

ওয়ার্ডপ্রেস



# সুচিপত্র

সম্পর্কিত	1
অধ্যায় 1: ওয়ার্ডপ্রেস দিয়ে শুরু করা	2
মন্তব্য	2
সংস্করণ	3
উদাহরণ	4
ওয়ার্ডপ্রেস পরিচিতি	4
ওয়ার্ডপ্রেস থিম	5
নির্দিষ্ট টেমপ্লেটে URL ম্যাপিং	5
মৌলিক থিম ডিরেক্টরি কাঠামো	5
একটি "একক" এর উদাহরণ (একটি পৃথক পোস্টের জন্য টেমপ্লেট)	6
একটি "আর্কাইভ" এর উদাহরণ (একাধিক পোস্টের তালিকার জন্য টেমপ্লেট)	7
পোস্ট, পৃষ্ঠা, কাস্টম পোস্ট প্রকার, এবং কাস্টম ক্ষেত্র	7
অধ্যায় 2: অ্যাকশন এবং ফিল্টার	8
বাক্য গঠন	8
পরামিতি	8
মন্তব্য	8
উদাহরণ	11
add_action - init	11
add_action - init - বেনামী ফাংশন	11
add_action - init - ক্লাস অবজেক্টের মধ্যে	11
add_action - init - স্ট্যাটিক ক্লাসের মধ্যে	12
অধ্যায় 3: শর্টকোড যোগ করুন	13
বাক্য গঠন	13
পরামিতি	13
মন্তব্য	13
উদাহরণ	13
সাম্প্রতিক পোস্টের জন্য সহজ শর্টকোড	13
সাম্প্রতিক পোস্টের জন্য উন্নত শর্টকোড	14
অধ্যায় 4: user_contactmethods ফিল্টার হুক সহ ব্যবহারকারীদের জন্য যোগাযোগের তথ্য যোগ করুন/সরান	15

উদাহরণ	15
সর্বাধিক জনপ্রিয় সামাজিক নেটওয়ার্ক সক্রিয় করা হচ্ছে।	15
যোগাযোগের পদ্ধতি সরানো হচ্ছে	16
<b>অধ্যায় 5: add_action()</b>	<b>17</b>
বাক্য গঠন	17
পরামিতি	17
উদাহরণ	17
সরাসরি ফাংশন কলব্যাক	17
ফাংশন নামের রেফারেন্স কলব্যাক	17
ক্লাস স্ট্যাটিক পদ্ধতি কলব্যাক	18
অবজেক্ট পদ্ধতি কলব্যাক	18
<b>অধ্যায় 6: add_editor_style()</b>	<b>19</b>
ভূমিকা	19
বাক্য গঠন	19
পরামিতি	19
উদাহরণ	19
একটি একক CSS ফাইল লোড হচ্ছে	19
<b>অধ্যায় 7: add_menu_page()</b>	<b>20</b>
ভূমিকা	20
বাক্য গঠন	20
পরামিতি	20
মন্তব্য	20
উদাহরণ	21
নেভি বারে "খিম পৃষ্ঠার শিরোনাম" আইটেম যোগ করা হচ্ছে	21
OOP এবং কিভাবে মেনু পৃষ্ঠায় স্ক্রিপ্ট/স্টাইল লোড করতে হয়	22
<b>অধ্যায় 8: add_submenu_page()</b>	<b>24</b>
ভূমিকা	24
বাক্য গঠন	24
পরামিতি	24
মন্তব্য	24
উদাহরণ	25

এনএভি বারে "সাবমেনু পৃষ্ঠা" "টুলস" এর একটি উপ-পৃষ্ঠা হিসাবে যুক্ত করা হচ্ছে	25
অধ্যায় 9: add_theme_support()	27
ভূমিকা	27
বাক্য গঠন	27
পরামিতি	27
মন্তব্য	27
উদাহরণ	27
পোস্ট ফরম্যাটের জন্য থিম সমর্থন যোগ করা হচ্ছে	27
পোস্টে থাম্বনেইলের জন্য থিম সমর্থন যোগ করা হচ্ছে	27
অধ্যায় 10: অ্যাডমিন ড্যাশবোর্ড উইজেট	29
ভূমিকা	29
বাক্য গঠন	29
পরামিতি	29
উদাহরণ	29
সহজ উইজেট (পাঠ্য প্রদর্শন করে)	29
অধ্যায় 11: AJAX	31
উদাহরণ	31
একটি JSON প্রতিক্রিয়া সহ AJAX অনুরোধ	31
AJAX এর সাথে .ajax() এবং WordPress Nonce	32
wp_ajax - মূল কার্যকারিতা + _wpnonce চেক	33
OOP Ajax জমা না দিয়ে একটি সাধারণ ক্লাস ব্যবহার করে	34
অধ্যায় 12: বিকল্প প্রধান লুপ (প্রি_গেট_পোস্ট ফিল্টার)	38
বাক্য গঠন	38
পরামিতি	38
মন্তব্য	38
উদাহরণ	38
এমনকি আরো নির্দিষ্ট লুপ টার্গেটিং	38
শুধুমাত্র একটি বিভাগ থেকে পোস্ট দেখান	39
প্রি পেতে পোস্ট ফিল্টার মৌলিক ব্যবহার	39
পোস্ট তালিকা সম্পাদনা শেয়ার থেকে বিভাগ বাদ দিন	39
প্রধান লুপের জন্য পোস্ট_প্রতি_পৃষ্ঠা পরিবর্তন করুন	40
শুধুমাত্র প্রধান ওয়ার্ডপ্রেস লুপ টার্গেটিং	40

অধ্যায় 13: চাইল্ড থিম বেসিক	41
বাক্য গঠন	41
মন্তব্য	41
উদাহরণ	41
2) উদ্দেশ্য	41
সংজ্ঞা এবং প্রয়োজনীয়তা	42
3) টেমপ্লেট ওভাররাইটিং	42
সম্পদ প্রতিস্থাপন	43
অধ্যায় 14: প্রোগ্রামগতভাবে একটি পোস্ট তৈরি করুন	45
বাক্য গঠন	45
পরামিতি	45
মন্তব্য	45
যুক্তি	45
ডুপ্লিকেটেড পোস্ট এড়িয়ে চলুন	46
ব্যাখ্যা	47
উদাহরণ	47
ভূমিকা	47
একটি মৌলিক পোস্ট তৈরি করুন	47
একটি মৌলিক পৃষ্ঠা তৈরি করুন	47
অধ্যায় 15: কাস্টম পোস্ট টাইপের জন্য টেমপ্লেট তৈরি করুন	49
উদাহরণ	49
কাস্টম পোস্ট টাইপ বইয়ের জন্য একটি কাস্টম টেমপ্লেট তৈরি করা	49
কাস্টম পোস্ট টাইপ টেমপ্লেট	49
কাস্টম পোস্ট টাইপ আর্কাইভ:	49
কাস্টম পোস্ট টাইপ একক টেমপ্লেট:	51
অধ্যায় 16: একটি কাস্টম টেমপ্লেট তৈরি করা	53
উদাহরণ	53
মৌলিক ফাঁকা টেমপ্লেট তৈরি করা হচ্ছে	53
আমাদের টেমপ্লেটে হেডার এবং ফুটার সহ	54
বিষয়বস্তু সহ কাস্টম টেমপ্লেট	54
অধ্যায় 17: excerpt_length এবং excerpt_more সহ কাস্টম এক্সপার্ট	56

উদাহরণ	56
উদ্ধৃতির দৈর্ঘ্য 50 শব্দের মধ্যে সীমাবদ্ধ করুন	56
উদ্ধৃতির শেষে আরও পড়ুন লিঙ্ক যুক্ত করা হচ্ছে	56
উদ্ধৃতির শেষে কয়েকটি বিল্ডু যোগ করা হচ্ছে	57
<b>অধ্যায় 18: কাস্টম পোস্ট প্রকার</b>	<b>59</b>
বাক্য গঠন	59
পরামিতি	59
উদাহরণ	59
একটি কাস্টম পোস্ট টাইপ নিবন্ধন	59
প্রধান ক্যোয়ারীতে কাস্টম পোস্ট প্রকার যোগ করুন	60
প্রধান RSS ফিডে কাস্টম পোস্টের ধরন যোগ করা হচ্ছে	61
কাস্টম পোস্ট টাইপ নিবন্ধন করুন	61
টোয়েন্টি ফিফটিন ওয়ার্ডপ্রেস থিম ব্যবহার করে কাস্টম পোস্ট টাইপ	62
ডিফল্ট অনুসন্ধানে কাস্টম পোস্ট প্রকার	63
<b>অধ্যায় 19: কাস্টমাইজার বেসিকস (প্যানেল, বিভাগ, সেটিং, নিয়ন্ত্রণ যোগ করুন)</b>	<b>64</b>
উদাহরণ	64
একটি কাস্টমাইজার প্যানেল যোগ করুন	64
মৌলিক সেটিংস এবং তাদের নিয়ন্ত্রণ সহ একটি কাস্টমাইজার বিভাগ যোগ করুন	64
<b>অধ্যায় 20: কাস্টমাইজার হ্যালো ওয়ার্ল্ড</b>	<b>68</b>
পরামিতি	68
উদাহরণ	68
হ্যালো ওয়ার্ল্ড উদাহরণ	68
<b>অধ্যায় 21: ডিবাগিং</b>	<b>70</b>
ভূমিকা	70
মন্তব্য	70
উদাহরণ	70
WP_DEBUG	70
WP_DEBUG_LOG	70
WP_DEBUG_DISPLAY	70
SCRIPT_DEBUG	71
সংরক্ষণ	71
উদাহরণ wp-config.php এবং ডিবাগিংয়ের জন্য ভাল অনুশীলন	71

একটি পৃথক ফাইলে লগ দেখুন	72
<b>অধ্যায় 22: সারিবদ্ধ স্ক্রিপ্ট</b>	<b>73</b>
বাক্য গঠন	73
পরামিতি	73
উদাহরণ	73
functions.php-এ স্ক্রিপ্ট সারিবদ্ধ করা	73
শুধুমাত্র IE এর জন্য সারিবদ্ধ স্ক্রিপ্ট	74
নির্দিষ্ট পৃষ্ঠাগুলির জন্য শর্তসাপেক্ষে স্ক্রিপ্টগুলি সারিবদ্ধ করা	74
<b>অধ্যায় 23: সারিবদ্ধ শৈলী</b>	<b>76</b>
বাক্য গঠন	76
পরামিতি	76
উদাহরণ	76
নির্ভরতা হিসাবে অন্য CSS ফাইলের সাথে অভ্যন্তরীণ CSS ফাইল সহ	76
অভ্যন্তরীণ CSS ফাইল সহ	76
বাহ্যিক CSS ফাইল সহ	76
শুধুমাত্র IE এর জন্য স্টাইলশীট সারিবদ্ধ করুন	77
আপনার প্লাগইন ক্লাসের জন্য অভ্যন্তরীণ CSS ফাইল সহ	77
বিকল্প স্টাইলশীট যোগ করুন	77
<b>অধ্যায় 24: ফাংশন : wp_trim_words()</b>	<b>79</b>
বাক্য গঠন	79
পরামিতি	79
উদাহরণ	79
পোস্টের বিষয়বস্তু ছাঁটাই	79
<b>অধ্যায় 25: ফাংশন: add_action()</b>	<b>80</b>
বাক্য গঠন	80
পরামিতি	80
মন্তব্য	80
উদাহরণ	80
বেসিক অ্যাকশন হুক	80
অ্যাকশন হুক অগ্রাধিকার	81
হকিং ক্লাস এবং অবজেক্ট মেথডস টু অ্যাকশন	81
<b>অবজেক্ট মেথড অ্যাকশন হুক</b>	<b>82</b>

ক্লাস মেথড অ্যাকশন হুক .....	82
অধ্যায় 26: get_bloginfo() .....	84
ভূমিকা .....	84
বাক্য গঠন .....	84
পরামিতি .....	84
মন্তব্য .....	84
উদাহরণ .....	86
সাইটের শিরোনাম হচ্ছে .....	86
সাইট ট্যাগলাইন পেয়ে .....	87
সক্রিয় থিম URL পাওয়া .....	৮৮
সাইটের ইউআরএল পান .....	৮৯
সাইট অ্যাডমিনিস্ট্রেটরের ইমেল ঠিকানা পান .....	৮৯
অধ্যায় 27: get_home_path() .....	90
ভূমিকা .....	90
পরামিতি .....	90
মন্তব্য .....	90
get_home_path() এবং ABSPATH-এর মধ্যে গুরুত্বপূর্ণ পার্থক্য .....	90
আপনার কোড এটি ব্যবহার করে .....	91
উদাহরণ .....	91
ব্যবহার .....	91
অধ্যায় 28: get_option() .....	92
ভূমিকা .....	92
বাক্য গঠন .....	92
পরামিতি .....	92
মন্তব্য .....	92
উদাহরণ .....	92
ব্লগ শিরোনাম দেখান .....	92
H1 স্টাইলে ব্লগের নাম .....	93
ক্যারেক্টার সেট দেখান .....	93
অ-বিদ্যমান বিকল্পগুলি পরিচালনা করা .....	93
অধ্যায় 29: get_permalink() .....	94

ভূমিকা	94
বাক্য গঠন	94
পরামিতি	94
মন্তব্য	94
উদাহরণ	94
get_parmalink এর সহজ ব্যবহার	94
লিঙ্ক পেতে পোস্ট উল্লেখ	94
পোস্টের নাম ছাড়া লিঙ্ক পান	95
<b>অধ্যায় 30: get_template_part()</b>	<b>96</b>
বাক্য গঠন	96
পরামিতি	96
উদাহরণ	96
একটি সাবফোল্ডার থেকে একটি টেমপ্লেট অংশ লোড করুন	96
একটি নির্দিষ্ট ফাইল পান	96
<b>অধ্যায় 31: get_template_part()</b>	<b>97</b>
ভূমিকা	97
বাক্য গঠন	97
পরামিতি	97
উদাহরণ	97
একটি কাস্টম টেমপ্লেট সহ	97
একটি ড্যাশ-বিচ্ছিন্ন ফাইলের নাম সহ একটি কাস্টম টেমপ্লেট সহ	97
একটি ডিরেক্টরির ভিতর থেকে একটি কাস্টম টেমপ্লেট সহ	98
একটি ডিরেক্টরির ভিতরে অবস্থিত একটি ড্যাশ-বিচ্ছিন্ন ফাইলের নাম সহ একটি কাস্টম টেমপ্লেট সহ	98
কাস্টম টেমপ্লেট সুযোগ পরিবর্তনশীল পাস	98
<b>অধ্যায় 32: get_template_part()</b>	<b>99</b>
বাক্য গঠন	99
পরামিতি	99
উদাহরণ	99
টেমপ্লেট অংশ লোড হচ্ছে	99
<b>অধ্যায় 33: get_the_category()</b>	<b>100</b>
ভূমিকা	100

বাক্য গঠন	100
পরামিতি	100
মন্তব্য	100
উদাহরণ	100
পোস্টের সব বিভাগের নাম পান	101
পোস্টের সমস্ত বিভাগের আইডি পান	101
অধ্যায় 34: get_the_title()	102
ভূমিকা	102
বাক্য গঠন	102
পরামিতি	102
মন্তব্য	102
উদাহরণ	102
get_the_title এর সহজ ব্যবহার	102
একটি নির্দিষ্ট পোস্ট আইডির শিরোনাম পান	102
অধ্যায় 35: home_url()	104
বাক্য গঠন	104
পরামিতি	104
উদাহরণ	104
হোম URL প্রাপ্ত করা হচ্ছে	104
সাইট URL টি	104
অধ্যায় 36: কিভাবে আমি অ্যাডভান্স কাস্টম ফিল্ডের রিপিটার অ্যাড-ও-এর সাথে মার্কডাউন সম্পাদককে সংহত করতে পারি	105
উদাহরণ	105
মার্কডাউন এডিটর যোগ করুন	105
অধ্যায় 37: init	106
বাক্য গঠন	106
মন্তব্য	106
উদাহরণ	106
\$_POST অনুরোধ ডেটা প্রক্রিয়া করা হচ্ছে	106
\$_GET অনুরোধ ডেটা প্রক্রিয়া করা হচ্ছে	106
একটি কাস্টম পোস্ট টাইপ নিবন্ধন	106
অধ্যায় 38: ইনস্টলেশন এবং কনফিগারেশন	107

উদাহরণ	107
LAMP এ Wordpress	107
MAMP-তে WP ইনস্টলেশন	108
অধ্যায় 39: HTTP API দিয়ে নেটওয়ার্ক অনুরোধ করা	110
বাক্য গঠন	110
পরামিতি	110
মন্তব্য	110
রিটার্নস	110
উদাহরণ	110
একটি দূরবর্তী JSON সম্পদ পান	110
অধ্যায় 40: মেটা বক্স	111
ভূমিকা	111
বাক্য গঠন	111
মন্তব্য	111
উদাহরণ	111
একটি নিয়মিত ইনপুট এবং একটি নির্বাচন ইনপুট সহ একটি সাধারণ উদাহরণ	111
অধ্যায় 41: অপশন এপিআই	114
ভূমিকা	114
বাক্য গঠন	114
মন্তব্য	114
উদাহরণ	114
get_option	114
add_option	115
ডিলিট অপশন	115
আপডেট_বিকল্প	115
অধ্যায় 42: প্লাগইন উন্নয়ন	116
বাক্য গঠন	116
পরামিতি	116
মন্তব্য	116
উদাহরণ	117
ছাঁকনি	117

কর্ম	117
প্লাগইন উন্নয়ন উদাহরণ: প্রিয় গান উইজেট	117
<b>অধ্যায় 43: পোস্ট ফরম্যাট</b>	<b>119</b>
মন্তব্য	119
উদাহরণ	119
থিমের পোস্টের ধরন যোগ করা হচ্ছে	119
পোস্ট_টাইপ 'পৃষ্ঠা'-তে পোস্ট-ফরম্যাট যোগ করুন	119
কাস্টম পোস্ট টাইপ 'my_custom_post_type' নিবন্ধন করুন	119
পোস্ট_টাইপ 'my_custom_post_type'-এ পোস্ট-ফরম্যাট যোগ করুন	120
'সমর্থন' প্যারামিটার সহ কাস্টম পোস্ট টাইপ 'my_custom_post_type' নিবন্ধন করুন	120
পোস্টের জন্য থিম সমর্থন যোগ করুন	120
ফাংশন কল	120
<b>অধ্যায় 44: প্রশ্ন করা পোস্ট</b>	<b>121</b>
বাক্য গঠন	121
পরামিতি	121
মন্তব্য	121
কখনই query_posts() ব্যবহার করবেন না	121
উদাহরণ	122
WP_Query() অবজেক্ট ব্যবহার করে	122
get_posts() ব্যবহার করে	122
<b>অধ্যায় 45: বিষয়বস্তু এবং উদ্ধৃতি থেকে স্বয়ংক্রিয় লাইন ব্রেকগুলি সরান</b>	<b>124</b>
ভূমিকা	124
মন্তব্য	124
উদাহরণ	124
ফিল্টারগুলি সরান	124
ফিল্টার অপসারণ ফাংশন	124
<b>অধ্যায় 46: ওয়ার্ডপ্রেস এবং স্টাইলশীট থেকে সংস্করণ সরান</b>	<b>126</b>
ভূমিকা	126
বাক্য গঠন	126
পরামিতি	126
মন্তব্য	126
উদাহরণ	126

CSS/js থেকে সংস্করণ নম্বরগুলি সরান	126
ওয়ার্ডপ্রেস থেকে সংস্করণ নম্বর সরান	127
<b>অধ্যায় 47: REST API</b>	<b>128</b>
ভূমিকা	128
মন্তব্য	128
উদাহরণ	128
সম্পূর্ণ কাজের উদাহরণ	129
<b>অধ্যায় 48: XAMPP দিয়ে ওয়ার্ডপ্রেস লোকাল চালান</b>	<b>130</b>
ভূমিকা	130
উদাহরণ	130
1. XAMPP ইনস্টল করা	130
2. XAMPP ইনস্টল করার পরে একটি ডাটাবেস সেটআপ করুন	130
3. ডাটাবেস সেটআপ করার পরে ওয়ার্ডপ্রেস ইনস্টল করা	130
<b>অধ্যায় 49: আপনার ইনস্টলেশন সুরক্ষিত করুন</b>	<b>132</b>
মন্তব্য	132
উদাহরণ	132
ফাইল এডিটর অক্ষম করুন	132
wp-config.php সরান	132
ওয়ার্ডপ্রেস টেবিলের জন্য একটি কাস্টম উপসর্গ সেট করুন	133
<b>অধ্যায় 50: ওয়ার্ডপ্রেসে নিরাপত্তা - পালানো</b>	<b>138</b>
বাক্য গঠন	138
মন্তব্য	138
উদাহরণ	138
এইচটিএমএল কোড এক্সেপ ডেটা	138
একটি url এক্সেপ	138
জেএস কোডে ডেটা এক্সেপ করুন	138
এক্সেপ বৈশিষ্ট্য	139
টেক্সটরিয়া এক্সেপ ডেটা	139
<b>অধ্যায় 51: ওয়ার্ডপ্রেসে নিরাপত্তা - স্যানিটাইজেশন</b>	<b>140</b>
বাক্য গঠন	140
মন্তব্য	140
উদাহরণ	140

পাঠ্য ক্ষেত্র স্যানিটাইজ করুন	140
স্যানিটাইজ শিরোনাম	140
ইমেইল স্যানিটাইজ করুন	141
এইচটিএমএল ক্লাস স্যানিটাইজ করুন	141
ফাইলের নাম স্যানিটাইজ করুন	141
ব্যবহারকারীর নাম স্যানিটাইজ করুন	141
<b>অধ্যায় 52: শর্টকোড</b>	<b>142</b>
উদাহরণ	142
শর্টকোড নিবন্ধন	142
ওয়ার্ডপ্রেস ব্যাকএন্ডে শর্টকোড ব্যবহার করা	142
নতুন শর্টকোড যোগ করা হচ্ছে	142
পিএইচপি কোডের ভিতরে শর্টকোড ব্যবহার করা (থিম এবং প্লাগইন)	143
উইজেটে শর্টকোড ব্যবহার করা	143
<b>অধ্যায় 53: বৈশিষ্ট্য সহ শর্টকোড</b>	<b>144</b>
বাক্য গঠন	144
পরামিতি	144
মন্তব্য	144
উদাহরণ	144
শর্টকোডের উদাহরণ	144
একটি স্ব-বন্ধ শর্টকোড তৈরি করা হচ্ছে	144
পরামিতি সহ একটি স্ব-বন্ধ শর্টকোড তৈরি করা	145
একটি আবদ্ধ শর্টকোড তৈরি করা হচ্ছে	145
উইজেটে শর্টকোড	146
<b>অধ্যায় 54: শর্টকোড</b>	<b>147</b>
উদাহরণ	147
শর্টকোড ভূমিকা	147
বোতাম শর্টকোড	147
<b>অধ্যায় 55: সাইডবার</b>	<b>149</b>
বাক্য গঠন	149
পরামিতি	149
মন্তব্য	149
উদাহরণ	149

সাইডবার নিবন্ধন	149
সাইডবার পান	150
অধ্যায় 56: সাইট মাইগ্রেশন	151
বাক্য গঠন	151
উদাহরণ	151
একটি নতুন URL দিয়ে টেবিল আপডেট করা হচ্ছে	151
অধ্যায় 57: শ্রেণীবিন্যাস	152
বাক্য গঠন	152
পরামিতি	152
উদাহরণ	152
জেনারগুলির জন্য একটি শ্রেণীবিন্যাস নিবন্ধনের উদাহরণ	152
পৃষ্ঠায় বিভাগ যোগ করুন	153
পৃষ্ঠাগুলিতে বিভাগ এবং ট্যাগ যুক্ত করুন এবং সেগুলিকে শ্রেণী হিসাবে সন্নিবেশ করুন।	153
পৃষ্ঠাগুলিতে বিভাগ এবং ট্যাগ যোগ করুন এবং ক্লাস হিসাবে সন্নিবেশ করুন	154
অধ্যায় 58: টেমপ্লেট অনুক্রম	156
মন্তব্য	156
উদাহরণ	156
ভূমিকা	156
ডিবাগিং	158
অধ্যায় 59: টেমপ্লেট_ইনক্লুড	159
পরামিতি	159
মন্তব্য	159
উদাহরণ	159
সহজ উদাহরণ	159
আরো Adv উদাহরণ	160
অধ্যায় 60: \$wpdb অবজেক্ট	161
মন্তব্য	161
উদাহরণ	161
একটি পরিবর্তনশীল নির্বাচন	161
একাধিক সারি নির্বাচন করা হচ্ছে	161
অধ্যায় 61: অ্যাডমিন বার (ওরফে "দ্য টুলবার")	163

মন্তব্য	163
উদাহরণ	163
প্রশাসক ছাড়া সকলের কাছ থেকে অ্যাডমিন টুলবার সরানো হচ্ছে	163
ফিল্টার ব্যবহার করে অ্যাডমিন টুলবার সরানো হচ্ছে	163
অ্যাডমিন বার থেকে কিভাবে ওয়ার্ডপ্রেস লোগো সরান	163
অ্যাডমিন লগইন পৃষ্ঠায় আপনার কাস্টম লোগো এবং কাস্টম লিঙ্ক যোগ করুন	164
অধ্যায় 62: লুপ (প্রধান ওয়ার্ডপ্রেস লুপ)	165
উদাহরণ	165
বেসিক ওয়ার্ডপ্রেস লুপ স্ট্রাকচার	165
বিকল্প লুপ সিনট্যাক্স	165
লুপে কোন আইটেম হ্যান্ডলিং	165
অধ্যায় 63: the_title()	167
ভূমিকা	167
বাক্য গঠন	167
পরামিতি	167
মন্তব্য	167
উদাহরণ	167
শিরোনামের সহজ ব্যবহার	167
আগে এবং পরে কোড সহ শিরোনাম প্রিন্ট করা	167
অধ্যায় 64: থিম	169
ভূমিকা	169
উদাহরণ	169
ওয়ার্ডপ্রেস থিম	169
কিভাবে একটি থিম চয়ন করুন	169
আপডেট উপলব্ধ	169
থিম ইনস্টল করুন	170
একটি কাস্টম থিম তৈরি করা	171
অধ্যায় 65: ম্যানুয়ালি ওয়ার্ডপ্রেস আপডেট করুন	172
উদাহরণ	172
এফটিপি মাধ্যমে	172
অধ্যায় 66: ওয়ার্ডপ্রেস প্লাগইন তৈরি	173

ভূমিকা	173
উদাহরণ	173
একটি প্লাগইন ফোল্ডার এবং ফাইলের ন্যূনতম সেটআপ	173
<b>অধ্যায় 67: ওয়ার্ডপ্রেস থিম এবং শিশু-থিম বিকাশ</b>	<b>175</b>
ভূমিকা	175
উদাহরণ	175
আপনার নিজস্ব থিম বিকাশ	175
<b>অধ্যায় 68: wp_get_current_user()</b>	<b>178</b>
বাক্য গঠন	178
উদাহরণ	178
বর্তমান ব্যবহারকারী পেয়ে	178
wp_get_current_user() থেকে ব্যবহারকারীর তথ্য পেতে foreach লুপ ব্যবহার করুন	178
<b>অধ্যায় 69: wp_get_current_user()</b>	<b>179</b>
উদাহরণ	179
বর্তমান লগইন ব্যবহারকারী তথ্য পান	179
<b>অধ্যায় 70: WP_Query() লুপ</b>	<b>180</b>
ভূমিকা	180
উদাহরণ	180
সর্বশেষ 10টি পোস্ট পুনরুদ্ধার করুন	180
<b>অধ্যায় 71: WP-CLI</b>	<b>181</b>
ভূমিকা	181
উদাহরণ	181
থিম পরিচালনা করুন	181
প্লাগইন পরিচালনা করুন	181
WP-CLI নিজেই পরিচালনা করুন	182
একটি ওয়ার্ডপ্রেস ইনস্টল ডাউনলোড, ইনস্টল, আপডেট এবং পরিচালনা করুন।	183
ব্যবহারকারীদের ম্যানেজ করুন	183
wp-config.php-এ সংরক্ষিত শংসাপত্র ব্যবহার করে মৌলিক ডাটাবেস অপারেশনগুলি সম্পাদন করুন	183
<b>অধ্যায় 72: WP-Cron</b>	<b>185</b>
উদাহরণ	185
wp_schedule_event() উদাহরণ	185
wp_schedule_event() এ কাস্টম পুনরাবৃত্তি ব্যবধান	185



---

## সম্পর্কিত

আপনি যে কারো সাথে এই পিডিএফটি শেয়ার করতে পারেন যাকে আপনি এটি থেকে উপকৃত হতে পারেন বলে মনে করেন, সর্বশেষ সংস্করণ ডাউনলোড করেছেন: wordpress

এটি একটি অনানুষ্ঠানিক এবং বিনামূল্যের ওয়ার্ডপ্রেস ইবুক যা শিক্ষামূলক উদ্দেশ্যে তৈরি করা হয়েছে। সমস্ত বিষয়বস্তু স্ট্যাক ওভারফ্লো ডকুমেন্টেশন থেকে বের করা হয়েছে, যা স্ট্যাক ওভারফ্লোতে অনেক পরিশ্রমী ব্যক্তি দ্বারা লেখা। এটি স্ট্যাক ওভারফ্লো বা অফিসিয়াল ওয়ার্ডপ্রেসের সাথে অনুমোদিত নয়।

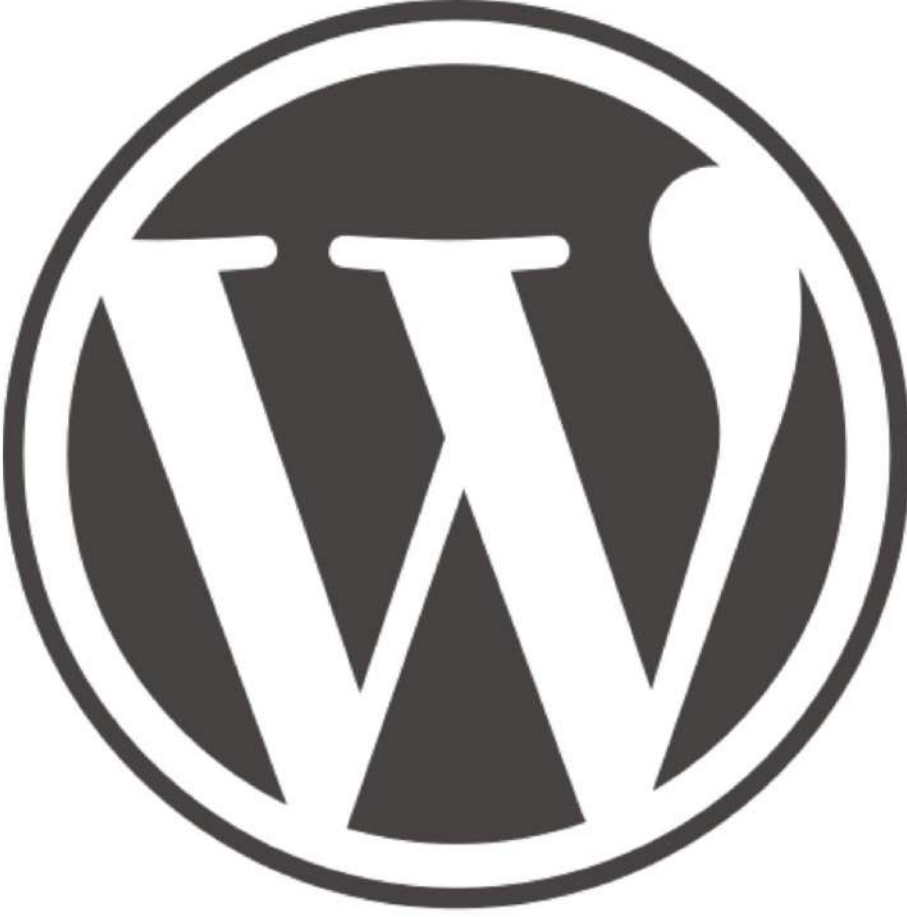
বিষয়বস্তু Creative Commons BY-SA-এর অধীনে প্রকাশিত হয়েছে, এবং প্রতিটি অধ্যায়ে অবদানকারীদের তালিকা এই বইয়ের শেষে ক্রেডিট বিভাগে দেওয়া হয়েছে। ছবিগুলি তাদের নিজ নিজ মালিকদের কপিরাইট হতে পারে যদি না অন্যথায় নির্দিষ্ট করা হয়। সমস্ত ট্রেডমার্ক এবং নিবন্ধিত ট্রেডমার্ক তাদের নিজ নিজ কোম্পানির মালিকদের সম্পত্তি।

আপনার নিজের ঝুঁকিতে এই বইটিতে উপস্থাপিত বিষয়বস্তু ব্যবহার করুন; এটি সঠিক বা নির্ভুল হওয়ার নিশ্চয়তা নেই, অনুগ্রহ করে আপনার প্রতিক্রিয়া এবং সংশোধনগুলি [info@zzzprojects.com](mailto:info@zzzprojects.com) এ পাঠান

---

# অধ্যায় 1: ওয়ার্ডপ্রেস দিয়ে শুরু করা

## মন্তব্য



ওয়ার্ডপ্রেস হল একটি ওপেন সোর্স কন্টেন্ট ম্যানেজমেন্ট সিস্টেম (CMS) যা ওয়েবসাইট তৈরি এবং পরিচালনা করতে ব্যবহৃত হয়। ওয়ার্ডপ্রেস হল ইন্টারনেটে সবচেয়ে জনপ্রিয় সিএমএস এক দেশের মাইল, প্রায় অর্ধেক শক্তি লেখার সময় সমস্ত CMS ওয়েবসাইট এবং সমস্ত ওয়েবসাইটের প্রায় এক চতুর্থাংশ ইন্টারনেটে।

ওয়ার্ডপ্রেস ব্লগিংয়ের জন্য একটি প্ল্যাটফর্ম হিসাবে জীবন শুরু করেছিল কিন্তু বছরের পর বছর ধরে এটি বেশিরভাগ ধরণের ওয়েবসাইটের জন্য উপযুক্ত হয়ে উঠেছে। ইন্টারফেসটি কোডিং জ্ঞান ছাড়াই ব্যবহার করা যেতে পারে এটি নতুনদের এবং বিকাশকারীদের জন্য জনপ্রিয় করে তোলে যারা তাদের ক্লায়েন্টদের তাদের নিজস্ব ওয়েবসাইট পরিচালনা করার ক্ষমতা দিতে চায়।

ওয়ার্ডপ্রেসের জনপ্রিয়তার আরেকটি বড় কারণ হল এর নমনীয়তা, বেশিরভাগই মূলের প্লাগইন এবং থিমিং সিস্টেমের কারণে। প্লাগইন সিস্টেম মূল কোড পরিবর্তন না করে মূল কার্যকারিতা প্রসারিত করা সহজ করে তোলে। একইভাবে, থিমিং সিস্টেম ওয়েবসাইটের লেআউট এবং নান্দনিকতা পরিবর্তন করা সহজ করে তোলে। এখন হাজার হাজার ফ্রি এবং প্রিমিয়াম ওয়ার্ডপ্রেস প্লাগইন এবং থিম পাওয়া যাচ্ছে। এর মধ্যে অনেকগুলি [wordpress.org](http://wordpress.org) প্লাগইন সংগ্রহস্থলে অবস্থিত এবং থিম সংগ্রহস্থল যথাক্রমে

ওয়ার্ডপ্রেস তার নিজস্ব সম্প্রদায় দ্বারা বিকশিত, কিন্তু দৃঢ়ভাবে কোম্পানি Automattic সঙ্গে যুক্ত, যা ওয়ার্ডপ্রেসের অনেক মূল বিকাশকারীকে নিয়োগ করে।

কোড

ওয়ার্ডপ্রেস পিএইচপি এর উপর নির্মিত সার্ভার স্ক্রিপ্টিং ভাষা এবং মাইএসকিউএল প্রশ্ন করার ভাষা।

ওয়ার্ডপ্রেস ব্যবহারকারীর সামগ্রী এবং কনফিগারেশনের জন্য একটি ডেটাস্টোর হিসাবে MySQL ব্যবহার করে। পিএইচপি ঝগড়া একটি HTML এ বিষয়বস্তু ডেটা সমস্ত প্রয়োজনীয় সম্পদ সহ ওয়েবপৃষ্ঠা।

wordpress.com বনাম wordpress.org

আপনি Automattic এর [wordpress.com](https://wordpress.com) এ সাইন আপ করে ওয়ার্ডপ্রেস ব্যবহার করতে পারেন সেবা এবং হোস্টিং আপনার তাদের সার্ভারে ওয়েবসাইট, অথবা আপনি [wordpress.org](https://wordpress.org) থেকে ওয়ার্ডপ্রেস সফটওয়্যার ডাউনলোড করতে পারেন এবং আপনার নিয়ন্ত্রণের অধীনে একটি সার্ভারে আপনার ওয়েবসাইট হোস্ট করুন। প্রথম বিকল্পটি সহজ কিন্তু আপনি কোনো সম্পাদনা করতে পারবেন না সাইটের কোড। আপনি শুধুমাত্র ওয়ার্ডপ্রেস ইন্টারফেসের মাধ্যমে পরিবর্তন করতে পারেন। দ্বিতীয় বিকল্প আরও কাজের প্রয়োজন কিন্তু আপনার ওয়েবসাইটের কোড দিয়ে যা খুশি তাই করার জন্য আপনাকে নমনীয়তা দেয়। আপনি যদি একজন StackOverflow ব্যবহারকারী, আপনি সম্ভবত দ্বিতীয় বিকল্পের সাথে যাচ্ছেন।

মুক্ত উৎস

ওয়ার্ডপ্রেস হল ওপেন সোর্স সফটওয়্যার যার অর্থ এটি বিনামূল্যে ব্যবহার করা যায় এবং যে কেউ উত্সটি দেখতে পারে কোড এবং এটি অবদান। সম্ভাব্য অবদানকারীরা এর অবদান পৃষ্ঠাটি পড়ে শুরু করতে পারেন [ওয়ার্ডপ্রেস কোডেক্স](#)..

ওয়ার্ডপ্রেস টিকিট ট্র্যাকারে একটি বাগ জমা দিয়ে বাগ রিপোর্ট করা যেতে পারে ।

ডকুমেন্টেশন

ওয়ার্ডপ্রেস আনুষ্ঠানিকভাবে ওয়ার্ডপ্রেস কোডেক্সে নথিভুক্ত WordPress.org এ । বিকাশকারীরা ওয়ার্ডপ্রেসের সাথে কাজ করলে ডেভেলপার কোডেক্সে বিশেষভাবে আগ্রহী হবে বিভাগ এবং [বিকাশকারী রেফারেন্স wordpress.org](#) এর বিভাগ ।

## সংস্করণ

সংস্করণ প্রকাশের তারিখ	
1.0	2004-01-03
1.2	2004-05-22
1.5	2005-02-17
2.0	2005-12-26
2.1	2007-01-22
2.2	2007-05-16
2.3	2007-09-24
2.5	2008-03-29
2.6	2008-07-15
2.7	2008-12-10

সংস্করণ প্রকাশের তারিখ	
2.8	2009-06-10
2.9	2009-12-18
3.0	2010-06-17
3.1	2011-02-23
3.2	2011-07-04
3.3	2011-12-12
3.4	2012-06-13
3.5	2012-12-11
3.6	2013-08-01
3.7	2013-10-24
3.8	2013-12-12
3.9	2014-04-16
4.0	2014-09-04
4.1	2014-12-17
4.2	2015-04-23
4.3	2015-08-18
4.4	2015-12-08
4.5	2016-04-12
4.6	2016-08-16
4.7	2016-12-06
4.8	2017-06-08

## উদাহরণ

ওয়ার্ডপ্রেস পরিচিতি

ওয়ার্ডপ্রেস [WP] অ্যাপ, ওয়েবসাইট এবং নির্মাণের জন্য একটি ওপেন সোর্স কন্টেন্ট ম্যানেজমেন্ট সিস্টেম  
ব্লগ WP PHP-তে লেখা এবং ব্যবহারকারীর বিষয়বস্তুর জন্য ডেটা স্টোর হিসাবে MySQL ব্যবহার করে

কনফিগারেশন. এটিতে প্লাগইনগুলির একটি সমৃদ্ধ ইকোসিস্টেম রয়েছে এবং থিম এবং একটি প্রাণবন্ত ওপেন সোর্স সম্প্রদায়, ভাল ডকুমেন্টেশন এবং প্রবেশে কম বাধা উপভোগ করে। ব্যবহারযোগ্যতা এবং বিকাশকারী ডকুমেন্টেশন WP কোডেব্লু পাওয়া যাবে।

ওয়ার্ডপ্রেসের একটি অংশ যা এটিকে অন্যান্য সিএমএস পণ্য থেকে আলাদা করে তোলে তা হল এর ইভেন্ট ড্রাইভেন প্রোগ্রামিং। এটি প্রোগ্রামিং এবং যুক্তি উপস্থাপনের একটি ভিন্ন উপায় তারপর MVC (মডেল ভিউ কন্ট্রোলার) আর্কিটেকচার যা বেশিরভাগ সিএমএস সিস্টেম দ্বারা ব্যবহৃত হয়। ওয়ার্ডপ্রেস অ্যাকশন এবং ফিল্টার ধারণা ব্যবহার করে। তারা ইভেন্টগুলির একটি সারি তৈরি করে যা প্লাগইন এবং থিমগুলিকে চূড়ান্ত ওয়েব অ্যাপ্লিকেশন পৃষ্ঠার অংশগুলি এবং/অথবা অংশগুলি সন্নিবেশ, সংশোধন বা এমনকি সরানোর অনুমতি দেয়। একটি অনুরূপ ধারণা JIT বা জাস্ট-ইন-টাইম কম্পাইলিং।

যদিও ঐতিহাসিকভাবে ওয়ার্ডপ্রেস একটি ব্লগিং প্ল্যাটফর্ম হিসাবে পরিচিত, এবং এটি এই কলক্লেট কখনই হারাতে পারে না, মূল ওয়ার্ডপ্রেস দলের ফোকাস স্পষ্টভাবে পরিবর্তিত হয়েছে। 2016 রাজ্যের শব্দের সাথে, প্রতিষ্ঠাতা ম্যাথিউ মুলেনওয়েগের দ্বারা, আমরা লক্ষ্য, দৃষ্টি এবং প্রচেষ্টার একটি স্পষ্ট পরিবর্তন দেখতে পাচ্ছি। 2016 সালে, আমরা আশ্চর্যজনক অগ্রগতি দেখেছি যখন ওয়ার্ডপ্রেস কোর বেশিরভাগ জনপ্রিয় REST API প্লাগইন গ্রহণ করেছে। এটি স্পষ্টতই মূল দলের উদ্দেশ্য ছিল প্রথম দিকে যখন তারা একটি ফ্রন্ট-এন্ড জাভাস্ক্রিপ্ট সিএমএস অ্যাডমিন প্যানেল তৈরি করার সাহসী প্রচেষ্টা শুরু করেছিল, যা আমরা এত বছর ধরে দেখেছি সোনালী মান থেকে দূরে চলে যায়; তারা একে ক্যালপিসো বলে।

ওয়ার্ডপ্রেস থিম

## নির্দিষ্ট টেমপ্লেটে URL ম্যাপিং

ওয়ার্ডপ্রেস থিমগুলি সম্পূর্ণরূপে উপলব্ধি করতে, আপনাকে অবশ্যই দুটি প্রাথমিক ধারণা বুঝতে হবে:

1. পারমালিঙ্কস 2.

টেমপ্লেট শ্রেণিবিন্যাস

একটি পারমালিঙ্ক হল একটি স্থায়ী, অ-পরিবর্তনকারী URL (বা লিঙ্ক, একটি নির্দিষ্ট সংস্থানের সাথে। উদাহরণস্বরূপ:

- example.com/about-us/ (WP-এ একটি পৃষ্ঠা) •
- example.com/services/ (একাধিক আইটেমের একটি তালিকা, যাকে WP লিংগোতে "আর্কাইভ"ও বলা হয়) •
- example.com/services/we-can- আপনার জন্য-করুন/ (একটি পৃথক আইটেম)

যখন একজন ব্যবহারকারী একটি URL এর জন্য অনুরোধ করেন, তখন কোন টেমপ্লেটটি তার লেআউটকে নিয়ন্ত্রণ করবে তা নির্ধারণ করতে ওয়ার্ডপ্রেস পারমালিঙ্ককে রিভার্স-ইঞ্জিনিয়ার করে। ওয়ার্ডপ্রেস বিভিন্ন টেমপ্লেট ফাইলের সন্ধান করে যা এই নির্দিষ্ট বিষয়বস্তুকে নিয়ন্ত্রণ করতে পারে এবং শেষ পর্যন্ত এটি খুঁজে পাওয়া সবচেয়ে নির্দিষ্টটিকে অগ্রাধিকার দেয়।

এটি টেমপ্লেট হায়ারার্কি নামে পরিচিত।

একবার WP অনুক্রমের মধ্যে মিলে যাওয়া ভিউ টেমপ্লেটটি খুঁজে পেলে, এটি সেই ফাইলটিকে পৃষ্ঠাটি প্রক্রিয়া ও রেন্ডার করতে ব্যবহার করে।

উদাহরণস্বরূপ: index.php (ডিফল্ট, "ক্যাচ-অল" টেমপ্লেট) archive.php (তালিকা-ভিত্তিক বিষয়বস্তুর জন্য ডিফল্ট টেমপ্লেট) দ্বারা ওভাররাইড করা হবে, যা archive-services.php (একটি টেমপ্লেট) দ্বারা ওভাররাইড করা হবে বিশেষভাবে "পরিষেবা" নামের সংরক্ষণাগারের জন্য ফাইল)।

এখানে টেমপ্লেট অনুক্রমের জন্য একটি দুর্দান্ত ভিজুয়াল রেফারেন্স রয়েছে

# মৌলিক থিম ডিরেক্টরি কাঠামো

একটি সাধারণ থিম এইরকম কিছু দেখায়:

```
// থিম সিএসএস
style.css

// আপনার থিমের জন্য কাস্টম কার্যকারিতা functions.php

// আংশিকগুলি পরবর্তী থিম ফাইলগুলিতে অন্তর্ভুক্ত করার জন্য header.php footer.php
sidebar.php
comments.php

// "আর্কাইভস", (একাধিক পোস্ট ধারণ করে এমন তালিকার দৃশ্য) archive.php author.php date.php
taxonomy.php
tag.php
category.php

// স্বতন্ত্র বিষয়বস্তু পৃষ্ঠাগুলি // মনে রাখবেন যে হোম
এবং ফ্রন্টপেজ টেমপ্লেটগুলি সুপারিশ করা হয় না // এবং সেগুলিকে পৃষ্ঠা টেমপ্লেটগুলি দ্বারা প্রতিস্থাপিত করা উচিত
singular.php single.php page.php front-page.php home.php

// বিবিধ। ইউটিলিটি পেজ index.php (অন্য
কিছু না মিললে একটি ক্যাচ-অল) search.php attachment.php image.php 404.php
```

## একটি "একক" এর উদাহরণ (একটি পৃথক পোস্টের জন্য টেমপ্লেট)

```
<?php get_header(); ?>

<?php if ( have_posts() ) যখন ( have_posts() ) <h1><?php the_title(); ?></h1> <?php : পোস্টটি(); ?>
the_content(); ?> <?php comments_template(
true); ?> <?php শেষ পর্যন্ত; ?>

<?php get_sidebar(); ?> <?php
get_footer(); ?>
```

এখানে কি হচ্ছে? প্রথমে, এটি header.php লোড করে (পিএইচপি অন্তর্ভুক্ত বা প্রয়োজনের অনুরূপ), লুপ সেট আপ করে, the\_title এবং the\_content প্রদর্শন করে, তারপর comments.php, sidebar.php এবং footer.php অন্তর্ভুক্ত করে।  
লুপ ভারী উত্তোলন করে, একটি পোস্ট অবজেক্ট সেট আপ করে, যা বর্তমানে দেখা বিষয়বস্তুর জন্য সমস্ত তথ্য ধারণ করে।

## একটি "আর্কাইভ" এর উদাহরণ (একাধিক পোস্টের তালিকার জন্য টেমপ্লেট)

```
<?php get_header(); ?>

<?php যদি ( have_posts() ) যখন ( have_posts() ) : পোস্টটি(); ?> <a href="<?php the_permalink(); ?>"><?php the_title(); ?></a> <?php the_excerpt(); ?> <?php শেষ পর্যন্ত; ?>

<?php
    next_posts_link( 'পুরানো এন্ট্রি', $the_query->max_num_pages); আগের_পোস্ট_লিঙ্ক ('নতুন এন্ট্রি');
?>

<?php get_sidebar(); ?> <?php
get_footer(); ?>
```

প্রথমত, এটি header.php অন্তর্ভুক্ত করে, লুপ সেট আপ করে এবং sidebar.php, এবং footer.php অন্তর্ভুক্ত করে। তবে এই ক্ষেত্রে লুপে একাধিক পোস্ট রয়েছে, তাই পরিবর্তে পৃথক পোস্টের লিঙ্ক সহ একটি উদ্ধৃতি দেখানো হয়েছে। পরবর্তী\_পোস্ট\_লিঙ্ক এবং পূর্ববর্তী\_পোস্ট\_লিঙ্কগুলিও অন্তর্ভুক্ত করা হয়েছে যাতে সংরক্ষণাগারটি ফলাফলের পৃষ্ঠায় স্থানান্তর করতে পারে।

## পোস্ট, পৃষ্ঠা, কাস্টম পোস্ট প্রকার, এবং অস্ত্রোপচার

বাক্সের বাইরে, ওয়ার্ডপ্রেস দুই ধরনের কন্টেন্ট সমর্থন করে: পোস্ট এবং পেজ। পোস্টগুলি সাধারণত ব্লগ পোস্টের মতো অ-শ্রেণীবিন্যাস বিষয়বস্তুর জন্য ব্যবহৃত হয়। পৃষ্ঠাগুলি স্ট্যাটিক, স্বতন্ত্র বিষয়বস্তুর জন্য ব্যবহার করা হয় যেমন একটি আমাদের সম্পর্কে পৃষ্ঠা, বা নীচে নেস্টেড সাব-পৃষ্ঠা সহ একটি কোম্পানির পরিষেবা পৃষ্ঠা।

ভার্সন 3.0 অনুযায়ী, ডেভেলপাররা ওয়ার্ডপ্রেসের কার্যকারিতাকে শুধুমাত্র মৌলিক বিষয়ের বাইরেও প্রসারিত করতে তাদের নিজস্ব কাস্টম পোস্টের ধরন সংজ্ঞায়িত করতে পারে। কাস্টম পোস্টের প্রকারগুলি ছাড়াও, আপনি আপনার পোস্ট/পৃষ্ঠা/কাস্টম পোস্টের ধরনগুলির সাথে সংযুক্ত করার জন্য আপনার নিজস্ব কাস্টম ক্ষেত্রগুলিও তৈরি করতে পারেন, যা আপনাকে আপনার টেমপ্লেটগুলির মধ্যে মেটাডেটা যোগ করার এবং অ্যাক্সেস করার একটি কাঠামোগত উপায় প্রদান করতে দেয়। দেখুন: উন্নত কাস্টম ক্ষেত্র।

## অধ্যায় 2: অ্যাকশন এবং ফিল্টার

### বাক্য গঠন

- `add_action( ট্যাগ, ফাংশন_টু_কল, অগ্রাধিকার, num_of_args);`
- `add_filter( ট্যাগ, ফাংশন_টু_কল, অগ্রাধিকার, num_of_args);`

### পরামিতি

প্যারামিটার ব্যাখ্যা	
\$ট্যাগ	(স্ট্রিং) (প্রয়োজনীয়) কর্মের নাম যেখানে \$ফাংশন হুক করা হয়েছে।
\$ফাংশন	(কলযোগ্য) (প্রয়োজনীয়) ফাংশনের নাম বা বেনামী ফাংশন ধারণকারী একটি স্ট্রিং প্রয়োজন। ক্লাসের মধ্যে ফাংশন যোগ করার জন্য উদাহরণ দেখুন।
\$ অগ্রাধিকার	(int) ডিফল্ট = 10. হুক/ফিল্টারের সাথে সংযুক্ত ফাংশনগুলি নির্ধারিত অগ্রাধিকারে চলবে। আপনার এমন পরিস্থিতি হতে পারে যেখানে আপনি অন্য কোনো অ্যাকশনের আগে কোডের সাথে কাজ করতে চান, অগ্রাধিকার =1 সেট করুন বা অন্যান্য সমস্ত সংযুক্ত ফাংশন অগ্রাধিকার = 100 ইত্যাদির পরে। সমস্ত পিএইচপি ফাংশনের মতো, আপনি একটি ডেরিয়েবলের জন্য একটি মান পাস না করে ফাংশনটি ব্যবহার করতে পারেন। যেখানে একটি ডিফল্ট মান সেট করা হয়েছে, কিন্তু আপনি যদি প্রত্যাবর্তিত পরামিতিগুলির সংখ্যা পরিবর্তন করতে চান তবে আপনাকে অবশ্যই নির্দিষ্ট করতে হবে!
\$প্যারামিটার	(int) ডিফল্ট = 1. আপনার সংযুক্ত ফাংশনে ফিরে আসা প্যারামিটারের সংখ্যা। ফিরে আসা পরামিতিগুলি সংযুক্ত নম্বরের উপর নির্ভর করবে যেখানে হুক তৈরি করা হয়েছিল। আরো বিস্তারিত জানার জন্য <code>apply_filters()</code> এবং <code>do_action()</code> দেখুন।

### মন্তব্য

#### ওয়ার্ডপ্রেস হুকস

ওয়ার্ডপ্রেসের সাথে কাজ শুরু করার সময় বিকাশকারীদেরকে বিভ্রান্ত করে এমন কিছু যা হল `apply_filters()` এবং `add_action()` এর ব্যবহার। আপনি প্রায়শই কোডে প্লাগইন/থিমগুলিকে এগুলি ব্যবহার করতে দেখতে পাবেন এবং আপনি যদি ধারণাটি বুঝতে না পারেন তবে তাদের সাথে কাজ করা আপনার পক্ষে কঠিন হবে।

সংক্ষেপে (খুব সংক্ষেপে, বিস্তারিতভাবে প্রক্রিয়ার জন্য ওয়ার্ডপ্রেস লোড ফ্লোচার্ট দেখুন), ওয়ার্ডপ্রেস নিম্নলিখিত উপায়ে লোড হয়:

1. `wp-load.php` - ফাংশন ইত্যাদি 2.

`mu-plugins` - `mu-plugins` ফোল্ডারে পাওয়া যেকোন ফাইল - প্রায়শই ক্যাশে করা বস্তু পরিবেশন করতে ব্যবহৃত হয় 3. প্লাগইন - কোন নির্দিষ্ট ক্রম নয়, কোন ইনস্টল করা এবং সক্রিয় প্লাগইনগুলি লোড করা হবে 4. সক্রিয় শিশু থিম / পিতামাতার থিম 5. `init` - বাকি ডেটা

6. টেমপ্লেট

আপনি যদি একজন বিকাশকারী হন এবং একটি ফাংশন ফাইলের সাথে কাজ করেন তবে আপনি দেখতে পাবেন যে দুটি ফাইলের সাথে আপনি কাজ করছেন তার চেয়ে প্রক্রিয়াজাত আগে লোড হয়েছে। এর অর্থ আপনি প্রক্রিয়াগুলি পরিবর্তন করতে পারবেন না (মনে রাখবেন যে আপনি ফাংশনগুলি ওভাররাইট করতে পারবেন না) বা ভেরিয়েবলগুলি যা পরে চলবে বা এখনও সংজ্ঞায়িত করা হয়নি। এছাড়াও থিম বিকাশকারীরা তাদের কোডে হুক স্থাপন করতে পারে যাতে প্লাগইনগুলিকে হুক করার অনুমতি দেওয়া হয় বা প্লাগইনগুলি অন্যান্য প্লাগইনগুলিকে তাদের ভেরিয়েবলগুলিকে ওভাররাইট করার অনুমতি দিতে পারে। এখন এই পর্যন্ত বিভ্রান্তিকর হতে পারে, কিন্তু সেখানে স্তর।

add\_filter() এবং add\_action() বোঝার জন্য আমাদের প্রথমে দেখতে হবে কিভাবে হুক তৈরি করা হয়।

```
$arga = 'হ্যালো';  
do_action('im_a_hook', $arga);
```

আপনি যখন ওয়ার্ডপ্রেসে উপরেরটির মুখোমুখি হন, তখন এটি হকের সাথে সংযুক্ত যেকোন ফাংশনকে কল করবে im\_a\_hook ( প্রক্রিয়া সম্পর্কে তথ্যের জন্য \$wp\_filter দেখুন )। আপনার সংযুক্ত ফাংশনে \$arga সংযুক্ত ফাংশনের সাথে কাজ করার জন্য উপলব্ধ হবে।

```
add_action('im_a_hook', 'সংযুক্ত_ফাংশন');  
  
ফাংশন সংযুক্ত_ফাংশন($আর্গা){ ইকো $আর্গা;  
  
}
```

এটি লোড প্রক্রিয়ার নির্দিষ্ট পয়েন্টগুলিতে ভেরিয়েবলগুলি সংশোধন করার শক্তিশালী নতুন সুযোগগুলি উন্মুক্ত করে।

মনে আছে আমরা আগে বলেছিলাম যে টেমপ্লেটগুলি প্লাগইন/থিমের পরে লোড হয়? একটি সাধারণ প্লাগইন হল WooCommerce যা প্রক্রিয়ার পরে স্ক্রিন তৈরি করে, আমি কীভাবে ডকুমেন্ট করতে যাচ্ছি না কিন্তু প্লাগইনে do\_action এর একটি উদাহরণ পাওয়া যাবে।

```
do_action('woocommerce_after_add_to_cart_button');
```

এখানে আমাদের একটি হুক তৈরি করা হয়েছে যা কোনও ভেরিয়েবলকে পাস করে না, তবে আমরা এখনও এটির সাথে মজা করতে পারি:

```
add_action('woocommerce_after_add_to_cart_button', 'special_offer');  
  
ফাংশন special_offer(){ echo '<h1>বিশেষ অফার!  
</h1>';  
  
}
```

উপরের অ্যাড\_অ্যাকশনটি বিশেষ অফারের একটি শিরোনাম প্রতিধ্বনিত করবে যেখানে

do\_action('woocommerce\_after\_add\_to\_cart\_button') অবস্থিত যা একটি WooCommerce স্ক্রিন তৈরি করার সময়। তাই আমরা html সন্নিবেশ করতে এই হুক ব্যবহার করতে পারেন। অন্যান্য ব্যবহারের মধ্যে সম্পূর্ণ আলাদা স্ক্রিনে পুনঃনির্দেশ করা অন্তর্ভুক্ত থাকতে পারে, ইত্যাদি।

এছাড়াও একাধিক ভেরিয়েবল ফাংশনে পাস করা যেতে পারে। আপনার থিম ফাংশন এটি চেষ্টা করুন। আমরা যে শেষ প্যারামিটারটি 3 সেট করছি তা নোট করুন, কারণ আমরা 3টি উপলব্ধ প্যারামিটারের সাথে কাজ করতে চাই। যদি আমরা এটিকে 2 তে পরিবর্তন করি, শুধুমাত্র 2টি ফেরত দেওয়া হবে এবং আমরা একটি অনির্ধারিত ত্রুটি পাব।

```
add_action('কাস্টম_হুক', 'সংযুক্ত_ফাংশন', 10, 3);  
  
ফাংশন সংযুক্ত_ফাংশন($a,$b,$c){
```

```

var_dump($a);
var_dump($b);
var_dump($c);

}

$args = 1;
$argsb = 2;
$argsc = 3;

do_action('custom_hook', $args, $argsb, $argsc); প্রস্থান

```

ফিল্টার নামে আরেকটি WP হুক টাইপ আছে। একটি ফিল্টার তার ব্যবহারের ক্ষেত্রে একটি অ্যাকশন থেকে আলাদা, একটি অ্যাকশন শুধুমাত্র ভেরিয়েবল গ্রহণ করতে পারে, স্পষ্টতই এই ভেরিয়েবলগুলি ফাংশন স্কোপের মধ্যে রয়েছে (আপনার জানা উচিত পিএইচপি স্কোপ কী, যদি গুগল না হয়)। ফিল্টারগুলি ফিরে আসা ডেটা পাস করে, তাই আপনি ভেরিয়েবলগুলি পরিবর্তন করতে ব্যবহার করতে পারেন।

```
$filter_me = apply_filters('im_a_filter', $variable_to_filter);
```

যেখানে আপনি উপরেরটি দেখতে পাচ্ছেন, আপনি \$filter\_me- এর মান পরিবর্তন করতে পারেন কারণ আপনি যে ডেটা ফেরত দেবেন তা ভেরিয়েবলে সংরক্ষিত মান হবে। সুতরাং উদাহরণস্বরূপ (মনে রাখবেন আমরা উদাহরণে \$variable\_to\_filter পরিবর্তন করে \$filter\_me করছি):

```

add_filter('im_a_filter', 'সংযুক্ত_ফাংশন', 100);

ফাংশন সংযুক্ত_ফাংশন($filter_me){

    $filter_me = 'রে';

    $filter_me ফেরত দিন;

}

$filter_me = 'বব'; $filter_me =
apply_filters('im_a_filter', $filter_me);

```

\$ filter\_me ভেরিয়েবলে এখন 'বব' এর পরিবর্তে 'রে' থাকবে, আমরা 100 এর অগ্রাধিকার নির্ধারণ করেছি তাই আমরা যুক্তিসঙ্গতভাবে আত্মবিশ্বাসী যে কেউ ব্যবহারের পরে মান পরিবর্তন করবে না (একই হুকে একাধিক ফিল্টার চলতে পারে) তাই আমরা যদি apply\_filters() উপস্থিত থাকে তবে প্রক্রিয়ায় পরে ব্যবহৃত ভেরিয়েবলগুলি এখন পরিবর্তন করতে পারে।

আপনি একাধিক পরামিতি পাস করতে পারেন, কিন্তু আপনি শুধুমাত্র একটি মান পরিবর্তন করতে পারেন। আপনাকে অবশ্যই একটি মান ফেরত দিতে হবে, অন্যথায় আপনার ভেরিয়েবলে কিছুই থাকবে না। ভেরিয়েবলের মান/অ্যারে/বস্তু বরাদ্দ করতে আপনি কীভাবে php ব্যবহার করেন তা আপনি বুঝতে পারলে এটি আপনার কাছে স্পষ্ট হবে, যেমন:

```

add_filter('im_a_filter', 'সংযুক্ত_ফাংশন', 100, 3);

ফাংশন সংযুক্ত_ফাংশন($filter_me, $args, $argsb){

    $filter_me = 'রে'।$args.$argsb;

```

```

    $arga = 'তুমি বোকা';

    $filter_me ফেরত দিন;
}

$filter_me = 'বব';

$arga = 'মধ্য নাম';
$argb = 'পদবি';

$filter_me = apply_filters('im_a_filter', $filter_me, $arga, $argb);

```

\$filter\_me ভেরিয়েবলে এখন 'রে মিডলনেম সার্ভে' রয়েছে। কিন্তু \$ arga সম্পর্কে কি? এটিতে এখনও 'মিডলনেম' রয়েছে, আমাদের ফাংশনের মধ্যে একটি \$আর্গাকে 'তুমি বোকা' তে পরিবর্তন করলে এর সুযোগের বাইরে সংজ্ঞায়িত মানের উপর কোন প্রভাব নেই (উপায় আছে, গুগল গ্লোবাল ইত্যাদি)

```
add_action($hook_name, $function, $priority, $parameters)
```

```
add_filter($hook_name, $function, $priority, $parameters);
```

## উদাহরণ

### add\_action - init

```

add_action('init', 'process_post');

ফাংশন process_post(){ if($_POST)
    var_dump($_POST);
}

```

### add\_action - init - বেনামী ফাংশন

```

add_action('init' echo 'আমি      , ফাংশন(){
    কিছু করেছি';
});

```

### add\_action - init - ক্লাস অবজেক্টের মধ্যে

```

ক্লাস নমুনা{

    পাবলিক ফাংশন __construct(){ add_action('init',
        array($this, 'samp') );
    }

    পাবলিক ফাংশন স্যাম্প(){ // অবশ্যই সর্বজনীন হতে হবে!! প্রতিধ্বনি 'আমি কিছু করেছি';
    }
}

নতুন নমুনা();

```

## add\_action - init - স্ট্যাটিক ক্লাসের মধ্যে

```
ক্লাস নমুনা{  
  
    পাবলিক স্ট্যাটিক ফাংশন add_action_func ()  
  
    }  
  
    পাবলিক স্ট্যাটিক ফাংশন samp(){ echo 'আমি কিছু করেছি';  
  
    }  
  
}  
  
নমুনা::add_action_func();
```

## অধ্যায় 3: শর্টকোড যোগ করুন

### বাক্য গঠন

- `add_shortcode( $tag , $func);`

### পরামিতি

পরামিতি বিবরণ	
\$ট্যাগ	(স্ট্রিং) (প্রয়োজনীয়) শর্টকোড ট্যাগ পোস্ট সামগ্রীতে অনুসন্ধান করতে হবে
\$func	(কলযোগ্য) (প্রয়োজনীয়) শর্টকোড পাওয়া গেলে চালানোর জন্য হুক

### মন্তব্য

- শর্টকোড কলব্যাক তিনটি আর্গুমেন্ট পাস করা হবে: শর্টকোড বৈশিষ্ট্য, শর্টকোড বিষয়বস্তু (যদি থাকে), এবং শর্টকোডের নাম।
- প্রতিটি শর্টকোডের জন্য শুধুমাত্র একটি হুক থাকতে পারে। যার মানে হল যে যদি অন্য একটি প্লাগইনের অনুরূপ শর্টকোড থাকে তবে এটি আপনারটিকে ওভাররাইড করবে বা আপনারটি তাদের ওভাররাইড করবে প্লাগইনগুলি অন্তর্ভুক্ত এবং/অথবা চালানো হয়েছে তার উপর নির্ভর করে।
- শর্টকোড অ্যাট্রিবিউটের নামগুলি হ্যান্ডলার ফাংশনে পাস করার আগে সর্বদা ছোট হাতের অক্ষরে রূপান্তরিত হয়। মূল্যবোধ অস্পৃশ্য।
- মনে রাখবেন যে শর্টকোড দ্বারা কল করা ফাংশনটি কখনই কোনও ধরণের আউটপুট তৈরি করবে না। শর্টকোড ফাংশন টেমপ্লেট ফেরত দিতে হবে যা শর্টকোড প্রতিস্থাপন করতে ব্যবহার করা হবে। সরাসরি আউটপুট উত্পাদন অপ্রত্যাশিত ফলাফল হতে হবে। এটি ফিল্টার ফাংশনগুলির আচরণের অনুরূপ, যাতে তারা কল থেকে প্রত্যাশিত পার্স প্রতিক্রিয়া তৈরি করবে না, যেহেতু কখন এবং কোথা থেকে কল করা হবে তা আপনি নিয়ন্ত্রণ করতে পারবেন না।

### উদাহরণ

#### সাম্প্রতিক পোস্টের জন্য সহজ শর্টকোড

`add_shortcode` হল `wp` কীওয়ার্ড।

```
// সাম্প্রতিক পোস্টগুলি আমাদের শর্টকোড হতে চলেছে। add_shortcode('সাম্প্রতিক-
পোস্ট', 'সাম্প্রতিক_পোস্ট_ফাংশন');

// সাম্প্রতিক পোস্ট শর্টকোড কল করা হলে এই ফাংশন পদক্ষেপ নিচ্ছে। ফাংশন সাম্প্রতিক_পোস্ট_ফাংশন() { query_posts(array('orderby'
=> 'date', 'order' => 'DESC' যদি (have_posts()):
, 'showposts' => 1));

while (have_posts()): the_post();
$return_string = '<a href="'.get_permalink().">'.get_the_title().</a>';
endwhile;
```

```
যদি শেষ;
wp_reset_query(); ফেরত
$return_string;
}
```

এই স্নিপেটটি আপনার থিম functions.php এ স্থাপন করা যেতে পারে।

[সাম্প্রতিক পোস্ট] এটি সাম্প্রতিক পোস্টের জন্য শর্টকোড। আমরা এই শর্টকোডটি ব্যাকএন্ডে প্রয়োগ করতে পারি (যেমন পৃষ্ঠা, পোস্ট, উইজেট)।

আমরা আমাদের কোডের ভিতরে একই শর্টকোড ব্যবহার করতে পারি। do\_shortcode এর সাহায্যে।

যেমন echo do\_shortcode(['সাম্প্রতিক পোস্ট']);

## সাম্প্রতিক পোস্টের জন্য উন্নত শর্টকোড

আপনি কতগুলি সাম্প্রতিক পোস্ট প্রদর্শন করতে চান তার জন্য এই ফাংশনটি প্যারামিটার নেয়।

যেমন: আপনি শুধুমাত্র পাঁচটি সাম্প্রতিক পোস্ট প্রদর্শন করতে চান। শুধু posts="5" দিয়ে আর্গুমেন্ট পাস করেছেন (আপনি প্রদর্শন করতে চান এমন যে কোনো সাম্প্রতিক পোস্ট পাস করতে পারেন)।

ফাংশন ডাটাবেস থেকে মাত্র পাঁচটি সাম্প্রতিক পোস্ট নিয়ে আসে।

```
// সাম্প্রতিক পোস্টগুলি আমাদের শর্টকোড হতে চলেছে। add_shortcode('সাম্প্রতিক-
পোস্ট', 'সাম্প্রতিক_পোস্ট_ফাংশন');

// ফাংশন প্যারামিটার নেয় যেমন posts="5"। ফাংশন সাম্প্রতিক_পোস্ট_ফাংশন($atts){

    extract(shortcode_atts(array(
        'পোস্ট' => 1,
    ), $atts));

    $return_string = '<ul>';
    query_posts(array('orderby' => 'date', 'order' => 'DESC' যদি (have_posts());
        , 'showposts' => $পোস্ট));

    while (have_posts()): the_post();
        $return_string .= '<li><a href="'.get_permalink().">'.get_the_title().</a></li>';
    endwhile; যদি
    শেষ;
    $return_string .= '</ul>';

    wp_reset_query(); ফেরত
    $return_string;
}
```

যেমন echo do\_shortcode(['সাম্প্রতিক পোস্টের পোস্ট="5"]');

## অধ্যায় 4: user\_contactmethods ফিল্টার হুক সহ ব্যবহারকারীদের জন্য যোগাযোগের তথ্য যোগ করুন/সরান

### উদাহরণ

সর্বাধিক জনপ্রিয় সামাজিক নেটওয়ার্ক সক্রিয় করা হচ্ছে।

```
ফাংশন social_profiles( $contactmethods )  
{  
  
    $contactmethods['facebook_profile'] = 'ফেসবুক প্রোফাইল URL'; $contactmethods['twitter_profile'] = 'টুইটার প্রোফাইল  
URL'; $contactmethods['google_profile'] $contactmethods['linkedin_profile'] = 'লিংকডিন প্রোফাইল URL';  
    $contactmethods['github_profile']  
        = 'গুগল প্রোফাইল URL';  
    $contactmethods['behance_profile'] = 'Behance প্রোফাইল URL'; $contactmethods['dribbble_profile'] = 'ড্রিবল প্রোফাইল  
URL'; $contactmethods['stack_profile']  
        = 'GitHub প্রোফাইল URL';  
    $contactmethods['twitch_profile'] $contactmethods['angellist_profile'] = 'AngelList প্রোফাইল URL';  
  
        = 'স্ট্যাক এক্সচেঞ্জ প্রোফাইল URL';  
        = 'টুইচ প্রোফাইল URL';  
  
    $contactmethods ফেরত দিন;  
}  
  
add_filter( 'user_contact methods', 'social_profiles', 10, 1);
```

আপনি আপনার ড্যাশবোর্ডে এই ফাইলগুলি পাবেন:

The screenshot shows the WordPress admin dashboard. On the left is a dark sidebar menu with the following items: Dashboard, Posts, Media, Pages, Comments, Cheat sheets, Tips & tricks, Appearance, Plugins (with a red notification bubble containing the number 2), Users (highlighted in blue), All Users, Add New, Your Profile, Tools, Settings, Custom Fields, and MailChimp for WP. The main content area is light gray and contains a list of profile URL settings, each with a corresponding empty text input field:

- Facebook Profile URL
- Twitter Profile URL
- Google Profile URL
- Linkedin Profile URL
- GitHub Profile URL
- Behance Profile URL
- Dribbble Profile URL
- Stack Exchange Profile URL
- Twitch Profile URL
- AngelList Profile URL

এবং এইভাবে আপনি কোডে এটি পুনরুদ্ধার করবেন

```
<?php $user_stack_exchange = get_the_author_meta( 'stack_profile' ); ?>
```

যোগাযোগের পদ্ধতি সরানো হচ্ছে

```
ফাংশন remove_contact_methods( $contactmethods ) {
    unset($contactmethods['facebook_profile']);
    unset($contactmethods['twitter_profile']);

    $contactmethods ফেরত দিন;
}

add_filter( 'user_contactmethods', 'remove_contact_methods', 10, 1);
```

# অধ্যায় 5: add\_action()

## বাক্য গঠন

- `add_action( $tag, $function_to_add )` •
- `add_action( $tag, $function_to_add, $priority )` •
- `add_action( $tag, $function_to_add, $priority, $accepted_args )`

## পরামিতি

প্যারামিটার	বিস্তারিত
<code>\$ট্যাগ</code>	(স্ট্রিং) কর্মের নাম যেখানে পদ্ধতিটি <code>\$function_to_add</code> হুক করা হবে।
<code>\$function_to_add</code> (কলযোগ্য)	কলযোগ্য ফাংশন/প্রক্রিয়া যা আপনি কল করতে চান।
<code>\$ অগ্রাধিকার</code>	(int) অগ্রাধিকার স্তর যেখানে <code>\$function_to_add</code> কার্যকর করা হবে। ঐচ্ছিক। ডিফল্ট 10।
<code>\$accepted_args</code>	(int) আর্গুমেন্টের সংখ্যা যা কলযোগ্য ফাংশন <code>\$function_to_add</code> গ্রহণ করে। ঐচ্ছিক। ডিফল্ট 1.

## উদাহরণ

### সরাসরি ফাংশন কলব্যাক

```
add_action( 'init', function() { // এখানে কিছু করুন  
  
});
```

নির্দেশাবলীর একটি সেট হুক করার জন্য একটি ফাংশন ব্লক ব্যবহার করে। `init` হকের সাহায্যে , ওয়ার্ডপ্রেস প্রয়োজনীয় উপাদানগুলি লোড করা শেষ করার পরেই নির্দেশাবলীর সেটটি কার্যকর করা হবে।

### ফাংশন নামের রেফারেন্স কলব্যাক

```
function my_init_function() { // এখানে কিছু করুন  
  
}  
  
add_action( 'init', 'my_init_function');
```

নির্দেশের একটি সেট হুক করতে ফাংশনের নাম ব্যবহার করে। `init` হকের সাহায্যে , ওয়ার্ডপ্রেস প্রয়োজনীয় উপাদানগুলি লোড করা শেষ করার পরেই নির্দেশাবলীর সেটটি কার্যকর করা হবে।

## ক্লাস স্ট্যাটিক পদ্ধতি কলব্যাক

```
ক্লাস MyClass { static
    function my_init_method() { // এখানে কিছু করুন
    }
}

add_action( 'init', অ্যারে ('MyClass', 'my_init_method') );
```

নির্দেশাবলীর একটি সেট হক করার জন্য একটি ক্লাসের একটি স্ট্যাটিক পদ্ধতি ব্যবহার করে। init হকের সাহায্যে , ওয়ার্ডপ্রেস প্রয়োজনীয় উপাদানগুলি লোড করা শেষ করার পরেই নির্দেশাবলীর সেটটি কার্যকর করা হবে।

## অবজেক্ট পদ্ধতি কলব্যাক

```
ক্লাস মাইক্লাস { ফাংশন
    my_init_method() {
        // এখানে কিছু করুন
    }
}

$obj = নতুন মাইক্লাস();

add_action( 'init', অ্যারে( $obj, 'my_init_method' ) );
```

নির্দেশাবলীর একটি সেট হক করার জন্য একটি বস্তুর একটি পদ্ধতি ব্যবহার করা। init হকের সাহায্যে , ওয়ার্ডপ্রেস প্রয়োজনীয় উপাদানগুলি লোড করা শেষ করার পরেই নির্দেশাবলীর সেটটি কার্যকর করা হবে।

## অধ্যায় 6: add\_editor\_style()

ভূমিকা

ফাংশনটি ব্যবহারকারীকে TinyMCE সম্পাদকের জন্য স্টাইলশীট লোড করতে দেয়

বাক্য গঠন

- add\_editor\_style( \$stylesheet )

পরামিতি

পরামিতি বিবরণ	
\$ স্টাইলশীট	(অ্যাংগে বা স্ট্রিং) (ঐচ্ছিক) স্টাইলশীটের নাম বা এর অ্যাংগে, থিম রুটের সাথে সম্পর্কিত। 'editor-style.css'-এ ডিফল্ট

উদাহরণ

একটি একক CSS ফাইল লোড হচ্ছে

কোড

```
ফাংশন add_new_style() {  
    add_editor_style('file-name-here.css');  
  
} add_action( 'admin_init', 'add_new_style');
```

ব্যাখ্যা

উপরের কোডে, আমরা css ফাইল লোড করার জন্য add\_editor\_style ব্যবহার করেছি। ওয়ার্ডপ্রেস আমাদের ফাংশন চালায় তা নিশ্চিত করতে আমরা add\_action ব্যবহার করেছি।

## অধ্যায় 7: add\_menu\_page()

### ভূমিকা

এই ফাংশনটি হল অ্যাডমিন প্যানেলের নেভি বারে একটি আইটেম যুক্ত করা।

### বাক্য গঠন

- `add_menu_page( $page_title, $menu_title, $capability, $menu_slug, $function, $icon_url, $position)`

### পরামিতি

পরামিতি	বিবরণ
<code>\$page_title</code>	(স্ট্রিং) মেনু নির্বাচন করা হলে পৃষ্ঠার শিরোনাম ট্যাগে প্রদর্শিত পাঠ্য।
<code>\$menu_title</code>	(স্ট্রিং) মেনুর জন্য ব্যবহৃত পাঠ্য।
<code>\$ সক্ষমতা</code>	(স্ট্রিং) ব্যবহারকারীর কাছে এই মেনু প্রদর্শনের জন্য প্রয়োজনীয় ক্ষমতা।
<code>\$menu_slug</code>	(স্ট্রিং) এই মেনুটি উল্লেখ করার জন্য স্লগ নাম (এই মেনুর জন্য অনন্য হওয়া উচিত)।
<code>\$ফাংশন</code>	(কলযোগ্য) (ঐচ্ছিক) এই পৃষ্ঠার বিষয়বস্তু আউটপুট করার জন্য যে ফাংশনটি কল করা হবে।
<code>\$icon_url</code>	(স্ট্রিং) (ঐচ্ছিক) এই মেনুর জন্য ব্যবহার করা আইকনের URL।
<code>\$পজিশন</code>	(int) (ঐচ্ছিক) মেনু ক্রমানুসারে অবস্থান এই একটি প্রদর্শিত হবে।

### মন্তব্য

এখানে ডিফল্ট অবস্থানের একটি তালিকা রয়েছে (\$ অবস্থানের জন্য)

- 2 - ড্যাশবোর্ড • 4 - বিভাজক • 5 - পোস্ট
- 10 - মিডিয়া • 15 - লিঙ্ক • 20 - পৃষ্ঠাগুলি • 25 - মন্তব্য • 59 - বিভাজক

- 60 – চেহারা • 65 –  
প্লাগইনস • 70 –  
ব্যবহারকারী
- 75 – টুলস
- 80 – সেটিংস • 99  
– বিভাজক

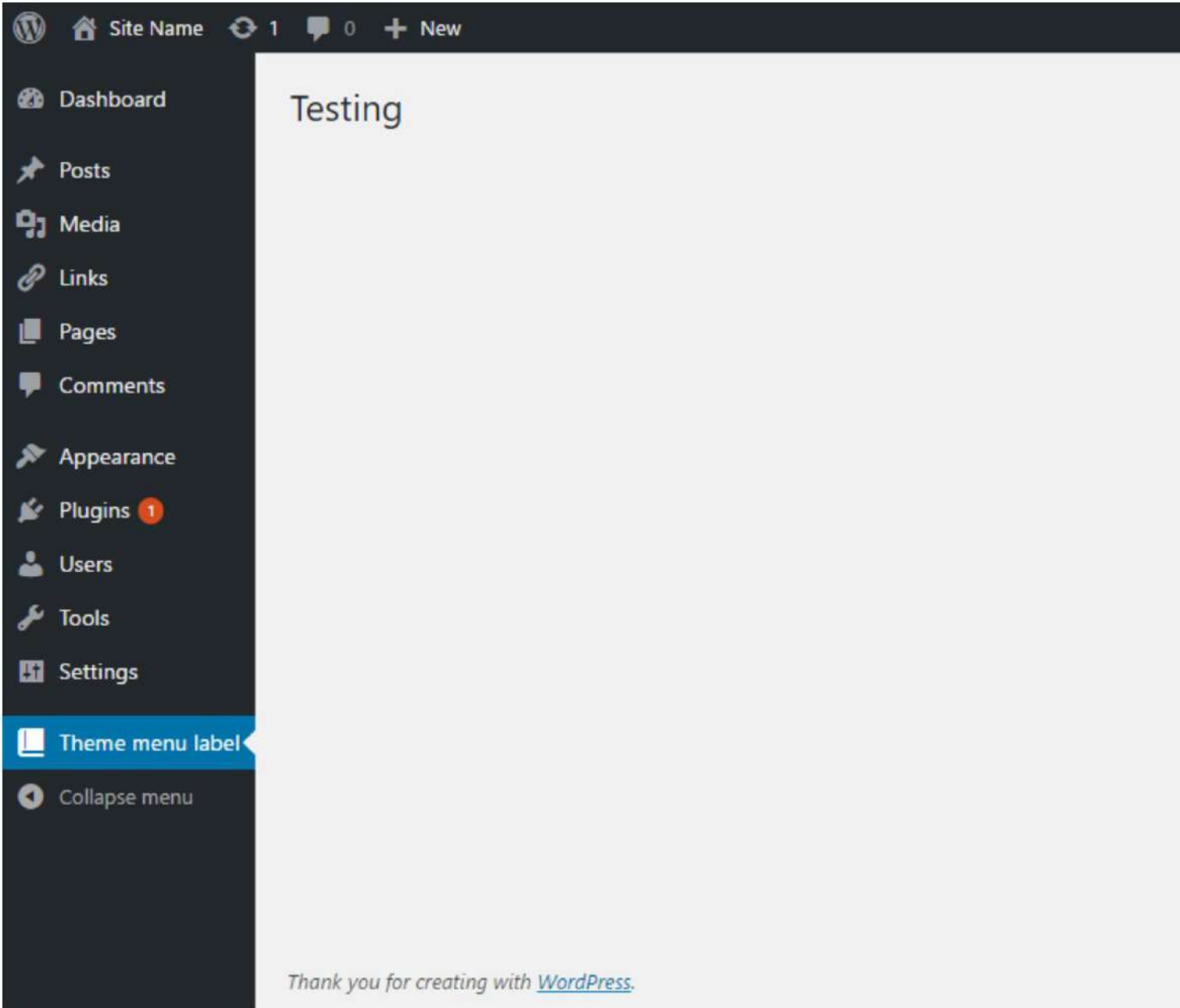
## উদাহরণ

নেভি বারে "থিম পৃষ্ঠার শিরোনাম" আইটেম যোগ করা হচ্ছে

কোড

```
ফাংশন add_the_theme_page(){  
    add_menu_page('থিম পৃষ্ঠার শিরোনাম', 'থিম মেনু লেবেল', 'ম্যানেজ_অপশন', 'থিম-বিকল্প',  
'পৃষ্ঠা_সামগ্রী', 'ড্যাশিকন-বুক-আল্ট'); } add_action('admin_menu',  
'add_the_theme_page'); ফাংশন page_content(){  
  
    echo '<div class="wrap"><h2>পরীক্ষা</h2></div>';  
}
```

আউটপুট



## ব্যখ্যা

কোডে, আমরা `add_the_theme_page` নামে একটি ফাংশন তৈরি করেছি এবং ন্যাভিগারে আইটেম যোগ করতে `add_menu_page` ব্যবহার করেছি। অনুগ্রহ করে এই পৃষ্ঠার প্যারামিটার অংশটি পরীক্ষা করে দেখুন আমরা যে আর্গুমেন্টে পাস করেছি সে সম্পর্কে জানতে। তারপরে আমরা আমাদের `add_the_theme_page` ফাংশন চালানোর জন্য `add_action` ব্যবহার করেছি। অবশেষে, আমরা পেজের বিষয়বস্তু প্রদর্শনের জন্য `page_content` ফাংশন তৈরি করেছি।

OOP এবং কিভাবে মেনু পৃষ্ঠায় স্ক্রিপ্ট/স্টাইল লোড করতে হয়

```
<?php /*  
  
 * প্লাগইন নাম: কাস্টম অ্যাডমিন মেনু */  
  
ক্লাস SO_WP_মেনু {  
  
    ব্যক্তিগত $plugin_url;
```

```

পাবলিক ফাংশন __construct() {
    $this->plugin_url = plugins_url('/', __FILE__); add_action( 'plugins_loaded', array( $this,
        'init' ));
}

পাবলিক ফাংশন init()
{ add_action( 'admin_menu', array( $this, 'add_menu' ));
}

পাবলিক ফাংশন add_menu() { $hook =
    add_menu_page(
        'আমার মেনু', // শিরোনাম, html মেটা ট্যাগ
        '<span style="color:#e57300;">আমার মেনু', // মেনু শিরোনাম, হার্ডকোড স্টাইল 'edit_pages', // ক্ষমতা

        'dummy-page-slug', // URL
        array( $this, 'content' ), // output // icon, ডিফল্ট // অবস্থান ব্যবহার
        খালি, // করে, অন্য সকলের উপরে দেখাচ্ছে
        1
    );
    add_action( "admin_print_scripts-$hook", array( $this, 'scripts' )); add_action( "admin_print_styles-$hook", array( $this,
        'styles' ));
}

পাবলিক ফাংশন বিষয়বস্তু() {
    ?>
    <div id="icon-post" class="icon32"></div> <h2>ডামি পৃষ্ঠা</h2> <p> Lorem
        ipsum</p>

    <?php
}

# সরাসরি মুদ্রণ, wp_enqueue_script পাবলিক ফাংশন স্ক্রিপ্ট() {

    ?><script>সতর্কতা('আমার পৃষ্ঠা');</script><?php
}

# প্লাগইন ডিরেক্টরি পাবলিক ফাংশন শৈলীতে একটি CSS ফাইল থেকে এনকুইং () {

    wp_enqueue_style( 'my-menu', $this->plugin_url . 'my-menu.css' );
}
}

নতুন SO_WP_Menu();

```

এই উদাহরণে যে বিষয়টি লক্ষ্য করা গুরুত্বপূর্ণ তা হল, `add_menu_page()` ব্যবহার করার সময় এটি একটি হুক প্রদান করে যা আমাদের সঠিক পৃষ্ঠাকে লক্ষ্য করতে এবং সেখানে শৈলী এবং স্ক্রিপ্ট লোড করতে ব্যবহার করা যেতে পারে।

একটি সাধারণ ভুল হল টার্গেট না করে সারিবদ্ধ করা এবং এটি স্ক্রিপ্ট এবং শৈলীকে `/wp` জুড়ে ছড়িয়ে দেয় অ্যাডমিন.

OOP ব্যবহার করে আমরা অভ্যন্তরীণ পদ্ধতির মধ্যে ব্যবহার করা সাধারণ ভেরিয়েবল সংরক্ষণ করতে পারি।

## অধ্যায় 8: add\_submenu\_page()

ভূমিকা

এই ফাংশনটি হল অ্যাডমিন প্যানেল নেভি বারে বিদ্যমান আইটেমে একটি উপ-আইটেম যোগ করা।

বাক্য গঠন

- `add_submenu_page( $parent_slug, $page_title, $menu_title, $capability, $menu_slug, $function )`

পরামিতি

পরামিতি বিবরণ	
<code>\$parent_slug</code>	(স্ট্রিং) প্যারেন্ট মেনুর জন্য স্লাগ নাম (বা একটি স্ট্যান্ডার্ড ওয়ার্ডপ্রেস অ্যাডমিন পৃষ্ঠার ফাইলের নাম)।
<code>\$page_title</code>	(স্ট্রিং) মেনু নির্বাচন করা হলে পৃষ্ঠার শিরোনাম ট্যাগে প্রদর্শিত পাঠ্য।
<code>\$menu_title</code>	(স্ট্রিং) মেনুর জন্য ব্যবহৃত পাঠ্য।
<code>\$ সক্ষমতা</code>	(স্ট্রিং) ব্যবহারকারীর কাছে এই মেনু প্রদর্শনের জন্য প্রয়োজনীয় ক্ষমতা।
<code>\$menu_slug</code>	(স্ট্রিং) এই মেনুটি উল্লেখ করার জন্য স্লাগ নাম (এই মেনুর জন্য অনন্য হওয়া উচিত)।
<code>\$ফাংশন</code>	(কলযোগ্য) (ঐচ্ছিক) এই পৃষ্ঠার বিষয়বস্তু আউটপুট করার জন্য যে ফাংশনটি কল করা হবে।

মন্তব্য

এখানে `$parent_slug`-এর জন্য স্লাগগুলির একটি তালিকা রয়েছে।

- ড্যাশবোর্ড: 'index.php' • পোস্ট:
- 'edit.php' • মিডিয়া:
- 'upload.php' • পৃষ্ঠা:
- 'edit.php?post\_type=page' • মন্তব্য: 'edit-
- comments.php' • কাস্টম পোস্টের ধরন :
- 'edit.php?post\_type=your\_post\_type' • চেহারা: 'themes.php' • প্লাগইন:
- 'plugins.php' • ব্যবহারকারী:
- 'users.php'

- টুল: 'tools.php' •
- সেটিংস: 'options-general.php' • নেটওয়ার্ক
- সেটিংস: 'settings.php'

## উদাহরণ

এনএভি বারে "সাবমেনু পৃষ্ঠা" "টুলস" এর একটি উপ-পৃষ্ঠা হিসাবে যুক্ত করা হচ্ছে

কোড

```
add_action('admin_menu', 'register_my_custom_submenu_page');

ফাংশন register_my_custom_submenu_page() {
    add_submenu_page( 'tools.php',
        'সাবমেনু পৃষ্ঠা', 'আমার
        কাস্টম সাবমেনু পৃষ্ঠা',
        'ম্যানেজ_অপশন', 'মাই-কাস্টম-সাবমেনু-পৃষ্ঠা',

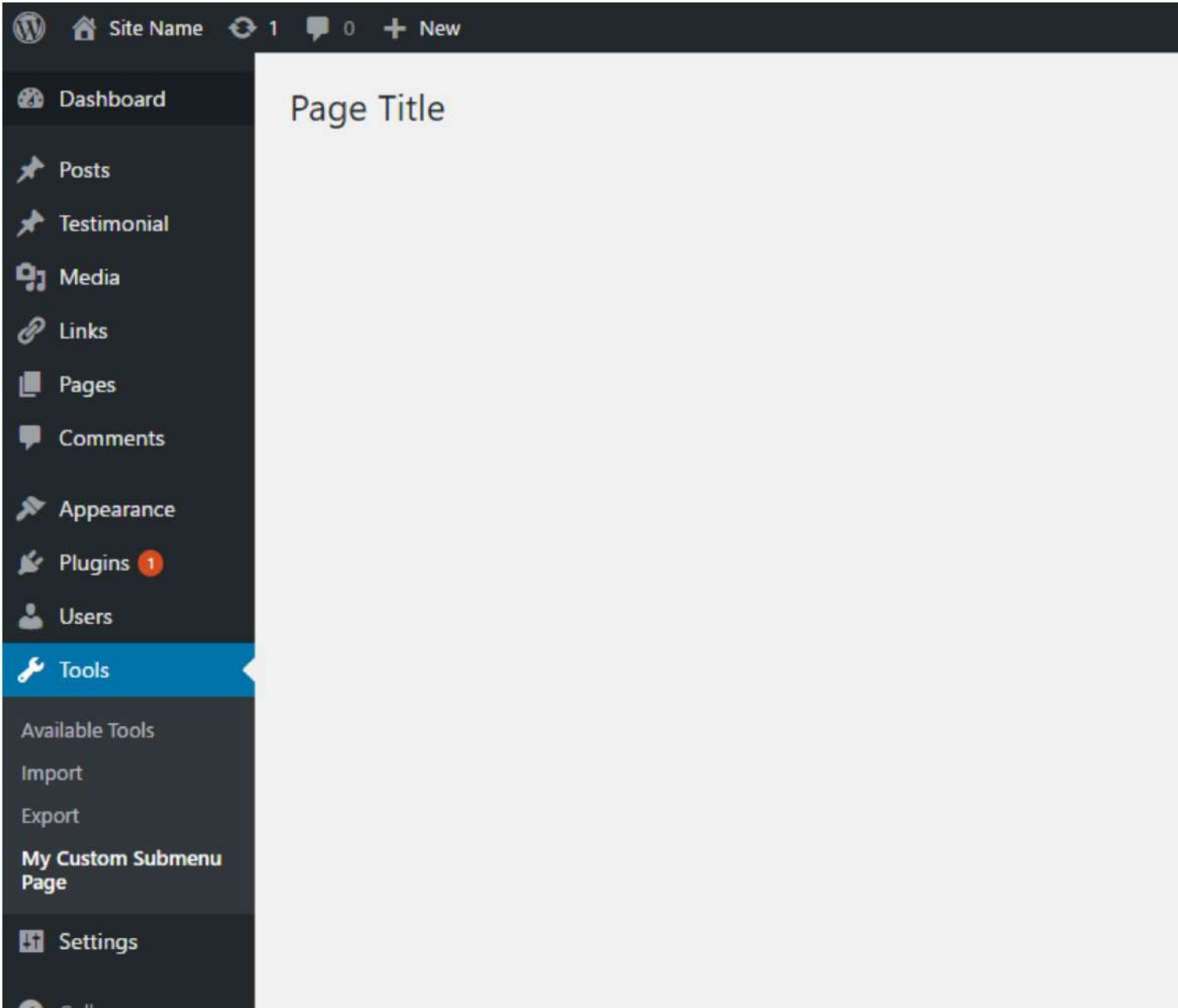
        'my_custom_submenu_page_content');
}

ফাংশন my_custom_submenu_page_content() { echo '<div class="wrap">';

    echo '<h2>পৃষ্ঠার শিরোনাম</h2>'; প্রতিধ্বনি '</div>';

}
```

আউটপুট



ব্যাখ্যা

কোডে, আমরা `register_my_custom_submenu_page` নামে একটি ফাংশন তৈরি করেছি এবং আমরা `add_submenu_page` আইটেমটিকে `tools.php`-এর একটি শিশু হিসাবে নেভিবারে যোগ করতে ব্যবহার করেছি, যা টুলস পৃষ্ঠা।

আমাদের পাস করা আর্গুমেন্ট সম্পর্কে জানতে অনুগ্রহ করে এই পৃষ্ঠার প্যারামিটার অংশ পরীক্ষা করুন। তারপর আমরা আমাদের `register_my_custom_submenu_page` ফাংশন চালানোর জন্য `add_action` ব্যবহার করেছি। অবশেষে, আমরা পৃষ্ঠায় বিষয়বস্তু প্রদর্শনের জন্য `my_custom_submenu_page_content` ফাংশন তৈরি করেছি।

# অধ্যায় 9: add\_theme\_support()

## ভূমিকা

এই ফাংশনটি থিম সমর্থন করে এমন বৈশিষ্ট্যগুলি নিবন্ধন করে।

## বাক্য গঠন

- add\_theme\_support(\$feature)

## পরামিতি

পরামিতি বিবরণ	
\$ বৈশিষ্ট্য	(স্ট্রিং) বৈশিষ্ট্য যোগ করা হচ্ছে.

## মন্তব্য

\$feature-এ ব্যবহার করা বৈশিষ্ট্যের তালিকা:

- 'পোস্ট-ফরম্যাট' •
- 'পোস্ট-থামনেল' • 'html5'
- 'কাস্টম-  
লোগো' • 'কাস্টম-  
হেডার-আপলোড' • 'কাস্টম-হেডার' •
- 'কাস্টম-ব্যাকগ্রাউন্ড' •
- 'টাইটেল-ট্যাগ' • 'স্টার্টার-কন্টেন্ট'

## উদাহরণ

পোস্ট ফরম্যাটের জন্য থিম সমর্থন যোগ করা হচ্ছে

```
add_theme_support('পোস্ট-ফরম্যাট', অ্যারে ('ফরম্যাটোন', 'ফরম্যাটটু'));
```

পোস্টে থামনেইলের জন্য থিম সমর্থন যোগ করা হচ্ছে

```
add_theme_support('পোস্ট-থামনেল', অ্যারে ('পোস্ট'));
```

উপরের কোডটি শুধুমাত্র সমস্ত পোস্টে পোস্ট থামনেইলের অনুমতি দেয়। সমস্ত পোস্ট প্রকারের বৈশিষ্ট্য সক্রিয় করতে, করুন:

```
add_theme_support('পোস্ট-থাম্বনেল');
```

# অধ্যায় 10: অ্যাডমিন ড্যাশবোর্ড উইজেট

## ভূমিকা

অ্যাডমিন ড্যাশবোর্ড উইজেটের সাহায্যে আপনি অ্যাডমিনের কাছে যেকোনো ধরনের তথ্য প্রদর্শন করতে পারবেন ড্যাশবোর্ড আপনি চাইলে একাধিক উইজেট তৈরি করতে পারেন। আপনি `functions.php` এ কোড যোগ করতে পারেন আপনার থিম বা আপনার প্লাগইনে।

## বাক্য গঠন

- `add_action($tag, $function_to_add, $priority, $accepted_args);`
- `wp_add_dashboard_widget($widget_id, $widget_name, $callback, $control_callback, $callback_args);`

## পরামিতি

প্যারামিটার	বিস্তারিত
\$ট্যাগ	(স্ট্রিং প্রয়োজন) কর্মের নাম যেখানে \$function_to_add হুক করা আছে
\$function_to_add (কলযোগ্য প্রয়োজন)	আপনি যে ফাংশনটি কল করতে চান তার নাম।
\$ অগ্রাধিকার	(int ঐচ্ছিক) কর্মের সমস্ত ফাংশনে ফাংশন কলের স্থান (ডিফল্ট = 10)
\$accepted_args (int-ঐচ্ছিক)	ফাংশন গ্রহণ করে প্যারামিটারের সংখ্যা (ডিফল্ট = 1)
\$widget_id	(স্ট্রিং প্রয়োজন) আপনার উইজেটের জন্য অনন্য স্লাগ
\$widget_name	(স্ট্রিং প্রয়োজন) আপনার উইজেটের নাম (মাথায় প্রদর্শিত)
\$কলব্যাক	(ক্যালেলবল প্রয়োজনীয়) ফাংশনের নাম যা ডাই কন্টেন্ট প্রদর্শন করে আপনার উইজেট
\$control_callback	(কলযোগ্য ঐচ্ছিক) ফাংশনের নাম যা উইজেট বিকল্পগুলি পরিচালনা করে ফর্ম
\$callback_args	(অ্যারে ঐচ্ছিক) \$control_callback ফাংশনের পরামিতি

## উদাহরণ

সহজ উইজেট (পাঠ্য প্রদর্শন করে)

এটি একটি সাধারণ উইজেট যোগ করবে যা শুধুমাত্র একটি ছোট বার্তা প্রদর্শন করে।

```
add_action('wp_dashboard_setup', 'register_my_dashboard_widgets');
```

```
ফাংশন register_my_dashboard_widgets() {
```

```
    wp_add_dashboard_widget('myInfo_widget', 'গুরুত্বপূর্ণ তথ্য', 'display_infoWidget');
```

```
}
```

```
ফাংশন display_infoWidget() {
```

```
    echo '<p>প্রথম ফেক্সারিতে এই সাইটটি একটি নতুন ডিজাইন পায়।
```

```
    তাই এই দিন পাওয়া যাবে না. বর্তমান অগ্রগতি দেখতে আপনি <a href="http://www.jstanexample.com" >এই সাইটে</a></p>' দেখতে পারেন;
```

```
}
```

# অধ্যায় 11: AJAX

## উদাহরণ

একটি JSON প্রতিক্রিয়া সহ AJAX অনুরোধ

functions.php:

```
// আমরা দুইবার অ্যাকশন যোগ করি, একবার লগ ইন করা ব্যবহারকারীদের জন্য এবং একবার নন-লগ ইন ব্যবহারকারীদের জন্য। add_action( 'wp_ajax_my_action',  
'my_action_callback'); add_action( 'wp_ajax_nopriv_my_action', 'my_action_callback');  
  
// সামনের প্রান্তে স্ক্রিপ্টটি সারিবদ্ধ করুন। add_action( 'wp_enqueue_scripts',  
'enqueue_my_action_script'); // স্ক্রিপ্টটিকে পিছনের প্রান্তে সারিবদ্ধ করুন (wp-admin) add_action( 'admin_enqueue_scripts',  
'enqueue_my_action_script');  
  
ফাংশন my_action_callback() { $json = array();  
  
    যদি ( isset( $_REQUEST['field2'] ) ) {  
        $json['message'] = 'সফল!'; wp_send_json_success($json);  
    }  
    অন্য {  
        $json['message'] = 'ক্ষেত্র 2 সেট করা হয়নি!'; wp_send_json_error($json);  
    }  
}  
  
ফাংশন enqueue_my_action_script() {  
    wp_enqueue_script( 'my-action-script', 'path/to/my-action-script.js', অ্যারে ('jquery'),  
    নাম, সত্য);  
    wp_localize_script('my-action-script', 'my_action_data', array(  
        'ajaxurl' => admin_url('admin-ajax.php'),  
    ));  
}
```

my-action-script.js:

```
(ফাংশন($) 'কঠোর ব্যবহার' {  
    'করুন';  
  
    $( ডকুমেন্ট .on( 'রেডি', ফাংশন() {  
        var ডেটা = {  
            অ্যাকশন: 'my_action', field2: 'Hello  
            World',  
            ক্ষেত্র3: 3  
        };  
  
        $.getJSON( my_action_data.ajaxurl, ডেটা, ফাংশন( json ) {  
            যদি ( json.success ) সতর্কতা ('হ্যাঁ!'); {  
  
            } অন্য {  
                সতর্কতা (json.data.message);  
            }  
        }  
    }  
});
```

```

    });
});

})(jQuery );

```

## AJAX এর সাথে .ajax() এবং WordPress Nonce

### functions.php

```

// AJAX URL এবং Nonce add_action ('wp_enqueue_scripts',
'example_localize_ajax') স্থানীয়করণ করুন; ফাংশন example_localize_ajax(){ wp_localize_script('jquery', 'ajax',
array( 'url' => admin_url('admin-ajax.php'), 'nonce' =>
wp_create_nonce('example_ajax_nonce'),

));
}

//উদাহরণ AJAX ফাংশন
add_action('wp_ajax_example_function', 'example_function'); add_action('wp_ajax_nopriv_example_function',
'example_function'); ফাংশন উদাহরণ_ফাংশন(){

যদি ( !wp_verify_nonce($_POST['nonce'], 'example_ajax_nonce') ){
die ('অনুমতি অস্বীকার!');
}

$firstname = sanitize_text_field($_POST['data']['firstname']); $lastname = sanitize_text_field($_POST['data']
['lastname']);

// এখানে ডেটা দিয়ে কিছু করুন প্রতিধ্বনি $firstname. .
$lastname; // প্রতিক্রিয়ার জন্য ইকো wp_die(); // এটি অবিলম্বে বন্ধ করতে এবং একটি সঠিক প্রতিক্রিয়া ফেরত
দিতে হবে:-
https://codex.wordpress.org/AJAX\_in\_Plugins }

```

### example.js

```

jQuery(document).on('ক্লিক টাচ ট্যাপ', '.example-নির্বাচক', function(){
jQuery.ajax({ প্রকার:
"POST", url: ajax.url,
ডেটা: {

nonce: ajax.nonce, কর্ম:
'example_function', ডেটা: {

প্রথম নাম: 'জন',
পদবি: 'ডো'
},
},
},
সফল্য: ফাংশন(প্রতিক্রিয়া){
//সফল
},
ত্রুটি: ফাংশন (XMLHttpRequest, টেক্সট স্ট্যাটাস, ত্রুটি থ্রো){
//ত্রুটি
},
সময়সীমা: 60000
});

```

```
    মিথ্যা ফেরত;  
});
```

## wp\_ajax - মূল কার্যকারিতা + \_wpnonce চেক

### functions.php:

```
ফাংশন rm_init_js() {  
    wp_enqueue_script('custom-ajax-script', get_template_directory_uri() . '/js/ajax.js',  
    অ্যারে('jquery', 'wp-util'), '1.0', true);  
    // JS-তে কাস্টম ডেরিয়েবল পাস করুন  
    wp_localize_script('কাস্টম-অ্যাজাক্স-স্ক্রিপ্ট', 'BEJS', অ্যারে(  
        'action' => 'কাস্টম_অ্যাকশন',  
        'nonce' => wp_create_nonce('test-nonce')  
    ));  
}  
  
add_action('wp_enqueue_scripts', 'rm_init_js');  
  
ফাংশন rm_ajax_handler() {  
    check_ajax_referer('test-nonce');  
  
    নির্যাস($_POST);  
    $ডেটা = কমপ্যাক্ট('first_name', 'last_name', 'email');  
  
    foreach ($data as $name => $value) {  
        পাল্টান ($name) {  
            কেস 'first_name':  
            কেস 'শেষ_নাম':  
                $data[$name] = ucfirst( sanitize_user($value));  
            বিরতি  
            কেস 'ইমেল':  
                $data[$name] = sanitize_email($value);  
            বিরতি  
        }  
    }  
  
    $userID = email_exists($data['email']);  
  
    যদি (!$userID) {  
        (wp_send_json_error(sprintf(__('কিছু ভুল হয়েছে! %s আবার চেষ্টা করুন!', 'টেক্সটডোমেইন'), $data['first_name'], $data['last_name'])));  
    }  
  
    wp_update_user( অ্যারে(  
        'ID' => $userID,  
        'display_name' => $ডেটা['first_name'] | 'first_name' => $data['last_name'],  
        $data['first_name'],  
        'নামের শেষাংশ' => $data['last_name'],  
    ));  
  
    wp_send_json_success(sprintf(__('Welcome Back %s!', 'textdomain'), $data['first_name']  
    . $data['last_name']));  
}  
  
add_action('wp_ajax_custom_action', 'rm_ajax_handler');  
add_action('wp_ajax_nopriv_custom_action', 'rm_ajax_handler');
```

## ajax.js

```
;ফাংশন() {
    wp.ajax.post(BEJS.action, {
        প্রথম_নাম: 'জন',
        শেষ_নাম: '%65doe',
        ইমেল: 'john.doe@example.com',
        _ajax_nonce: BEJS.nonce
    }); সম্পন্ন ( ফাংশন ( প্রতিক্রিয়া ) {
        সতর্কতা ('সফল: ${response}');
    });fail( ফাংশন( প্রতিক্রিয়া ) {
        সতর্কতা ('ত্রুটি: ${response}');
    });
};
})();
```

## OOP Ajax জমা না দিয়ে একটি সাধারণ ক্লাস ব্যবহার করে

আপনি এটি চেষ্টা করার জন্য এই পুরো প্লাগইনটি অনুলিপি এবং পেস্ট করতে পারেন। এখান থেকে শ্রেণী কঙ্কাল ব্যবহৃত হয়।

## class-oop-ajax.cpp

```
<?php
/**
 * প্লাগইন বুটস্ট্র্যাপ ফাইল
 *
 * এই ফাইলটি ওয়ার্ডপ্রেস দ্বারা পঠিত হয় প্লাগইনে প্লাগইন তথ্য তৈরি করতে
 * ড্যাশবোর্ড। এই ফাইলটি একটি ফাংশন সংজ্ঞায়িত করে যা প্লাগইন শুরু করে।
 *
 * @wordpress-প্লাগইন
 * প্লাগইন নাম:                উফ এজাক্স
 * প্লাগইন ইউআরআই:         http://
 * বর্ণনা:                     থেকে একটি ফর্ম জমা দিতে OOP নীতিগুলি ব্যবহার করার একটি সহজ উদাহরণ।
সামনের অংশ.
 * সংস্করণ:                 1.0.0
 * লেখক:                     দিগ্বিজয় নারুকা
 * লেখক URI:                 http://
 * লাইসেন্স:                 GPL-2.0+
 * লাইসেন্স ইউআরআই:       http://www.gnu.org/licenses/gpl-2.0.txt
 * টেক্সট ডোমেন:            oop-এজাক্স
 * ডোমেন পথ:                /ভাষা
 */

// যদি এই ফাইলটি সরাসরি বলা হয়, তাহলে বাতিল করুন।
যদি (! সংজ্ঞায়িত ('WPINC') মারা যায়; ) {

}

ক্লাস Oop_Ajax {

    // আপনার সমস্ত add_action, add_shortcode, add_filter ফাংশন __construct() এ রাখুন
    // কলব্যাক নামের জন্য, এটি ব্যবহার করুন: array($this,'<function name>')
    // <function name> এই ক্লাসের মধ্যে ফাংশনের নাম, তাই বিশ্বব্যাপী হওয়ার দরকার নেই
```

অন্য

```
// কিছু নমুনা সাধারণত ব্যবহৃত ফাংশন নীচে অন্তর্ভুক্ত করা হয়
```

```
পাবলিক ফাংশন __construct() {
```

```
    // ফ্রন্ট-এন্ড ডিসপ্লে add_action('wp_enqueue_scripts', array($this,'enqueue')) এর জন্য  
    Javascript এবং CSS যোগ করুন;
```

```
    // ফ্রন্ট-এন্ড ফর্ম প্রদর্শনের জন্য শর্টকোড যোগ করুন add_action( 'init', অ্যারে( $this,  
    'add_form_shortcode' ));
```

```
    // Ajax ফাংশন যোগ করুন যা লগ ইন করা ব্যবহারকারীদের জন্য কল ব্যাক গ্রহণ করবে add_action( 'wp_ajax_my_action', array( $this,  
    'my_action_callback' ));
```

```
    // Ajax ফাংশন যোগ করুন যা গেস্টের জন্য কল ব্যাক রিসিভ করবে অথবা লগইন না করে ব্যবহারকারীদের add_action( 'wp_ajax_nopriv_my_action', array( $this,  
    'my_action_callback' ));
```

```
}
```

```
// এটি একটি জাভাস্ক্রিপ্ট ফাইল এবং সামনে ব্যবহারের জন্য একটি CSS ফাইল সারিবদ্ধ করার একটি উদাহরণ
```

শেষ প্রদর্শন পাবলিক

```
ফাংশন সারি () {
```

```
    // প্রকৃত সারি, নোট করুন ফাইলগুলি জেএস এবং সিএসএস ফোল্ডারে রয়েছে
```

```
    // স্ক্রিপ্টগুলির জন্য, নিশ্চিত করুন যে আপনি প্রাসঙ্গিক নির্ভরতা অন্তর্ভুক্ত করছেন (এতে jquery
```

মামলা)

```
wp_enqueue_script('my-ajax-script', plugins_url('js/ooop-ajax.js', __FILE__), array('jquery'), '1.0', true);
```

```
// কখনও কখনও আপনি আপনার জাভাস্ক্রিপ্ট ফাইলের সামনের প্রান্তে ডেটা অ্যাক্সেস করতে চান // এই কলের প্রয়োজন হয়। সর্বদা এগিয়ে যান এবং ajaxurl অন্তর্ভুক্ত করুন। অন্য কেউ
```

চলক,

```
// অ্যারে যোগ করুন।
```

```
// তারপর জাভাস্ক্রিপ্ট ফাইলে, আপনি এটিকে এইভাবে উল্লেখ করতে পারেন: my_php_variables.ajaxurl wp_localize_script( 'my-  
ajax-script', 'my_php_variables', array(
```

```
    'ajaxurl' => admin_url('admin-ajax.php'), 'nonce' =>
```

```
    wp_create_nonce('_wpnonce')
```

```
));
```

```
}
```

```
/**
```

```
 * ফর্মের জন্য শর্টকোড নিবন্ধন করে। *
```

```
পাবলিক ফাংশন add_form_shortcode() {
```

```
    add_shortcode( "oop-ajax-add-form", array( $this, "add_form_front_end" ));
```

```
}
```

```
/**
```

```
 * শর্টকোড ওপ-অ্যাজাক্স-অ্যাড-ফর্ম প্রসেস করে  
 *
```

```
 * @পরম অ্যারে $atts শর্টকোড থেকে গণাবলী  
 *
```

```
 * @রিটার্ন */ মিশ্রিত $আউটপুট বাফারের আউটপুট
```

```
ফাংশন add_form_front_end($atts, $content) {
```

```
    প্রতিক্রিয়া "<form id='my_form'>";
```

```
    echo "<label for='name'>Name: </label>"; echo "<input id='name'
```

```
    type='text' name='name'";
```

```

প্রতিধ্বনি "<br>";

echo "<label id='email' for='email'>ইমেল: </label>"; echo "<input type='text' name='email'>";

প্রতিধ্বনি "<br>";

প্রতিধ্বনি "<ইনপুট প্রকার='লুকানো' নাম='অ্যাকশন' মান='my_action' >"; echo "<input id='submit_btn' type='submit'
name='submit' value='submit'> ";

প্রতিধ্বনি "</form><br><br>"; echo "<div
id='response'>AJAX রেসপন্স এখানে থাকবে</div> ";
}

/**
 * ফর্ম ব্যবহৃত my_action-এর জন্য কলব্যাক ফাংশন।
 *
 * ফর্ম থেকে প্রাপ্ত ডেটা প্রক্রিয়া করে এবং আপনি এটি দিয়ে যা চান তা করতে পারেন।
 *
 * @রিটার্ন */
ইকো রেসপন্স স্ট্রিং এজাক্স কলের সমাপ্তি সম্পর্কে।

ফাংশন my_action_callback() { // echo wp_die('<pre>' .
    print_r($_REQUEST) "<pre>");

    check_ajax_referer( '_wpnonce', 'সুরক্ষা');

    যদি(! খালি($_POST)){

        যদি ( isset( $_POST['name'] ) ) {

            $name = sanitize_text_field($_POST['name'] ) ;

        }

        if( isset($_POST['email'] ) ) {

            $email = sanitize_text_field( $_POST['email'] ) ;

        }

        /////////////////////////////////// মান সহ জিনিসপত্র করুন / উদাহরণ: ডাটাবেসে যাচাই
        করুন এবং সংরক্ষণ করুন // প্রক্রিয়া এবং
        আউটপুট ///////////////////////////////////

        $response = "Wow <strong style='color:red'>"! $নাম "!</strong> তুমি
        রক, আপনি এইমাত্র ওপ দিয়ে AJAX কাজ করেছেন!";
        //এটি js ফাংশনে ডেটা ফেরত পাঠাবে: echo $response;

    } অন্য {

        প্রতিধ্বনি "ওহ! মনে হচ্ছে আমি আজ খাইনি";

    }

    wp_die(); // কলটি বন্ধ করতে হবে, অন্যথায় ওয়ার্ডপ্রেস শুরু করবে
    সমাপ্তি এবং শেষে অঙ্কিত '0' আউটপুট।

}

}
//আমাদের প্লাগইন শুরু করুন

```

```
বিশ্বব্যাপী $প্লাগইন;
```

```
// পুরো জিনিসটি শুরু করতে আমাদের ক্লাসের একটি উদাহরণ তৈরি করুন $plugin = new Oop_Ajax();
```

## oop-ajax.js

js ফাইলটি js ডিরেক্টরির ভিতরে রাখুন অর্থাৎ oop-ajax/js/oop-ajax.js

```
(ফাংশন($){ 'ব্যবহার কর্তার';

    $("#submit_btn").on('ক্লিক', ফাংশন(){
        // ডেটা সেট করুন
        var ডেটা = {
            কর্ম: 'my_action', নিরাপত্তা:
            my_php_variables.nonce, নাম: $("#name").val(), ইমেল: $
            ("#email").val()

        }

        $.ajax({ প্রকার:
            'পোস্ট', url:
            my_php_variables.ajaxurl,
            তথ্য: উপাত্ত,
            সাফল্য: ফাংশন(প্রতিক্রিয়া){ //সাফল্যের উপর প্রতিক্রিয়া
                আউটপুট করুন $("#response").html(response);

            },
            ত্রুটি: ফাংশন(ত্রুটি){
                console.log(err);
            }
        });

        মিথ্যা ফেরত;
    });
})(jQuery);
```

## অধ্যায় 12: বিকল্প প্রধান লুপ (প্রি\_গেট\_পোস্ট ফিল্টার)

### বাক্য গঠন

- `add_action( 'pre_get_posts', 'callback_function_name' );` • ফাংশন `callback_function_name( $query )` {} • // PHP 5.3.0 বা তার বেশির জন্য
- `add_action( 'pre_get_posts', ফাংশন( $query ) {} );`

### পরামিতি

পরামিতি বিবরণ	
<code>\$query</code>	(WP_Query) লুপ অবজেক্ট

### মন্তব্য

আপনি যদি PHP 5.3.0 বা তার বেশি ব্যবহার করেন, তাহলে আপনি ক্লোজার ব্যবহার করতে পারেন (বেনামী ফাংশন)

```
add_action( 'pre_get_posts', ফাংশন( $query )  
            if( !$query->is_main_query() || is_admin() ) রিটার্ন;  
  
            // এই কোডটি তখনই চলবে যখন // - এই প্রমিটি প্রধান  
            প্রম // - এবং এটি অ্যাডমিন স্ক্রীন নয়  
  
});
```

### উদাহরণ

এমনকি আরো নির্দিষ্ট লুপ টার্গেটিং

ধরা যাক আমরা প্রধান লুপ পরিবর্তন করতে চাই, শুধুমাত্র নির্দিষ্ট শ্রেণীবিন্যাস বা পোস্ট টাইপের জন্য।

বই পোস্ট টাইপ আর্কাইভ পৃষ্ঠায় শুধুমাত্র প্রধান লুপ টার্গেটিং .

```
add_action( 'pre_get_posts', 'my_callback_function' );  
  
ফাংশন my_callback_function( $query )  
            if( !$query->is_main_query() || is_admin() ) রিটার্ন; if( !is_post_type_archive('book') ) রিটার্ন;  
  
            // এই কোডটি তখনই চলবে যদি // - এই প্রমিটি প্রধান প্রম  
  
            // - এবং এটি অ্যাডমিন স্ক্রীন নয়
```

```
// - এবং আমরা 'বই' পোস্ট টাইপ আর্কাইভ পেজে আছি  
}
```

এছাড়াও আপনি `is_category()`, `is_tag()` এবং `is_tax()` ব্যবহার করে বিভাগ, ট্যাগ বা কাস্টম শ্রেণীবিন্যাস সংরক্ষণাগার পৃষ্ঠা পরীক্ষা করতে পারেন।

আপনি ওয়ার্ডপ্রেসে উপলব্ধ যেকোনো শর্তসাপেক্ষ ট্যাগ ব্যবহার করতে পারেন।

## শুধুমাত্র একটি বিভাগ থেকে পোস্ট দেখান

```
add_action( 'pre_get_posts', 'single_category');  
  
ফাংশন single_category( $query ) if( !$query->is_main_query(){  
    || is_admin() ) রিটার্ন;  
  
    $query->সেট( 'বিভাগ', '1' );  
    প্রত্যাবর্তন  
}
```

## প্রি পেতে পোস্ট ফিল্টার মৌলিক ব্যবহার

কখনও কখনও আপনি প্রধান ওয়ার্ডপ্রেস ক্যোয়ারী পরিবর্তন করতে চান।

প্রি\_গেট\_পোস্টগুলি ফিল্টার করার উপায়।

উদাহরণস্বরূপ `pre_get_posts` ব্যবহার করে আপনি প্রধান লুপ বলতে পারেন মাত্র 5টি পোস্ট দেখানোর জন্য। অথবা শুধুমাত্র একটি বিভাগ থেকে পোস্ট দেখাতে, বা কোন বিভাগ বাদ দিয়ে ইত্যাদি।

```
add_action( 'pre_get_posts', 'my_callback_function');  
  
ফাংশন my_callback_function( $query ) // এখানে আপনার ফিল্টারের {  
    যুক্তি আছে  
}
```

আপনি দেখতে পাচ্ছেন, আমরা প্রধান লুপ পাস করছি আমাদের কলব্যাক ফাংশন আর্গুমেন্ট মধ্যে প্রথম বস্তু।

এখানে গুরুত্বপূর্ণ নোট: আমরা একটি রেফারেন্স হিসাবে যুক্তি পাস করছি। এর মানে হল যে এটি কাজ করার জন্য আমাদের কোয়েরি ফেরত দেওয়ার বা কোনো গ্লোবাল সেট করার দরকার নেই। যেহেতু `$query` হল মূল ক্যোয়ারী অবজেক্টের একটি রেফারেন্স, তাই আমরা আমাদের অবজেক্টে যে সমস্ত পরিবর্তন করি তা অবিলম্বে মূল লুপ অবজেক্টে প্রতিফলিত হয়।

## পোস্ট তালিকা সম্পাদনা শেয়ার থেকে বিভাগ বাদ দিন

```
add_action( 'pre_get_posts', 'single_category_exclude');  
  
ফাংশন single_category_exclude( $query ) {  
    if( !$query->is_main_query() || is_admin() ) রিটার্ন;  
  
    $query->সেট( 'বিভাগ', '-1' ); প্রত্যাবর্তন  
}
```

প্রধান লুপের জন্য পোস্ট\_প্রতি\_পৃষ্ঠা পরিবর্তন করুন

আমাদের যা করতে হবে তা হল \$query অবজেক্টের সেট() পদ্ধতি ব্যবহার করা।

এটি দুটি আর্গুমেন্ট লাগে, প্রথম আমরা কি সেট করতে চাই এবং দ্বিতীয় কোন মান সেট করতে চাই।

```
add_action( 'pre_get_posts', 'change_posts_per_page');

ফাংশন change_posts_per_page( $query ) if (!$query->is_main_query() {
    || is_admin() ) রিটার্ন;

    $query->সেট('posts_per_page', 5);
    প্রত্যাবর্তন
}
```

শুধুমাত্র প্রধান ওয়ার্ডপ্রেস লুপ টার্গেটিং

ওয়ার্ডপ্রেস আক্ষরিক অর্থে যে কোনও লুপ তৈরি করতে প্রি\_গেট\_পোস্ট ফিল্টার প্রয়োগ করছে। এর মানে হল যে সমস্ত পরিবর্তন আমরা আমাদের কলব্যাক ফাংশনে করছি সমস্ত প্রস্থানকারী লুপে প্রয়োগ করা হয়।

স্পষ্টতই আমরা বেশিরভাগ পরিস্থিতিতে যা চাই তা নয়।

বেশিরভাগ ক্ষেত্রে আমরা শুধুমাত্র প্রধান লুপ লক্ষ্য করতে চাই, এবং শুধুমাত্র নন-অ্যাডমিন স্ক্রিনের জন্য।

আমরা সঠিক জায়গায় আছি কিনা তা পরীক্ষা করতে আমরা is\_main\_query() পদ্ধতি এবং is\_admin() গ্লোবাল ফাংশন ব্যবহার করতে পারি।

```
add_action( 'pre_get_posts', 'my_callback_function');

ফাংশন my_callback_function( $query ) {
    if (!$query->is_main_query() || is_admin() ) রিটার্ন;

    // এই কোডটি তখনই চলবে যখন // - এই প্রশ্নটি প্রধান
    প্রশ্ন // - এবং এটি অ্যাডমিন স্ক্রীন নয়
}
```

# অধ্যায় 13: চাইল্ড থিম বেসিক

## বাক্য গঠন

- টেমপ্লেট -- হল মূল ওয়ার্ডপ্রেস থিমের নাম, প্যারেন্ট। • চাইল্ড-থিম -- হল প্যাকেজ যা টেমপ্লেটকে ওভাররাইড করে।

## মন্তব্য

আমি বিজ্ঞাপন দিচ্ছি যে একটি শিশু থিম ব্যবহার করা সবসময় একটি ভাল জিনিস কিন্তু সবসময় একটি কিন্তু আছে

...

আমাদের টেমপ্লেট ওভাররাইটিং উদাহরণে কল্পনা করা যাক যে একটি থিমের লেখক সাইডবার টেমপ্লেটে তার নিজস্ব উন্নতি যোগ করছেন এবং সেখানে একটি নতুন হবে

/themes/template/sidebar.php

```
<?php /
** *
প্রধান উইজেট এলাকা ধারণকারী সাইডবারের জন্য টেমপ্লেট
*
* @link https://developer.wordpress.org/themes/basics/template-files/#template-partials */

যদি ( is_active_sidebar ('sidebar-1') ) :?>
<aside id="secondary" class="sidebar widget-area" role="complementary">
<?php dynamic_sidebar('sidebar-1'); ?>
</aside><!-- .sidebar .widget-area --> <?php endif; ?>
```

এখন আমাদের ওয়েবসাইট নতুন role="complementary" spec থেকে উপকৃত হবে না কারণ আমাদের চাইল্ড থিম এখনও /themes/child-theme/sidebar.php- এ নিজস্ব ফাইল দিয়ে টেমপ্লেটটি ওভাররাইট করছে

ওয়েবসাইট রক্ষণাবেক্ষণকারী হিসাবে এটি আমাদের কর্তব্য যে আমরা কোন টেমপ্লেটগুলিকে ওভাররাইট করি সে সম্পর্কে একটি ট্র্যাক রাখা এবং, একটি আপডেটের আসন্ন ক্ষেত্রে, চেঞ্জলগটি সাবধানে দেখুন যাতে আপনি প্রয়োজনে চাইল্ড থিম ফাইলগুলি আপডেট করেন।

## উদাহরণ

### 2) উদ্দেশ্য

চাইল্ড থিমগুলিকে থিম আপডেটে হারানোর ভয় না করে মূল টেমপ্লেটের কাস্টমাইজেশন রাখার একটি নিরাপদ উপায় বোঝানো হয়েছে।

মূলত, আপনি যখনই আপনার ওয়েবসাইট থেকে সক্রিয় টেমপ্লেটের ভিতরে একটি ফাইল সম্পাদনা করতে চান তখন আপনাকে নিজেই জিজ্ঞাসা করতে হবে "আমি যখন থিম আপডেট করব তখন কী ঘটবে?"

এবং উত্তরটি সহজ: আপনি সেগুলি হারাবেন কারণ আপডেটটি একটি সম্পূর্ণ নতুন থিম নিয়ে আসবে

## ফোল্ডার

সুতরাং আসুন একটি চাইল্ড থিমকে ফাইল সহ একটি ফোল্ডার হিসাবে দেখি যা মূল থিমের একই পথ দিয়ে ফাইলগুলিকে ওভাররাইট করবে। এখন কিছু বাস্তব উদাহরণ নিয়ে আসা যাক:

## সংজ্ঞা এবং প্রয়োজনীয়তা

নিম্নলিখিত ফাইলগুলির সাথে /wp-content/themes/ এর ভিতরে একটি ডিরেক্টরি (উদাহরণস্বরূপ চাইল্ড-থিম) থাকলে ওয়ার্ডপ্রেস দ্বারা একটি চাইল্ড থিম সনাক্ত করা হয় :

- style.css

এই ফাইলটি অবশ্যই এই মত একটি মন্তব্য টেমপ্লেট দিয়ে শুরু করতে হবে:

```
/*  
থিমের নাম: উদাহরণ শিশু  
লেখক: আমি  
লেখক URI: https://example.com/ টেমপ্লেট: উদাহরণ টেমপ্লেট  
ডোমেন: example-child-theme  
ডোমেন পথ: /languages/ */
```

এখানে বিবেচনা করা সবচেয়ে গুরুত্বপূর্ণ বিষয় হল:

- টেমপ্লেটের নামটি অবশ্যই সেই ফোল্ডারের নাম হতে হবে যা মূল থিম (ওরফে প্যারেন্ট থিম স্লাগ) ধারণ করে
- আপনার সন্তানের থিমের নাম এমনভাবে রাখুন যাতে আপনি সহজেই ড্যাশবোর্ডে এটি সনাক্ত করতে পারেন (সাধারণত শুধুমাত্র পিতামাতার নামের সাথে শিশু যোগ করুন, যেমন উদাহরণ শিশু)

- index.php
- functions.php

## 3) টেমপ্লেট ওভাররাইটিং

একটি চাইল্ড থিমের সবচেয়ে সাধারণ ব্যবহার হল টেমপ্লেট অংশগুলিকে ওভাররাইড করা। উদাহরণস্বরূপ, একটি সাইডবার, যদি আমাদের কাছে নিম্নলিখিত ফাইল সহ একটি থিম থাকে

/themes/template/sidebar.php

```
<?php /  
**  
* প্রধান উইজেট এলাকা ধারণকারী সাইডবার।  
*  
* @link https://developer.wordpress.org/themes/basics/template-files/#template-partials */  
  
যদি ( ! is_active_sidebar('sidebar-1') ) {  
    প্রত্যাবর্তন  
}>  
<div id="sidebar" class="widget-area">  
    <?php dynamic_sidebar('sidebar-1'); ?>
```

```
</div>
```

আমরা অবশ্যই চাইল্ড থিমে আমাদের নিজস্ব ফাইল যুক্ত করতে পারি (প্রথম উদাহরণের স্পেসিফিকেশন সহ) নিম্নলিখিত ফাইলের সাথে

```
/themes/child-theme/sidebar.php
```

```
<?php /
** *
প্রধান উইজেট এলাকা ধারণকারী সাইডবার। */

যদি ( !is_active_sidebar('sidebar-1') ) {
    প্রত্যাবর্তন
}>
<div id="my-sidebar" class="my-own-awesome-class widget-area"> <div class="my-wrapper"> <?php
dynamic_sidebar('sidebar-1'); ?> </div> </div>
div>
```

এখন আমার-নিজের-অসাধারণ-ক্লাস চাইল্ড থিমে নিরাপদ এবং এটি কোনো থিম আপডেটে সরানো হবে না এবং ওয়ার্ডপ্রেস সবসময় চাইল্ড থিম থেকে একটি টেমপ্লেট বেছে নেবে যখন এটি একই পথে একটি খুঁজে পায়।

## সম্পদ প্রতিস্থাপন

এমনকি এটি একটি সর্বোত্তম অনুশীলন না হলেও, কখনও কখনও আপনাকে CSS বা JS ফাইল বা লাইব্রেরির মতো সম্পদ প্রতিস্থাপন করতে হবে।

মনে রাখবেন যে ওয়ার্ডপ্রেস টেমপ্লেট ওভাররাইটিং সিস্টেম .php ফাইল ছাড়া অন্য কিছুতে কাজ করে না, তাই আমরা যখন সম্পদের কথা বলি তখন আমরা নিবন্ধিত উল্লেখ করি সম্পদ

একটি উদাহরণ আপনার পছন্দসই সংস্করণের সাথে jQuery লাইব্রেরির প্রতিস্থাপন হতে পারে। আমাদের চাইল্ড থিম functions.php ফাইলে আমাদের একটি ফাংশন যোগ করতে হবে যা বর্তমান jQuery সংস্করণটি সরিয়ে দেয় এবং CDN থেকে আমাদের নিজস্ব যোগ করে (মনে রাখবেন একটি উদাহরণ মাত্র)।

```
/**
 * jQuery স্ক্রিপ্ট ডিকুয় করুন এবং আমাদের নিজস্ব সংস্করণ যোগ করুন।
 */

* wp_print_scripts অ্যাকশনে আবদ্ধ, দেরিতে অগ্রাধিকার সহ (100),
* যাতে এটি স্ক্রিপ্ট সারিবদ্ধ করার পরে হয়।
*/

ফাংশন my_own_theme_scripts() {
    // বর্তমান সংস্করণ সরান
    wp_dequeue_script('jquery'); // আমার কাস্টম সংস্করণ
    নিবন্ধন করুন wp_register_script( 'jquery', 'https://
code.jquery.com/jquery-3.1.0.min.js', মিথ্যা,
'3.1.0');
    // আমার সংস্করণ লোড করুন, এখানে বা অন্য কোথাও wp_enqueue_script('jquery');
} add_action( 'wp_print_scripts', 'my_own_theme_scripts');
```



## অধ্যায় 14: প্রোগ্রামগতভাবে একটি পোস্ট তৈরি করুন

### বাক্য গঠন

• wp\_insert\_post(অ্যারে \$args, bool \$wp\_error);

### পরামিতি

প্যারামিটার	বর্ণনা
\$args (অ্যারে প্রয়োজনীয়)	নীচের উপাদানগুলির একটি মূল মান অ্যারে।
\$wp_error (বুলিয়ান ঐচ্ছিক) ব্যর্থতার ক্ষেত্রে	একটি WP_Error ফেরত দিন।

### মন্তব্য

## যুক্তি

পরবর্তী টেবিল আপনাকে উপাদানগুলির একটি তালিকা দেখায় যা আপনি প্রথম প্যারামিটার (অ্যারে) এর ভিতরে ব্যবহার করতে পারেন।

প্যারামিটার	বর্ণনা
আইডি	(Int) পোস্ট আইডি। 0 ছাড়া অন্য কিছু সমান হলে, সেই আইডি সহ পোস্ট আপডেট করা হবে। ডিফল্ট 0।
পোস্ট_লেখক	(Int) পোস্ট যোগ করা ব্যবহারকারীর আইডি। ডিফল্ট হল বর্তমান ব্যবহারকারী আইডি।
পোস্ট_তারিখ	(স্ট্রিং) পোস্টের তারিখ। ডিফল্ট বর্তমান সময়।
পোস্ট_তারিখ_জিএমটি	(স্ট্রিং) GMT টাইমজোনে পোস্টের তারিখ। ডিফল্ট \$post_date এর মান।
পোস্ট_কন্টেন্ট	(মিশ্র) পোস্ট বিষয়বস্তু। ডিফল্ট খালি।
post_content_filtered	(স্ট্রিং) ফিল্টার করা পোস্টের বিষয়বস্তু। ডিফল্ট খালি।
post_title	(স্ট্রিং) পোস্টের শিরোনাম। ডিফল্ট খালি।
পোস্ট_শ্রেণী	(অ্যারে) পোস্ট বিভাগের মানগুলির অ্যারে।
post_excerpt (স্ট্রিং) পোস্ট উদ্ধৃতি	ডিফল্ট খালি।

প্যারামিটার	বর্ণনা
পোস্ট_স্ট্যাটাস	(স্ট্রিং) পোস্টের অবস্থা। ডিফল্ট খসড়া।
পোস্ট_টাইপ	(স্ট্রিং) পোস্টের ধরন। ডিফল্ট পোস্ট।
মন্তব্য_স্থিতি	(স্ট্রিং) পোস্টটি মন্তব্য গ্রহণ করতে পারে কিনা। গ্রহণ করে খোলা বা বন্ধ। ডিফল্ট এর মান default_comment_status বিকল্প।
ping_status	(স্ট্রিং) পোস্টটি পিংস গ্রহণ করতে পারে কিনা। খোলা স্বীকার করে বা বন্ধ। ডিফল্ট হল default_ping_status এর মান বিকল্প।
পোস্ট_পাসওয়ার্ড	(স্ট্রিং) পোস্ট অ্যাক্সেস করার পাসওয়ার্ড। ডিফল্ট খালি।
পোস্ট_নাম	(স্ট্রিং) পোস্টের নাম বা স্লাগ। ডিফল্ট স্যানিটাইজ করা হয় একটি নতুন পোস্ট তৈরি করার সময় পোস্ট শিরোনাম।
পিং করতে	(স্ট্রিং) স্পেস বা ক্যারেজ ফেরত-বিচ্ছিন্ন ইউআরএল-এর তালিকা পিং ডিফল্ট খালি।
pinged	(স্ট্রিং) স্পেস বা ক্যারেজ রিটার্ন-বিচ্ছিন্ন URL-এর তালিকা যে pinged করা হয়েছে। ডিফল্ট খালি।
post_modified	(স্ট্রিং) পোস্টটি শেষবার পরিবর্তন করার তারিখ। ডিফল্ট বর্তমান সময়।
post_modified_gmt	(স্ট্রিং) যে তারিখে পোস্টটি শেষবার পরিবর্তন করা হয়েছিল GMT টাইমজোন। ডিফল্ট বর্তমান সময়।
post_parent	(Int) এটি যে পোস্টের, যদি থাকে তার জন্য এটি সেট করুন। ডিফল্ট 0।
মেনু_অর্ডার	(Int) পোস্টটি যে অর্ডারে প্রদর্শিত হবে। ডিফল্ট 0।
পোস্ট_মাইম_টাইপ	(স্ট্রিং) পোস্টের মাইম প্রকার। ডিফল্ট খালি।
নির্দেশিকা	(স্ট্রিং) পোস্টের রেফারেন্সের জন্য গ্লোবাল ইউনিক আইডি। ডিফল্ট খালি।
ট্যাক্স_ইনপুট	(অ্যারে) শ্রেণীবিন্যাস পদের বিন্যাস তাদের শ্রেণীবিন্যাস দ্বারা কী করা হয়েছে নাম ডিফল্ট খালি।
meta_input	(অ্যারে) পোস্ট মেটা মানের অ্যারে তাদের পোস্ট মেটা দ্বারা কী করা হয় চাবি। ডিফল্ট খালি।

## ডুপ্লিকেটেড পোস্ট এড়িয়ে চলুন

আপনি যখন এই ফাংশনটি চালান, আপনি সম্ভবত একটি সদৃশ পোস্ট পেতে পারেন, অন্তত আমার সাথে এটি ঘটেছে। (আপনি পোস্ট ওয়ার্ডপ্রেস বিভাগে এটি পরীক্ষা করতে পারেন)

আমি একটি সমাধান খুঁজে পেয়েছি:

```
if( !get_page_by_title( $title, 'OBJECT', 'post' ) ){ $my_post = array('post_title' => $title,

    'post_content' => 'সামগ্রী', 'tags_input' => $tags,
    'post_category' => অ্যারে(2),
    'post_status' => 'প্রকাশ করুন'

);

$result = wp_insert_post( $my_post);
}
```

## ব্যাখ্যা

আপনি একটি নতুন পোস্ট সংরক্ষণ করার আগে, একটি প্যারামিটার হিসাবে পোস্ট শিরোনাম ব্যবহার করে নতুন পোস্টটি ইতিমধ্যেই বিদ্যমান কিনা তা যাচাই করুন, যদি পোস্টের শিরোনাম না থাকে তবে আপনি আপনার নতুন পোস্ট সংরক্ষণ করতে পারেন।

get\_page\_by\_title এর ডকুমেন্টেশন এখানে দেখুন।

## উদাহরণ

### ভূমিকা

কখনও কখনও আমাদের কাছে TinyMCE (Wordpress Default Editor) এর পরিবর্তে অন্য কোথাও অন্য সম্পাদক থাকে। এটি ঘটে যখন আমরা আমাদের নিজস্ব থিম, প্লাগইন বা নির্দিষ্ট কিছু তৈরি করি; এবং আমাদের একটি ধরনের পোস্ট লিখতে এবং ম্যানিপুলেট করতে হবে এবং আমাদের WP ডেটাবেসে সংরক্ষণ করতে হবে।

সুতরাং, আপনি যদি সেই পরিস্থিতিতে থাকেন তবে আপনি একটি ওয়ার্ডপ্রেস ফাংশন ব্যবহার করতে পারেন যার নাম:

```
wp_insert_post( অ্যারে $args, bool $wp_error);
```

একটি মৌলিক পোস্ট তৈরি করুন

```
$basic_post_args = array( 'post_title' =>
    'আমার মৌলিক পোস্ট', 'post_content' => 'এটি একটি
    মৌলিক বিষয়বস্তু', 'post_status' => 'প্রকাশ করুন', 'post_author' => 1,
    'post_category' => অ্যারে(8, 59)

);

wp_insert_post( $basic_post_args);
```

একটি মৌলিক পৃষ্ঠা তৈরি করুন

```
$basic_page_args = অ্যারে( 'post_title' =>
    'আমার মৌলিক পৃষ্ঠা', 'post_content' => 'এটি একটি মৌলিক
    বিষয়বস্তু', 'post_type' => 'পৃষ্ঠা', 'post_status' => 'প্রকাশ', 'post_author' => ১

);

wp_insert_post( $basic_page_args);
```

# অধ্যায় 15: কাস্টম পোস্টের জন্য টেমপ্লেট তৈরি করুন টাইপ

## উদাহরণ

কাস্টম পোস্ট টাইপ বইয়ের জন্য একটি কাস্টম টেমপ্লেট তৈরি করা

আমাদের কাস্টম পোস্ট টাইপের একক পোস্টের জন্য একটি টেমপ্লেট তৈরি করতে, আমাদের একক-পোস্ট\_টাইপ\_নাম.php নামে একটি ফাইল তৈরি করতে হবে যেখানে post\_type\_name আমাদের কাস্টম পোস্ট টাইপের নাম।

উদাহরণস্বরূপ, যদি আমাদের কাস্টম পোস্টের ধরনটিকে "বুকস" বলা হয়, তাহলে আমাদের একক book.php নামে একটি পিএইচপি ফাইল তৈরি করতে হবে। মনে রাখবেন যে আমরা আমাদের কাস্টম পোস্ট টাইপের একক নাম ব্যবহার করেছি।

থিম ফোল্ডার থেকে single.php এর বিষয়বস্তু কপি করুন এবং নতুন টেমপ্লেটে পেস্ট করুন এবং সংরক্ষণ করুন তাহলে টেমপ্লেটটি কাস্টম পোস্ট টাইপ পৃষ্ঠার জন্য প্রয়োগ করা হবে।

## কাস্টম পোস্ট টাইপ টেমপ্লেট

## কাস্টম পোস্ট টাইপ আর্কাইভ:

একটি কাস্টম পোস্ট টাইপের জন্য একটি সংরক্ষণাগার টেমপ্লেট তৈরি করতে আপনাকে আপনার রেজিস্টার\_পোস্ট\_টাইপ()- এ সত্যের সমান has\_archive আর্গুমেন্ট সেট করতে হবে ফাংশন নিচের উদাহরণে একটি ইভেন্ট পোস্ট টাইপের জন্য একটি কাস্টম পোস্ট টাইপ তৈরি করা হয়েছে।

```
add_action('init', 'create_events_post_type'); ফাংশন create_events_post_type()
{ register_post_type('event',

    অ্যারে(
        'লেবেল' => অ্যারে(
            'নাম' => __('ঘটনা'), 'একবচন_নাম' =>
                __('ইভেন্ট')
        ),
        'public' => সত্য, 'has_archive'
            => সত্য,
    )
);
}
```

একটি টেমপ্লেট তৈরি করতে নতুন কাস্টম পোস্ট প্রকারের জন্য আপনাকে একটি নতুন টেমপ্লেট ফাইল তৈরি করতে হবে। একক পোস্ট পৃষ্ঠাগুলির জন্য একটি টেমপ্লেট তৈরি করতে আপনি এটিকে একক নাম দেবেন-{post\_type}.php এবং সংরক্ষণাগারের জন্য-{post\_type}.php।

আমাদের সংরক্ষণাগার টেমপ্লেটের ফাইলের নাম হবে archive-event.php এবং ইভেন্ট পৃষ্ঠার জন্য এটি হবে একক-event.php। উভয় ফাইলই আপনার থিম রুট ডিরেক্টরিতে থাকা উচিত।



```

        </main><!-- #main -->
    </div><!-- #primary -->
    <?php get_sidebar(); ?>
</div><!-- .wrap -->

<?php get_footer();

```

## কাস্টম পোস্ট টাইপ একক টেমপ্লেট:

এখানে একটি একক টেমপ্লেটের উদাহরণ। টুয়েন্টিস্টিন থিম থেকে টানা।

```

<?php
/**
 * সমস্ত একক পোস্ট প্রদর্শনের জন্য টেমপ্লেট
 *
 * @লিংক https://developer.wordpress.org/themes/basics/template-hierarchy/#single-post
 *
 * @প্যাকেজ ওয়ার্ডপ্রেস
 * @সাবপ্যাকেজ টুয়েন্টি_সেভেন্টিন
 * @১.০ থেকে
 * @সংস্করণ 1.0
 */

get_header(); ?>

<div class="wrap">
    <div id="primary" class="content-area">
        <main id="main" class="site-main" role="main">

            <?php
                /* লুপ শুরু করুন */
                যখন ( আছে_পোস্ট() ) : পোস্টটি();

                get_template_part('template-parts/post/content', get_post_format());

                // যদি মন্তব্য খোলা থাকে বা আমাদের অন্তত একটি মন্তব্য থাকে, তাহলে লোড করুন
                মন্তব্য টেমপ্লেট।

                যদি ( comments_open() || get_comments_number() ) comments_template();

                যদি শেষ;

                the_post_navigation( array(
                    'prev_text' => '<span class="screen-reader-text">' . __( 'আগে
                    পোস্ট', 'একুশ' ) . '<span aria-hidden="true" class="nav-subtitle">' . __(
                    'আগের', 'twentyseventeen' ) . ' icon-wrapper' ) . '<span class="nav-title"><span class="nav-title
                    twentyseventeen_get_svg( array( 'icon' => 'তীর-বাম' ) ) . '%title',
                    ) .

                    'next_text' => '<span class="screen-reader-text">' . __( 'পরবর্তী পোস্ট',
                    'twentyseventeen' ) . '<span aria-hidden="true" class="nav-subtitle">' . __( 'পরবর্তী',
                    'twentyseventeen' ) . ' wrapper' ) . '<span class="nav-title">%title<span class="nav-title-icon'
                    twentyseventeen_get_svg( array( 'icon' => 'তীর-ডান' ) . '%title',
                    ) .

                    );

                endwhile; // লুপের শেষ।
            ?>

        </main><!-- #main -->

```

```
</div><!-- #primary --> <?php  
get_sidebar(); ?> </div><!-- .wrap -->
```

```
<?php get_footer();
```

উভয় টেমপ্লেট উদাহরণ আংশিক মধ্যে টানা হয় ভিতরের বিষয়বস্তু প্রদর্শন [করতে](#)।

যদি আপনার সন্তানের/অভিভাবকের থিমের একটি একক/আর্কাইভ টেমপ্লেট থাকে তাহলে আপনার নতুন টেমপ্লেটের জন্য সেই কোডটি বয়লারপ্লেট হিসেবে ব্যবহার করা উচিত।

# অধ্যায় 16: একটি কাস্টম টেমপ্লেট তৈরি করা

## উদাহরণ

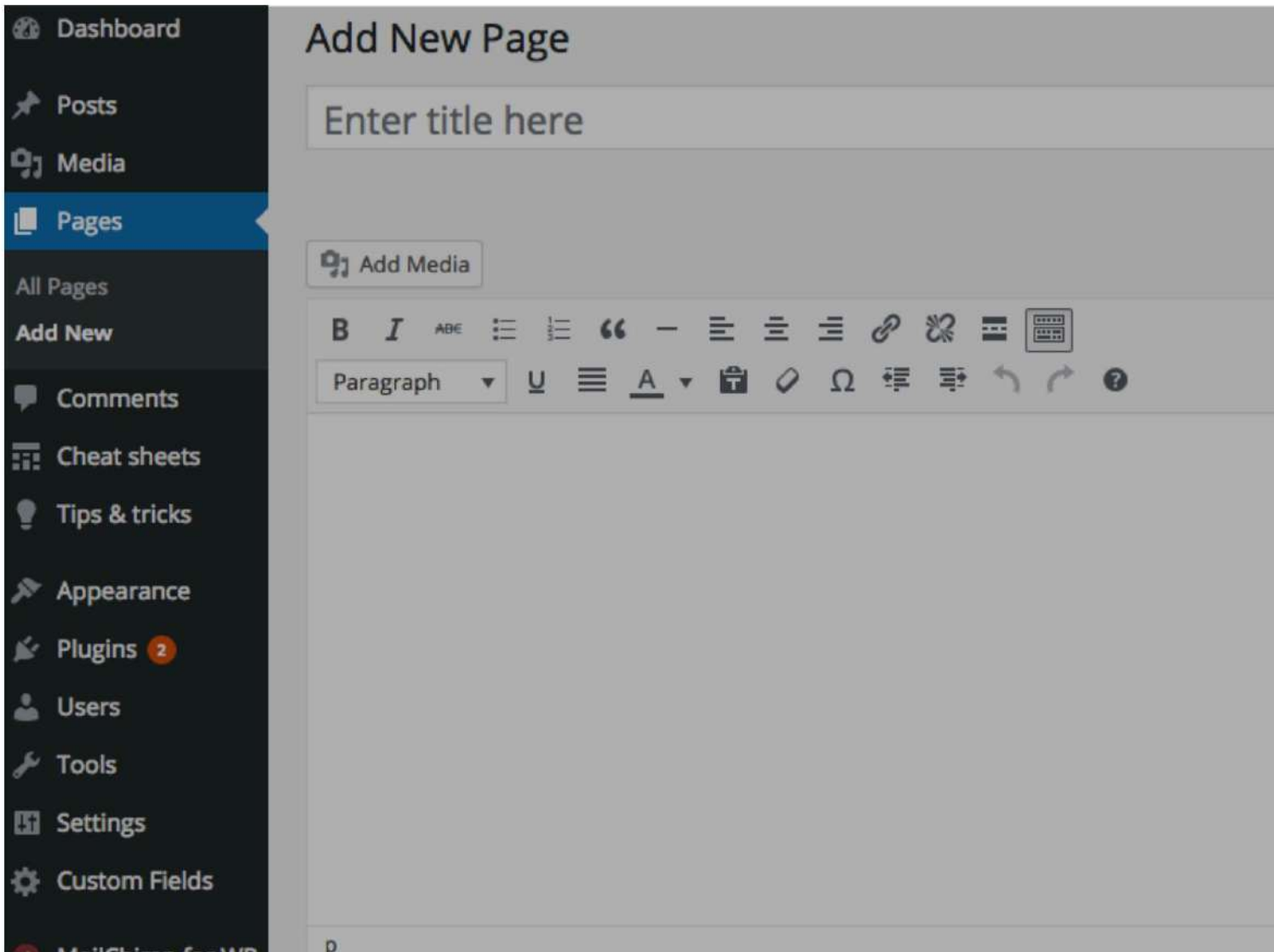
মৌলিক ফাঁকা টেমপ্লেট তৈরি করা হচ্ছে

একটি কাস্টম টেমপ্লেট তৈরি করতে আমাদের প্রথমে একটি থিম ডিরেক্টরিতে php ফাইল তৈরি করতে হবে। আপনি প্রায় যে কোন উপায়ে এটি নাম দিতে পারেন। এই উদাহরণের জন্য আমরা example.php তৈরি করব

ওয়ার্ডপ্রেসকে টেমপ্লেট হিসেবে স্বীকৃতি দেওয়ার জন্য আমাদের example.php-এর ভিতরে একটি এবং একমাত্র জিনিসটি সংজ্ঞায়িত করতে হবে, তা হল টেমপ্লেটের নাম। আমরা একটি ফাইলের শীর্ষে বিশেষ মন্তব্য রেখে এটি করি, যেমন:

```
<?php /  
*  
টেমপ্লেট নাম: উদাহরণ */  
?>
```

এবং এখন যখন আমরা পেজ অ্যাট্রিবিউট বক্সে টেমপ্লেট ড্রপডাউনে তালিকাভুক্ত আমাদের টেমপ্লেট দেখতে পাব



আমাদের টেমপ্লেটে হেডার এবং ফুটার সহ

আসুন উপরে থেকে আমাদের টেমপ্লেট প্রসারিত করি এবং header.php এবং footer.php থেকে সামগ্রী অন্তর্ভুক্ত করি

শিরোনাম সহ:

আমরা টেমপ্লেট নামের মন্তব্যের ঠিক পরে শিরোনাম অন্তর্ভুক্ত করব

এটি করার দুটি সাধারণ উপায় রয়েছে। উভয়ই সঠিক এবং একই কাজ করে, এটি কেবল আপনার শৈলী এবং কোডটি কীভাবে দেখায় সে সম্পর্কে

প্রথম উপায়:

```
<?php
/*
টেমপ্লেটের নাম: উদাহরণ
*/
get_header();
?>
```

দ্বিতীয় উপায়:

```
<?php
/*
টেমপ্লেটের নাম: উদাহরণ
*/
?>
<?php get_header(); ?>
```

ফুটার সহ:

পাদচরণ সহ একইভাবে কাজ করে, শুধুমাত্র একটি জিনিস আমাদের যত্ন নিতে হবে, এবং তা হল আমরা শিরোনাম অন্তর্ভুক্ত করার পরে ফুটার অন্তর্ভুক্ত করি। তাই চূড়ান্ত টেমপ্লেট এই মত কিছু দেখতে হবে.

```
<?php
/*
টেমপ্লেটের নাম: উদাহরণ
*/
get_header();
?>

<?php get_footer(); ?>
```

বিষয়বস্তু সহ কাস্টম টেমপ্লেট

আমরা আমাদের টেমপ্লেট আরও প্রসারিত করব এবং পৃষ্ঠার শিরোনাম এবং একটি বিষয়বস্তু অন্তর্ভুক্ত করব

```
<?php /
*
টেমপ্লেট নাম: উদাহরণ */
```

```
get_header();

শিরোনাম();
সূচিপত্র();

get_footer();
```

এবং যদি আপনি চান আপনি এই মত HTML উপাদান দিয়ে মোড়ানো করতে পারেন

```
<?php /
*
টেমপ্লেট নাম: উদাহরণ */

get_header();

প্রতিধ্বনি '<h1>' . শিরোনাম() . '</h1>'; প্রতিধ্বনি '<বিভাগ>'।
'সূচিপত্র()' . '</section>';

get_footer();
```

অথবা আপনি যদি ইকো ব্যবহার না করে সাধারণ HTML ফাইলের মতো কাজ করতে পছন্দ করেন

```
<?php /
*
টেমপ্লেট নাম: উদাহরণ */

get_header();
?>

<h1><?php the_title(); ?></h1> <বিভাগ><?php the_content(); ?
></section>

<?php get_footer(); ?>
```

---

## অধ্যায় 17: excerpt\_length এবং excerpt\_more সহ কাস্টম এক্সপার্ট

### উদাহরণ

উদ্ধৃতির দৈর্ঘ্য 50 শব্দের মধ্যে সীমাবদ্ধ করুন

নিম্নলিখিত কোডটি functions.php এ রাখুন:

```
ফাংশন themify_custom_excerpt_length($length) রিটার্ন 50; {  
  
} add_filter( 'excerpt_length', 'themify_custom_excerpt_length', 999);
```

ফাংশনটি ডিফল্ট ওয়ার্ডপ্রেস ফিল্টারের পরে চলে তা নিশ্চিত করতে অগ্রাধিকার হিসাবে 999 ব্যবহার করুন, অন্যথায় এটি এখানে যা সেট করা আছে তা ওভাররাইড করবে।

উদ্ধৃতির শেষে আরও পড়ুন লিঙ্ক যুক্ত করা হচ্ছে

এটি করার জন্য, functions.php এ নিম্নলিখিত কোডটি রাখুন :

```
ফাংশন custom_excerpt_more($ more) {  
    '<a href="" ফেরত দিন। get_permalink($post->ID)। "">আরো পড়ুন</a>';  
  
} add_filter('excerpt_more', 'custom_excerpt_more');
```

ফলাফল এই মত হওয়া উচিত:





## Utisci sa prvog iOS meetupa

Mar 4, 2016

Utisci sa prvog iOS meetupa su fantastični, što bi u našem narodu rekli-prvo pa muško! Ovo okupljanje dokazalo je da iOS zajednica u Srbiji i te kako postoji i da uopšte nije mala kao što se mislilo. Na događaju je bilo nešto više od 100 ljudi, a predavanja su bila izuzetno zanimljiva i korisna. Organizatori.....

[Ostavi komentar](#)



# অধ্যায় 18: কাস্টম পোস্ট প্রকার

বাক্য গঠন

- register\_post\_type( \$post\_type, \$args);

পরামিতি

পরামিতি বিবরণ	
\$post_type (স্ট্রিং) (প্রয়োজনীয়)	
\$args	(অ্যারে/স্ট্রিং) (ঐচ্ছিক)

উদাহরণ

একটি কাস্টম পোস্ট টাইপ নিবন্ধন

বলুন আপনার একটি লাইব্রেরি ওয়েবসাইট আছে এবং আপনি বই নামে একটি কাস্টম পোস্ট টাইপ রাখতে চান। এটি হিসাবে নিবন্ধিত করা যেতে পারে

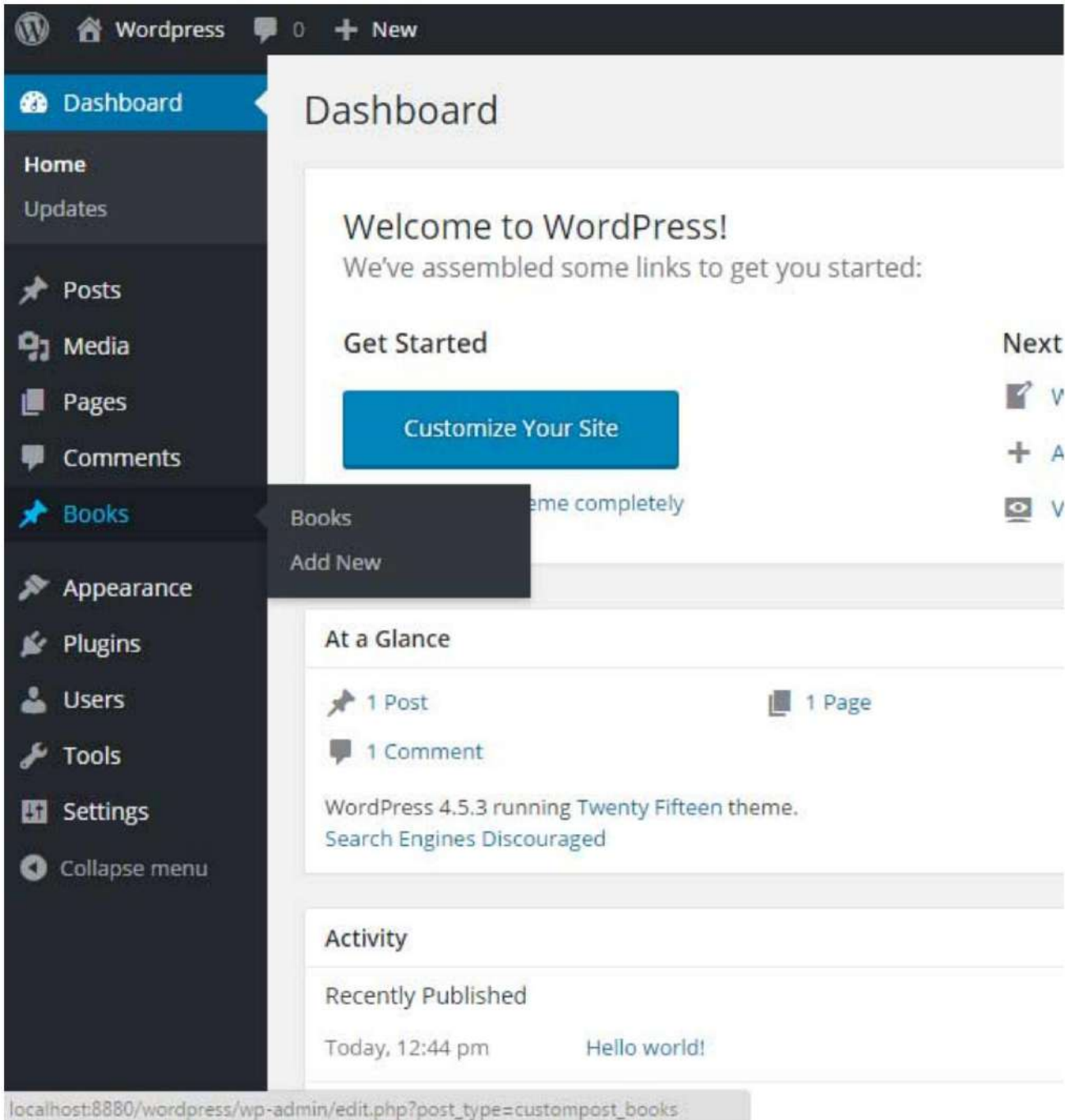
```
ফাংশন create_bookpostype() { $args = array(

    'পাবলিক' => সত্য, 'লেবেল' =>
    অ্যারে( 'নাম' => __( 'বই' ),
        'একবচন_নাম' => __( 'বই' )

    ),
);
register_post_type('custompost_books', $args);
}

add_action ('init', 'create_bookpostype');
```

এবং, যতটা সহজ, আপনার কাছে এখন একটি কাস্টম পোস্ট টাইপ নিবন্ধিত আছে।



এই স্লিপেটটি আপনার থিম functions.php ফাইলে বা একটি প্লাগইন কাঠামোর মধ্যে স্থাপন করা যেতে পারে।

প্রধান ক্যোয়ারীতে কাস্টম পোস্ট প্রকার যোগ করুন

একটি কাস্টম পোস্ট টাইপ নিবন্ধন করার অর্থ এই নয় যে এটি স্বয়ংক্রিয়ভাবে মূল ক্যোয়ারীতে যোগ হয়ে যাবে। প্রধান ক্যোয়ারীতে কাস্টম পোস্টের ধরন যোগ করতে আপনাকে pre\_get\_posts ফিল্টার ব্যবহার করতে হবে।

```
// হোম পেজে 'পোস্ট' এবং 'বুক' কাস্টম পোস্ট প্রকারের পোস্টগুলি দেখান add_action( 'pre_get_posts',  
'add_my_post_types_to_query');  
  
ফাংশন add_my_post_types_to_query( $query ) if ( is_home() && $query->is_main_query() )  
{  
    $query->set( 'post_type', array('post', 'book' ) );  
    ফেরত $query;  
}
```

## প্রধান RSS ফিডে কাস্টম পোস্টের ধরন যোগ করা হচ্ছে

একটি কাস্টম পোস্ট টাইপ নিবন্ধন করার অর্থ এই নয় যে এটি প্রধান RSS ফিডে যোগ করা হবে।

স্বয়ংক্রিয়ভাবে। প্রধান RSS ফিডে কাস্টম পোস্ট প্রকার যোগ করতে আপনাকে অনুরোধ ফিল্টার ব্যবহার করতে হবে।

```
// প্রধান RSS ফিডে 'বই' কাস্টম পোস্ট প্রকার যোগ করুন
ফাংশন add_book_post_types_to_rss($qv) {
    যদি (isset($qv['feed']) && isset($qv['post_type']))
        $qv['post_type'] = অ্যারে('পোস্ট', 'বই');
    ফেরত $qv;
}
add_filter('request', 'add_book_post_types_to_rss');
```

## কাস্টম পোস্ট টাইপ নিবন্ধন করুন

```
যদি ( ! function_exists('products_post_type') )
{
    ফাংশন পণ্য_পোস্ট_টাইপ() {
        $লেবেল = অ্যারে(
            'নাম' => _x( 'পণ্য', 'পোস্ট টাইপ সাধারণ নাম', 'টেক্সট_ডোমেইন' ),
            'একবচন_নাম' => _x( 'পণ্য', 'পোস্ট টাইপ সিস্টেম নাম', 'টেক্সট_ডোমেইন' ),
            'নাম_অ্যাডমিন_বার' => __( 'পণ্য', 'টেক্সট_ডোমেইন' ),
            'আর্কাইভ' => __( 'আইটেম আর্কাইভস', 'টেক্সট_ডোমেইন' ),
            'গুণাবলী' => __( 'আইটেম অ্যাট্রিবিউটস', 'টেক্সট_ডোমেইন' ),
            'parent_item_colon' => __( 'মূল পণ্য:', 'টেক্সট_ডোমেইন' ),
            'add_new_item' => __( 'সমস্ত পণ্য', 'টেক্সট_ডোমেইন' ),
            'add_new' => __( 'নতুন পণ্য যোগ করুন', 'টেক্সট_ডোমেইন' ),
            'edit_item' => __( 'নতুন পণ্য', 'টেক্সট_ডোমেইন' ),
            'update_item' => __( 'নতুন আইটেম', 'টেক্সট_ডোমেইন' ),
            'view_item' => __( 'পণ্য সম্পাদনা করুন', 'টেক্সট_ডোমেইন' ),
            'view_items' => __( 'আপডেট পণ্য', 'টেক্সট_ডোমেইন' ),
            'search_items' => __( 'পণ্য দেখুন', 'টেক্সট_ডোমেইন' ),
            'not_found' => __( 'আইটেম দেখুন', 'টেক্সট_ডোমেইন' ),
            'not_found_in_trash' => __( 'সার্চ প্রোডাক্ট', 'টেক্সট_ডোমেইন' ),
            'dim_feature' => __( 'কোন পণ্য পাওয়া যায়নি', 'টেক্সট_ডোমেইন' ),
            'dim_feature' . 'dim_feature' => __( 'ট্র্যাশে কোনো পণ্য পাওয়া যায়নি', 'টেক্সট_ডোমেইন' ),
            'বৈশিষ্ট্যযুক্ত ছবি সরান', => __( 'বৈশিষ্ট্যযুক্ত চিত্র', 'টেক্সট_ডোমেইন' ),
            'text_domain' ), => __( 'বৈশিষ্ট্যযুক্ত চিত্র সেট করুন', 'টেক্সট_ডোমেইন' ),
            'ব্যবহার_বিশিষ্ট_চিত্র' => __( 'বৈশিষ্ট্যযুক্ত চিত্র হিসাবে ব্যবহার করুন', 'টেক্সট_ডোমেইন' ),
            'আইটেম-এ_ডোকান' => __( 'আইটেমটিতে ডোকান', 'টেক্সট_ডোমেইন' ),
            'uploaded_to_this_item' => __( 'এই আইটেমটিতে আপলোড করা হয়েছে', 'টেক্সট_ডোমেইন' ),
            'items_list' => __( 'আইটেম তালিকা', 'টেক্সট_ডোমেইন' ),
            'items_list_navigation' => __( 'আইটেম তালিকা নেভিগেশন', 'টেক্সট_ডোমেইন' ),
            'ফিল্টার_আইটেম_তালিকা' => __( 'ফিল্টার আইটেম তালিকা', 'টেক্সট_ডোমেইন' ),
        );
        $args = অ্যারে(
            'লেবেল' => __( 'পণ্য', 'টেক্সট_ডোমেইন' ),
            'বর্ণনা' => __( 'পণ্য তথ্য পৃষ্ঠা', 'টেক্সট_ডোমেইন' ),
            'লেবেল' => $লেবেল,
            'সমর্থন করে' => 'মন্তব্য', => অ্যারে ( 'শিরোনাম', 'সম্পাদক', 'উদ্ধৃতি', 'লেখক', 'থাম্বনেল',
            'কাস্টম-ক্ষেত্র', ),
            'শ্রেণীবিন্যাস' => অ্যারে ( 'বিভাগ', 'পোস্ট_ট্যাগ' ),
            'অনুক্রমিক' => মিথ্যা,
            'পাবলিক' => সত্য,
            'শো_ইউআই' => সত্য,
        );
    }
}
```

```

        'শো_ইন_মেনু' 'মেনু_পজিশন'           => সত্য,
        'মেনু_আইকন'                         => 5,
        'শো_ইন_অ্যাডমিন_বার'               => 'ড্যাশিকন-পণ্য',
        'শো_ইন_নেভ_মেনু'                   => সত্য,
                                                => সত্য,
        'ক্যান_রপ্তানি_করতে'                 => সত্য,
        পারে 'আর্কাইভ_আছে'               => সত্য,
        'exclude_from_search' => মিথ্যা,
        'publicly_queryable' 'capability_type' => সত্য,
        'show_in_rest'                       => 'পৃষ্ঠা',
                                                => সত্য,
    );
    register_post_type('product', $args);
}
add_action('init', 'products_post_type', 0);
}

```

## টোয়েন্টি ফিফটিন ওয়ার্ডপ্রেস থিম ব্যবহার করে কাস্টম পোস্ট টাইপ

আপনি ফাংশনের জন্য যেকোনো নাম ব্যবহার করতে পারেন।

```

ফাংশন custom_postype(){
    register_post_type('cus_post', array(

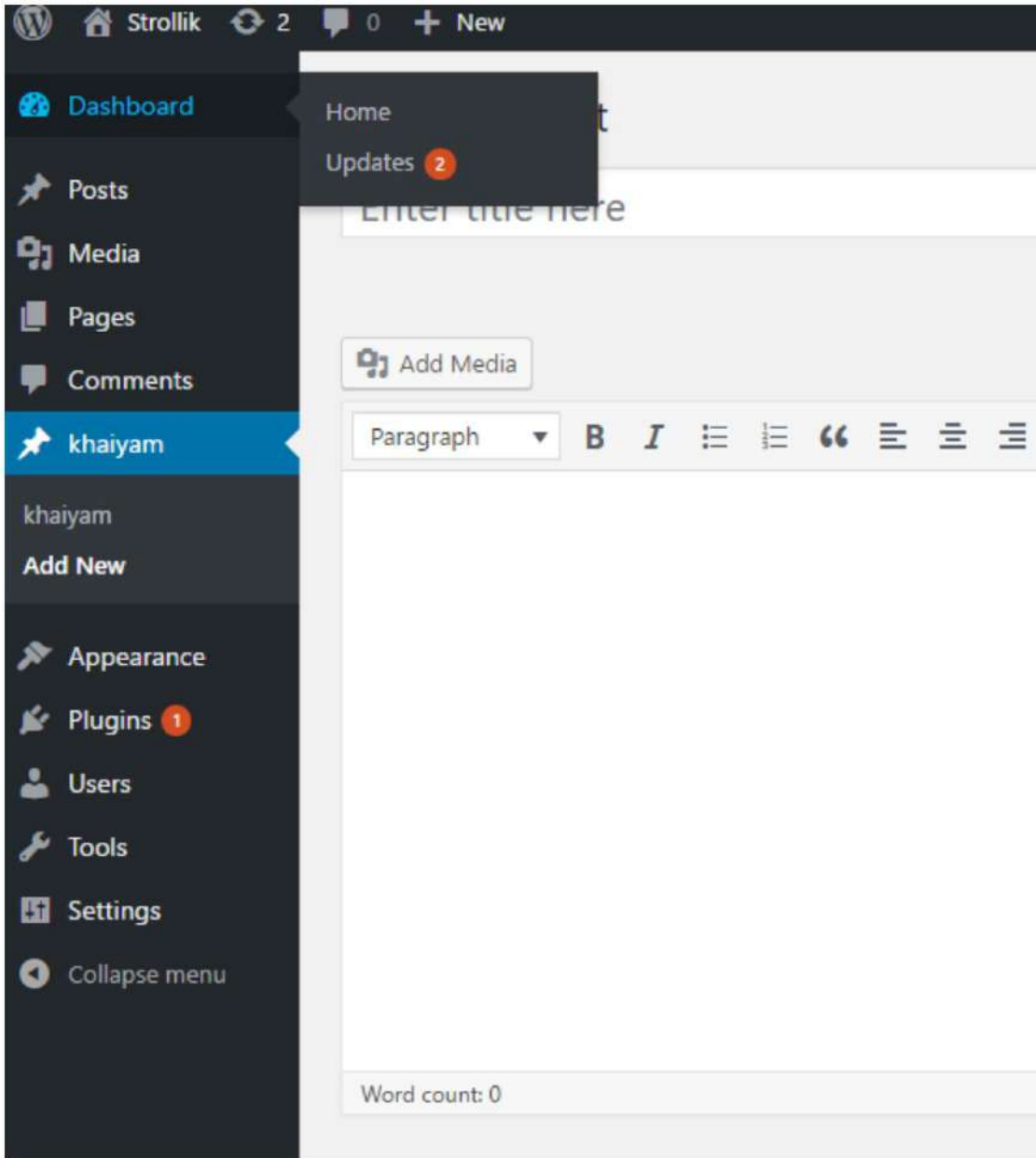
        'লেবেল' => অ্যারে(
            'name' => 'খাইয়াম'// আপনি আপনার ব্যবহারকারীদের জন্য মেনুতে দেখাতে চান এমন যেকোনো নাম ব্যবহার করুন
        ),
        'public' => সত্য, // ** আবশ্যিক
        'supports' => অ্যারে('শিরোনাম', 'সম্পাদক', 'থাম্বনেইল')// বৈশিষ্ট্যগুলি যা আপনি আপনার উপর প্রদান করতে চান
    পোস্ট
    ));
}
add_action('after_setup_theme', 'custom_postype');

```

বা

```
add_action('init', 'custom_postype');
```

আপনি যে কোনো ছক ব্যবহার করতে পারেন তবে অবশ্যই তাদের আলাদা অর্থ এবং ব্যবহার রয়েছে।



ডিফল্ট অনুসন্ধানে কাস্টম পোস্ট প্রকার

আপনি ডিফল্ট ওয়ার্ডপ্রেস অনুসন্ধানে কাস্টম পোস্ট টাইপ পোস্ট যোগ করতে পারেন, থিম functions.php-এ নীচের কোড যোগ করুন

```
ফাংশন my_search_filter($query) { if ( !is_admin() && $query->is_main_query() ) {  
    যদি ($query->is_search) { $query->set('post_type', array('news','post','article' ));  
    }  
}  
  
} add_action('pre_get_posts','my_search_filter');
```

# অধ্যায় 19: কাস্টমাইজার বেসিকস (প্যানেল যোগ করুন, বিভাগ, সেটিং, নিয়ন্ত্রণ)

## উদাহরণ

একটি কাস্টমাইজার প্যানেল যোগ করুন

```
<?php
/**
 * প্যানেল: WPC কাস্টমাইজ
 *
 * মৌলিক নিয়ন্ত্রণ সহ মৌলিক কাস্টমাইজার প্যানেল।
 *
 * @ থেকে * @ 1.0.0
 প্যাকেজ WPC
 */

// সরাসরি অ্যাক্সেস করা হলে প্রস্থান করুন।
যদি (! সংজ্ঞায়িত ('ABSPATH')) {

}

// কাস্টমাইজ ফাংশন।
যদি (! function_exists( 'wpc_panel_wpcustomize' )) {
    // কাস্টমাইজ রেজিস্টার অ্যাকশন।
    add_action( 'কাস্টমাইজ_রেজিস্টার', 'wpc_panel_wpcustomize' );
    /**
     * প্যানেল কাস্টমাইজ করুন।
     *
     * মৌলিক নিয়ন্ত্রণ সহ একটি প্যানেল, বিভাগ যোগ করে।
     *
     * @param অবজেক্ট WP_Customize $wp_customize WP_Customize_Manager ক্লাসের উদাহরণ।
     * @ ১.০.০ থেকে
     */
    ফাংশন wpc_panel_wpcustomize( $wp_customize ) {
        // প্যানেল: মৌলিক।
        $wp_customize->add_panel( 'wpc_panel_wpcustomize', অ্যারে(
            'অগ্রাধিকার' => 10,
            'শিরোনাম' => __('WPC কাস্টমাইজ প্যানেল শিরোনাম', 'WPC'),
            'বর্ণনা' 'ক্ষমতা' => __('প্যানেল বিবরণ', 'WPC'),
            => 'সম্পাদনা_থিম_বিকল্প'
        ));
    }
}
```

মৌলিক সেটিংস এবং তাদের নিয়ন্ত্রণ সহ একটি কাস্টমাইজার বিভাগ যোগ করুন

প্যানেলে বিভাগ থাকতে পারে, বিভাগে সেটিংস থাকতে পারে এবং সেটিংসে নিয়ন্ত্রণ থাকতে পারে। সেটিংস হল ডাটাবেসে সংরক্ষিত, যখন নির্দিষ্ট সেটিংসের জন্য নিয়ন্ত্রণ শুধুমাত্র তাদের প্রদর্শনের জন্য ব্যবহৃত হয় ব্যবহারকারীর সাথে সংশ্লিষ্ট সেটিং।

এই কোডটি উপরে থেকে প্যানেলে একটি মৌলিক বিভাগ তৈরি করে। ভিতরে কয়েকটি মৌলিক সেটিংস আছে নিয়ন্ত্রণ করে

```

<?php
/**
 * বিভাগ: মৌলিক
 *
 * মৌলিক নিয়ন্ত্রণ সহ মৌলিক কাস্টমাইজার বিভাগ।
 *
 * @ থেকে 1.0.0
 * @ প্যাকেজ WPC
 */

// সরাসরি অ্যাক্সেস করা হলে প্রস্থান করুন।
যদি (! সংজ্ঞায়িত ('ABSPATH') প্রস্থান করে; ) {

}

// কাস্টমাইজ ফাংশন।
যদি (! function_exists( 'wpc_customize_panel_basic' ) ) {
    // কাস্টমাইজ রেজিস্টার অ্যাকশন।
    add_action( 'কাস্টমাইজ_রেজিস্টার', 'wpc_customize_panel_basic' );

    /**
     * প্যানেল কাস্টমাইজ করুন।
     *
     * মৌলিক নিয়ন্ত্রণ সহ একটি প্যানেল, বিভাগ যোগ করে।
     *
     * @param অবজেক্ট WP_Customize $wp_customize WP_Customize_Manager ক্লাসের উদাহরণ।
     * @ ১.০.০ থেকে
     */
    ফাংশন wpc_customize_panel_basic( $wp_customize ) {
        // বিভাগ: মৌলিক।
        $wp_customize->add_section( 'wpc_section_basic', array(
            'অগ্রাধিকার' => 10,
            'প্যানেল' => 'wpc_panel_wpcustomize',
            'শিরোনাম' => __( 'মৌলিক বিভাগের শিরোনাম', 'WPC' ),
            'বর্ণনা' => __( 'বিভাগের বর্ণনা', 'WPC' ),
            'সম্পাদনা_খিম_বিকল্প' => 'সম্পাদনা_খিম_বিকল্প'
        ) );

        // সেটিং: পাঠ্য।
        $wp_customize->add_setting( 'wpc_text', অ্যারে(
            'টাইপ' => 'খিম_মোড',
            'ডিফল্ট' => 'প্লেসহোল্ডার',
            'পরিবহন' => 'সম্পাদনা_খিম_বিকল্প', // বিকল্প: রিফ্রেশ বা পোস্ট মেসেজ।
            'স্যানিটাইজ_কলব্যাক' => 'সম্পাদনা_খিম_বিকল্প',
            'esc_attr' => 'esc_attr'
        ) );

        // নিয়ন্ত্রণ: পাঠ্য।
        $wp_customize->add_control( 'wpc_text', অ্যারে(
            'লেবেল' => __( 'টেক্সট', 'WPC' ),
            'description' => __( 'বিবরণ', 'WPC' ),
            'অধ্যায়' => 'wpc_section_basic',
            'টাইপ' => 'পাঠ্য'
        ) );

        // সেটিং: Textarea.
        $wp_customize->add_setting( 'wpc_textarea', অ্যারে(
            'টাইপ' => 'খিম_মোড',
            'ডিফল্ট' => 'প্লেসহোল্ডার টেক্সটেরিয়া',

```

```

'পরিবহন' 'ক্ষমতা' => 'রিফ্রেশ', // বিকল্প: রিফ্রেশ বা পোস্ট মেসেজ।
'স্যানিটাইজ_কলব্যাক' => 'সম্পাদনা_থিম_বিকল্প',
=> 'exc_textarea'

);

// নিয়ন্ত্রণ: Textarea.
$wp_customize->add_control( 'wpc_textarea', অ্যারে(
'লেবেল' => __('Textarea', 'WPC'),
'description' => __('বিবরণ', 'WPC'),
'অধ্যায়' => 'wpc_section_basic',
'টাইপ' => 'টেক্সটেরিয়া'
);

// সেটিং: চেকবক্স।
$wp_customize->add_setting( 'wpc_checkbox', array(
'টাইপ' => 'থিম_মোড',
'ডিফল্ট' => 'সক্ষম',
'পরিবহন' 'ক্ষমতা' => 'রিফ্রেশ', // বিকল্প: রিফ্রেশ বা পোস্ট মেসেজ।
'sanitize_callback' => 'সম্পাদনা_থিম_বিকল্প',
কাস্টমাইজার-sanitization.php => 'wpc_sanitize_checkbox' // কাস্টম ফাংশন ইন
ফাইল।
);

// নিয়ন্ত্রণ: চেকবক্স।
$wp_customize->add_control( 'wpc_checkbox', array(
'লেবেল' => __('চেকবক্স', 'WPC'),
'description' => __('বিবরণ', 'WPC'),
'অধ্যায়' => 'wpc_section_basic',
'টাইপ' => 'চেকবক্স'
);

// সেটিং: রেডিও।
$wp_customize->add_setting( 'wpc_radio', array(
'টাইপ' => 'থিম_মোড',
'ডিফল্ট' => 'চালু',
'পরিবহন' 'ক্ষমতা' => 'রিফ্রেশ', // বিকল্প: রিফ্রেশ বা পোস্ট মেসেজ।
'sanitize_callback' => 'সম্পাদনা_থিম_বিকল্প',
sanitization.php ফাইল => 'wpc_sanitize_select', // কাস্টমাইজারে কাস্টম ফাংশন
);

// নিয়ন্ত্রণ: রেডিও।
$wp_customize->add_control( 'wpc_radio', array(
'লেবেল' => __('রেডিও', 'WPC'),
'description' => __('বিবরণ', 'WPC'),
'অধ্যায়' => 'wpc_section_basic',
'টাইপ' => 'রেডিও',
'চয়েস' => অ্যারে(
'enable' => 'সক্রিয় করুন',
'disable' => 'অক্ষম করুন'
)
);

// সেটিং: নির্বাচন করুন।
$wp_customize->add_setting( 'wpc_select', array(
'টাইপ' => 'থিম_মোড',
'ডিফল্ট' => 'সক্ষম',
'পরিবহন' 'ক্ষমতা' => 'রিফ্রেশ', // বিকল্প: রিফ্রেশ বা পোস্ট মেসেজ।
'sanitize_callback' => 'সম্পাদনা_থিম_বিকল্প',
sanitization.php ফাইল => 'wpc_sanitize_select' // কাস্টমাইজারে কাস্টম ফাংশন
);

```

```
);  
  
// নিয়ন্ত্রণ: নির্বাচন করুন।  
$wp_customize->add_control( 'wpc_select', array(  
    'লেবেল' => __( 'নির্বাচন', 'WPC' ),  
    'description' => __( 'বিবরণ', 'WPC' ), 'বিভাগ'  
        => 'wpc_section_basic',  
    'টাইপ' => 'নির্বাচন',  
    'চয়েস' => অ্যারে(  
        'enable' => 'সক্রিয় করুন',  
        'disable' => 'অক্ষম করুন'  
    )  
);  
}  
}
```

# অধ্যায় 20: কাস্টমাইজার হ্যালো ওয়ার্ল্ড

## পরামিতি

### পরামিতি বিবরণ

mytheme আপনার থিমের (বা চাইল্ড থিম) জন্য একটি অনন্য শনাক্তকারী। এটি আপনার থিম স্লাগ হতে পারে

## উদাহরণ

### হ্যালো ওয়ার্ল্ড উদাহরণ

কাস্টমাইজারের মৌলিক ধারণা হল প্রশাসকরা তাদের সাইটের পরিবর্তনগুলি লাইভ প্রিভিউ করতে পারে এবং তারপরে স্থায়ীভাবে তাদের যোগ করতে পারে।

একটি থিমের functions.php ফাইলে নিম্নলিখিতটি কপি এবং পেস্ট করা যেতে পারে

- মাই ফার্স্ট সেকশন নামে একটি কাস্টমাইজার বিভাগ যোগ করুন •
- হ্যালো ওয়ার্ল্ড কালার নামে একটি কাস্টমাইজার সেটিং যুক্ত করুন যা অ্যাডমিনকে একটি রঙ চয়ন করতে দেয়। . সেটিংটি
- `<head>` এর শেষে একটি `<style>` ট্যাগে রাখা হবে।

```
ফাংশন mytheme_customize_register( $wp_customize )  
  
    {  
  
        $wp_customize->add_section( 'my_first_section_id' , array(  
            'শিরোনাম' => __( 'আমার প্রথম বিভাগ', 'মিথিম' ),  
            'অগ্রাধিকার' => 30,  
        ));  
  
        $wp_customize->add_setting( 'hello_world_color' , অ্যারে(  
            'ডিফল্ট' => '#000000', 'ট্রান্সপোর্ট'  
            => 'রিফ্রেশ',  
        ));  
  
        $wp_customize->add_control( নতুন WP_Customize_Color_Control( $wp_customize, 'link_color', array(  
            'লেবেল' => __( 'Hello World Color', 'mytheme' ), => 'my_first_section_id',  
            'অধ্যায়'  
            'settings' => 'hello_world_color', ));  
        )  
  
    } অ্যাড_অ্যাকশন ( 'কাস্টমাইজ_রেজিস্টার', 'মিথিম_কাস্টমাইজ_রেজিস্টার' );  
  
ফাংশন mytheme_customize_css() {  
  
    ?>  
    <style type="text/css">
```

```
.hello-world { color: #<?php echo get_theme_mod('hello_world_color', '000000'); ?>; } </style> <?php  
}  
add_action( 'wp_head', 'mytheme_customize_css');
```

---

## অধ্যায় 21: ডিবাগিং

ভূমিকা

[https://codex.wordpress.org/Debugging\\_in\\_WordPress](https://codex.wordpress.org/Debugging_in_WordPress)

ডিবাগিং পিএইচপি কোড যেকোন প্রকল্পের অংশ, তবে ওয়ার্ডপ্রেস নির্দিষ্ট ডিবাগ সিস্টেমের সাথে আসে যা প্রক্রিয়াটিকে সহজ করার পাশাপাশি মূল, প্লাগইন এবং থিম জুড়ে কোড প্রমিত করার জন্য ডিজাইন করা হয়েছে।

### মন্তব্য

ওয়ার্ডপ্রেসে ডিবাগিংয়ের জন্য প্লাগইন:

- <https://wordpress.org/plugins/query-monitor/> • <https://wordpress.org/plugins/debug-bar/> • <https://wordpress.org/plugins/debug-bar-console/> • <https://wordpress.org/plugins/kint-debugger/> • <https://wordpress.org/plugins/rest-api-console/>

### উদাহরণ

#### WP\_DEBUG

**WP\_DEBUG** এটি একটি PHP ধ্রুবক (একটি স্থায়ী গ্লোবাল ভেরিয়েবল) যা ওয়ার্ডপ্রেস জুড়ে "ডিবাগ" মোড ট্রিগার করতে ব্যবহার করা যেতে পারে। এটিকে ডিফল্টরূপে মিথ্যা বলে ধরে নেওয়া হয় এবং সাধারণত wp-config.php এ সত্যে সেট করা হয় ওয়ার্ডপ্রেসের উন্নয়ন অনুলিপি ফাইল।

```
সংজ্ঞায়িত করুন ('WP_DEBUG', সত্য); সংজ্ঞায়িত করুন ('WP_DEBUG', মিথ্যা);
```

#### WP\_DEBUG\_LOG

**WP\_DEBUG\_LOG** WP\_DEBUG-এর একটি সঙ্গী যা /wp-content/ ডিরেক্টরির ভিতরে একটি debug.log লগ ফাইলে সমস্ত ত্রুটি সংরক্ষিত করে। আপনি যদি পরে সমস্ত নোটিশ পর্যালোচনা করতে চান বা অফ-স্ক্রীনে জেনারেট করা নোটিশগুলি দেখতে চাইলে এটি দরকারী (যেমন AJAX অনুরোধ বা wp-cron রানের সময়)।

```
//সক্ষম করুন  
সংজ্ঞায়িত করুন ('WP_DEBUG_LOG', সত্য);  
  
// নিষ্ক্রিয় করুন  
সংজ্ঞায়িত করুন ('WP_DEBUG_LOG', মিথ্যা);
```

#### WP\_DEBUG\_DISPLAY

**WP\_DEBUG\_DISPLAY** WP\_DEBUG এর আরেকটি সহচর যা ডিবাগ বার্তাগুলি কিনা তা নিয়ন্ত্রণ করে

পৃষ্ঠাগুলির HTML এর ভিতরে দেখানো হয়েছে বা না। ডিফল্ট হল 'সত্য' যা তৈরি হওয়ার সাথে সাথে ক্রটি এবং সতর্কতা দেখায়। এটি মিথ্যাতে সেট করা সমস্ত ক্রটি লুকাবে। এটি WP\_DEBUG\_LOG এর সাথে ব্যবহার করা উচিত যাতে ক্রটিগুলি পরে পর্যালোচনা করা যায়।  
দ্রষ্টব্য: WP\_DEBUG\_DISPLAY কিছু করার জন্য, WP\_DEBUG অবশ্যই সক্ষম হতে হবে (সত্য)।

```
// define
enable('WP_DEBUG_DISPLAY', true);

// ডিজেবল
ডিফাইন('WP_DEBUG_DISPLAY', মিথ্যা);
```

## SCRIPT\_DEBUG

**SCRIPT\_DEBUG** এটি একটি সম্পর্কিত ধ্রুবক যা ওয়ার্ডপ্রেসকে সাধারণত লোড করা সংক্ষিপ্ত সংস্করণগুলির পরিবর্তে মূল CSS এবং জাভাস্ক্রিপ্ট ফাইলগুলির "dev" সংস্করণগুলি ব্যবহার করতে বাধ্য করবে। আপনি যখন কোনো বিল্ট-ইন .js বা .css ফাইলে পরিবর্তন পরীক্ষা করছেন তখন এটি কার্যকর। ডিফল্ট মিথ্যা।

```
//সক্ষম করুন
সংজ্ঞায়িত করুন ('SCRIPT_DEBUG', সত্য);

// নিষ্ক্রিয় করুন
সংজ্ঞায়িত করুন ('SCRIPT_DEBUG', মিথ্যা);
```

## সংরক্ষণ

SAVEQUERIES সংজ্ঞা একটি অ্যারেতে ডাটাবেস প্রশ্নগুলি সংরক্ষণ করে এবং সেই অ্যারেটি সেই প্রশ্নগুলি বিশ্লেষণ করতে সাহায্য করার জন্য প্রদর্শিত হতে পারে। সত্য হিসাবে সংজ্ঞায়িত ধ্রুবক প্রতিটি ক্যোয়ারী সংরক্ষিত হওয়ার কারণ, সেই ক্যোয়ারীটি কার্যকর করতে কতক্ষণ সময় নেয় এবং এটিকে কোন ফাংশন বলে। দ্রষ্টব্য: এটি আপনার সাইটের পারফরম্যান্সের উপর প্রভাব ফেলবে, তাই আপনি যখন ডিবাগ করছেন না তখন এটি বন্ধ করতে ভুলবেন না।

```
সংজ্ঞায়িত করুন ('সংরক্ষণ', সত্য);
```

অ্যারে সংরক্ষণ করা হয়

```
বিশ্বব্যাপী $wpdb->কোয়েরি;
```

## উদাহরণ wp-config.php এবং ডিবাগিংয়ের জন্য ভাল অনুশীলন

আপনার wp-config.php ফাইলে চোকানো নিম্নলিখিত কোডটি wp-content ডিরেক্টরিতে debug.log নামক একটি ফাইলে সমস্ত ক্রটি, নোটিশ এবং সতর্কতা লগ করবে। এটি ক্রটিগুলিও লুকিয়ে রাখবে যাতে তারা পৃষ্ঠা তৈরিতে বাধা না দেয়।

```
// WP_DEBUG মোড সক্ষম করুন
সংজ্ঞায়িত করুন ('WP_DEBUG', সত্য);

// wp-content/debug.log ফাইলে ডিবাগ লগিং সক্ষম করুন
সংজ্ঞায়িত করুন ('WP_DEBUG_LOG', সত্য);

// ক্রটি এবং সতর্কতা প্রদর্শন অক্ষম করুন
```

```
সংজ্ঞায়িত করুন ('WP_DEBUG_DISPLAY', মিথ্যা);
```

```
@ini_set('display_errors', 0);
```

```
// কোর JS এবং CSS ফাইলগুলির ডেভ সংস্করণ ব্যবহার করুন (কেবলমাত্র আপনি যদি এই মূল ফাইলগুলি পরিবর্তন করেন) সংজ্ঞায়িত করুন('SCRIPT_DEBUG', true);
```

ভালো অভ্যাস আপনি যদি ডিবাগ লগে কাস্টম বার্তা যোগ করতে চান তাহলে আপনার প্লাগইন বা থিমে নিম্নলিখিত কোড যোগ করুন।

```
// পরীক্ষা করা হচ্ছে ফাংশন_এক্সিস্ট
যদি ( !function_exists('print_to_log') ) {
    ডিবাগিং চালু থাকলে //function debug.log এ একটি বার্তা লেখে। ফাংশন print_to_log( $message )

    {
        যদি ( সত্য === WP_DEBUG ) {
            যদি ( is_array( $message ) || is_object( $message ) ) {
                error_log( print_r($message, true ));
            } অন্য {
                error_log($message);
            }
        }
    }
}
```

## একটি পৃথক ফাইলে লগ দেখুন

যখন আপনার একটি AJAX কল থাকে, তখন কলব্যাক ফাংশনের ভিতর থেকে একটি লগ পাওয়া অত্যন্ত কঠিন।

কিন্তু আপনি যদি ডিবাগিং সক্ষম করেন

```
define('WP_DEBUG', true);
```

এবং তারপর যে যোগ করুন

```
ini_set('log_errors',TRUE); ini_set('error_reporting',
E_ALL); ini_set('error_log', dirname(__FILE__) . '/error_log.txt');
```

আপনার রুট ফোল্ডারে একটি error.log.txt ফাইল থাকবে যেখানে আপনার সমস্ত লগ অবস্থিত। আপনি এমনকি তাদের লগ করতে পারেন

```
error_log( print_r( 'আমি যা পরীক্ষা করতে চাই তা এখানে যায়', সত্য) );
```

আপনার কোডের ভিতরে। এটি আপনার জীবনকে অনেক সহজ করে তুলবে।

## অধ্যায় 22: সারিবদ্ধ স্ক্রিপ্ট

### বাক্য গঠন

- wp\_enqueue\_script( \$handle, \$src, \$deps, \$ver, \$in\_footer)

### পরামিতি

পরামিতি বিবরণ	
\$হ্যান্ডেল	(স্ট্রিং) (প্রয়োজনীয়) স্ক্রিপ্টের নাম। অনন্য হতে হবে।
\$src	(স্ট্রিং) (ঐচ্ছিক) স্ক্রিপ্টের সম্পূর্ণ URL, বা ওয়ার্ডপ্রেস রুট ডিরেক্টরির সাথে সম্পর্কিত স্ক্রিপ্টের পথ। ডিফল্ট মান: মিথ্যা
\$deps	(অ্যারে) (ঐচ্ছিক) নিবন্ধিত স্ক্রিপ্টের একটি অ্যারে এই স্ক্রিপ্টের উপর নির্ভর করে। ডিফল্ট মান: অ্যারে()
\$ver	(স্ট্রিং   বুল   নাল) (ঐচ্ছিক) স্ট্রিং নির্দিষ্ট করে স্ক্রিপ্ট সংস্করণ নম্বর, যদি এটিতে একটি থাকে, যা ক্যাশে বাস্টিংয়ের উদ্দেশ্যে একটি কোয়ারী স্ট্রিং হিসাবে URL-এ যোগ করা হয়। সংস্করণ মিথ্যা সেট করা হলে, বর্তমান ইনস্টল করা ওয়ার্ডপ্রেস সংস্করণের সমান একটি সংস্করণ নম্বর স্বয়ংক্রিয়ভাবে যোগ করা হয়। নাল সেট করা হলে, কোনো সংস্করণ যোগ করা হয় না। ডিফল্ট মান: মিথ্যা
\$in_footer	(bool) (ঐচ্ছিক) <head> এর পরিবর্তে </body> এর আগে স্ক্রিপ্ট সারিবদ্ধ করা হবে কিনা। ডিফল্ট মান: মিথ্যা

### উদাহরণ

#### functions.php-এ স্ক্রিপ্ট সারিবদ্ধ করা

আপনি যদি আপনার থিমের js/ ফোল্ডারে অবস্থিত custom.js স্ক্রিপ্ট যোগ করতে চান তবে আপনাকে এটি সারিবদ্ধ করতে হবে।

functions.php এ যোগ করুন

```
<?php

add_action( 'after_setup_theme', 'yourtheme_theme_setup' );

যদি ( ! function_exists( 'yourtheme_theme_setup' ) ফাংশন yourtheme_theme_setup() {      ) {

    add_action( 'wp_enqueue_scripts', 'yourtheme_scripts' ); add_action( 'admin_enqueue_scripts',
    'yourtheme_admin_scripts' );

}

}
```

```

যদি !function_exists('yourtheme_scripts') ) {
    ( ফাংশন yourtheme_scripts() {

        wp_enqueue_script('yourtheme_custom', get_template_directory_uri().'/js/custom.js', array('jquery'), '1.0.0', true );

    }
}

যদি ( !function_exists('yourtheme_admin_scripts') ফাংশন yourtheme_admin_scripts() { ) {

        wp_enqueue_script('yourtheme_custom', get_template_directory_uri().'/js/custom.js',
অ্যারে ('jquery-ui-autocomplete', 'jquery'), '1.0.0', true );

    }
}

```

## শুধুমাত্র IE এর জন্য সারিবদ্ধ স্ক্রিপ্ট

```

add_action( 'wp_enqueue_scripts', 'enqueue_my_styles_and_scripts');

/**
 * শর্তসাপেক্ষে সারিবদ্ধ স্ক্রিপ্ট (বা শৈলী)।
 *
 * বিশেষ করে IE এর জন্য স্ক্রিপ্ট (বা স্টাইলশীট) লোড করুন। IE10 এবং তার উপরে * স্ট্যান্ডার্ড মোডে শর্তসাপেক্ষ মন্তব্য সমর্থন করে না।
 *
 * @link https://gist.github.com/wpscholar/4947518 * @link https://msdn.microsoft.com/en-us/library/ms537512(v=vs.85).aspx */
ফাংশন enqueue_my_styles_and_scripts() {

    // ইন্টারনেট এক্সপ্লোরার HTML5 সমর্থন
    wp_enqueue_script( 'html5shiv', get_template_directory_uri().'/js/html5shiv.js', array(), '3.7.3', মিথ্যা); wp_script_add_data( 'html5shiv', 'শর্তাধীন', 'lt IE 9');

    // ইন্টারনেট এক্সপ্লোরার 8 মিডিয়া ক্যায়াবী সমর্থন wp_enqueue_script('respond',
    get_template_directory_uri().'/js/respond.js', array(), '1.4.2', false); wp_script_add_data( 'প্রতিক্রিয়া', 'শর্তাধীন', 'lt IE 9');

}

```

## নির্দিষ্ট পৃষ্ঠাগুলির জন্য শর্তসাপেক্ষে স্ক্রিপ্টগুলি সারিবদ্ধ করা

আপনি আপনার ওয়েবসাইটের নির্দিষ্ট পৃষ্ঠাগুলিতে স্ক্রিপ্টগুলি সারিবদ্ধ করতে ওয়ার্ডপ্রেসে শর্তসাপেক্ষ অপারেটর ব্যবহার করতে পারেন।

```

ফাংশন load_script_for_single_post(){ if(is_single()){ wp_enqueue_script(

        'some',
        get_template_directory_uri().'/js/some.js', array('jquery'), '1.0.0',

```

```
        );  
    }  
}  
add_action('wp_enqueue_scripts','load_script_for_single_post');
```

উপরের উদাহরণে, যদি বর্তমান ওয়েবপৃষ্ঠাটি একক পোস্ট হয়, তাহলে স্ক্রিপ্টটি সারিবদ্ধ হবে। অন্যথায় wp\_enqueue\_script ফাংশন কার্যকর করা হবে না।

# অধ্যায় 23: সারিবদ্ধ শৈলী

## বাক্য গঠন

1. `wp_enqueue_style($handle, $src, $dependency, $version, $media);`

## পরামিতি

পরামিতি বিবরণ	
\$হ্যান্ডেল	(স্ট্রিং) (প্রয়োজনীয়) স্টাইলশীটের অনন্য নাম।
\$src	(স্ট্রিং) (ঐচ্ছিক) স্টাইলশীটের URL যা লিঙ্ক ট্যাগের src অ্যাট্রিবিউটের ভিতরে ব্যবহার করা হবে।
\$deps	(স্ট্রিং) (ঐচ্ছিক) স্টাইলশীটের একটি অ্যারে এই স্টাইলশীটটির উপর নির্ভর করে।
\$ver	(স্ট্রিং) (ঐচ্ছিক) স্টাইলশীটের স্টাইলশীট সংস্করণ নির্দিষ্ট করে।
\$মিডিয়া	(স্ট্রিং) (ঐচ্ছিক) মিডিয়া যার জন্য এই স্টাইলশীট তৈরি করা হয়েছে। যেমন 'সমস্ত', 'প্রিন্ট', 'স্ক্রিন' ইত্যাদি

## উদাহরণ

### নির্ভরতা হিসাবে অন্য CSS ফাইলের সাথে অভ্যন্তরীণ CSS ফাইল সহ

```
ফাংশন themeSlug_enqueue_scripts() {  
    wp_enqueue_style('themeSlug-reset', get_template_directory_uri() .'/css/reset.css',  
    '1.0.0');  
    wp_enqueue_style('themeSlug-style', get_template_directory_uri() .'/style.css',  
    'themeSlug-রিসেট', '1.0.0'); }  
  
add_action('wp_enqueue_scripts', 'themeSlug_enqueue_scripts');
```

### অভ্যন্তরীণ CSS ফাইল সহ

এই ক্ষেত্রে style.css থিমের ফোল্ডারের রুটে অবস্থিত

```
ফাংশন themeSlug_enqueue_scripts() {  
    wp_enqueue_style('themeSlug-style', get_template_directory_uri() .'/style.css', '1.0.0');  
  
} add_action('wp_enqueue_scripts', 'themeSlug_enqueue_scripts');
```

### বাহ্যিক CSS ফাইল সহ

এই উদাহরণে আমরা ফন্টের অসাধারণ আইকন ফন্ট অন্তর্ভুক্ত করতে চাই

```

ফাংশন themeSlug_enqueue_scripts() {
    wp_enqueue_style( 'font-Awesome', '//cdnjs.cloudflare.com/ajax/libs/font
awesome/4.6.3/css/font-awesome.css');
}
add_action('wp_enqueue_scripts', 'themeSlug_enqueue_scripts');

```

## শুধুমাত্র IE এর জন্য স্টাইলশীট সারিবদ্ধ করুন

```

add_action( 'wp_enqueue_scripts', 'enqueue_my_styles_and_scripts');

/**
 * শর্তসাপেক্ষে সারিবদ্ধ শৈলী (বা স্ক্রিপ্ট)।
 *
 * বিশেষ করে IE এর জন্য স্টাইলশীট (বা স্ক্রিপ্ট) লোড করুন। IE10 এবং তার উপরে * স্ট্যান্ডার্ড মোডে শর্তসাপেক্ষ মন্তব্য সমর্থন করে না।
 *
 * @link https://gist.github.com/wpscholar/4947518 * @link https://msdn.microsoft.com/
en-us/library/ms537512(v=vs.85).aspx */

ফাংশন enqueue_my_styles_and_scripts() {

    // ইন্টারনেট এক্সপ্লোরার নির্দিষ্ট স্টাইলশীট। wp_enqueue_style( 'thename-ie',
get_stylesheet_directory_uri() . '/css/ie.css', অ্যারে(
'কুড়ি পনের-শৈলী', '20141010' ); wp_style_add_data( 'থিমের নাম-অর্থং',
'শর্তাধীন', 'lte IE 9');

    // ইন্টারনেট এক্সপ্লোরার 7 নির্দিষ্ট স্টাইলশীট। wp_enqueue_style('thename-ie7',
get_stylesheet_directory_uri() . '/css/ie7.css', অ্যারে(
'কুড়ি পনের-শৈলী', '20141010' ); wp_style_add_data('thename-ie7',
'শর্তাধীন', 'lt IE 8');

}

```

## আপনার প্লাগইন ক্লাসের জন্য অভ্যন্তরীণ CSS ফাইল সহ

```

ক্লাস My_Plugin() { ফাংশন
    __construct() {
        add_action( 'wp_enqueue_scripts', array( $this, 'init_fe_assets' ) );
    }

    পাবলিক ফাংশন init_fe_assests() {
        wp_enqueue_style( 'my-plugin', plugin_dir_url( __FILE__ )
'assets/css/frontend/plugin.css', অ্যারে(), '0.0.1', সত্য );
    }
}

নতুন My_Plugin();

```

## বিকল্প স্টাইলশীট যোগ করুন

```

<?php wp_enqueue_style('থিম-ফাইল', get_template_directory_uri()।
'/path/to/additional/css'); wp_style_add_data('থিম-
ফাইল', 'alt', সত্য); wp_style_add_data('থিম-ফাইল', 'টাইটেল', __('থিম-ফাইল.সিএসএস',
'আপনার-থিম-নাম')); ?>

```



## অধ্যায় 24: ফাংশন : wp\_trim\_words()

### বাক্য গঠন

```
<?php $trimmed_text = wp_trim_words( $text, $num_words = 55, $more = null); ?>
```

### পরামিতি

পরামিতি বিবরণ	
\$টেক্সট	(স্ট্রিং) (প্রয়োজনীয়) পাঠ্য যা ছোট বা ছাঁটা হবে।
\$num_words	(পূর্ণসংখ্যা) (প্রয়োজনীয়) শব্দের সংখ্যা যেখানে পাঠ্য সীমাবদ্ধ থাকবে।
আরো \$	(স্ট্রিং) (ঐচ্ছিক) যদি \$টেক্সট ট্রিম করতে হয় তাহলে কি যোগ করতে হবে।

### উদাহরণ

#### পোস্টের বিষয়বস্তু ছাঁটাই

এই ফাংশনটি একটি নির্দিষ্ট সংখ্যক শব্দে পাঠ্যকে সংক্ষিপ্ত করে এবং সংক্ষিপ্ত পাঠ ফেরত দেয়।

```
<?php echo wp_trim_words( get_the_content(), 40, '...' ); ?>
```

উপরের উদাহরণে আমরা পোস্টের বিষয়বস্তু ফাংশনে প্রেরণ করছি। এটি বিষয়বস্তুর দৈর্ঘ্য 40 শব্দের মধ্যে সীমাবদ্ধ করবে এবং বাকি শব্দগুলিকে ছাঁটাই করবে।

# অধ্যায় 25: ফাংশন: add\_action()

## বাক্য গঠন

- `add_action( $tag, $function_to_add )` •
- `add_action( $tag, $function_to_add, $priority )` •
- `add_action( $tag, $function_to_add, $priority, $accepted_args )`

## পরামিতি

প্যারামিটার	বিস্তারিত
\$ট্যাগ	(স্ট্রিং) (প্রয়োজনীয়) কর্মের নাম যেখানে \$function_to_add হুক করা হয়েছে
\$function_to_add	(আহ্বানযোগ্য) (প্রয়োজনীয়) যে ফাংশনটিকে কল করা উচিত যখন \$tag দ্বারা নির্দেশিত ক্রিয়াটি কার্যকর করা হয়।
\$ অগ্রাধিকার	(int) (ঐচ্ছিক) ডিফল্ট মান: 10 একটি নির্দিষ্ট কর্মের সাথে যুক্ত ফাংশনগুলি যে ক্রমানুসারে সম্পাদিত হয় তা নির্দিষ্ট করতে ব্যবহৃত হয়। নিম্ন সংখ্যাগুলি পূর্বের নির্বাহের সাথে সামঞ্জস্যপূর্ণ, এবং একই অগ্রাধিকার সহ ফাংশনগুলি সেই ক্রমে সম্পাদিত হয় যে ক্রমে সেগুলিকে অ্যাকশনে যুক্ত করা হয়েছিল।
\$accepted_args	(int) (ঐচ্ছিক) ডিফল্ট মান: 1 ফাংশন গ্রহণ করে আর্গুমেন্টের সংখ্যা।

## মন্তব্য

`add_action()` ফাংশন একটি অ্যাকশন হুক তৈরি করে, একটি পিএইচপি ফাংশনকে একটি নির্দিষ্ট অ্যাকশন "ট্যাগ" বা নামের সাথে যুক্ত করে। একটি নির্দিষ্ট ট্যাগ সহ `do_action()` (বা `do_action_ref_array()`) কলের মাধ্যমে অ্যাকশনটি "ট্রিগার" হলে, সেই ট্যাগের সাথে "হুকড" সমস্ত ফাংশন চালানো হবে।

বেশিরভাগ ক্ষেত্রে, এই ফাংশনটি একটি থিমের `functions.php` ফাইলে বা একটি প্লাগইন ফাইলে ব্যবহার করা উচিত - অথবা অন্য একটি সোর্স ফাইল যেটির দ্বারা লোড করা হয়েছে।

এই ফাংশনটি প্লাগইন API এর একটি অংশ

## উদাহরণ

বেসিক অ্যাকশন হুক

`add_action()` এর সবচেয়ে মৌলিক প্রয়োগ হল একটি ওয়ার্ডপ্রেস ইন্সটলেশনের সোর্স-কোড-এ একটি নির্দিষ্ট স্থানে কার্যকর করার জন্য কাস্টম কোড যোগ করা - হয় ওয়ার্ডপ্রেসের কোর দ্বারা সরবরাহ করা অ্যাকশন ব্যবহার করে, অথবা তৃতীয় পক্ষের কোড দ্বারা তৈরি করা যেমন প্লাগইন এবং থিম

সাইটের <head></head> বিভাগে বিষয়বস্তু যোগ করতে - সাইটের কপিরাইট তথ্য কোথায় পাওয়া যাবে তা নির্দেশ করার জন্য একটি <link> মেটা উপাদান যোগ করতে বলুন - add\_action() একটি ফাংশন সংযুক্ত করতে ব্যবহার করা যেতে পারে যা প্রিন্ট করে 'wp\_head' অ্যাকশনের উপযুক্ত মার্কাআপ (যা যখন "ট্রিগার" করে ওয়ার্ডপ্রেস <head> বিভাগ তৈরি করে):

```
ফাংশন add_copyright_meta_link() {
    echo('<link rel="copyright" href="'. get_home_url() . '/copyright">');
}

add_action( 'wp_head', 'add_copyright_meta_link');
```

## অ্যাকশন হুক অগ্রাধিকার

যেকোন সংখ্যক ফাংশন যে কোন প্রদত্ত কর্মের সাথে "হুক" হতে পারে। কিছু কিছু ক্ষেত্রে একটি হুকড ফাংশন অন্যদের আগে বা পরে চালানো গুরুত্বপূর্ণ, যেখানে তৃতীয় প্যারামিটার add\_action (), \$priority কার্যকর হয়।

যদি \$priority আর্গুমেন্ট বাদ দেওয়া হয়, ফাংশনটি 10 এর ডিফল্ট অগ্রাধিকারের সাথে সংযুক্ত করা হবে।

যখন অ্যাকশনটি "ট্রিগার" হয়, তখন "হুকড" ফাংশনগুলিকে বলা হবে যেগুলি সবচেয়ে ছোট \$ অগ্রাধিকারের সাথে যুক্ত করা হয়েছে এবং সবচেয়ে বড় \$ অগ্রাধিকারের সাথে ফাংশনে অগ্রসর হচ্ছে। যেকোন হুকড ফাংশন যা একই অগ্রাধিকার শেয়ার করে সেই ক্রমানুসারে ডাকা হবে (যে ক্রমে তাদের নিজ নিজ add\_action() কল করা হয়েছিল)।

উদাহরণস্বরূপ, বলুন একটি তৃতীয় পক্ষের প্লাগইন একটি বহিরাগত ই-কমার্স সাইটের জন্য একটি অনুমোদিত লিঙ্কে দর্শকদের ডেইলি-ডিল পৃষ্ঠায় ফরোয়ার্ড করার জন্য 'টেমপ্লেট\_রিডাইরেস্ট' অ্যাকশনের সাথে যুক্ত একটি ফাংশন ব্যবহার করছে, কিন্তু আপনি পুনঃনির্দেশ করতে চান শুধুমাত্র লগ ইন ব্যবহারকারীদের জন্য ঘটতে। লগ-আউট হওয়া দর্শকদের সাইন-ইন পৃষ্ঠায় পাঠাতে আপনাকে আপনার নিজস্ব 'টেমপ্লেট\_রিডাইরেস্ট' হুক ব্যবহার করতে হবে। তৃতীয় পক্ষের প্লাগইনটি 10 এর ডিফল্ট \$পিওরিটি সহ এটির ফাংশন সংযুক্ত করে তা নির্ধারণ করার পরে, আপনার লগ-ইন চেকটি প্রথমে ঘটে তা নিশ্চিত করতে আপনি 9 এর অগ্রাধিকার দিয়ে আপনার ফাংশনকে হুক করতে পারেন :

```
ফাংশন redirect_deal_visitors_to_login() {
    if( is_page('দৈনিক-ডিল') && !user_is_logged_in() ) {
        wp_redirect( wp_login_url() ); প্রস্থান ();
    }
}

add_action( 'টেমপ্লেট_রিডাইরেস্ট', 'রিডাইরেস্ট_ডিল_ভিজিটরস_টু_লগইন', 9);
```

## হুকিং ক্লাস এবং অবজেক্ট মেথডস টু অ্যাকশন

পিএইচপি ক্লাস কোড সংগঠনের উন্নতি এবং নামকরণের সংঘর্ষ কমানোর জন্য শক্তিশালী হাতিয়ার।

কোনো না কোনো সময়ে, একটি ক্লাস পদ্ধতির জন্য কীভাবে একটি অ্যাকশন হুক তৈরি করা যায় সেই প্রশ্নটি অনিবার্যভাবে উঠে আসে।

\$function\_to\_add আর্গুমেন্টটি প্রায়ই ফাংশনের নাম সম্বলিত একটি স্ট্রিং হিসাবে দেখানো হয়, তবে আর্গুমেন্টের ডেটা-টাইপ আসলে একটি "কলযোগ্য", যা আমাদের উদ্দেশ্যে "একটি ফাংশন বা পদ্ধতির একটি রেফারেন্স" হিসাবে সংক্ষিপ্ত করা যেতে পারে।

অনেকগুলি কলযোগ্য বিন্যাস রয়েছে যা ক্লাসে এবং রেফারেন্স পদ্ধতিতে ব্যবহার করা যেতে পারে

বস্তু যাইহোক, সমস্ত ক্ষেত্রে উল্লেখিত পদ্ধতিটি সর্বজনীনভাবে দৃশ্যমান হতে হবে। একটি পদ্ধতি সর্বজনীন হয় যখন এটি হয় সর্বজনীন কীওয়ার্ডের সাথে প্রিফিক্স করা হয়, বা কোনও দৃশ্যমানতা কীওয়ার্ড না থাকে (যে ক্ষেত্রে পদ্ধতিটি সর্বজনীনের জন্য ডিফল্ট হয়)।

## অবজেক্ট মেথড অ্যাকশন হুক

অবজেক্ট পদ্ধতিগুলি একটি ক্লাসের একটি নির্দিষ্ট উদাহরণে কার্যকর করা হয়।

```
ক্লাস আমার_ক্লাস {  
  // কনস্ট্রাক্টর  
  ফাংশন My_Class() {  
    // ইনস্ট্যান্সিয়েশন লজিক  
  }  
  
  // ইনিশিয়ালাইজেশন ফাংশন  
  পাবলিক ফাংশন ইনিশিয়ালাইজ() {  
    // সূচনা যুক্তি  
  }  
}
```

নিম্নোক্ত শ্রেণীটি ইনস্ট্যান্সিয়েট করার পরে,

```
$my_class_instance = নতুন My_Class();
```

ইনিশিয়ালাইজ () পদ্ধতিটি সাধারণত \$my\_class\_instance->initialize(); কল করে অবজেক্টে চালু করা হবে। 'init' ওয়ার্ডপ্রেস অ্যাকশনের সাথে মেথড হুক করা একটি অ্যারে পাস করে ইনস্ট্যান্সের রেফারেন্স এবং অবজেক্ট মেথডের নাম সম্বলিত একটি স্ট্রিং করা হয়:

```
add_action('init', [ $my_class_instance, 'initialize' ]);
```

যদি add\_action() একটি অবজেক্ট পদ্ধতির মধ্যে বলা হয়, \$this pseudo-variable ব্যবহার করা যেতে পারে:

```
ক্লাস আমার_ক্লাস {  
  // কনস্ট্রাক্টর  
  ফাংশন My_Class() {  
    // ইনস্ট্যান্সিয়েশন লজিক add_action( 'init',  
    [ $this, 'initialize' ] );  
  }  
  
  // ইনিশিয়ালাইজেশন ফাংশন  
  পাবলিক ফাংশন ইনিশিয়ালাইজ() {  
    // সূচনা যুক্তি  
  }  
}
```

## ক্লাস মেথড অ্যাকশন হুক

ক্লাস পদ্ধতিগুলি কোনও নির্দিষ্ট উদাহরণের পরিবর্তে একটি ক্লাসে স্ট্যাটিকভাবে কার্যকর করা হয়। দেওয়া

নিম্নলিখিত ক্লাস,

```
ক্লাস আমার_ক্লাস {  
    // ইনিশিয়ালাইজেশন ফাংশন পাবলিক স্ট্যাটিক  
    ফাংশন ইনিশিয়ালাইজ() {  
        // (সূচনা যুক্তি)  
    }  
}
```

ইনিশিয়ালাইজ () পদ্ধতিটি সাধারণত :: স্কোপ-রেজোলিউশন অপারেটর ব্যবহার করে আহ্বান করা হবে , যেমন  
মাই\_ক্লাস::ইনিশিয়ালাইজ();। একটি ওয়ার্ডপ্রেসে একটি স্ট্যাটিক ক্লাস পদ্ধতি হুক করা কয়েকটি ভিন্ন উপায়ে করা যেতে পারে:

- ক্লাসের নাম সম্বলিত একটি স্ট্রিং এবং পদ্ধতির নাম ধারণকারী একটি স্ট্রিং দ্বারা গঠিত একটি অ্যারে ব্যবহার করা:

```
add_action( 'init', [ 'My_Class', 'initialize' ] );
```

- :: অপারেটর সহ পদ্ধতিটির সম্পূর্ণ রেফারেন্স সহ একটি স্ট্রিং পাস করা :

```
add_action('init', 'My_Class::initialize');
```

- যদি একটি স্ট্যাটিক ক্লাস পদ্ধতির মধ্যে add\_action() কল করা হয়, তাহলে ক্লাস নামের জায়গায় সেলফ কীওয়ার্ড বা \_\_CLASS\_\_ ম্যাজিক কনস্ট্যান্ট ব্যবহার করা যেতে পারে। মনে রাখবেন যে এটি সাধারণত অব্যাহিত কারণ এই আইটেমগুলির মানগুলি শ্রেণী উত্তরাধিকারের ক্ষেত্রে কিছুটা প্রতি-স্বজ্ঞাত হয়ে ওঠে।

```
ক্লাস আমার_ক্লাস {  
    // সেটআপ ফাংশন পাবলিক  
    স্ট্যাটিক ফাংশন setup_actions() {  
        add_action('init', 'self::initialize');  
    }  
  
    // ইনিশিয়ালাইজেশন ফাংশন পাবলিক স্ট্যাটিক  
    ফাংশন ইনিশিয়ালাইজ() {  
        // (সূচনা যুক্তি)  
    }  
}
```

# অধ্যায় 26: get\_bloginfo()

## ভূমিকা

বর্তমান সাইট সম্পর্কে তথ্য পুনরুদ্ধার করে.

## বাক্য গঠন

- `get_bloginfo( $শো , $ফিল্টার)`

## পরামিতি

পরামিতি বিবরণ	
\$শো	(স্ট্রিং) পুনরুদ্ধার করার জন্য সাইট সেটিং তথ্য।
\$ফিল্টার	(স্ট্রিং) ফিল্টার করা মান ফেরত দিতে হবে কি না তার স্পেসিফিকেশন।

## মন্তব্য

\$শো

মূল্যবোধ	বর্ণনা	উদাহরণ
'নাম' (ডিফল্ট)	সাইট শিরোনাম	'ম্যাট মুলেনওয়েগ'
'বর্ণনা'	সাইটের ট্যাগলাইন	'ওধু আরেকটি ওয়ার্ডপ্রেস সাইট'
'wpurl'	ওয়ার্ডপ্রেস এর URL স্থাপন. একই site_url() ফাংশন	'http://example.com' 'http:// localhost/wordpress'
'ইউআরএল'	সাইটের URL. একই রকম home_url () ফাংশন	'http://example.com' 'http:// localhost/wordpress'
'admin_email'	এর ইমেল ঠিকানা প্রধান প্রশাসক	'matt@mullenweg.com'
'বর্ণমালা'	এর অক্ষর এনকোডিং পেজ এবং ফিড	'UTF-8'
'সংস্করণ'	বর্তমান সংস্করণ ওয়ার্ডপ্রেস ইনস্টলেশন	'4.5'
'html_type'	'টেক্সট/এইচটিএমএল'- এর কন্টেন্ট-টাইপ মান	

মূল্যবোধ	বর্ণনা	উদাহরণ
	এইচটিএমএল	
'লেখার দিকবিন্যাস'	লেখার দিকবিন্যাস সাইটের দ্বারা নির্ধারিত হয় ভাষা	'ltr'
'ভাষা'	ISO 639-1 ভিত্তিক ভাষার কোড	'en-US'
'স্টাইলশীট_ইউআরএল'	এর স্টাইলশীটের URL সক্রিয় থিম। মূল্য অগ্রাধিকার: শিশু থিম » অভিভাবক থিম।	'http://example.com/wp content/ themes/twentytysixteen/style.css'
'স্টাইলশীট_ডিরেক্টরি'	এর সম্পদ অবস্থান সক্রিয় থিম। মান অগ্রাধিকার: চাইল্ড থিম » অভিভাবক থিম।	'http://example.com/wp content/ themes/twentytysixteen'
'টেমপ্লেট_ইউআরএল'	এর URL ডিরেক্টরি সক্রিয় থিম। মান অগ্রাধিকার: অভিভাবক থিম » শিশু থিম।	'http://example.com/wp content/ themes/twentytysixteen'
'টেমপ্লেট_ডিরেক্টরি'	'টেমপ্লেট_ইউআরএল' এর মতো	
'pingback_url'	Pingback XML-RPC ফাইল	'http://example/xmlrpc.php'
'এটম_ইউআরএল'	পরমাণু ফিড URL	'http://example/feed/atom/'
'rdf_url'	RDF/RSS 1.0 ফিড URL	'http://example/feed/rdf/'
'rss_url'	RSS 0.92 ফিড ইউআরএল	'http://example/feed/rss/'
'rss2_url'	RSS 2.0 ফিড URL	'http://example/feed/'
'মন্তব্য_অ্যাটম_ইউআরএল'	মন্তব্য পরমাণু ফিড URL	'http://example/comments/feed/atom/'
'সাইট URL টি'	(অপ্রচলিত) 'url' ব্যবহার করুন পরিবর্তে	
'বাড়ি'	(অপ্রচলিত) 'url' ব্যবহার করুন পরিবর্তে	

\$ফিল্টার

মূল্যবোধ	বর্ণনা	উদাহরণ
'raw' (ডিফল্ট) কোনো ফিল্টার প্রয়োগ করা হবে না		মূল তথ্য
'প্রদর্শন'	\$show 'url' না হলে রিটার্ন মানের ক্ষেত্রে ফিল্টার প্রয়োগ করা হবে , 'ডিসক্রেটরি' , 'বাড়ি'	ফিল্টার করা তথ্য

## উদাহরণ

সাইটের শিরোনাম হচ্ছে

```
<?php echo get_bloginfo('name'); ?>
```

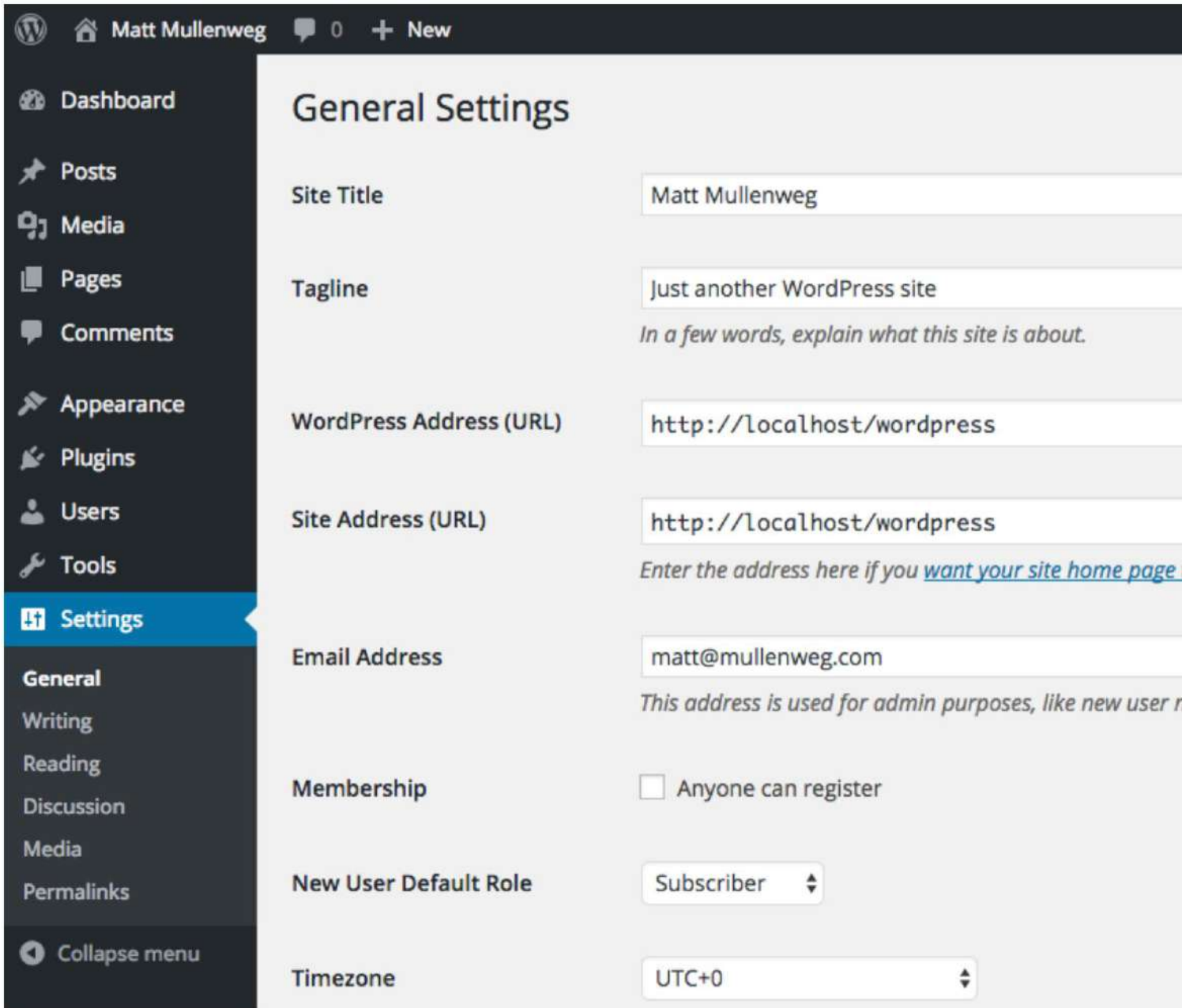
বা

```
<?php echo get_bloginfo(); ?>
```

আউটপুট

ম্যাট মুলেনওয়েগ

এই নমুনা সেটিংস উপর ভিত্তি করে



The image shows the WordPress dashboard for a user named Matt Mullenweg. The left sidebar contains navigation links for Dashboard, Posts, Media, Pages, Comments, Appearance, Plugins, Users, Tools, and Settings. The Settings menu is expanded, showing sub-categories: General, Writing, Reading, Discussion, Media, and Permalinks. The main content area is titled "General Settings" and contains several configuration options:

- Site Title:** Matt Mullenweg
- Tagline:** Just another WordPress site. Below it, a note says "In a few words, explain what this site is about."
- WordPress Address (URL):** http://localhost/wordpress
- Site Address (URL):** http://localhost/wordpress. Below it, a note says "Enter the address here if you want your site home page..."
- Email Address:** matt@mullenweg.com. Below it, a note says "This address is used for admin purposes, like new user r..."
- Membership:**  Anyone can register
- New User Default Role:** Subscriber
- Timezone:** UTC+0

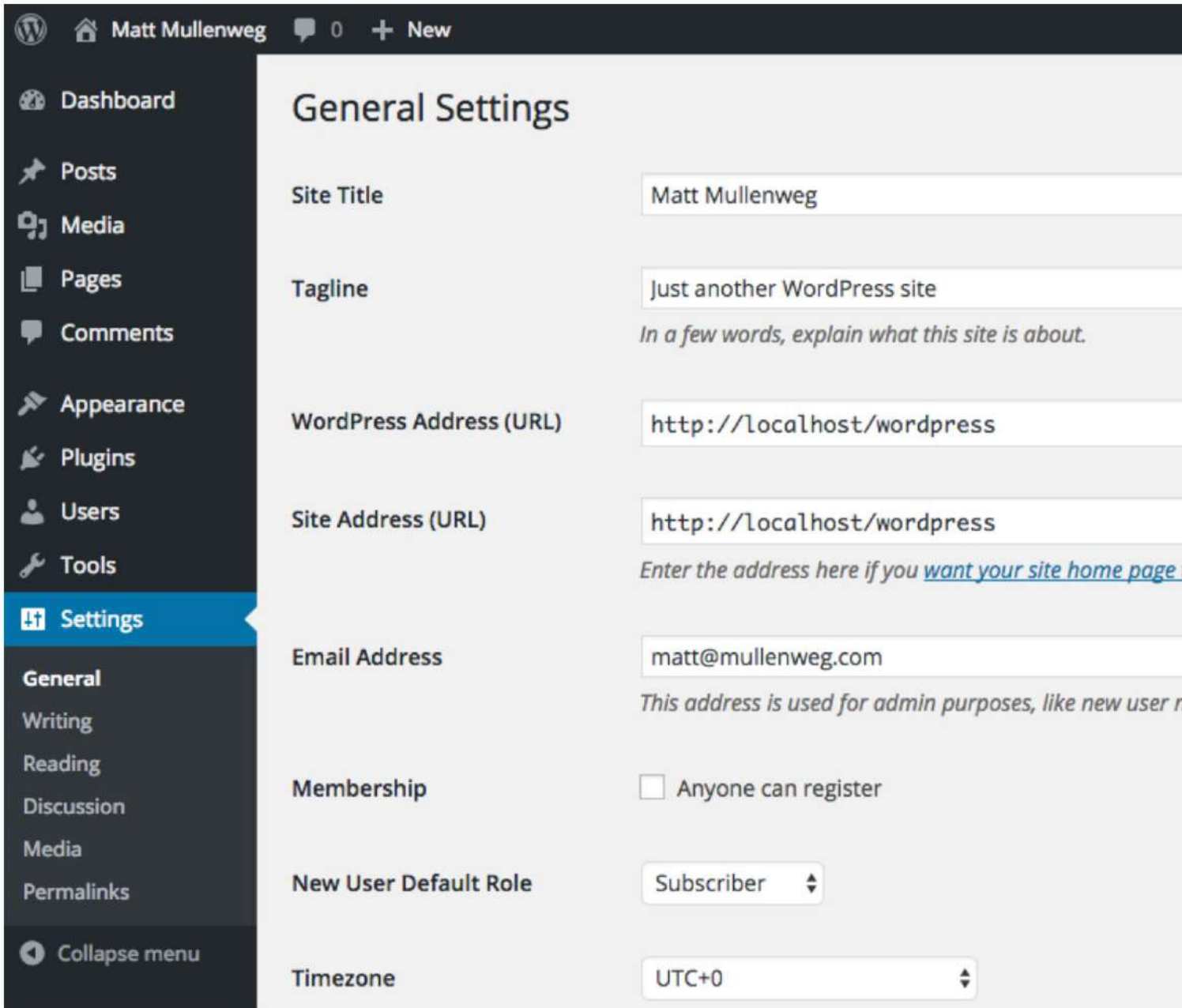
সাইট ট্যাগলাইন পেয়ে

```
<?php echo get_bloginfo('description'); ?>
```

আউটপুট

শুধু আরেকটি ওয়ার্ডপ্রেস সাইট

এই নমুনা সেটিংস উপর ভিত্তি করে



The image shows the WordPress administration interface for a user named Matt Mullenweg. The left sidebar contains navigation links for Dashboard, Posts, Media, Pages, Comments, Appearance, Plugins, Users, Tools, and Settings. The Settings menu is expanded to show sub-categories: General, Writing, Reading, Discussion, Media, and Permalinks. The main content area is titled 'General Settings' and contains several configuration options:

- Site Title:** Matt Mullenweg
- Tagline:** Just another WordPress site. Below the text is a note: "In a few words, explain what this site is about."
- WordPress Address (URL):** http://localhost/wordpress
- Site Address (URL):** http://localhost/wordpress. Below the text is a note: "Enter the address here if you want your site home page..."
- Email Address:** matt@mullenweg.com. Below the text is a note: "This address is used for admin purposes, like new user r..."
- Membership:**  Anyone can register
- New User Default Role:** Subscriber
- Timezone:** UTC+0

## সক্রিয় থিম URL পাওয়া

```
<?php echo esc_url( get_bloginfo('stylesheet_directory') ); ?>
```

### আউটপুট

```
http://example.com/wp-content/themes/twentysixteen
```

### বিকল্প

অভ্যন্তরীণভাবে, `get_bloginfo('stylesheet_directory')` কল করে `get_stylesheet_directory_uri()`, তাই আপনি এর পরিবর্তে এটি ব্যবহার করতে চাইতে পারেন:

```
<?php echo esc_url( get_stylesheet_directory_uri() ); ?>
```

অনেক ডেভেলপার এই ডেভিকেটেড ফাংশনগুলি ব্যবহার করতে পছন্দ করেন কারণ তাদের এবং `get_bloginfo()` এর মধ্যে অসামঞ্জস্যপূর্ণ নামকরণের নিয়ম রয়েছে। উদাহরণস্বরূপ, `get_stylesheet_directory()` চাইল্ড থিম পাথ প্রদান করে; যাইহোক, আমাদের পূর্ববর্তী উদাহরণ হিসাবে দেখায়, `get_bloginfo('stylesheet_directory')` চাইল্ড থিমের URL প্রদান করে। আপনি যদি এর পরিবর্তে `get_stylesheet_directory_uri()` ব্যবহার করেন, তাহলে আপনি একটি পাথ বা URL পুনরুদ্ধার করছেন কিনা তা নিয়ে বিভ্রান্তির সম্ভাবনা কম থাকে।

সাইটের ইউআরএল পান

```
<?php echo esc_url(get_bloginfo('url')); ?>
```

অথবা যদি আপনি একটি সাব পৃষ্ঠা লিঙ্ক করতে চান

```
<?php echo esc_url(get_bloginfo('url'). '/some-sub-page'); ?>
```

সাইট অ্যাডমিনিস্ট্রেটরের ইমেল ঠিকানা পান

আমরা সাইট অ্যাডমিনিস্ট্রেটরের ইমেল ঠিকানা পুনরুদ্ধার করতে `get_bloginfo` ফাংশন ব্যবহার করতে পারি।

```
<?php echo get_bloginfo('admin_email'); ?>
```

# অধ্যায় 27: get\_home\_path()

ভূমিকা

ওয়ার্ডপ্রেস ইনস্টলেশনের মূলে পরম ফাইল সিস্টেম পথ পান।

পরামিতি

পরামিতি বিবরণ	
কোনোটিই নয়	এই ফাংশন কোনো পরামিতি গ্রহণ করে না।

মন্তব্য

মধ্যে গুরুত্বপূর্ণ পার্থক্য

get\_home\_path()

এবং ABSPATH

অনুগ্রহ করে ABSPATH এবং get\_home\_path() এর মধ্যে পার্থক্যটি মনে রাখবেন যদি আপনি একটি সাবফোল্ডারে ওয়ার্ডপ্রেস ইনস্টল করে থাকেন।

get\_home\_path () ফাংশন সর্বদা সাবফোল্ডার ছাড়া একটি পথ ফেরত দেবে :

- <http://www.example.com> - /var/www/htdocs/example • <http://www.example.com/wp> - /var/www/htdocs/example

এইভাবে এটি ABSPATH থেকে আলাদা, যা বিভিন্ন মান প্রদান করবে:

- <http://www.example.com> - /var/www/htdocs/example • <http://www.example.com/wp> - /var/www/htdocs/example/wp

ABSPATH প্রথমে wp-load.php এ সংজ্ঞায়িত করা হয়েছে যা /var/www/htdocs/example/wp/wp load.php এ অবস্থিত হবে তাই এখানেই ABSPATH এর সংজ্ঞা নেবে।

get\_home\_path() সাইট\_url এবং home\_url আলাদা কিনা তা পরীক্ষা করে এবং পথ থেকে সাবস্ট্রিং সরিয়ে দেয়। অন্যথায় এটি ABSPATH মান প্রদান করে:

```
ফাংশন get_home_path() { $home =  
    set_url_scheme( get_option('home'), 'http'); $siteurl = set_url_scheme( get_option('siteurl'), 'http');  
    যদি ( ! খালি ( $home ) && 0 !== strcasecmp( $home, $siteurl )  
        ) {  
        $wp_path_rel_to_home = str_ireplace( $home, "", $siteurl); /* $siteurl - $home */ $pos = stripos( str_replace( '\\', '/',  
            $_SERVER['SCRIPT_FILENAME'] ),  
            trailingslashit( $wp_path_rel_to_home ) );  
        $home_path = substr( $_SERVER['SCRIPT_FILENAME'], 0, $pos); $home_path =  
            trailingslashit( $home_path ); } অন্য { $home_path = ABSPATH;  
    }  
}
```

```
str_replace('\\', '/', $home_path);  
}
```

## আপনার কোড এটি ব্যবহার করে

get\_home\_path() কল করা অবশ্যই এমন একটি প্রেক্ষাপটে করা উচিত যেখানে wp-admin/includes/file.php ইতিমধ্যেই অন্তর্ভুক্ত করা হয়েছে।

উদাহরণস্বরূপ admin\_init হকের মধ্যে get\_home\_path() ব্যবহার করা ভাল, কিন্তু init এর মধ্যে এটি ব্যবহার করা হয় না এবং এর ফলে একটি PHP মারাত্মক ত্রুটি হবে:

```
অনির্ধারিত ফাংশনে কল করুন get_home_path()
```

এই ফাইলটি শুধুমাত্র প্রশাসক (ড্যাশবোর্ড) প্রসঙ্গ থেকে অন্তর্ভুক্ত করা হয়, যদি আপনার একেবারে এই প্রসঙ্গের বাইরে এটির প্রয়োজন হয় তবে ফাংশনটি কল করার আগে আপনাকে ফাইলটি অন্তর্ভুক্ত করতে হবে:

```
প্রয়োজন_একবার (ABSPATH | 'wp-admin/includes/file.php');
```

## উদাহরণ

### ব্যবহার

```
$পথ = get_home_path();
```

ফেরত মূল্য:

স্ট্রিং

ওয়ার্ডপ্রেস ইনস্টলেশনের রুটে সম্পূর্ণ ফাইল সিস্টেম পথ, এমনকি যদি এটি একটি সাবফোল্ডারে ইনস্টল করা থাকে।

উদাহরণ:

```
/var/www/htdocs/উদাহরণ
```

# অধ্যায় 28: get\_option()

## ভূমিকা

একটি বিকল্প নামের উপর ভিত্তি করে একটি বিকল্প মান পুনরুদ্ধার করে।

## বাক্য গঠন

- `get_option( $option, $default )`

## পরামিতি

পরামিতি বিবরণ	
\$বিকল্প	(স্ট্রিং) পুনরুদ্ধার করার বিকল্পের নাম। এসকিউএল-এস্কেপড হবে না বলে আশা করা হচ্ছে।
\$ডিফল্ট	(মিশ্র) (ঐচ্ছিক) বিকল্পটি না থাকলে ফেরত দিতে ডিফল্ট মান।

## মন্তব্য

\$option-এর জন্য আর্গুমেন্টের তালিকা

- 'অ্যাডমিন\_ইমেল' •
- 'ব্লগনাম' • 'ব্লগ
- বিবরণ' • 'ব্লগ\_চারসেট' •
- 'তারিখ\_ফরম্যাট' •
- 'ডিফল্ট\_ক্যাটাগরি' •
- 'হোম' • 'siteurl' • 'টেমপ্লেট'
- 

'সাপ্তাহিক\_শুরু'

- 'আপলোড\_পথ' •

'ব্যবহারকারীদের\_ক্যান\_রেজিস্টার'

- 'posts\_per\_page' •

'posts\_per\_rss'

## উদাহরণ

ব্লগ শিরোনাম দেখান

কোড

```
<h1><?php echo get_option('blogname'); ?></h1>
```

আউটপুট

---

# H1 স্টাইলে ব্লগের নাম

ক্যারেক্টার সেট দেখান

কোড

```
<p><?php echo esc_html( sprintf( __( 'অক্ষর সেট: %s', 'টেমপ্লেটডোমেন' ), get_option( 'blog_charset' ) ); ?></p>
```

আউটপুট

অক্ষর সেট: UTF-8

অ-বিদ্যমান বিকল্পগুলি পরিচালনা করা

কোড

```
<?php echo get_option('something_bla_bla_bla'); ?>
```

আউটপুট

মিথ্যা

---

কোড

```
<?php echo get_option('something_bla_bla_bla', 'ওহ না!'); ?>
```

আউটপুট

ওহ না!

---

## অধ্যায় 29: get\_permalink()

### ভূমিকা

এই ফাংশনটি বর্তমান পোস্ট বা মনোনীত পোস্টের সম্পূর্ণ প্যারালিংক প্রদান করে।

### বাক্য গঠন

- get\_permalink( \$post, \$leavename )

### পরামিতি

পরামিতি বিবরণ	
\$পোস্ট	(int) (ঐচ্ছিক) পোস্ট আইডি বা পোস্ট অবজেক্ট। ডিফল্ট হল বর্তমান পোস্টের আইডি।
\$leavename (বুল) (ঐচ্ছিক)	পোস্টের নাম বা পৃষ্ঠার নাম রাখবেন কিনা।

### মন্তব্য

\$leavename প্যারামিটারের জন্য, এটি ডিফল্টরূপে মিথ্যা।

### উদাহরণ

#### get\_permalink এর সহজ ব্যবহার

##### কোড

```
echo get_permalink();
```

##### আউটপুট

বর্তমান পৃষ্ঠার লিঙ্ক, উদাহরণস্বরূপ: <http://website.com/category/name-of-post/>

#### লিঙ্ক পেতে পোস্ট উল্লেখ

##### কোড

```
echo get_permalink(44);
```

##### আউটপুট

পোস্ট আইডির লিঙ্ক: 44, উদাহরণস্বরূপ: <http://website.com/category/name-of-post/>

পোস্টের নাম ছাড়া লিঙ্ক পান

কোড

```
echo get_permalink( 44 , 'মিথ্যা');
```

আউটপুট

পোস্টের নাম ছাড়া পোস্ট আইডি'র লিঙ্ক: 44, উদাহরণস্বরূপ:  
<http://website.com/category/%postname%/>

## অধ্যায় 30: get\_template\_part()

বাক্য গঠন

- `get_template_part( $slug, $name);` •  
`get_template_part($slug);`

পরামিতি

পরামিতি বিবরণ	
<code>\$slug</code>	(স্ট্রিং) জেনেরিক টেমপ্লেট স্লাগ নাম
<code>\$নাম</code>	(স্ট্রিং) বিশেষায়িত টেমপ্লেট নাম

উদাহরণ

একটি সাবফোল্ডার থেকে একটি টেমপ্লেট অংশ লোড করুন

```
get_template_part('foo/bar', 'পৃষ্ঠা');
```

'foo' ডিরেক্টরি থেকে 'bar-page.php' প্রয়োজন হবে।

একটি নির্দিষ্ট ফাইল পান

আপনি এই ফাংশন ব্যবহার করে নির্দিষ্ট ফাইল পেতে পারেন।

```
get_template_part('template-parts/layout');
```

আপনার থিম ফোল্ডারের রুটে রাখা টেমপ্লেট-পার্টস সাবডিরেক্টরি থেকে layout.php ফাইল অন্তর্ভুক্ত করুন।

# অধ্যায় 31: get\_template\_part()

## ভূমিকা

এই ফাংশনের উদ্দেশ্য হল মূল থিম টেমপ্লেটে একটি থিমের আংশিক বা উপাদান আমদানি করার উপায়কে প্রমিত করা। আপনি একটি স্ট্যান্ডার্ড PHP SSI ব্যবহার করতে পারেন (সার্ভার সাইড অন্তর্ভুক্ত), তবে, get\_template\_part() ব্যবহার করার কিছু সুবিধা রয়েছে। এই ফাংশনটি ব্যবহার করে সার্ভারে সম্পূর্ণ যোগ্য পথ সনাক্ত করার চেষ্টা করা কম অভিজ্ঞ বিকাশকারীদের ত্রুটির প্রবণতা হ্রাস করে। এছাড়াও, ফাইলগুলি বিদ্যমান না থাকলে এটি সুন্দরভাবে ব্যর্থ হয় এবং একটি কাস্টম হায়ারার্কি ফলব্যাক সিস্টেম ওরফে "ফজি টেমপ্লেট অনুসন্ধান" পরিচালনা করে।

## বাক্য গঠন

- get\_template\_part( \$slug ) •
- get\_template\_part( \$slug, \$name )

## পরামিতি

পরামিতি বিবরণ	
\$slug	(স্ট্রিং) কাস্টম টেমপ্লেটের স্লাগ নাম।
\$নাম	(স্ট্রিং) বিশেষায়িত টেমপ্লেটের নাম। ঐচ্ছিক

## উদাহরণ

### একটি কাস্টম টেমপ্লেট সহ

```
<?php get_template_part('foo'); ?>
```

#### অন্তর্ভুক্ত

```
../wp-content/themes/your-theme-slug/foo.php
```

### একটি ড্যাশ-বিচ্ছিন্ন ফাইলের নাম সহ একটি কাস্টম টেমপ্লেট সহ

```
<?php get_template_part('foo','bar'); ?>
```

#### অন্তর্ভুক্ত

```
../wp-content/themes/your-theme-slug/foo-bar.php
```

একটি ডিরেক্টরির ভিতর থেকে একটি কাস্টম টেমপ্লেট সহ

```
<?php get_template_part('dir/foo'); ?>
```

অন্তর্ভুক্ত

```
../wp-content/themes/your-theme-slug/dir/foo.php
```

একটি ডিরেক্টরির ভিতরে অবস্থিত একটি ড্যাশ-বিচ্ছিন্ন ফাইলের নাম সহ একটি কাস্টম টেমপ্লেট সহ

```
<?php get_template_part( 'dir/foo', 'bar' ); ?>
```

অন্তর্ভুক্ত

```
../wp-content/themes/your-theme-slug/dir/foo-bar.php
```

কাস্টম টেমপ্লেট সুযোগ পরিবর্তনশীল পাস

```
<?php  
set_query_var('passed_var', $my_var); get_template_part('foo',  
'বার');  
?>
```

foo-bar.php- এ এটি অ্যাক্সেস করুন

```
<?php echo $passed_var; ?>
```

# অধ্যায় 32: get\_template\_part()

## বাক্য গঠন

- `get_template_part('file-name-no-extension');`

## পরামিতি

প্যারামিটার	বর্ণনা
ফাইল-নাম-কোন এক্সটেনশন নেই	কোনো এক্সটেনশন ছাড়া টেমপ্লেট অংশের নাম। যেমন 'foo.php'-এর পরিবর্তে 'foo'

## উদাহরণ

### টেমপ্লেট অংশ লোড হচ্ছে

একটি নির্দিষ্ট ফাইল থেকে কোডটি অন্য ফাইলে টেনে নেয় যেখানে কলটি করা হয়েছিল।

যেমন example.php এর ভিতরে

```
<h1>হ্যালো ওয়ার্ল্ড!</h1>
```

page.php এর ভিতরে

```
// হেডার কোড  
get_template_part('উদাহরণ'); // বাকি পৃষ্ঠা কোড
```

আউটপুট:

```
// হেডার কোড <h1>হ্যালো  
ওয়ার্ল্ড</h1> // বাকি পৃষ্ঠা কোড
```

## অধ্যায় 33: get\_the\_category()

### ভূমিকা

এই ফাংশনটি বর্তমান পোস্ট বা পৃষ্ঠা বা মনোনীত পোস্ট বা পৃষ্ঠার একটি অ্যারে হিসাবে সমস্ত বিভাগ প্রদান করে।

### বাক্য গঠন

- `get_the_category($id)`

### পরামিতি

পরামিতি বিবরণ	
<code>\$id</code>	(int) (ঐচ্ছিক) বর্তমান পোস্ট আইডিতে ডিফল্ট। পোস্ট আইডি।

### মন্তব্য

অনুগ্রহ করে মনে রাখবেন `get_the_category()` একটি অ্যারে প্রদান করে, যার মানে আপনি প্রত্যাবর্তিত মানটিকে সরাসরি প্রতিধ্বনিত করতে পারবেন না।

এখানে প্রতিটি বিভাগের বস্তুর একটি তালিকা রয়েছে যা আপনি মুদ্রণ করতে পারেন:

- `term_id`
- নাম
- স্লাগ
- `term_group`
- `term_taxonomy_id`
- শ্রেণীবিন্যাস
- বিবরণ
- অভিভাবক
- গণনা
- 

- অবজেক্ট\_আইডি
- ফিল্টার • `cat_ID` • বিভাগ\_গণনা
- বিভাগ\_বিবরণ
- বিভাগের\_নাম • বিভাগ\_সু নাম
- বিভাগ\_পিতা

### উদাহরণ

## পোস্টের সব বিভাগের নাম পান

কোড

```
$শ্রেণী = get_the_category();  
foreach ( $category হিসাবে $category ) {  
    echo $category->name. '<br />';  
}
```

আউটপুট

বর্তমান পৃষ্ঠার সমস্ত বিভাগের নাম, প্রতিটি লাইনে একটি।

## পোস্টের সমস্ত বিভাগের আইডি পান

কোড

```
$শ্রেণী = get_the_category();  
foreach ( $category হিসাবে $category ) {  
    echo $category->term_id. '<br />';  
}
```

আউটপুট

বর্তমান পৃষ্ঠার বিভাগগুলির সমস্ত আইডি, প্রতিটি লাইনে একটি।

---

# অধ্যায় 34: get\_the\_title()

## ভূমিকা

এই ফাংশনটি বর্তমান পোস্ট বা মনোনীত পোস্টের শিরোনাম প্রদান করে।

## বাক্য গঠন

- get\_the\_title( \$post )

## পরামিতি

পরামিতি বিবরণ	
\$পোস্ট	(int) (ঐচ্ছিক) পোস্ট আইডি বা পোস্ট অবজেক্ট। ডিফল্ট হল বর্তমান পোস্টের আইডি।

## মন্তব্য

আপনি যদি একটি পোস্ট লুপ ব্যবহার করে একটি পোস্ট বা পৃষ্ঠার শিরোনাম পাওয়ার পরিকল্পনা করেন, তাহলে এর পরিবর্তে the\_title() ব্যবহার করার পরামর্শ দেওয়া হয়।

## উদাহরণ

get\_the\_title এর সহজ ব্যবহার

কোড

```
get_the_title();
```

আউটপুট

বর্তমান পোস্ট বা পৃষ্ঠার শিরোনাম

একটি নির্দিষ্ট পোস্ট আইডির শিরোনাম পান

কোড

```
get_the_title( 44);
```

আউটপুট

পোস্টের শিরোনাম আইডি: 44.



## অধ্যায় 35: home\_url()

### বাক্য গঠন

- home\_url( \$path, \$scheme);

### পরামিতি

পরামিতি বিবরণ	
\$পাথ	(স্ট্রিং, ঐচ্ছিক) হোম ইউআরএলের পরে আরও সেগমেন্ট যোগ করতে।
\$স্কিম	(স্ট্রিং, ঐচ্ছিক) হোম ইউআরএল প্রসঙ্গ দিতে স্কিম। 'http', 'https', 'আপেক্ষিক', 'বিশ্রাম', বা শূন্য গ্রহণ করে।

### উদাহরণ

হোম URL প্রাপ্ত করা হচ্ছে

`home_url()` বর্তমান সাইটের হোম ইউআরএল পুনরুদ্ধার করার জন্য ব্যবহৃত হয়।

```
<?php echo esc_url( home_url( '/' ) ); ?>
```

#### আউটপুট

```
http://www.example.com
```

#### সাইট URL টি

উপযুক্ত প্রোটোকল সহ 'site\_url' বিকল্পটি ফেরত দেয় (https://)

```
<?php echo site_url(); ?>
```

#### আউটপুট

```
http://www.example.com
```

## অধ্যায় 36: অ্যাডভান্স কাস্টম ফিল্ডের রিপোর্টার অ্যাড-অনের সাথে আমি কিভাবে মার্কডাউন সম্পাদককে সংহত করতে পারি।

### উদাহরণ

মার্কডাউন এডিটর যোগ করুন

আমি সমাধান খুঁজে পেয়েছি। অনুগ্রহ করে নিচে উল্লেখ করা ধাপগুলো বিবেচনা করুন।

[wp মার্কডাউন এডিটর ইনস্টল করুন](#) প্লাগ লাগানো.

তারপর "acf-wp-wysiwyg" ইনস্টল করুন রিপোর্টার ফিল্ডের জন্য। এখন আপনাকে কিছু ফাইল আপডেট করতে হবে। এই ফাইলটি খুলুন এবং লাইন নম্বর "180" এ যান বা "create\_field" ফাংশনে যান

```
echo '<script> var simplemde = new SimpleMDE({element: document.getElementById("'.  
$.id.'});jQuery("quicktags toolbar").css("display","none"); </script>';
```

এখন "acf-রিপোর্টার" প্লাগইনের অধীনে "input.js" ফাইল খুলুন, লাইন নম্বর "142"

প্রতিস্থাপন

```
new_field_html = this.$el.find('> টেবিল > tbody > tr.row-clone').html().replace(/(("[\w- \[\]]*?)(acfcloneindex )/g, '$1' + new_id),
```

সঙ্গে

```
new_field_html = this.$el.find('> টেবিল > tbody > tr.row-clone').html().replace(/(("[\w- \[\]]*?)(acfcloneindex) /g, '$1' + new_id),
```

# অধ্যায় 37: init

বাক্য গঠন

```
1. add_action('init', কলযোগ্য $function)
```

মন্তব্য

init হল একটি অ্যাকশন হুক যা ওয়ার্ডপ্রেস লোডিং শেষ করার পরে কিন্তু কোনো HTTP শিরোনাম পাঠানোর আগেই বরখাস্ত হয়ে যায়।

উদাহরণ

\$\_POST অনুরোধ ডেটা প্রক্রিয়া করা হচ্ছে

```
add_action('init', 'process_post_data');
```

```
ফাংশন process_post_data() { if( isset( $_POST ) // এখানে  
    $_POST ডেটা প্রক্রিয়া করুন    ) {  
  
    }  
}
```

\$\_GET অনুরোধ ডেটা প্রক্রিয়া করা হচ্ছে

```
add_action('init', 'process_get_data');  
  
ফাংশন process_get_data() { if( isset( $_GET ) // এখানে  
    $_GET ডেটা প্রক্রিয়া করুন    ) {  
  
    }  
}
```

একটি কাস্টম পোস্ট টাইপ নিবন্ধন

```
add_action( 'init', ফাংশন() {  
    register_post_type( 'ইভেন্ট', অ্যারে( 'public' => সত্য,  
  
        'label' => 'ঘটনা'  
    );  
});
```

একটি লেবেল ইভেন্ট এবং একটি স্লাগ ইভেন্ট সহ একটি নতুন কাস্টম পোস্ট প্রকার নিবন্ধন করে।

---

# অধ্যায় 38: ইনস্টলেশন এবং কনফিগারেশন

## উদাহরণ

### LAMP এ Wordpress

আমি উবুন্টু 14.04 এর সাথে Amazon EC2 এ নিম্নলিখিত পদক্ষেপগুলি পরীক্ষা করেছি

এখানে LAMP প্রযুক্তি স্ট্যাক ইনস্টল করার জন্য কমান্ড লাইনগুলির একটি সারাংশ রয়েছে (ওয়ার্ডপ্রেস ইনস্টল করার আগে):

1: LAMP প্রযুক্তি স্ট্যাক ইনস্টল করুন

লিনাক্স আপডেট করুন

```
#> sudo apt-get update -> আপডেট প্যাকেজ সংগ্রহস্থলের তথ্য
```

```
#> sudo apt-get upgrade -> প্যাকেজ আপগ্রেড ইনস্টল করুন
```

apache2 ইনস্টল করুন

```
#> sudo apt-get install apache2
```

php ইনস্টল করুন

```
#> sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt
```

মাইএসকিউএল ইনস্টল করুন

```
#> sudo apt-get install mysql-server php5-mysql
```

phpmyadmin ইনস্টল করুন

```
#> sudo apt-get install phpmyadmin
```

যদি জিনিসটি ভুল হয়ে যায়, ইনস্টল করা প্যাকেজ এবং এর কনফিগারেশন ফাইল(গুলি) সরান

```
#> sudo apt-get purge package_name
```

স্ট্যাক ইনস্টল হয়ে গেলে এখানে ওয়ার্ডপ্রেস ইনস্টলেশন সম্পূর্ণ করার ধাপগুলি হল:

2: ওয়ার্ডপ্রেস ইনস্টল করুন

1. `wget https://wordpress.org/latest.zip`

2. `sudo mkdir /var/www/wordpress`

3. জিপ ফাইলকে /var/www/wordpress-এ সরান এবং এক্সট্রাক্ট করুন।

4. `cd /etc/apache2/sites-available/`

5. `sudo cp 000-default.conf wordpress.conf`

wordpress.conf পরিবর্তন করুন যাতে apache2 কোন অনুরোধ বা আপনার ডোমেনকে 6. wordpress অ্যাপ ফোল্ডারে ফরওয়ার্ড করতে পারে।

7. sudo a2ensite wordpress.conf

8. sudo পরিষেবা apache2 পুনরায় চালু করুন

9. youdomain.com/phpmyadmin দ্বারা আপনার ব্রাউজার এর মাধ্যমে PhpMyAdmin-এ যান। একটি নতুন ব্যবহারকারী "ওয়ার্ডপ্রেস" তৈরি করুন এবং সংশ্লিষ্ট ডাটাবেস তৈরি করতে পরীক্ষা করুন। cd/var/www/wordpress

10. sudo cp wp-config-example.php wp-config.php

11. আপনার মাইএসকিউএল ওয়ার্ডপ্রেস ডাটাবেস তথ্য যোগ করতে কনফিগার ফাইল পরিবর্তন করুন।

12. sudo chown -R www-data:www-data /var/www/wordpress

13. sudo chmod -R 755 /var/www/wordpress/

14. ব্রাউজার খুলুন এবং আপনার ডোমেনে টাইপ করুন। আপনি ওয়ার্ডপ্রেস ইনস্টলেশন পৃষ্ঠা দেখতে হবে। নির্দেশ অনুসরণ করুন এবং আপনি সম্পন্ন!

## MAMP-তে WP ইনস্টলেশন

MAMP এ ওয়ার্ডপ্রেস ইনস্টল করা বেশ সহজ।

আপনাকে কয়েকটি সহজ পদক্ষেপ অনুসরণ করতে হবে:

1 - এখান থেকে MAMP ডাউনলোড করুন, এটি বিনামূল্যে এবং আপনার সম্ভবত প্রো সংস্করণের প্রয়োজন নেই।

2 - আপনার পিসি বা ম্যাকে ইনস্টল করুন।

3 - প্রোগ্রাম চালান -> আপনি একটি ছোট উইন্ডো খোলা দেখতে পাবেন, সেখান থেকে আপনি MAMP সেট করতে পারেন। এই মুহুর্তে সমস্ত প্রাক-সেট মান ছেড়ে দিন, প্রথম ইনস্টলেশনের জন্য আপনাকে আপনার জীবনকে জটিল করতে হবে না!

ভবিষ্যতে মনে রাখবেন মাইএসকিউএল ডাটাবেসের জন্য আপনার পাসওয়ার্ড এবং ব্যবহারকারীর নাম পরিবর্তন করা ভাল অভ্যাস। ডিফল্টরূপে এটি রুট।

4 - "ওয়েবস্টার্ট পৃষ্ঠা খুলুন"-এ ক্লিক করুন - এখানে আপনি আপনার ডেটা তথ্য যেমন পাসওয়ার্ড, অ্যাডমিনের নাম এবং MAMP সম্পর্কে তথ্য দেখতে পাবেন।

5 - টুল -> phpMyAdmin-এ ক্লিক করুন এবং আপনাকে ডাটাবেস পৃষ্ঠায় পুনঃনির্দেশিত করা হবে।

6 - একটি নতুন ডাটাবেস তৈরি করুন, নতুন এ ক্লিক করুন এবং আপনি যে নামটি চান তা দিন, আপনার এই নামটি পরে প্রয়োজন হবে।

7 - এখন একটি ফোল্ডার কল "htdocs" সন্ধান করুন, এটি আপনার পিসি বা ম্যাকের MAMP ফোল্ডারের ভিতরে রয়েছে। আপনি যদি একটি ম্যাক ব্যবহার করেন তবে আপনাকে ফাইল্ডারে একটি অ্যাপ্লিকেশন করতে হবে এবং "প্যাকেজ সামগ্রী দেখান" ব্যবহার করে অ্যাপটি খুলতে হবে। ভিতরে আপনি htdocs ফোল্ডার পাবেন।

8 - Wordpress ফোল্ডার নিন এবং htdocs এর ভিতরে কপি করুন, আপনাকে সমস্ত ফোল্ডারের ভিতরে রাখতে হবে, জিপ ফাইলটি রাখতে হবে না। ফোল্ডারের নাম পরিবর্তন করুন, আপনি আপনার প্রকল্পের নাম দিয়ে কল করতে পারেন।

9 - আপনার ব্রাউজারে MAMP এর উইন্ডোতে ফিরে আসুন এবং "আমার সাইট" এ ক্লিক করুন - এটি একটি খুলবে

URL কল <http://localhost:8888> (8888 হল ডিফল্ট পোর্ট, আপনি চাইলে এটি পরিবর্তন করতে পারেন)। এখানে আপনি htdocs ফোল্ডারের ভিতরে যে ফোল্ডারটি রেখেছেন সেটি দেখতে পাবেন , ফোল্ডারটিতে আপনার দেওয়া নামের উপর ক্লিক করুন।

10 - এখন ওয়ার্ডপ্রেসের একটি সাধারণ ইনস্টলেশন শুরু করুন, আপনাকে অ্যাডমিন এবং পাসওয়ার্ড ব্যবহার করতে হবে। ডিফল্টরূপে এটি রুট এবং রুট, এবং আপনি আগে তৈরি করা ডাটাবেসের নাম ব্যবহার করুন।

11 - ইনস্টলেশন চালান এবং শেষ!

আপনি <http://localhost:8888/namefolder-> এ আপনার ওয়েবসাইট দেখতে পাবেন অবশ্যই আপনাকে MAMP চালু রাখতে হবে।

# অধ্যায় 39: এর সাথে নেটওয়ার্ক অনুরোধ করা HTTP API

## বাক্য গঠন

- \$response = wp\_remote\_get( \$url, \$args);
- \$response = wp\_remote\_post( \$url, \$args);
- \$response = wp\_safe\_remote\_post( \$url, \$args);

## পরামিতি

পরামিতি বিবরণ	
\$url	(স্ট্রিং) (প্রয়োজনীয়) পুনরুদ্ধার করার জন্য সাইট URL।
\$args	(অ্যারে) (ঐচ্ছিক) আর্গুমেন্টের অনুরোধ করুন।

## মন্তব্য

## রিটার্নস

(WP\_Error | array) একটি অ্যারে হিসাবে প্রতিক্রিয়া, অথবা ব্যর্থতার উপর WP\_Error।

## উদাহরণ

একটি দূরবর্তী JSON সম্পদ পান

এই স্নিপেটটি একটি JSON ফরম্যাটেড রিসোর্স ধরবে, এটিকে ডিকোড করবে এবং পিএইচপি অ্যারে ফরম্যাটে প্রিন্ট করবে।

```
// আনা
$response = wp_remote_get('http://www.example.com/resource.json');

যদি ( ! is_wp_error( $response ) ) {
    $headers = wp_remote_retrieve_headers($response);

    যদি ( isset( $headers[ 'content-type' ] ) ) {
        যদি ( $headers[ 'content-type' ] == 'application/json' ) {
            print_r( json_decode( wp_remote_retrieve_body( $response ) ) );
        }
    }
}
```

# অধ্যায় 40: মেটা বক্স

## ভূমিকা

wp-admin বিষয়বস্তু সম্পাদকগুলিতে একটি মেটা বক্সের সহজ ব্যবহার

## বাক্য গঠন

- `_x('টেক্সট', 'বিবরণ', 'টেক্সটডোমেন')` `__('টেক্সট', 'টেক্সটডোমেন')` এর পরিবর্তে অনুবাদ পরিষেবার জন্য একটি বিবরণ যোগ করতে ব্যবহৃত হয় যা শুধুমাত্র অনুবাদ
- `_ex('টেক্সট', 'ডেসক্রিপশন', 'টেক্সটডোমেন')` বর্ণনার সাথে অনুবাদ করা টেক্সটকে প্রতিনিধিত্ব করতে ব্যবহৃত হয়

## মন্তব্য

রেন্ডার মেটা বক্সের ভিতরের বিষয়বস্তু যেকোনো কিছু হতে পারে। মানগুলি সরাসরি একত্রিত হওয়ার পরিবর্তে, আপনি একটি PHP টেমপ্লেটের সাথে একটি অন্তর্ভুক্ত ব্যবহার করতে পারেন এবং এতে ডেটা পাস করতে `set_query_var` পদ্ধতি ব্যবহার করতে পারেন। সংরক্ষণ একই ভাবে কাজ করবে।

## উদাহরণ

একটি নিয়মিত ইনপুট এবং একটি নির্বাচন ইনপুট সহ একটি সাধারণ উদাহরণ

```
/**
 * পোস্টের ধরনে মেটা বক্স যোগ করুন।
 *
 * @১.০.০ থেকে
 */
ফাংশন myplugin_add_meta_box() {
    // ডিফল্ট পোস্ট প্রকার সেট আপ করুন/ $types = অ্যারে ('পোস্ট',

);

    // আরো ধরনের মেটা বক্স যোগ করার জন্য ঐচ্ছিক ফিল্টার. ব্যবহার করতে uncomment. // $types = apply_filters('myplugin_meta_box_types',
    $types);

    // পৃষ্ঠায় মেটা বক্স যোগ করুন add_meta_box(

        'myplugin-meta-box', // Meta Box Id. কিছু হতে পারে। _x( 'কাস্টম মেটা', 'কাস্টম মেটা বক্স', 'মাইপ্লাগইন' ), //
        মেটা বক্সের শিরোনাম।
        অনুবাদ ঐচ্ছিক। শুধু স্ট্রিং হতে পারে.
        'myplugin_render_meta_box', // রেন্ডার মেটা বক্স ফাংশন। $types, // মেটা বক্স যোগ করতে পোস্ট প্রকার যোগ করুন।
        'সাইড', // সম্পাদনার পাশে দেখান। 'উচ্চ' // সম্পাদনার শীর্ষে দেখান।

    );
}

add_action( 'add_meta_boxes', 'myplugin_add_meta_box');
```

```

/**
 * মেটা বক্স রেন্ডার করুন।
 *
 * এটি একটি মেটা বক্সের ভিতরে একটি মৌলিক ইনপুট এবং একটি নির্বাচনের উদাহরণ দেখায়। এগুলো যেকোনো কিছু হতে পারে।
 *
 * @১.০.০ থেকে
 *
 * @param $post WP_Post পোস্টটি সম্পাদনা করা হচ্ছে। */ ফাংশন

myplugin_render_meta_box( $post )
{
    // আমাদের কাস্টম মেটা বক্সের জন্য বর্তমান পোস্ট মেটা মান। $city = get_post_meta( $post->ID, 'city', true); // একটি
    একক ফেরত দেওয়ার জন্য সত্য
    মান
    $country = get_post_meta( $post->ID, 'country', true); // একটি একক ফেরত দেওয়ার জন্য সত্য
    মান

    // নিরাপত্তার জন্য WP Nonce ফিল্ড যোগ করুন।
    wp_nonce_field( plugin_basename( __FILE__ ), 'myplugin_meta_nonce' );
    ?>

<p>
<label for="city"> <?php_ex( 'City',
    'Custom Meta Box Template', 'myplugin' ); ?> </label> <input name="city" id="city" value="<?php echo $city; ?>" /> </p>
<p>

<label for="country"> <?php
    _ex( 'Country', 'Custom Meta Box Template', 'myplugin' ); ?> </label> <select name="country" id="country">।

    <option value="United States" <?php selected( $country, 'United States' ); ?>><?php
    _ex( 'মার্কিন যুক্তরাষ্ট্র', 'কাস্টম মেটা বক্স টেমপ্লেট', 'মাইপ্লাগইন' ); ?></option>
    <বিকল্প মান="মেক্সিকো" <?php নির্বাচিত( $country, 'Mexico' ); ?>><?php_ex( 'মেক্সিকো',
    'কাস্টম মেটা বক্স টেমপ্লেট', 'myplugin' ); ?></option>
    <বিকল্প মান="কানাডা" <?php নির্বাচিত( $country, 'Canada' ); ?>><?php_ex( 'কানাডা',
    'কাস্টম মেটা বক্স টেমপ্লেট', 'myplugin' ); ?></option> </select> </p>

<?php
}

/**
 * মেটা বক্স ডেটা সংরক্ষণ করুন।
 *
 * @১.০.০ থেকে
 *
 * @param $post_id int পোস্টের আইডি সংরক্ষণ করা হচ্ছে। */ ফাংশন myplugin_save_meta_data( $post_id )

{
    // যাচাই করুন এটি একটি স্বয়ংক্রিয় সংরক্ষণ নয়। যদি ( সংজ্ঞায়িত
    ('DOING_AUTOSAVE') && DOING_AUTOSAVE )
    {
        প্রত্যাবর্তন
    }

    // নিরাপত্তার জন্য মেটা বক্স ননস যাচাই করুন। যদি (
        ! isset( $_POST['myplugin_meta_nonce'] ) || ! wp_verify_nonce(
    $_POST['myplugin_meta_nonce'], plugin_basename( __FILE__ )
        ) ) {
        প্রত্যাবর্তন
    }
}

```

```

// ফর্ম থেকে নতুন মান পান।
$city = $_POST['শহর'];
$country = $_POST['দেশ'];

// update_post_meta মান যোগ করবে যদি এটি বিদ্যমান না থাকে বা যদি এটি থাকে তবে এটি আপডেট করবে।
update_post_meta( $post_id, 'city', $city);
update_post_meta( $post_id, 'দেশ', $country);

/*
* ঐচ্ছিক বিকল্প
*
* শুধু update_post_meta ব্যবহার করার পরিবর্তে, আপনি মানগুলিও পরীক্ষা করতে পারেন এবং
* পোস্ট মেটা মান তৈরি/আপডেট/মুছে ফেলতে সমস্যা।
*/

// $current_city_value = get_post_meta( $post_id, 'city', true); // ফেরার জন্য সত্য
একটি একক মান।
//
// // পোস্ট মেটা যোগ করুন যদি এটি বিদ্যমান না থাকে।
// যদি ( $city && $city !== $current_city_value ) // add_post_meta( $post_id, {
'city', $city, true); // সত্য মানে কী অনন্য
পোস্টটি, মিথ্যা ডিফল্ট এবং একাধিক যোগ করা যেতে পারে।
// }
// // পোস্ট মেটা সম্পাদনা করুন যদি এটি বিদ্যমান থাকে এবং একটি নতুন মান থাকে।
// elseif ( $city && $city !== $current_city_value ) // update_post_meta( $post_id, 'city', {
$city);
// }
// // কোন নতুন মান না থাকলে পোস্ট মেটা মুছুন।
// elseif ( " //
=== $city && $current_city_value ) {
delete_post_meta( $post_id, 'city', $current_city_value); // $current_city_value
ঐচ্ছিক এবং একই কী দিয়ে অন্যান্য মানের মধ্যে পার্থক্য করতে ব্যবহৃত হয়।
// }
}

add_action('save_post', 'myplugin_save_meta_data');

```

---

# অধ্যায় 41: অপশন এপিআই

## ভূমিকা

বিকল্পগুলি হল ডেটার টুকরো যা ওয়ার্ডপ্রেস বিভিন্ন পছন্দ এবং কনফিগারেশন সেটিংস সংরক্ষণ করতে ব্যবহার করে। অপশন এপিআই হল ডাটাবেসে ডেটা সংরক্ষণের একটি সহজ এবং প্রমিত উপায়। API তৈরি করা, অ্যাক্সেস করা, আপডেট করা এবং অপশন মুছে ফেলা সহজ করে তোলে।

## বাক্য গঠন

- // WordPress add\_option এর মধ্যে নতুন বিকল্প তৈরি করুন( \$option, \$value = , \$deprecated = , \$autoload = 'হ্যাঁ');
- // ডাটাবেস থেকে একটি বিকল্প সরিয়ে দেয়। ডিলিট\_অপশন (\$ বিকল্প);
- // একটি সংরক্ষিত বিকল্প পুনরুদ্ধার করুন get\_option( \$option, \$default = false);
- // ইতিমধ্যে যোগ করা একটি বিকল্পের মান আপডেট করুন। update\_option( \$option, \$newvalue);
- // এই ফাংশনগুলির \*\_site\_option() সংস্করণগুলিও রয়েছে, // ওয়ার্ডপ্রেস মাল্টিসাইট-এ নেটওয়ার্ক-ওয়াইড বিকল্পগুলি পরিচালনা করতে
- // নতুন নেটওয়ার্ক বিকল্প তৈরি করুন add\_site\_option( \$option, \$value = , \$deprecated = , \$autoload = 'হ্যাঁ');
- // একটি নেটওয়ার্ক বিকল্প সরিয়ে দেয় delete\_site\_option( \$option);
- // একটি সংরক্ষিত নেটওয়ার্ক বিকল্প পুনরুদ্ধার করুন get\_site\_option( \$option, \$default = false );
- // ইতিমধ্যে যোগ করা একটি বিকল্পের মান আপডেট করুন। update\_site\_option( \$option, \$newvalue);

## মন্তব্য

Options API হল MySQL ডাটাবেসের বিকল্প সারণীতে সংরক্ষিত ডেটা নিয়ে কাজ করার একটি সহজ এবং মানসম্মত উপায়। API বিকল্পগুলি তৈরি করা, পড়া, আপডেট করা এবং মুছে ফেলা সহজ করে তোলে।

## উদাহরণ

### get\_option

get\_option ফাংশন বিকল্প নামের উপর ভিত্তি করে বিকল্প টেবিল থেকে একটি মান পুনরুদ্ধার করতে ব্যবহার করা হয়।

আপনি ওয়ার্ডপ্রেস সাইট অ্যাডমিনিস্ট্রেটরের ইমেল ঠিকানা পেতে নিম্নলিখিত কোডটি ব্যবহার করতে পারেন।

```
<?php echo get_option('admin_email'); ?>
```

get\_option() এর একটি ঐচ্ছিক 2য় আর্গুমেন্ট রয়েছে, যা আপনাকে অনুরোধ করা বিকল্পটি সেট না করার ক্ষেত্রে রিটার্ন করার জন্য একটি ডিফল্ট মান সেট করতে দেয়। ডিফল্টরূপে, এই যুক্তি মিথ্যা।

একটি পাঠ্য স্ট্রিং পুনরুদ্ধার করতে, এবং বিকল্প টেবিলে পাঠ্য সেট না থাকলে একটি বয়লারপ্লেট স্ট্রিং ব্যবহার করুন, আপনি এটি করতে পারেন:

```
<?php get_option('my_text', "আমার কাছে কিছু লেখা নেই। এখনো!"); ?>
```

## add\_option

add\_option ফাংশন ins অপশন টেবিলে নতুন সারি ঢোকাতে ব্যবহৃত হয়।

এটি বিকল্প সারণীতে একটি নতুন সারি সন্নিবেশ করবে যার নাম some\_option\_name এবং মান হিসেবে some\_option\_value

```
<?php add_option('some_option_name', 'some_option_value'); ?>
```

## ডিলিট\_অপশন

অপশন টেবিল থেকে অপশন মুছে ফেলার জন্য delete\_option ফাংশন ব্যবহার করা হয়।

এটি বিকল্প টেবিল থেকে my\_custom\_option মুছে ফেলবে।

```
<?php delete_option('my_custom_option'); ?>
```

## আপডেট\_বিকল্প

update\_option ফাংশনটি বিকল্প টেবিলে বিদ্যমান একটি মান আপডেট করতে ব্যবহৃত হয়। যদি অপশনটি বিদ্যমান না থাকে, তাহলে বিকল্পটির মান দিয়ে বিকল্পটি যোগ করা হবে।

এটি ডিফল্ট মন্তব্যের স্থিতিকে 'বন্ধ' এ সেট করবে:

```
update_option('default_comment_status', 'closed');
```

## অধ্যায় 42: প্লাগইন উন্নয়ন

### বাক্য গঠন

- `add_action(স্ট্রিং $tag, callable $function_to_add, int $priority = 10, int $accepted_args = 1)`
- `add_filter(string $tag, callable $function_to_add, int $priority = 10, int $accepted_args = 1)`

### পরামিতি

পরামিটার	বিস্তারিত
\$ট্যাগ	(স্ট্রিং) (প্রয়োজনীয়) \$function_to_add কলব্যাককে হুক করার জন্য ফিল্টারের নাম।
\$function_to_add (কলযোগ্য)	(প্রয়োজনীয়) ফিল্টার প্রয়োগ করা হলে কলব্যাক চালানো হবে।
\$ অগ্রাধিকার	(int) (ঐচ্ছিক) একটি নির্দিষ্ট কর্মের সাথে যুক্ত ফাংশনগুলি যে ক্রমানুসারে সম্পাদিত হয় তা নির্দিষ্ট করতে ব্যবহৃত হয়। নিম্ন সংখ্যাগুলি পূর্বের নির্বাহের সাথে সামঞ্জস্যপূর্ণ, এবং একই অগ্রাধিকার সহ ফাংশনগুলি সেই ক্রমে কার্যকর করা হয় যে ক্রমে সেগুলিকে অ্যাকশনে যুক্ত করা হয়েছিল। ডিফল্ট মান: 10
\$accepted_args	(int) (ঐচ্ছিক) ফাংশন গ্রহণ করে আর্গুমেন্টের সংখ্যা। ডিফল্ট মান: 1

### মন্তব্য

প্লাগইন হুকগুলি যেভাবে কাজ করে তা হল যে ওয়ার্ডপ্রেস চলাকালীন বিভিন্ন সময়ে, ওয়ার্ডপ্রেস পরীক্ষা করে যে সেই সময়ে চালানোর জন্য কোনও প্লাগইন রেজিস্টার্ড ফাংশন আছে কিনা এবং যদি তাই হয়, ফাংশনগুলি চালানো হয়। এই ফাংশনগুলি ওয়ার্ডপ্রেসের ডিফল্ট আচরণ পরিবর্তন করে।

দুই ধরনের হুক আছে:

ফিল্টারগুলি আপনাকে ওয়ার্ডপ্রেস কার্যকর করার সময় ডেটার একটি অংশের মান পরিবর্তন করার ক্ষমতা দেয়। ফিল্টারগুলির জন্য কলব্যাক ফাংশনগুলি পরিবর্তনশীল, পরিবর্তিত এবং তারপরে ফেরত দেওয়া হবে। এগুলি একটি বিচ্ছিন্নভাবে কাজ করার জন্য বোঝানো হয় এবং কখনই গ্লোবাল ভেরিয়েবল বা ফাংশনের বাইরে অন্য কিছুকে প্রভাবিত করা উচিত নয়।

বিপরীতে, অ্যাকশন আপনাকে ওয়ার্ডপ্রেস কীভাবে কাজ করে তাতে যোগ করতে বা পরিবর্তন করতে দেয়। আপনার কলব্যাক ফাংশনটি ওয়ার্ডপ্রেসের কার্য সম্পাদনের একটি নির্দিষ্ট বিন্দুতে চলবে এবং ব্যবহারকারীর কাছে আউটপুট প্রতিক্ষণিত করা বা ডাটাবেসে কিছু সন্নিবেশ করার মতো কিছু কাজ সম্পাদন করতে পারে।

[ফিল্টার রেফারেন্স](#)

[অ্যাকশন রেফারেন্স](#)

হ্যান্ডবুক

প্লাগইন API

ফিল্টার বনাম অ্যাকশন

## উদাহরণ

ছাঁকনি

```
add_filter('comment_text','fore_comment');
add_filter('comment_text','after_comment'); ফাংশন
before_comment($comment_text){
    'মন্তব্যের আগে ইনপুট' ফেরত দিন।$comment_text;
}
ফাংশন after_comment($comment_text){
    $comment_text ফেরত দিন।'মন্তব্যের পর ইনপুট';
}
```

কর্ম

```
add_action('wp_head','hook_javascript');

ফাংশন হুক_জাভাস্ক্রিপ্ট() {
    $output="<script> সতর্কতা('পৃষ্ঠা লোড হচ্ছে...'); </script>"; echo $output;
}
}
```

প্লাগইন উন্নয়ন উদাহরণ: প্রিয় গান উইজেট

```
<?php
ফাংশন wpshout_register_widgets() {
    register_widget('Prevorite_Song_Widget');
}

add_action( 'widgets_init', 'wpshout_register_widgets');

ক্লাস ফেভারিট_সং_উইজেট WP_Widget প্রসারিত করে {

ফাংশন প্রিয়_গান_উইজেট() {
    // প্যারেন্ট অবজেক্ট প্যারেন্টকে ইনস্ট্যান্সিয়েট করুন::__construct(

        'favorite_song_widget', // বেস আইডি
        __('পছন্দের গান', 'টেবুট_ডোমেন'), // নাম অ্যারে( 'বিবরণ' => __( 'বাজানো যায় এমন
        প্রিয় গানের জন্য উইজেট', 'টেবুট_ডোমেন'),
    )// আর্গস
    );
}

ফাংশন উইজেট ($args, $instance) echo $args['before_widget']; {
    echo '<h3>প্রিয় গানের তালিকা:</h3>'; echo
    $instance['songinfo']; echo '<a href="' . $instance['link'] . "'>এটি
    ডাউনলোড করুন</a><br>';
```

```

প্রতিধ্বনি $instance['description']; echo
        $args['after_widget'];
}

ফাংশন আপডেট($new_abc,$old_abc) { $instance = $old_abc;

// ক্ষেত্র
$instance['link'] = strip_tags($new_abc['link']); $instance['songinfo'] =
strip_tags($new_abc['songinfo']);
        $instance['description'] = strip_tags($new_abc['description']); ফেরত $উদাহরণ;

}

// উইজেট ফর্ম ক্রিয়েশন ফাংশন ফর্ম($ইনস্ট্যান্স)
{ $link = ""; $songinfo = ""; $description = "";

// মান পরীক্ষা করুন
যদি( $instance) { $link =
        esc_attr($instance['link']); $songinfo =
        esc_textarea($instance['songinfo']); $description =
                esc_textarea($instance['description']);
} ?>

<p>
        <label for="<?php echo $this->get_field_id('link'); ?>"><?php _e('লিঙ্ক', 'wp_widget_plugin'); ?></label> <input class="widefat"
id="<?php echo $this->get_field_id('link'); ?>"
        name="<?php
echo $this->get_field_name('link'); ?>" type="text" value="<?php echo $link; ?>" />
</p>

<p>
        <label for="<?php echo $this->get_field_id('songinfo'); ?>"><?php _e('গানের তথ্য:', 'wp_widget_plugin'); ?></label> <input class="widefat"
id="<?php echo $this->get_field_id('songinfo'); ?>"
        name="<?php
echo $this->get_field_name('songinfo'); ?>" type="text" value="<?php echo $songinfo; ?>" />
</p>

        <p>
<label for="<?php echo $this->get_field_id('description'); ?>"><?php _e('বিবরণ:', 'wp_widget_plugin'); ?></label>

        <textarea class="widefat" id="<?php echo $this->get_field_id('description'); ?>" name="<?php echo $this->get_field_name('description'); ?
>" প্রকার="text" value="<?php echo $description; ?>"></textarea> </p>

        <p><a href="#" id="add-more-tabs"><?php _e('আরো ট্যাব যোগ করুন', 'wp_widget_plugin');
?></a></p>

<?php }
}

```

# অধ্যায় 43: পোস্ট ফরম্যাট

## মন্তব্য

নিম্নলিখিত পোস্ট ফরম্যাটগুলি ব্যবহারকারীদের থেকে বেছে নেওয়ার জন্য উপলব্ধ, যদি থিম তাদের জন্য সমর্থন সক্ষম করে।

মনে রাখবেন যে প্রকৃত পোস্টের বিষয়বস্তু এন্ট্রি পরিবর্তন হবে না, থিমটি বেছে নেওয়া বিন্যাসের উপর ভিত্তি করে পোস্টটি ভিন্নভাবে প্রদর্শন করতে এই ব্যবহারকারীর পছন্দ ব্যবহার করতে পারে। উদাহরণস্বরূপ, একটি থিম একটি "স্থিতি" পোস্টের জন্য শিরোনামের প্রদর্শন বন্ধ করে দিতে পারে। জিনিসগুলি কীভাবে প্রদর্শিত হবে তা সম্পূর্ণভাবে থিমের উপর নির্ভর করে, তবে এখানে কিছু সাধারণ নির্দেশিকা রয়েছে।

- একপাশে - সাধারণত শিরোনাম ছাড়াই স্টাইল করা হয়। একটি ফেসবুক নোট আপডেট অনুরূপ।
- গ্যালারি - ছবির একটি গ্যালারি। পোস্টে সম্ভবত একটি গ্যালারি শর্টকোড থাকবে এবং এতে ছবি সংযুক্তি থাকবে।
- লিঙ্ক - অন্য সাইটের একটি লিঙ্ক। থিমগুলি সেই পোস্টের জন্য বাহ্যিক লিঙ্ক হিসাবে পোস্টের বিষয়বস্তুর প্রথম ট্যাগটি ব্যবহার করতে পারে। একটি বিকল্প পদ্ধতি হতে পারে যদি পোস্টটিতে শুধুমাত্র একটি URL থাকে, তাহলে সেটি হবে URL এবং শিরোনাম (post\_title) হবে অ্যাঙ্করের সাথে সংযুক্ত নাম। চিত্র - একটি একক চিত্র। পোস্টের প্রথম ট্যাগটি চিত্রটিকে বিবেচনা করা যেতে পারে। বিকল্পভাবে, যদি পোস্টে শুধুমাত্র একটি URL থাকে, তাহলে সেটি হবে ছবির URL এবং পোস্টের শিরোনাম (post\_title) হবে ছবির শিরোনাম বৈশিষ্ট্য। উদ্ধৃতি - একটি উদ্ধৃতি। সম্ভবত উদ্ধৃতি সামগ্রী ধারণ করে একটি ব্লককোট থাকবে।
- বিকল্পভাবে, উৎস/লেখক শিরোনাম সহ উদ্ধৃতি শুধুমাত্র বিষয়বস্তু হতে পারে।
- স্ট্যাটাস - একটি ছোট স্ট্যাটাস আপডেট, টুইটার স্ট্যাটাস আপডেটের মতো। ভিডিও - একটি একক ভিডিও বা ভিডিও প্লেলিস্ট। পোস্টের বিষয়বস্তুর প্রথম ট্যাগ বা অবজেক্ট/এম্বেডকে ভিডিও হিসেবে বিবেচনা করা যেতে পারে। বিকল্পভাবে, যদি পোস্টে শুধুমাত্র একটি URL থাকে, তাহলে সেটি হবে ভিডিও URL। ব্লগে ভিডিও সমর্থন সক্ষম করা থাকলে (যেমন একটি প্লাগইনের মাধ্যমে) পোস্টের সংযুক্তি হিসাবে ভিডিওটি থাকতে পারে। অডিও - একটি অডিও ফাইল বা প্লেলিস্ট। পডকাস্টিংয়ের জন্য ব্যবহার করা যেতে পারে।
- চ্যাট - একটি চ্যাট ট্রান্সক্রিপ্ট

## উদাহরণ

থিমে পোস্টের ধরন যোগ করা হচ্ছে

পোস্ট\_টাইপ 'পূষ্ঠা'-তে পোস্ট-ফরম্যাট যোগ করুন

```
add_post_type_support('পূষ্ঠা', 'পোস্ট-ফরম্যাট');
```

পরবর্তী উদাহরণ কাস্টম পোস্ট টাইপ 'my\_custom\_post\_type' নিবন্ধন করে এবং পোস্ট ফরম্যাট যোগ করে।

কাস্টম পোস্ট টাইপ 'my\_custom\_post\_type' নিবন্ধন করুন

```
add_action('init', 'create_my_post_type'); ফাংশন create_my_post_type() {  
  
    রেজিস্টার_পোস্ট_টাইপ ('আমার_কাস্টম_পোস্ট_টাইপ',  
        অ্যারে(  
            'লেবেল' => অ্যারে( 'নাম' => __('পণ্য' ) ), 'পাবলিক' => সত্য  
        )  
    );  
}
```

পোস্ট\_টাইপ 'my\_custom\_post\_type'-এ পোস্ট-ফরম্যাট যোগ করুন

```
add_post_type_support('my_custom_post_type', 'post-formats');
```

অথবা register\_post\_type() ফাংশনে, 'সাপোর্ট করে' প্যারামিটার অ্যাারেতে 'পোস্ট-ফরম্যাট' যোগ করুন। পরবর্তী উদাহরণ উপরের একের সমতুল্য।

'সমর্থন' প্যারামিটার সহ কাস্টম পোস্ট টাইপ 'my\_custom\_post\_type' নিবন্ধন করুন

```
add_action('init', 'create_my_post_type'); ফাংশন create_my_post_type() {  
  
    রেজিস্টার_পোস্ট_টাইপ ('আমার_কাস্টম_পোস্ট_টাইপ',  
        অ্যারে(  
            'লেবেল' => অ্যারে( 'নাম' => __('পণ্য' ) ), 'পাবলিক' => সত্য, 'সমর্থন করে' => অ্যারে( 'শিরোনাম',  
            'সম্পাদক', 'পোস্ট-ফরম্যাট' )  
        )  
    );  
}
```

পোস্টের জন্য থিম সমর্থন যোগ করুন

ফাংশন কল

```
add_theme_support ('পোস্ট-ফরম্যাট')
```

# অধ্যায় 44: প্রশ্ন করা পোস্ট

## বাক্য গঠন

- `$the_query = নতুন WP_Query($args);` •  
`$posts_array = get_posts($args);`

## পরামিতি

পরামিতি বর্ণনা	
<code>\$args</code>	(অ্যারে) একটি প্রশ্নের জন্য প্রয়োজনীয় আর্গুমেন্টের একটি বিন্যাস - আপনার প্রয়োজন অনুসারে কাস্টম করা যেতে পারে, যেমন শুধুমাত্র একটি বিভাগ থেকে পোস্ট জিজ্ঞাসা করা, কাস্টম পোস্টের ধরন থেকে বা এমনকি নির্দিষ্ট শ্রেণিবিন্যাস জিজ্ঞাসা করা

## মন্তব্য

কোয়্যারী আর্গুমেন্ট অনেক. `WP_Query()` কোডেব্ল পৃষ্ঠায় পরামিতিগুলির একটি তালিকা রয়েছে। তাদের মধ্যে কিছু হয়

- [লেখক পরামিতি](#)
- [বিভাগ পরামিতি](#) • [ট্যাগ](#)  
[প্যারামিটার](#) • [শ্রেণীবিন্যাস](#)  
[পরামিতি](#) • [অনুসন্ধান পরামিতি](#) •  
[পোস্ট](#) এবং [পৃষ্ঠা প্যারামিটার](#)
- [পাসওয়ার্ড প্যারামিটার](#)
- [টাইপ প্যারামিটার](#) • [স্থিতি](#)  
[পরামিতি](#) • [পেজিনেশন](#)  
[প্যারামিটার](#) • [অর্ডার](#) এবং [অর্ডারবি](#)  
[পরামিতি](#) • [তারিখ পরামিতি](#) • [কাস্টম ফিল্ড](#)  
[প্যারামিটার](#) • [অনুমতি](#)  
[পরামিতি](#) • [মাইম টাইপ প্যারামিটার](#) •  
[ক্যাশিং পরামিতি](#) • [রিটার্ন ফিল্ড](#)  
[প্যারামিটার](#)

মনে রাখা সবচেয়ে গুরুত্বপূর্ণ জিনিস এক হল:

কখনই `query_posts()` ব্যবহার করবেন না

`query_posts()` প্রধান কোয়্যারীকে ওভাররাইড করে এবং আপনার বাকি থিমে সমস্যা সৃষ্টি করতে পারে। যে কোন

আপনার প্রধান ক্যোয়ারী (অথবা সেই বিষয়ে যেকোন প্রশ্ন) পরিবর্তন করার সময় হল `pre_get_posts` ব্যবহার করা ছাঁকনি.

এটি আপনাকে এটি চালানোর আগে অনুসন্ধানটি সংশোধন করার অনুমতি দেবে।

এছাড়াও আপনি যখন পোস্টগুলি জিজ্ঞাসা করছেন, আপনার সর্বদা `wp_reset_postdata()` ব্যবহার করে এটি পুনরায় সেট করা উচিত। এটি মূল ক্যোয়ারী লুপের গ্লোবাল `$post` ডেরিয়েবলকে পুনরুদ্ধার করবে এবং পরবর্তীতে আপনার কোন সমস্যা হবে না (যেমন বিভাগগুলি বাদ দেওয়া হচ্ছে, কারণ আপনার সেকেন্ডারী লুপে আপনি সেগুলি বাদ দিয়েছেন এবং ক্যোয়ারী রিসেট করতে ভুলে গেছেন)।

## উদাহরণ

`WP_Query()` অবজেক্ট ব্যবহার করে

`WP_Query` অবজেক্টের একটি পৃথক উদাহরণ তৈরি করা সহজ:

```
$query_args = অ্যারে(
    'post_type' => 'পোস্ট', 'পোস্ট_পার_পৃষ্ঠা'
    => 10
);

$my_query = নতুন WP_Query($query_args);

যদি( $my_query->have_posts() ):
    যখন( $my_query->have_posts() ): $my_query->the_post();
        //আমার কাস্টম ক্যোয়ারী লুপ শেষ পর্যন্ত;
    যদি শেষ;
```

`wp_reset_postdata();`

লক্ষ্য করুন যে আপনাকে আপনার স্পেসিফিকেশনে ক্যোয়ারী আর্গুমেন্ট অ্যারে তৈরি করতে হবে। আরো বিস্তারিত জানার জন্য, `WP_Query` কোডেক্স পৃষ্ঠা দেখুন ।

`get_posts()` ব্যবহার করে

`get_posts()` হল একটি `WP_Query` অবজেক্টের একটি পৃথক উদাহরণের জন্য একটি মোড়ক। প্রত্যাবর্তিত মান হল পোস্ট অবজেক্টের একটি অ্যারে।

```
বিশ্বব্যাপী $পোস্ট;
```

```
$args =
    অ্যারে( 'numberposts' => 5, 'offset'
    => 1, 'category' => 1
);

$myposts = get_posts($args);

foreach ($myposts $post হিসাবে) :
    setup_postdata($post); ?> <h2><a href="<?
    php the_permalink(); ?>"><?php the_title(); ?></a></h2> <?php endforeach; wp_reset_postdata(); ?>
```



## অধ্যায় 45: থেকে অটো লাইন ব্রেকগুলি সরান

### বিষয়বস্তু এবং উদ্ভূতি

#### ভূমিকা

যে সাইটগুলো এডিটর বা উদ্ভূতাংশে হাত দিয়ে HTML এর উপর নির্ভর করে, যেগুলোকে আপনি নিজে কোড করতে চান, অটো লাইন ব্রেক একটি বিরক্তিকর হতে পারে। আপনি এই ফিল্টার অপসারণ করে তাদের নিষ্ক্রিয় করতে পারেন।

#### মন্তব্য

এগুলি অবশ্যই অন্তর্ভুক্ত ফাইলে সরাসরি কার্যকর করা উচিত। এটি functions.php-এ হোক বা অন্য ইনক্লুড ফাইলে হোক, এগুলোকে হকে মোড়ানো যাবে না। তারা init বা অন্য কোন কাজ করবে না যা আমি এখন পর্যন্ত পেয়েছি।

শুধুমাত্র সেই টেমপ্লেটের জন্য চালানোর জন্য page.php-এর মতো একটি টেমপ্লেটে সরাসরি অন্তর্ভুক্ত করা যেতে পারে।

দ্রষ্টব্য: এটিকে ডিস্ট্রিবিউটেড থিম বা প্লাগইনে অন্তর্ভুক্ত করবেন না (যদি না এটি ডিফল্টরূপে অক্ষম করা হয়, যেমন ব্যবহারকারী নির্দিষ্ট না করা পর্যন্ত অন্তর্ভুক্ত ফাইলটি অন্তর্ভুক্ত না করে)।

এটি এমন একটি সাইটে অন্তর্ভুক্ত করা খারাপ অভ্যাস যা আপনি নিয়ন্ত্রণ করেন না কারণ এটি অন্য কোনো থিম বা প্লাগইনগুলির আউটপুটকে ভেঙে দিতে পারে।

#### উদাহরণ

##### ফিল্টারগুলি সরান

```
// সামগ্রী থেকে অটো-অনুচ্ছেদ এবং অটো-লাইন-ব্রেক সরান remove_filter( 'the_content', 'wpautop' );

// উদ্ভূতি থেকে অটো-অনুচ্ছেদ এবং অটো-লাইন-ব্রেক সরান remove_filter( 'the_excerpt', 'wpautop' );
```

##### ফিল্টার অপসারণ ফাংশন

```
/**
 * বিষয়বস্তু এবং উদ্ভূতিগুলি থেকে স্বয়ংক্রিয় লাইন বিরতিগুলি সরান।
 *
 * @ ১.০.০ থেকে
 */
ফাংশন remove_content_auto_line_breaks() {
    // সামগ্রী থেকে অটো-অনুচ্ছেদ এবং অটো-লাইন-ব্রেক সরান remove_filter( 'the_content', 'wpautop' );

    // উদ্ভূতি থেকে অটো-অনুচ্ছেদ এবং অটো-লাইন-ব্রেক সরান remove_filter( 'the_excerpt', 'wpautop' );
}
}
```

```
// रिमुड_कन्टेक्ट_अटो_लाइन_ब्रेकस();
```

# অধ্যায় 46: ওয়ার্ডপ্রেস এবং স্টাইলশীট থেকে সংস্করণ সরান

## ভূমিকা

অন্যদের জন্য আপনার ওয়েবসাইট হ্যাক করা আরও কঠিন করার জন্য আপনি আপনার সাইট, আপনার CSS এবং js থেকে ওয়ার্ডপ্রেস সংস্করণ নম্বর মুছে ফেলতে পারেন। সেই সংখ্যাটি ছাড়া আপনি পুরানো সংস্করণগুলি থেকে বাগগুলিকে কাজে লাগাতে বর্তমান সংস্করণটি চালান না কিনা তা দেখা সম্ভব নয়।

উপরন্তু এটি আপনার সাইটের লোডিং গতি উন্নত করতে পারে, কারণ URL-এ কেয়ারী স্ট্রিং ছাড়াই CSS এবং js ফাইলগুলি ক্যাশে করা যেতে পারে।

## বাক্য গঠন

- `add_filter( $tag, $function_to_add, $priority, $accepted_args)`

## পরামিতি

প্যারামিটার	বিস্তারিত
<code>\$ট্যাগ</code>	(স্ট্রিং প্রয়োজনীয়) ফিল্টারের নাম যেখানে <code>\$function_to_add</code> হক করা আছে
<code>\$function_to_add</code>	(কলযোগ্য প্রয়োজন) ফিল্টার প্রয়োগ করার সময় যে ফাংশনটি চলে তার নাম
<code>\$ অগ্রাধিকার</code>	একটি ক্রিয়ায় অন্যান্য ফাংশনের মধ্যে <code>\$function_to_add</code> এর স্থান (ডিফল্ট = 10)
<code>\$accepted_args</code>	(int ঐচ্ছিক) প্যারামিটারের সংখ্যা যা <code>\$function_to_add</code> গ্রহণ করে (ডিফল্ট = 1)

## মন্তব্য

সাইটের গতি এবং নিরাপত্তা উন্নত করার উদ্দেশ্যে।

## উদাহরণ

CSS/js থেকে সংস্করণ নম্বরগুলি সরান

শুধু আপনার `functions.php` এই ফাংশন যোগ করুন. এটি সমস্ত সারিবদ্ধ জেএস এবং সিএসএস ফাইল থেকে সংস্করণটি সরিয়ে দেবে।

```
ফাংশন remove_cssjs_ver( $src ) if( strpos( $src, '?
ver=' ) $src = remove_query_arg('ver',
$src); ফেরত $src;

}

add_filter('style_loader_src', 'remove_cssjs_ver', 999); add_filter('script_loader_src',
'remove_cssjs_ver', 999);
```

ওয়ার্ডপ্রেস থেকে সংস্করণ নম্বর সরান

আপনি যদি এটি আপনার functions.php এ যোগ করেন তবে এটি আরএসএস ফিড এবং হেডার থেকে ওয়ার্ডপ্রেস সংস্করণ নম্বর সরিয়ে দেয়।

```
ফাংশন remove_wordpress_ver() {
    ফেরত "";
} add_filter('the_generator', 'remove_wordpress_ver', 999);
```

# অধ্যায় 47: REST API

## ভূমিকা

ওয়ার্ডপ্রেস REST API ওয়ার্ডপ্রেস ডেটা প্রকারের জন্য API এন্ডপয়েন্ট প্রদান করে যা ডেভেলপারদের JSON (জাভাস্ক্রিপ্ট অবজেক্ট নোটেশন) অবজেক্ট পাঠানো এবং গ্রহণ করার মাধ্যমে দূর থেকে সাইটগুলির সাথে ইন্টারঅ্যাক্ট করতে দেয়।

আপনি যখন এপিআই-এ কন্টেন্ট পাঠান বা অনুরোধ করেন, তখন প্রতিক্রিয়া JSON-এ ফেরত দেওয়া হবে। এটি ডেভেলপারদের ক্লায়েন্ট-সাইড জাভাস্ক্রিপ্ট বা বাহ্যিক অ্যাপ্লিকেশনগুলি থেকে ওয়ার্ডপ্রেস সামগ্রী তৈরি করতে, পড়তে এবং আপডেট করতে সক্ষম করে, এমনকি PHP-এর বাইরের ভাষায় লেখা।

## মন্তব্য

আপনার জন্য কাজ করার জন্য এই ওয়ার্ডপ্রেস REST API সহজ উদাহরণ পেতে, আপনাকে আরও বিস্তারিতভাবে এটি কীভাবে কাজ করে তা শিখতে হবে। অফিসিয়াল ডকুমেন্টেশন এই বিষয়ে শেখার সুপারিশ করে:

1. রুট/এন্ডপয়েন্ট - যা "এন্ডপয়েন্ট" নামে পরিচিত রুটে পৃথক HTTP পদ্ধতির ম্যাপিং - আপনি এটি `register_rest_route()` ফাংশন ব্যবহার করে করেন, এবং এখানে আপনি রুট এবং এন্ডপয়েন্ট সম্পর্কে আরও জানতে পারেন।
2. অনুরোধ - ওয়ার্ডপ্রেস REST API `WP_REST_Request` ক্লাস সংজ্ঞায়িত করে যা বর্তমান অনুরোধের জন্য তথ্য সংরক্ষণ এবং পুনরুদ্ধার করতে ব্যবহৃত হয়। আপনি যখনই একটি নিবন্ধিত রুটে একটি HTTP অনুরোধ করেন তখনই `WP_REST_Request` বস্তুগুলি আপনার জন্য স্বয়ংক্রিয়ভাবে তৈরি হয়। অনুরোধে উল্লিখিত ডেটা নির্ধারণ করবে যে আপনি API থেকে কী প্রতিক্রিয়া ফিরে পাবেন। এখানে `WP_REST_Request` ক্লাস সম্পর্কে আরও জানতে পারবেন।
3. প্রতিক্রিয়া - আপনি API থেকে ফিরে পেতে ডেটা। `WP_REST_Response` এন্ডপয়েন্ট দ্বারা প্রত্যাবর্তিত প্রতিক্রিয়া ডেটার সাথে ইন্টারঅ্যাক্ট করার একটি উপায় প্রদান করে। আপনার এন্ডপয়েন্ট সংজ্ঞায় আপনি কলব্যাক (প্রতিক্রিয়া) ফাংশনের নাম দেন আপনার মিথস্ক্রিয়া পরিবেশনের জন্য। এখানে `WP_REST_Response` ক্লাস সম্পর্কে আরও জানতে পারবেন।
4. স্কিমা - প্রতিটি এন্ডপয়েন্টের প্রয়োজন হয় এবং কিছুটা আলাদা ডেটা স্ট্রাকচার প্রদান করে এবং সেই স্ট্রাকচারগুলি API স্কিমাতে সংজ্ঞায়িত করা হয়। আপনি যদি রক্ষণাবেক্ষণযোগ্য, আবিষ্কারযোগ্য এবং সহজে এক্সটেনসিবল এন্ডপয়েন্ট চান তবে স্কিমা ব্যবহার করার পরামর্শ দেওয়া হয়। এখানে আপনি স্কিমা সম্পর্কে আরও জানতে পারেন।
5. কন্ট্রোলার ক্লাস - তারা সমস্ত উপাদানকে এক জায়গায় একত্রিত করে। একটি কন্ট্রোলার ক্লাসের সাহায্যে আপনি রুট এবং এন্ডপয়েন্টগুলির নিবন্ধন পরিচালনা করতে পারেন, অনুরোধগুলি পরিচালনা করতে পারেন, স্কিমা ব্যবহার করতে পারেন এবং API প্রতিক্রিয়াগুলি তৈরি করতে পারেন। আপনি ইতিমধ্যে দুটি কন্ট্রোলার ক্লাস সম্পর্কে শিখেছেন: `WP_REST_Request` এবং `WP_REST_Response`। এখানে আপনি কন্ট্রোলার ক্লাস সম্পর্কে আরও জানতে পারবেন।

দ্রষ্টব্য: এই তথ্যের কিছু অফিসিয়াল Wordpress REST API হ্যান্ডবুক থেকে নেওয়া হয়েছে

## উদাহরণ

## সম্পূর্ণ কাজের উদাহরণ

```
add_action('rest_api_init', 'my_rest_validate_endpoint'); ফাংশন my_rest_validate_endpoint() {

    // আমাদের নামস্থান ঘোষণা করুন $namespace
    = 'myrest/v1';

    // রুট নিবন্ধন করুন
    // এই রুটের সাথে মিলে যাওয়া ইউআরএলের উদাহরণ: // http://
    yourdomain/wp-json/myrest/v1/guides/tag=europe/price=29 register_rest_route($namespace,

        // রেগুলার এক্সপ্রেশন ব্যবহার করে আমরা প্রাথমিকভাবে ইনপুট যাচাই করতে পারি '/guides/tag=(?P<tag>[a-zA-Z0-9-]+)/
        price=(?P<price>[0-9]+ )', // আমাদের একটি রুট অ্যারের জন্য একাধিক এন্ডপয়েন্ট থাকতে পারে( array( 'methods' =>
        'GET', 'callback' => 'my_get_guides_handler'

        )
    ),
    // আমরা আমাদের স্কিমা কলব্যাক নিবন্ধন করতে পারি // 'schema' =>
    'my_get_guide_schema',

    );

    // আপনি এখানে একইভাবে অন্য রুট নিবন্ধন করতে পারেন
}

// এন্ডপয়েন্ট ফাংশনের জন্য কলব্যাক হ্যান্ডলার
my_get_guides_handler(WP_REST_Request $request) {

    // প্যারামিটার পান: $tag = $request-
    >get_param('tag'); $price = $request->get_param('price');

    // পরামিতিগুলির সাথে কিছু করুন // উদাহরণস্বরূপ: DB থেকে একটি
    অ্যারেতে মিলিত গাইডগুলি পান - $results // ...

    // প্রতিক্রিয়া প্রস্তুত করুন $message =
    "আমরা পেয়েছি " $message.=" (একটি ট্যাগ . গণনা ($ ফলাফল)। "গাইড"; . $ট্যাগ। ", একটি মূল্য ট্যাগ সহ:
    অনুসন্ধান করা হচ্ছে: " . $মূল্য। "));

    // প্রতিক্রিয়া স্বয়ংক্রিয়ভাবে JSON ফর্ম্যাটে রূপান্তরিত হয়ে নতুন WP_REST_Response( array( 'results' => $results,
    'message' => $message, 'status' => 'OK'

    ),
    200 );

}
```

# অধ্যায় 48: এর সাথে ওয়ার্ডপ্রেস স্থানীয় চালান XAMPP

## ভূমিকা

XAMPP এর মাধ্যমে আপনি আপনার স্থানীয় পিসিতে একটি ওয়েব সার্ভার ইনস্টল করতে পারেন। এটিতে একটি অ্যাপাচি ওয়েব সার্ভার রয়েছে, ডাটাবেস মারিয়াডিবি (মাইএসকিউএল) এবং এটি পার্ল এবং পিএইচপি এর সাথে কাজ করে। ইনস্টলেশনের পরে আপনি আপনার স্থানীয় পিসিতে ওয়ার্ডপ্রেসের মতো বিষয়বস্তু ব্যবস্থাপনা সিস্টেমগুলি চালাতে এবং ডিবাগ করতে সক্ষম হবেন। আপনি এটি উইন্ডোজ, লিনাক্স, ম্যাক ওএস এক্স এবং সোলারিসের সাথে ব্যবহার করতে পারেন।

## উদাহরণ

### 1. XAMPP ইনস্টল করা

1. বর্তমান সংস্করণের জন্য ইনস্টলার ডাউনলোড করুন। XAMPP 2 এর।

ইনস্টলেশন উইজার্ডের মাধ্যমে যান এবং আপনি কি পরিবর্তন করছেন তা নিশ্চিত না হলে কিছুই পরিবর্তন করবেন না। 3.

ইনস্টলেশনের পরে আপনি XAMPP কন্ট্রোল প্যানেল দেখতে পাবেন। শুধুমাত্র Apache এবং MySQL-এর স্টার্ট বোতামে ক্লিক করুন উভয়ই মৌলিক কনফিগারেশনের সাথে চালানোর জন্য। 4.

যদি আপনার কম্পিউটার একটি স্থানীয় নেটওয়ার্কের সাথে সংযুক্ত থাকে, এবং আপনি অন্য কম্পিউটার থেকেও ওয়েব সার্ভার অ্যাক্সেস করতে চান, তাহলে নিশ্চিত হন যে আপনার ফায়ারওয়াল এটির অনুমতি দেয়।

### 2. XAMPP ইনস্টল করার পরে একটি ডাটাবেস সেটআপ করুন

আপনার কম্পিউটারে ওয়ার্ডপ্রেস চালানোর জন্য আপনাকে প্রথমে একটি ডাটাবেস কনফিগার করতে হবে। নিশ্চিত হন যে Apache এবং MySQL চলছে ( XAMPP ইনস্টল করা দেখুন ধাপ 3)

1. আপনার পছন্দের একটি ওয়েব ব্রাউজার শুরু করুন এবং ঠিকানায় "লোকালহোস্ট" লিখুন।
2. আপনার পছন্দের ভাষা চয়ন করুন এবং "সরঞ্জাম" -> phpMyAdmin বিভাগে নির্বাচন করুন।
3. ডাটাবেস ট্যাব নির্বাচন করুন।
4. "ডেটাবেস তৈরি করুন" এর নীচে আপনার ডাটাবেসের নাম লিখুন (আপনার এই নামটি পরে প্রয়োজন হবে) এবং ড্রপডাউন "কোলেশন" থেকে utf8\_general\_ci বেছে নিন।
5. বাম দিকে আপনার তৈরি ডাটাবেসে ক্লিক করুন এবং ট্যাব ব্যবহারকারী নির্বাচন করুন।
6. ব্যবহারকারী যোগ করুন এ ক্লিক করে একটি নতুন ব্যবহারকারী যোগ করুন। 7.  
আপনার ব্যবহারকারীর নাম লিখুন, "হোস্ট" ড্রপডাউনে স্থানীয় চয়ন করুন (যদি আপনি নেটওয়ার্কের অন্যান্য কম্পিউটার থেকে ডাটাবেস অ্যাক্সেস করতে চান তবে যে কোনও হোস্ট চয়ন করুন) এবং আপনার পাসওয়ার্ড লিখুন (পরে আপনার এটিরও প্রয়োজন হবে)।
8. "ব্যবহারকারীর জন্য ডাটাবেস" নীচে ওয়াইল্ডকার্ড নামের উপর সমস্ত সুবিধা প্রদান করুন চেক করা আছে কিনা তা পরীক্ষা করুন।
9. Go টিপুন।

### 3. ডাটাবেস সেটআপ করার পরে ওয়ার্ডপ্রেস ইনস্টল করা

আপনি XAMPP ইনস্টল করার পরে এবং MySQL ডাটাবেস সেটআপ করার পরে আপনি ওয়ার্ডপ্রেস ইনস্টল করতে পারেন।

1. ডাউনলোড করুন ওয়ার্ডপ্রেসের সর্বশেষ সংস্করণ।

ফাইলটি আনজিপ করুন এবং আপনার ওয়েব সার্ভারের রুট ডিরেক্টরিতে ফোল্ডারটি সন্নিবেশ করুন (যদি আপনি XAMPP সেটআপের সময় 2. প্রস্তাবিত পথটি ব্যবহার করেন তবে এটি "c:\xampp\htdocs")।

3. সমস্ত ফাইল যা বিদ্যমান দুইবার প্রতিস্থাপন করা যেতে পারে।

4. আপনার ওয়েব ব্রাউজারের ঠিকানা ক্ষেত্রে লিখুন "localhost/wordpress/wp-admin/install.php"।

5. আপনার ভাষা বেছে নেওয়ার পর Continue- এ ক্লিক করুন এবং তারপর চলুন। 6.

আপনার নির্বাচিত কনফিগারেশনের সাথে সমস্ত ডিফল্ট পাঠ্য ক্ষেত্র প্রতিস্থাপন করুন ( "ডেটাবেস সেটআপ করুন" দেখুন)।

"ডাটাবেস হোস্ট" এবং "টেবিল উপসর্গ" পরিবর্তন করবেন না, আপনি আপনার স্থানীয় ওয়ার্ডপ্রেস ইনস্টলেশন অনলাইনে স্থানান্তর করতে চান না। তারপর মনে রাখবেন যে নিরাপত্তা গুরুত্বপূর্ণ। আপনি নতুন ওয়ার্ডপ্রেস ইনস্টলেশনে সেট প্রিফিক্সে আরও তথ্য পাবেন।

7. সাবমিট টিপুন এবং ইনস্টল করার চেয়ে রান করুন।

8. এখন আপনাকে আপনার সাইটের নাম, একটি ব্যবহারকারীর নাম, পাসওয়ার্ড এবং একটি বৈধ ই-মেইল লিখতে হবে।

9. সেটআপ শেষ করতে ওয়ার্ডপ্রেস ইনস্টল করুন ক্লিক করুন এবং আপনার নতুন স্থানীয় ওয়ার্ডপ্রেস সাইটে লগ ইন করুন।

## অধ্যায় 49: আপনার ইনস্টলেশন সুরক্ষিত করুন

### মন্তব্য

ওয়ার্ডপ্রেস ওয়েবসাইটগুলি প্রায়শই হ্যাক হয়। এই বিষয়টি এমন কৌশল এবং অনুশীলনের জন্য যা আপনার ওয়ার্ডপ্রেস ইনস্টলেশনের নিরাপত্তা বেস ইন্সটলে অর্জনের চেয়েও বাড়িয়ে দেয়।

এই বিষয়টি ছাড়াও, একটি ওয়ার্ডপ্রেস ইনস্টলেশন সুরক্ষিত করার বিষয়ে পড়ার জন্য আরেকটি ভাল জায়গা হল হার্ডনিং ওয়ার্ডপ্রেস কোডেক্স পৃষ্ঠা।

### উদাহরণ

ফাইল এডিটর অক্ষম করুন

ওয়ার্ডপ্রেসের সাথে যে ফাইল এডিটর পাঠানো হয় সেটি একটি নিরাপত্তা ঝুঁকি। যদি কোনো আক্রমণকারী আপনার ওয়ার্ডপ্রেস ওয়েবসাইটে প্রশাসক অ্যাক্সেস লাভ করে তবে তারা সহজেই থিম এবং প্লাগইন ফাইলগুলিতে দূষিত কোড সন্নিবেশ করতে সক্ষম হবে। এটি এমন ক্লায়েন্টদের সাথেও একটি ঝুঁকি যারা তারা জানেন না তারা কী করছেন। একবার ফাইল এডিটরে কোলন ভুল স্থানান্তরিত হলে একটি সাইট ভেঙ্গে যেতে পারে এবং ব্রাউজার থেকে এটিকে অ্যাক্সেসযোগ্য করে তুলতে পারে।

আপনার ওয়ার্ডপ্রেস wp-config.php ফাইলে, কোডের নিম্নলিখিত লাইন যোগ করে ফাইল সম্পাদক নিষ্ক্রিয় করুন।

```
সংজ্ঞায়িত ('DISALLOW_FILE_EDIT', সত্য);
```

আপনার থিমের functions.php ফাইলে যোগ করা হলে সেই লাইনটি পছন্দসই প্রভাব ফেলবে তবে wp-config.php এ যোগ করা ভাল।

আপনি যদি ওয়ার্ডপ্রেস CLI ব্যবহার করেন ওয়ার্ডপ্রেস ইন্সটল করতে আপনি নিচের কমান্ডটি ব্যবহার করে একটি wp-config.php ফাইল তৈরি করতে পারেন যেখানে ফাইল এডিটিং অক্ষম আছে।

```
/* আগে থেকেই ডেরিয়েবল ঘোষণা করুন বা */ wp core config --dbname="$MYSQL_DBNAME" --
dbuser="$MYSQL_USERNAME" --dbpass="$MYSQL_PASS" --dbprefix="$WP_DBPREFIX" --locale-এ স্থিতিগুলি প্রতিস্থাপন করুন =en_AU --extra-php <<PHP
define('DISALLOW_FILE_EDIT', true);
```

পিএইচপি

আপনি যদি স্ক্রিপ্ট সহ ওয়ার্ডপ্রেস ইনস্টল করেন তবে এই পদ্ধতিটি কার্যকর।

### wp-config.php সরান

ওয়ার্ডপ্রেস ইন্সটলের সবচেয়ে সংবেদনশীল তথ্য wp-config.php ফাইলে সংরক্ষিত থাকে। যদি কোনও হ্যাকার এই ফাইলটিতে অ্যাক্সেস পায় তবে তাদের আপনার ওয়েবসাইটের সম্পূর্ণ নিয়ন্ত্রণ থাকবে।

ডিফল্টরূপে wp-config.php ওয়ার্ডপ্রেস ইনস্টল ফোল্ডারে সংরক্ষণ করা হয়। এই ফাইলটিকে চুরি করা কঠিন করতে আপনি এটিকে ওয়েব অ্যাক্সেসযোগ্য ফোল্ডার থেকে সরাতে পারেন। আপনি যদি এটিকে উপরে একটি ফোল্ডার সরান তবে ওয়ার্ডপ্রেস স্বয়ংক্রিয়ভাবে এটি খুঁজে পাবে। আপনি যদি wp-config.phpকে অন্য জায়গায় নিয়ে যান, তাহলে ওয়ার্ডপ্রেস ইনস্টলেশন ফোল্ডারে wp-config.php নামে একটি খালি ফাইল তৈরি করুন। তারপর নিম্নলিখিত যোগ করুন:

```
define('ABSPATH', dirname(__FILE__) . '/'); // '../wp-config.php' ইনস্টলেশন  
ফোল্ডারের উপরে দুটি ফোল্ডারের অবস্থান নির্ধারণ করে।  
// প্রয়োজনে wp-config.php ফাইলের প্রকৃত অবস্থান সহ বিকল্প। need_once(ABSPATH . '../wp-config.php');
```

আপনি যে ফোল্ডারে wp-config.php রাখবেন সেই ফোল্ডারে আপনাকে php এক্সিকিউটেবল করতে হতে পারে। আপনার যতটা সম্ভব কম ফোল্ডারে php এক্সিকিউটেবল করা উচিত। একটি ভালো সিস্টেম ওয়ার্ডপ্রেস ইন্সটলকে /path/to/wordpress/install/ এবং কনফিগারেশনকে /path/to/wordpress/config এ রাখা। আপনি নিশ্চিত করবেন যে কনফিগার ফোল্ডারটি ওয়েব অ্যাক্সেসযোগ্য নয় এবং ফোল্ডারের অনুক্রমে /path/to/ বা উচ্চতর কোনও সংবেদনশীল তথ্য স্থাপন করবেন না। সেক্ষেত্রে আপনি আপনার php.ini তে নিচের মত একটি লাইন লিখবেন:

```
open_basedir = "/path/to/wordpress/install;/path/to/wordpress/config"
```

এই কৌশলটি বিতর্কিত এবং কিছু লোক মনে করে না যে এটি নিরাপত্তা বাড়ায়। বিষয়ের উপর বিস্তৃত আলোচনা এই ওয়ার্ডপ্রেস স্ট্যাকএক্সচেঞ্জ প্রশ্নে পড়া যেতে পারে।

ওয়ার্ডপ্রেস টেবিলের জন্য একটি কাস্টম উপসর্গ সেট করুন

আপনি যখন আপনার সার্ভারে ওয়ার্ডপ্রেস ইনস্টল করবেন, তখন ইনস্টলেশন স্ক্রিপ্টটি সমস্ত ওয়ার্ডপ্রেস মাইএসকিউএল টেবিলের নামের সামনে একটি উপসর্গ স্থাপন করবে। এই উপসর্গটি ডিফল্টরূপে 'wp\_' এ সেট করা আছে। ওয়ার্ডপ্রেস পোস্ট টেবিল উদাহরণস্বরূপ বলা হবে wp\_posts . টেবিল উপসর্গ পরিবর্তন করে আপনি অস্পষ্টতা দ্বারা কিছু নিরাপত্তা তৈরি করতে পারেন। এইভাবে যখন একজন হ্যাকার এসকিউএল ইনজেকশন আক্রমণের চেষ্টা করে, তখন তাদের শুধু 'wp\_' ব্যবহার না করে আপনার টেবিলের উপসর্গটি অনুমান করতে হবে। আপনি এই উপসর্গটি আপনার পছন্দ মতো সেট করতে পারেন।

নতুন ওয়ার্ডপ্রেস ইন্সটলেশনে প্রিফিক্স সেট করুন

বিখ্যাত 5 মিনিট ইনস্টলেশন ব্যবহার করলে ইনস্টলেশনের সময় স্ক্রিপ্টের উপসর্গ পরিবর্তন করুন।



Below you should enter your database connection details. If you're not sure about these, contact your host.

<b>Database Name</b>	<input type="text" value="wordpress"/>	The name of the database you want to use with WordPress.
<b>Username</b>	<input type="text" value="username"/>	Your database username.
<b>Password</b>	<input type="text" value="password"/>	Your database password.
<b>Database Host</b>	<input type="text" value="localhost"/>	You should be able to get this info from your web host, if localhost doesn't work.
<b>Table Prefix</b>	<input type="text" value="wp_"/>	If you want to run multiple WordPress installations in a single database, change this.

ওয়ার্ডপ্রেস CLI এর মাধ্যমে ইনস্টল করা হলে নিম্নলিখিত কমান্ডটি ব্যবহার করুন:

```
// উপরে অন্যান্য ভেরিয়েবল সেট করুন, অথবা আপনার স্ট্রিংগুলি প্রতিস্থাপন করুন।
WP_DBPREFIX=foo wp
core config --dbname="$MYSQL_DBNAME" --dbuser="$MYSQL_USERNAME" --dbpass="$MYSQL_PASS" --dbprefix="$WP_DBPREFIX_" --locale=en_AU
```

বিদ্যমান ইনস্টলেশনে উপসর্গ পরিবর্তন করুন

উপসর্গ পরিবর্তন করা একটু বেশি কঠিন। config.php ফাইল এডিট করতে প্রথমে FileZilla এর মত FTP প্রোগ্রাম ব্যবহার করুন। এন্ট্রি wp পরিবর্তন করুন \$table\_prefix = 'wp\_'; to \$table\_prefix = 'foo\_'; আপনার পছন্দসই উপসর্গের জন্য 'foo' প্রতিস্থাপন করা হচ্ছে।

পরবর্তীতে আমাদের ডাটাবেস সম্পাদনা করতে হবে। আপনার যদি phpMyAdmin অ্যাক্সেস থাকে, লগইন করুন এবং নিম্নলিখিতগুলি করুন:

- ওয়ার্ডপ্রেস ডাটাবেস নির্বাচন করুন

<input type="checkbox"/> wp_links	Browse Structure
<input type="checkbox"/> wp_options	Browse Structure
<input type="checkbox"/> wp_postmeta	Browse Structure
<input type="checkbox"/> wp_posts	Browse Structure

- সমস্ত টেবিল নির্বাচন করুন এবং ড্রপডাউনে টেবিল প্রিফিক্স প্রতিস্থাপন করুন।

The screenshot shows the phpMyAdmin interface with a dropdown menu open. The menu is titled 'With selected:' and contains the following options: 'With selected:', 'Export', 'Print view', 'Empty', 'Drop', 'Check table', 'Optimize table', 'Repair table', 'Analyze table', 'Add prefix to table', 'Replace table prefix' (highlighted in blue), and 'Copy table with prefix'. In the background, there is a 'Create table' form with a 'Name:' field and a 'Check All' checkbox.

- "থেকে" টাইপ করুন 'wp\_'। "টু" এ আপনার উপসর্গ টাইপ করুন, এই উদাহরণে 'foo\_' এবং "জমা দিন" টিপুন।

The screenshot shows the 'Replace table prefix' dialog box. It has a title 'Replace table prefix:'. Below the title, there are two input fields: 'From' with the value 'wp\_' and 'To' with the value 'foo\_'. At the bottom right of the dialog, there is a 'Submit' button.

- টেবিলগুলি এখন এইরকম হওয়া উচিত:

<input type="checkbox"/> foo_links	Browse Structure
<input type="checkbox"/> foo_options	Browse Structure
<input type="checkbox"/> foo_postmeta	Browse Structure
<input type="checkbox"/> foo_posts	Browse Structure

আপনি যদি phpMyAdmin ব্যবহার করতে না পারেন তবে নিম্নলিখিত MySQL কমান্ডটি ব্যবহার করুন:

সারণী RENAME করুন 'wp\_comments' থেকে 'foo\_comments'


আপনাকে প্রতিটি টেবিলের জন্য সেই কমান্ডটি চালাতে হবে, অন্য টেবিলের নামের জন্য 'মন্তব্য' প্রতিস্থাপন করতে হবে।

পরবর্তীতে আমাদের কিছু টেবিলে কয়েকটি এন্ট্রি পরিবর্তন করতে হবে। 'foo\_options' টেবিলে এই প্রশ্নটি চালান





































foo\_options থেকে \* নির্বাচন করুন যেখানে বিকল্প\_নাম লাইক '%user\_roles%'

'wp\_user\_roles'-এর option\_name সহ একটি এন্ট্রি উপস্থিত হওয়া উচিত। সেই এন্ট্রিতে 'option\_name' এন্ট্রিকে wp\_user\_roles থেকে foo\_user\_roles এ পরিবর্তন করুন।

তারপর 'foo\_usermeta' টেবিল খুলুন এবং সামনে 'wp\_' সহ প্রতিটি এন্ট্রি খুঁজুন।

<input type="checkbox"/>	 Edit	 Copy	 Delete	10	1	wp_capabilities	a:1:s
<input type="checkbox"/>	 Edit	 Copy	 Delete	26	2	wp_capabilities	a:1:s
<input type="checkbox"/>	 Edit	 Copy	 Delete	75	3	wp_capabilities	a:1:s
<input type="checkbox"/>	 Edit	 Copy	 Delete	14	1	wp_dashboard_quick_press_last_post_id	2595
<input type="checkbox"/>	 Edit	 Copy	 Delete	29	2	wp_dashboard_quick_press_last_post_id	1283
<input type="checkbox"/>	 Edit	 Copy	 Delete	11	1	wp_user_level	10
<input type="checkbox"/>	 Edit	 Copy	 Delete	27	2	wp_user_level	10
<input type="checkbox"/>	 Edit	 Copy	 Delete	76	3	wp_user_level	0
<input type="checkbox"/>	 Edit	 Copy	 Delete	15	1	wp_user-settings	advl
<input type="checkbox"/>	 Edit	 Copy	 Delete	41	2	wp_user-settings	editor
<input type="checkbox"/>	 Edit	 Copy	 Delete	16	1	wp_user-settings-time	14753
<input type="checkbox"/>	 Edit	 Copy	 Delete	42	2	wp_user-settings-time	14134

এবং 'foo\_' এ পরিবর্তন করুন। আপনার কতজন ব্যবহারকারী আছে তার উপর নির্ভর করবে আপনাকে কতগুলি এন্ট্রি পরিবর্তন করতে হবে।

<input type="checkbox"/>	 Edit	 Copy	 Delete	10	1	foo_capabilities	a:1
<input type="checkbox"/>	 Edit	 Copy	 Delete	26	2	foo_capabilities	a:1
<input type="checkbox"/>	 Edit	 Copy	 Delete	75	3	foo_capabilities	a:1
<input type="checkbox"/>	 Edit	 Copy	 Delete	14	1	foo_dashboard_quick_press_last_post_id	25
<input type="checkbox"/>	 Edit	 Copy	 Delete	29	2	foo_dashboard_quick_press_last_post_id	12
<input type="checkbox"/>	 Edit	 Copy	 Delete	11	1	foo_user_level	10
<input type="checkbox"/>	 Edit	 Copy	 Delete	27	2	foo_user_level	10
<input type="checkbox"/>	 Edit	 Copy	 Delete	76	3	foo_user_level	0
<input type="checkbox"/>	 Edit	 Copy	 Delete	15	1	foo_user-settings	ad
<input type="checkbox"/>	 Edit	 Copy	 Delete	41	2	foo_user-settings	ed
<input type="checkbox"/>	 Edit	 Copy	 Delete	16	1	foo_user-settings-time	14
<input type="checkbox"/>	 Edit	 Copy	 Delete	42	2	foo_user-settings-time	14

একটি বিদ্যমান ইনস্টলেশনে উপসর্গটি পরিবর্তন করার জন্য আপনার যা দরকার তা হওয়া উচিত

# অধ্যায় 50: ওয়ার্ডপ্রেসে নিরাপত্তা - পালানো

## বাক্য গঠন

- esc\_html( স্ট্রিং \$টেক্সট ) •
- esc\_url( স্ট্রিং \$url, অ্যারে \$ প্রোটোকল, স্ট্রিং \$ \_context ) • esc\_js ( স্ট্রিং \$টেক্সট ) • wp\_json\_encode ( মিশ্রিত \$data, int \$options, int \$depth = 512 ) • esc\_at \$text ) • esc\_textarea( স্ট্রিং \$text )

## মন্তব্য

উন্নয়নের সময় নিরাপত্তা সবসময় মাথায় রাখা উচিত। নিরাপত্তা ছাড়াই একটি অ্যাপ বিভিন্ন আক্রমণের জন্য উন্মুক্ত থাকে যেমন SQL ইনজেকশন, XSS, CSRF, RFI ইত্যাদি যা গুরুতর সমস্যার কারণ হতে পারে।

অবিশ্বস্ত ডেটা অনেক উত্স (ব্যবহারকারী, তৃতীয় পক্ষের সাইট, আপনার নিজস্ব ডাটাবেস!, ...) থেকে আসে এবং এটির সমস্ত ইনপুট এবং আউটপুট উভয় ক্ষেত্রেই যাচাই করা প্রয়োজন। (সূত্র: ওয়ার্ডপ্রেস কোডেক্স)

ব্যবহার এবং উদ্দেশ্যের উপর নির্ভর করে ডেটা যাচাই করা উচিত, স্যানিটাইজ করা বা পালানো উচিত।

যাচাই করার জন্য ব্যবহারকারীর কাছে আপনার অনুরোধ করা ডেটা তাদের জমা দেওয়া তথ্যের সাথে মেলে তা নিশ্চিত করা। (সূত্র: ওয়ার্ডপ্রেস কোডেক্স)

স্যানিটাইজেশন ব্যবহারকারীর ডেটা গ্রহণ করার পদ্ধতির কিছুটা বেশি উদার। গ্রহণযোগ্য ইনপুটের একটি পরিসর থাকলে আমরা এই পদ্ধতিগুলি ব্যবহার করতে ফিরে যেতে পারি। (সূত্র: ওয়ার্ডপ্রেস কোডেক্স)

পালানোর জন্য আপনার কাছে ইতিমধ্যে থাকা ডেটা গ্রহণ করা এবং শেষ ব্যবহারকারীর জন্য রেন্ডার করার আগে এটিকে সুরক্ষিত করতে সহায়তা করা। (সূত্র: ওয়ার্ডপ্রেস কোডেক্স)

## উদাহরণ

এইচটিএমএল কোড এক্সেপ ডেটা

আমরা যখনই HTML কোডের মধ্যে ডেটা আউটপুট করছি তখন esc\_html ব্যবহার করা উচিত।

```
<h4><?php echo esc_html( $title); ?></h4>
```

একটি url এক্সেপ

```
<a href="<?php echo esc_url( home_url( '/' )); ?>">হোম</a>
```

```

```

জেএস কোডে ডেটা এক্সেপ করুন

esc\_js() একটি ট্যাগ অ্যাট্রিবিউটের ভিতরে ইনলাইন JS-এর জন্য ব্যবহার করার উদ্দেশ্যে করা হয়েছে।

একটি <script> ট্যাগের ভিতরে ডেটার জন্য wp\_json\_encode() ব্যবহার করুন।

```
<input type="text" onfocus="if( this.value == '<?php echo esc_js( $fields['input_text']); ?>' ) { this.value = "};" name="name">
```

wp\_json\_encode() একটি ভেরিয়েবলকে JSON-এ এনকোড করে, কিছু স্যানিটিং চেক সহ।

মনে রাখবেন wp\_json\_encode() স্বয়ংক্রিয়ভাবে স্ট্রিং-ডিলিমিটিং উদ্ধৃতি অন্তর্ভুক্ত করে।

```
<?php
$book = array( "title"
    => "JavaScript: The Definitive Guide",
    "লেখক" => "স্ট্যাক ওভারফ্লো",
);
?>

<script type="text/javascript"> var বই = <?php echo
wp_json_encode($book) ?>; /* var বই = { "শিরোনাম": "ওয়ার্ডপ্রেসের নিরাপত্তা",

    "লেখক" => "স্ট্যাক ওভারফ্লো",
}; */ </
script>
```

বা

```
<script type="text/javascript">|
var শিরোনাম = <?php echo wp_json_encode($title); ?>; var সামগ্রী = <?php echo
wp_json_encode( $content); ?>; var comment_count = <?php echo
wp_json_encode( $comment_count); ?>; </script>
```

## এস্কেপ বৈশিষ্ট্য

```
<input type="text" value="<?php echo esc_attr($_POST['username']); ?>" />
```

## টেক্সটেরিয়া এস্কেপ ডেটা

```
<textarea><?php echo esc_textarea( $text); ?></textarea>
```

---

## অধ্যায় 51: ওয়ার্ডপ্রেসে নিরাপত্তা - স্যানিটাইজেশন

### বাক্য গঠন

- `sanitize_text_field( স্ট্রিং $str )` •  
`sanitize_title( স্ট্রিং $title, স্ট্রিং $fallback_title, স্ট্রিং $context )` • `sanitize_email( string $email )` • `sanitize_html_class( string $class, string $fallback )` • `sanitize_file_name( স্ট্রিং $_us) স্ট্রিং $username, বুলিয়ান $strict )`

### মন্তব্য

উন্নয়নের সময় নিরাপত্তা সবসময় মাথায় রাখা উচিত। নিরাপত্তা ছাড়াই একটি অ্যাপ বিভিন্ন আক্রমণের জন্য উন্মুক্ত থাকে যেমন SQL ইনজেকশন, XSS, CSRF, RFI ইত্যাদি যা গুরুতর সমস্যার কারণ হতে পারে।

অবিশ্বস্ত ডেটা অনেক উত্স (ব্যবহারকারী, তৃতীয় পক্ষের সাইট, আপনার নিজস্ব ডাটাবেস!, ...) থেকে আসে এবং এটির সমস্ত ইনপুট এবং আউটপুট উভয় ক্ষেত্রেই যাচাই করা প্রয়োজন। (সূত্র: ওয়ার্ডপ্রেস কোডেক্স)

ব্যবহার এবং উদ্দেশ্যের উপর নির্ভর করে ডেটা যাচাই করা উচিত, স্যানিটাইজ করা বা পালানো উচিত।

যাচাই করার জন্য ব্যবহারকারীর কাছে আপনার অনুরোধ করা ডেটা তাদের জমা দেওয়া তথ্যের সাথে মেলে তা নিশ্চিত করা। (সূত্র: ওয়ার্ডপ্রেস কোডেক্স)

স্যানিটাইজেশন ব্যবহারকারীর ডেটা গ্রহণ করার পদ্ধতির কিছুটা বেশি উদার। গ্রহণযোগ্য ইনপুটের একটি পরিসর থাকলে আমরা এই পদ্ধতিগুলি ব্যবহার করতে ফিরে যেতে পারি। (সূত্র: ওয়ার্ডপ্রেস কোডেক্স)

পালানোর জন্য আপনার কাছে ইতিমধ্যে থাকা ডেটা গ্রহণ করা এবং শেষ ব্যবহারকারীর জন্য রেন্ডার করার আগে এটিকে সুরক্ষিত করতে সহায়তা করা। (সূত্র: ওয়ার্ডপ্রেস কোডেক্স)

### উদাহরণ

পাঠ্য ক্ষেত্র স্যানিটাইজ করুন

```
$title = sanitize_text_field($_POST['title']);
```

স্যানিটাইজ শিরোনাম

প্রত্যাবর্তিত মানটি একটি URL-এ ব্যবহারের জন্য উপযুক্ত হওয়ার উদ্দেশ্যে তৈরি করা হয়েছে, মানুষের-পাঠযোগ্য শিরোনাম হিসাবে নয়। পরিবর্তে `sanitize_text_field` ব্যবহার করুন।

```
$new_url = sanitize_title($title);
```

ইমেইল স্যানিটাইজ করুন

```
$sanitized_email = sanitize_email('admin@example.com!');
```

এইচটিএমএল ক্লাস স্যানিটাইজ করুন

```
$post_class = sanitize_html_class($post->post_title); প্রতিস্থাপন '<div class="" | $post_class | ">';
```

ফাইলের নাম স্যানিটাইজ করুন

```
$incfile = sanitize_file_name($_REQUEST["file"]); অন্তর্ভুক্ত ($incfile . ".php");
```

ফাইলের নাম স্যানিটাইজ না করে একজন আক্রমণকারী সহজভাবে পাস করতে পারে [http://attacker\\_site/malicious\\_page](http://attacker_site/malicious_page) ইনপুট হিসাবে এবং আপনার সার্ভারে যাই হোক না কেন কোড চালান।

ব্যবহারকারীর নাম স্যানিটাইজ করুন

```
$user = sanitize_user("আক্রমণকারী ব্যবহারকারীর নাম<script>console.log(document.cookie)</script>");
```

স্যানিটাইজ করার পরে \$ব্যবহারকারীর মান হল "আক্রমণকারী ব্যবহারকারীর নাম"

# অধ্যায় 52: শর্টকোড

## উদাহরণ

### শর্টকোড নিবন্ধন

শর্টকোড হল কোডের একটি ছোট অংশ যা ওয়ার্ডপ্রেস এডিটরে যোগ করা যেতে পারে এবং পৃষ্ঠাটি প্রকাশ বা পূর্বরূপ দেখার পরে ভিন্ন কিছু আউটপুট করবে।

প্রায়শই, শর্টকোডগুলি থিম functions.php ফাইলে যোগ করা হয়, তবে এটি একটি ভাল অনুশীলন নয় যেহেতু শর্টকোডগুলি থিম পরিবর্তন করার পরে কাজ চালিয়ে যেতে পারে বলে আশা করা হচ্ছে। পরিবর্তে, একটি প্লাগইন লিখুন এই কার্যকারিতা যোগ করতে।

শর্টকোড নিবন্ধনের জন্য কাঠামো হল:

```
ফাংশন new_shortcode($atts, $content = null){
    // যদি শর্টকোডে প্যারামিটারের প্রয়োজন হয় // প্যারামিটারগুলিকে ডিফল্ট হিসাবে সেট
    করা যেতে পারে কিছু নির্ধারিত( shortcode_atts( array( 'param_one' => 'h1' ), $atts ));
    $shortcode = '<!.param_one>!.content!</!';
        $param_one.'>'; $shortcode
    ফেরত দিন;

} // এটিই ওয়ার্ডপ্রেস অ্যাড_শর্টকোড ('ডেমো-শর্টকোড', 'নতুন_শর্টকোড') দিয়ে শর্টকোড নিবন্ধন করে।
```

ওয়ার্ডপ্রেস সম্পাদকের ভিতরে, আপনি টাইপ করতে পারেন:

```
[demo-shortcode param_one="h2"]Demo[/demo-shortcode] // যদি এটির একটি ডিফল্ট মান থাকে তাহলে আপনাকে এডিটরে param_one টোকাতে হবে না! // এটি সম্পাদকে থাকা ডিফল্টটিকে ওভাররাইড করবে
```

পৃষ্ঠাটি প্রকাশিত হলে, এটি পরিণত হবে

```
<h2>ডেমো</h2>
```

### ওয়ার্ডপ্রেস ব্যাকএন্ডে শর্টকোড ব্যবহার করা

```
[footag foo="value of 1" attribute-2="value of 2"]
```

ওয়ার্ডপ্রেস অ্যাডমিনে আমরা বর্গাকার বন্ধনীর ভিতরে শর্টকোড নাম লিখে এবং স্থান দ্বারা আলাদা করে বিকল্পভাবে বৈশিষ্ট্য যোগ করে পূর্বনির্ধারিত শর্টকোড ব্যবহার করি।

নতুন শর্টকোড যোগ করা হচ্ছে

```
ফাংশন footag_func ( $atts ) {
    রিটার্ন "foo = {$atts['foo']}";
}
```

```
}  
add_shortcode('footag', 'footag_func');
```

প্লাগইনে আমরা add\_shortcode ফাংশন ব্যবহার করে শর্টকোড যোগ করতে পারি।

শর্টকোডটি শুধুমাত্র বর্গাকার বক্সনীতে আবদ্ধ করে যেকোনো Wordpress পৃষ্ঠা বা পোস্টে ব্যবহার করা যেতে পারে।

[ফুট্যাগ]

পিএইচপি কোডের ভিতরে শর্টকোড ব্যবহার করা (থিম এবং প্লাগইন)

```
<?php echo do_shortcode("[footag foo='হাই! আমি একটি foo আউটপুট']"); ?>
```

php ব্যবহার করে একটি শর্টকোড প্রিন্ট করতে do\_shortcode ফাংশন ব্যবহার করুন এবং প্রত্যাবর্তিত মানটি প্রতিধ্বনিত করুন।

উইজেটে শর্টকোড ব্যবহার করা

```
add_filter('widget_text', 'shortcode_unautop'); add_filter('widget_text', 'do_shortcode');  
এখানে কোড লিখুন
```

উইজেটগুলিতে শর্টকোডগুলি সক্ষম করতে এটি একটি প্লাগইন বা functions.php ফাইলে যুক্ত করুন। কোডটি প্রথমে ওয়ার্ডপ্রেসের টার্নিং লাইনকে প্যারাগ্রাফ ট্যাগে পরিণত করা বন্ধ করে এবং তারপর শর্টকোডগুলিকে উইজেটের জন্য পার্স করতে দেয়।

দুটি লাইনের ক্রম গুরুত্বপূর্ণ।

## অধ্যায় 53: বৈশিষ্ট্য সহ শর্টকোড

### বাক্য গঠন

- `add_shortcode('your_short_code', 'your_function_name');`

### পরামিতি

পরামিতি বর্ণনা এবং ব্যবহার	
\$ট্যাগ	(স্ট্রিং) (প্রয়োজনীয়) শর্টকোড ট্যাগ পোস্টের বিষয়বস্তুতে অনুসন্ধান করতে হবে ডিফল্ট: কোনোটিই নয়
\$func	(কলযোগ্য) (প্রয়োজনীয়) শর্টকোড পাওয়া গেলে চালানোর জন্য হুক ডিফল্ট: কোনোটিই নয়

### মন্তব্য

গুরুত্বপূর্ণ - আপনার বৈশিষ্ট্যের জন্য ক্যামেলকেস বা UPPER-CASE ব্যবহার করবেন না

আপনি এখানে [অ্যাট্রিবিউট সহ একটি শর্টকোড](#) তৈরি করতে পারেন

### উদাহরণ

#### শর্টকোডের উদাহরণ

ওয়ার্ডপ্রেস শর্টকোড 2.5 সালে চালু করা হয়েছিল

এখানে শর্টকোড উদাহরণ আসা

```
[বোতাম]
```

থিমে সরাসরি শর্টকোড ব্যবহার করতে আপনাকে `do_shortcode()` ব্যবহার করতে হবে

```
<?php echo do_shortcode("[বোতাম]); ?>
```

বোতামটি কাস্টমাইজ করতে, আমরা সহজভাবে কিছু যোগ করতে পারি:

```
[বোতাম টাইপ="টুইটার"]
```

অথবা এটিকে আরও ভালো করার জন্য, আমরা একটি আবদ্ধ শর্টকোড ব্যবহার করতে পারি:

```
[বোতাম প্রকার="twitter"]টুইটারে আমাকে অনুসরণ করুন![button]
```

একটি স্ব-বন্ধ শর্টকোড তৈরি করা হচ্ছে

সবচেয়ে সহজ শর্টকোড হল স্ব-ক্লোজিং। আমরা আমাদের টুইটার অ্যাকাউন্টে একটি সহজ লিঙ্ক তৈরি করতে যাচ্ছি, এবং তারপর এটি একটি ব্লগ পোস্টে যোগ করব। সমস্ত কোড functions.php-এ যায়, যা /wp content/themes/your-theme/- এ অবস্থিত। যদি আপনার কাছে না থাকে তবে এটি তৈরি করুন এবং এতে কোডটি রাখুন।

```
<?php
function button_shortcode() { return '<a href="http://
twitter.com/rupomkhondaker" class="twitter-button">টুইটারে আমাকে অনুসরণ করুন!</a>';
}
add_shortcode('বোতাম', 'বোতাম_শর্টকোড');
?>
```

ব্যবহার: [বোতাম]

## পরামিতি সহ একটি স্ব-বন্ধ শর্টকোড তৈরি করা

পরামিতি সহ একটি স্ব-বন্ধ শর্টকোড তৈরি করা

```
<?php
ফাংশন button_shortcode( $type ) {

    নির্ধারিত( shortcode_atts(
        অ্যারে( 'টাইপ'
            => 'মান' ), $টাইপ ) );

    // ব্যবহারকারী কি ধরনের সুইচ প্রবেশ করেছে তা পরীক্ষা করুন
    ($type) {

        কেস 'টুইটার':
            '<a href="http://twitter.com/rupomkhondaker" class="twitter-button">অনুসরণ করুন
আমি টুইটারে!</a>'; বিরতি

        কেস 'আরএসএস':
            ফিরুন '<a href="http://example.com/rss" class="rss-button">এ সদস্যতা নিন
খাওয়ানো!</a>'
            বিরতি

    }

} add_shortcode( 'বোতাম', 'বোতাম_শর্টকোড');
?>
```

এখন আপনি আপনার শর্টকোডের ধরন সংজ্ঞায়িত করে কোন বোতামটি প্রদর্শন করবেন তা চয়ন করতে পারেন।

```
[বোতাম টাইপ="টুইটার"] [বোতাম
টাইপ="আরএসএস"]
```

## একটি আবদ্ধ শর্টকোড তৈরি করা হচ্ছে

সংক্ষিপ্ত কোড

এনক্লোসিং শর্টকোড আপনাকে আপনার শর্টকোডের মধ্যে বিষয়বস্তু এম্বেড করতে দেয়, ঠিক BBCode এর মতো যদি আপনি এটি ব্যবহার করে থাকেন।

```
<?php
ফাংশন button_shortcode( $attr, $content = null ) রিটার্ন '<a href="http://twitter.com/
filipstefansson" class="twitter-button">'। $সামগ্রী। '</a>';

}
add_shortcode('বোতাম', 'বোতাম_শর্টকোড');
?>
```

এই শর্টকোডটি ব্যবহার করতে, আপনি যে পাঠ্যটি ব্যবহার করতে চান সেটি এমবেড করুন:

```
[button]টুইটারে আমাকে অনুসরণ করুন![/button]
```

এই বোতামটিকে আরও ভালো করার জন্য, আমরা আগের উদাহরণের মতোই প্যারামিটার যোগ করতে পারি।

```
<?php
ফাংশন button_shortcode( $atts, $content = null )
নির্যাস( shortcode_atts( array(
'account' => 'অ্যাকাউন্ট',
'style' => 'স্টাইল' ), $atts ));

'<a href="http://twitter.com/" . esc_attr($account) . esc_attr($style) . $content! '</a>';
"" class="twitter-boton'

}
add_shortcode('বোতাম', 'বোতাম_শর্টকোড');
?>
```

ব্যবহার:

```
[button account="rupomkhondaker" style="simple"]টুইটারে আমাকে অনুসরণ করুন![/button]
```

## উইজেটে শর্টকোড

ডিফল্টরূপে, ওয়ার্ডপ্রেস সাইডবার উইজেটের মধ্যে শর্টকোড সমর্থন করে না। এটি শুধুমাত্র একটি পোস্ট, পৃষ্ঠা বা কাস্টম পোস্ট প্রকারের বিষয়বস্তুর মধ্যে শর্টকোডগুলিকে প্রসারিত করে। সাইডবার উইজেটগুলিতে শর্টকোড সমর্থন যোগ করতে, আপনি একটি প্লাগইন ইনস্টল করতে পারেন, বা নীচের কোডটি ব্যবহার করতে পারেন:

```
add_filter('widget_text', 'shortcode_unautop'); add_filter('widget_text', 'do_shortcode');
```

এই ক্রমে এই লাইন যোগ করা গুরুত্বপূর্ণ। প্রথম লাইনটি ওয়ার্ডপ্রেসকে লাইন ব্রেককে প্যারাগ্রাফ ট্যাগে পরিণত করতে বাধা দেয়, কারণ এটি শর্টকোডগুলিকে কাজ করা থেকে বিরত রাখে। দ্বিতীয় লাইন হল শর্টকোডগুলিকে কাজ করে।

# অধ্যায় 54: শর্টকোড

## উদাহরণ

শর্টকোড ভূমিকা

শর্টকোডগুলি উপযোগী হয় যখন আপনি ইনলাইনে আরও জটিল উপাদান যোগ করতে সক্ষম হতে চান। সাধারণ বিষয়বস্তু সম্পাদক।

এটির মধ্যে একটি শর্টকোড এটির মতো দেখতে সবচেয়ে সহজ:

```
ফাংশন my_shortcode( ){
    "এটি একটি শর্টকোড" ফেরত;
}
add_shortcode('my_shortcode', 'my_shortcode');
```

এটি "এটি একটি শর্টকোড" পাঠ্যটি আউটপুট করবে এবং আপনি এটিতে [my\_shortcode] লিখে এটি ব্যবহার করবেন বিষয়বস্তু সম্পাদক।

বোতাম শর্টকোড

এখানে একটি সাধারণ বোতাম শর্ট কোডের একটি উদাহরণ রয়েছে:

```
<?php
ফাংশন my_button_shortcode( $atts ) {

    // এর জন্য ইনপুট বৈশিষ্ট্য এবং assign ডিফল্ট মানগুলি পার্স করুন
    // বৈশিষ্ট্য যা শর্টকোডে নির্দিষ্ট করা নেই
    $a = shortcode_atts( অ্যারে
        'id' => "",
        'url' => '#',
        'class' => "",
        'টেক্সট' => ""
    ), $atts );

    // অ্যাক্সর ট্যাগ খুলুন এবং আরও ভাল অ্যাক্সেসযোগ্যতার জন্য ভূমিকা= বোতাম যোগ করুন
    $btn_html = '<a role="button";

    // href(লিঙ্ক) অ্যাট্রিবিউট যোগ করুন
    $btn_html .= ' href="' . $a['url'] . '"';

    // নির্দিষ্ট করা থাকলে আউটপুটে আইডি অ্যাট্রিবিউট যোগ করুন
    যদি ( !খালি ($a['id']) ) {
        $btn_html .= ' id="' . $a['id'] . '"';
    }

    $btn_classes = 'বোতাম';

    // আউটপুটে ক্লাস অ্যাট্রিবিউট যোগ করুন
    $btn_html .= ' class="বোতাম' . ( !খালি($a['class']) ? $a['class'] : '' ) . '"';

    // খোলার অ্যাক্সর ট্যাগ বন্ধ করুন
```

```
$btn_html .= '>|
// বোতাম টেক্সট $a['টেক্সট'] যোগ
করুন।
// ক্লোজিং অ্যাক্সর ট্যাগ '</a>' যোগ করুন।"\n";

ফেরত $btn_html;
}
add_shortcode( 'বোতাম', 'my_button_shortcode');
```

এই শর্টকোডটি এডিটরে [button url="/my-other-page" id="my-other-page-button" class="dark" text="Click me!"]  
লিখে ব্যবহার করা যেতে পারে এবং আউটপুট দেবে নিম্নলিখিত HTML:

```
<a role="button" href="/my-other-page" id="my-other-page-button" class="button dark">আমাকে ক্লিক করুন!
</a>
```

# অধ্যায় 55: সাইডবার

## বাক্য গঠন

- register\_sidebar( \$args ) •
- get\_sidebar (স্ট্রিং \$name = null)

## পরামিতি

পরামিতি বিবরণ	
\$args	(স্ট্রিং   অ্যারে) (ঐচ্ছিক) এর উপর ভিত্তি করে সাইডবার তৈরি করে নাম এবং id values
\$নাম	*(স্ট্রিং) (ঐচ্ছিক) বিশেষ সাইডবারের নাম। ডিফল্ট মান: নাল

## মন্তব্য

যুক্তি বিকল্পগুলি হল:

- নাম - সাইডবারের নাম (ডিফল্ট: স্থানীয় 'সাইডবার' এবং সংখ্যাসূচক আইডি)।
- id - সাইডবার আইডি - সমস্ত ছোট হাতের অক্ষরে হতে হবে, কোন স্পেস ছাড়াই (ডিফল্ট: একটি সংখ্যাসূচক স্বয়ংক্রিয় বর্ধিত আইডি)। আপনি আইডি আর্গুমেন্ট মান সেট না করলে, আপনি ডিবাগ মোডে E\_USER\_NOTICE বার্তা পাবেন, সংস্করণ 4.2 থেকে শুরু করে। বর্ণনা - সাইডবার কি/কোথায়
- আছে তার পাঠ্য বিবরণ। উইজেট ম্যানেজমেন্ট স্ক্রিনে দেখানো হয়েছে। (2.9 থেকে) (ডিফল্ট: খালি) ক্লাস - সিএসএস ক্লাস সাইডবারে অ্যাসাইন করার জন্য চেহারা -> উইজেট অ্যাডমিন
- পৃষ্ঠা। এই ক্লাসটি শুধুমাত্র ওয়ার্ডপ্রেস উইজেট অ্যাডমিন পৃষ্ঠার উত্সে প্রদর্শিত হবে। এটি আপনার ওয়েবসাইটের সামনের প্রান্তে অন্তর্ভুক্ত করা হবে না। দ্রষ্টব্য: মান সাইডবারটি ক্লাস মানের সাথে প্রিপেন্ড করা হবে। উদাহরণস্বরূপ, tal- এর একটি শ্রেণির ফলে সাইডবার-tal-এর একটি শ্রেণি মান হবে। (ডিফল্ট: খালি)। before\_widget - প্রতিটি উইজেটের আগে HTML বসাতে হবে (ডিফল্ট: <li id="%1\$s" class="widget %2\$s">) দ্রষ্টব্য: পরিবর্তনশীল প্রতিস্থাপনের জন্য স্প্রিন্টএফ ব্যবহার করে •
- after\_widget - প্রতিটি উইজেটের পরে বসানোর জন্য HTML (ডিফল্ট : </li>\n)। • before\_title - প্রতিটি শিরোনামের আগে HTML রাখতে হবে (ডিফল্ট: <h2 class="widgettitle">)। • after\_title - প্রতিটি শিরোনামের পরে HTML বসাতে হবে (ডিফল্ট: </h2>\n)।

## উদাহরণ

### সাইডবার নিবন্ধন

আপনার functions.php- এ আপনি এই কোড দিয়ে নতুন সাইডবার নিবন্ধন করতে পারেন

```
/**
```

\* সাইডবার নিবন্ধন করে  
\*

\* @param অ্যারে ডিফল্ট বা নির্দিষ্ট অ্যারে মান সহ অ্যারে \* @sence 1.0.0 \*/

```
যদি ( function_exists ('register_sidebar') register_sidebar( array (
    'নাম' => esc_html__( 'প্রাথমিক সাইডবার', 'mytheme' ), => 'প্রাথমিক-উইজেট-এরিয়া', 'বিবরণ'
    'আইডি' => esc_html__( 'প্রাথমিক উইজেট এলাকা',
    'mytheme' ), 'before_widget' => '<div id="%1$s" class="widget %2$s">', 'after_widget' => '</div>', 'before_title' => '<div
class="sidebar-widget-heading" <h3>', 'after_title' => '</h3></div>',

);

রেজিস্টার সাইডবার( অ্যারে (
    'নাম' <div id="%1$s" class="widget %2$s">', 'after_widget' => '</div>', 'before_title' =>
    'আইডি' '<div class="sidebar-widget-heading" <h3>',
    'after_title' => '</h3></div>',

);
}
```

আপনি যত খুশি সাইডবার যোগ করতে পারেন।

## সাইডবার পান

বিভিন্ন টেমপ্লেটে কল করার জন্য আপনি থিমে আপনার নিজস্ব সাইডবার ফাইল তৈরি করতে পারেন। বর্তমান থিমের sidebar.php কপি এবং পেস্ট করুন এবং নাম পরিবর্তন করুন (যেমন sidebar-book.php)

টেমপ্লেটে আপনি get\_sidebar('book') ব্যবহার করে এই সাইডবারটিকে কল করতে পারেন। এটি ব্যবহার করে আপনি বিভিন্ন পৃষ্ঠায় বিভিন্ন সাইডবার কল করতে পারেন।

---

# অধ্যায় 56: সাইট মাইগ্রেশন

## বাক্য গঠন

- OLD\_SITE - পুরানো সাইট স্থানান্তরিত হচ্ছে (যেমন: <http://localhost/example>)
- NEW\_SITE - নতুন সাইট যেখানে স্থানান্তরিত হবে (যেমন: <https://example.com>)

## উদাহরণ

একটি নতুন URL দিয়ে টেবিল আপডেট করা হচ্ছে

```
আপডেট wp_options SET option_value = প্রতিস্থাপন করুন(option_value, 'OLD_SITE', 'NEW_SITE') যেখানে option_name = 'হোম' বা option_name = 'siteurl';
```

```
আপডেট wp_posts SET guid = প্রতিস্থাপন (গাইড, 'OLD_SITE', 'NEW_SITE');
```

```
আপডেট wp_posts SET post_content = প্রতিস্থাপন করুন(post_content, 'OLD_SITE', 'NEW_SITE');
```

# অধ্যায় 57: শ্রেণীবিন্যাস

## বাক্য গঠন

- register\_taxonomy( \$taxonomy, \$object\_type, \$args);

## পরামিতি

পরামিতি বিবরণ	
\$শ্রেণীবিন্যাস	(স্ট্রিং) (প্রয়োজনীয়) শ্রেণীবিন্যাস নাম। নাম শুধুমাত্র থাকা উচিত ছোট হাতের অক্ষর এবং আন্ডারস্কোর অক্ষর, এবং 32 এর বেশি হবে না অক্ষর দীর্ঘ (ডাটাবেস গঠন সীমাবদ্ধতা)।
\$object_type	(অ্যারে/স্ট্রিং) (প্রয়োজনীয়) ট্যাক্সোনমি অবজেক্টের জন্য অবজেক্ট টাইপের নাম। অবজেক্ট-টাইপ বিল্ট-ইন পোস্ট টাইপ বা যেকোনো কাস্টম পোস্ট টাইপ হতে পারে নিবন্ধিত
\$args	(অ্যারে/স্ট্রিং) (ঐচ্ছিক) আর্গুমেন্টের একটি অ্যারে।

## উদাহরণ

### জেনারগুলির জন্য একটি শ্রেণীবিন্যাস নিবন্ধনের উদাহরণ

```
<?php
// init অ্যাকশনে হুক করুন এবং এটি ফায়ার হলে create_book_taxonomies কল করুন
add_action( 'init', 'create_book_taxonomies', 0);

// পোস্ট টাইপ "বই" এর জন্য শ্রেণীবিন্যাস জেনার তৈরি করুন
ফাংশন create_book_taxonomies() {
    // নতুন শ্রেণীবিন্যাস যোগ করুন, এটিকে শ্রেণিবিন্যাস করুন (বিভাগের মতো)
    $লেবেল = অ্যারে(
        'নাম' => _x('জেনারস', 'টেক্সোনমি সাধারণ নাম'),
        'একবচন_নাম' => _x('শৈলী', 'শ্রেণীবিন্যাস একবচন নাম'),
        'সার্চ_আইটেম' => __('সার্চ জেনারস'),
        'সকল_প্রকার' => __('সমস্ত জেনারস'),
        'parent_item' => __('প্যারেন্ট জেনার'),
        'parent_item_colon' => __('প্যারেন্ট জেনার:'),
        'সম্পাদনা_আইটেম' => __('এডিট জেনার'),
        'আপডেট_আইটেম' => __('আপডেট জেনার'),
        'নতুন আইটেম যোগ করুন' => __('নতুন জেনার যোগ করুন'),
        'নতুন_আইটেম_নাম' => __('নতুন ঘরানার নাম'),
        'মেনু_নাম' => __('জেনার'),
    );

    $args = অ্যারে(
        'অনুক্রমিক' => সত্য,
        'লেবেল' => $লেবেল,
        'শো_ইউআই' => সত্য,
    );
}
```

```

        'show_admin_column' => সত্য,
        'query_var' 'পুনর্লিখন'          => সত্য,
                                         => অ্যারে( 'slug' => 'জেনার' ),
    );

    register_taxonomy ('জেনার', অ্যারে ('বই'), $args );

}
?>

```

আপনি একটি থিমের functions.php টেমপ্লেট ফাইলে কাস্টম ট্যাক্সোনমি সংজ্ঞায়িত করতে পারেন:

## পৃষ্ঠায় বিভাগ যোগ করুন

আপনি নীচের কোড ব্যবহার করে পোস্ট টাইপ পৃষ্ঠায় একই কাস্টম তৈরি ট্যাক্সোনমি যোগ করতে পারেন।

```

ফাংশন add_taxonomies_to_pages()
    { register_taxonomy_for_object_type('genre', 'page');
    }
add_action('init', 'add_taxonomies_to_pages');

```

আপনার থিমের functions.php ফাইলে উপরের কোড যোগ করুন। একইভাবে আপনি পোস্ট টাইপ পৃষ্ঠায় কাস্টম বা ডিফল্ট post\_tag যোগ করতে পারেন।

কাস্টম ট্যাক্সোনমি কোয়েরি ব্যবহার করে পৃষ্ঠাগুলি পেতে একই ফাইলে নীচের কোড যোগ করতে হবে।

```

যদি ( ! is_admin() )
    {
        add_action('pre_get_posts', 'category_and_tag_archives');
    }

ফাংশন বিভাগ_এন্ড_ট্যাগ_আর্কাইভস( $wp_query ) $my_post_array = array('page'); যদি
    ( $wp_query->get( 'category_name' ) || $wp_query-
    >get( 'cat' ) $wp_query->set( 'post_type', $my_post_array);
}

```

পৃষ্ঠাগুলিতে বিভাগ এবং ট্যাগ যুক্ত করুন এবং সেগুলিকে শ্রেণী হিসাবে সন্নিবেশ করুন।

```

// পৃষ্ঠাগুলিতে ট্যাগ এবং বিভাগ যোগ করুন

ফাংশন add_taxonomies_to_pages() {
    register_taxonomy_for_object_type('post_tag', 'page'); register_taxonomy_for_object_type('বিভাগ',
    'পৃষ্ঠা'); }

add_action('init', 'add_taxonomies_to_pages');
যদি ! is_admin()
    {
        ( add_action ( 'pre_get_posts', 'category_and_tag_archives');

    }
ফাংশন বিভাগ_এন্ড_ট্যাগ_আর্কাইভস( $wp_query ) $my_post_array = array('post','page');
যদি ( $wp_query->get( 'category_name' ) || $wp_query->get( 'cat' ) $wp_query->set( 'post_type', $my_post_array);

```

```

যদি ( $wp_query->get( 'tag' ) $wp_query-
>সেট( 'post_type', $my_post_array);
}

// <body> এ ক্লাস হিসাবে ট্যাগ এবং বিভাগ যোগ করুন

ফাংশন add_categories_and_tags( $classes = if( is_page() ) $categories =
get_the_category(); {

    foreach( $category as $category ) $classes[] = 'category-'. {
        $category->slug;
    }
    $tags = get_the_tags();
    foreach( $tags as $tag ) $classes[] = 'tag-'. {
        $tag->slug;
    }
}
}
$ক্লাস ফেরত দিন;
}
add_filter('body_class', 'add_categories_and_tags');

```

পৃষ্ঠাগুলিতে বিভাগ এবং ট্যাগ যোগ করুন এবং ক্লাস হিসাবে সন্নিবেশ করুন

আপনি এই কোডটি আপনার কাস্টম functions.php ফাইলে যোগ করতে পারেন:

```

// পৃষ্ঠাগুলিতে ট্যাগ এবং বিভাগ যোগ করুন

ফাংশন add_taxonomies_to_pages() {
    register_taxonomy_for_object_type('post_tag', 'page');
    register_taxonomy_for_object_type('বিভাগ', 'পৃষ্ঠা');
}
add_action('init', 'add_taxonomies_to_pages');

যদি ( ! is_admin() ) {
    add_action( 'pre_get_posts', 'category_and_tag_archives' );
}

ফাংশন বিভাগ_এবং_ট্যাগ_আর্কাইভস( $wp_query ) $my_post_array = array('post','page'); {

    যদি ( $wp_query->get( 'category_name' ) || $wp_query->get( 'cat' ) $wp_query->set( 'post_type', $my_post_array); )

    যদি ( $wp_query->get( 'tag' ) $wp_query-
>সেট( 'post_type', $my_post_array);
}

// <body> এ ক্লাস হিসাবে ট্যাগ এবং বিভাগ যোগ করুন

ফাংশন add_categories_and_tags( $classes = if( is_page() ) $categories =
get_the_category(); {

    foreach( $category as $category ) $classes[] = 'category-'. {
        $category->slug;
    }
    $tags = get_the_tags();
    foreach( $tags as $tag ) $classes[] = 'tag-'. {
        $tag->slug;
    }
}
}

```

```
    }  
  }  
  $ক্লাস ফেরত দিন;  
}  
add_filter('body_class', 'add_categories_and_tags');
```

নিখুঁত কাজ করে, ওয়ার্ডপ্রেস 4.8 এ পরীক্ষিত

---

## অধ্যায় 58: টেমপ্লেট অনুক্রম

### মন্তব্য

ওয়ার্ডপ্রেসে ডিবাগিংয়ের জন্য প্লাগইন:

- <https://wordpress.org/plugins/query-monitor/> •
- <https://wordpress.org/plugins/debug-bar/> •
- <https://wordpress.org/plugins/debug-bar-console/> • <https://wordpress.org/plugins/kint-debugger/> • <https://wordpress.org/plugins/rest-api-console/>

### উদাহরণ

ভূমিকা

আপনি একটি ওয়ার্ডপ্রেস থিম তৈরি করার সময় সবচেয়ে গুরুত্বপূর্ণ বিষয়গুলির মধ্যে একটি হল থিমের জন্য ওয়ার্ডপ্রেস টেমপ্লেট শ্রেণিবিন্যাস। টেমপ্লেট অনুক্রম নির্ধারণ করে কোন টেমপ্লেট ফাইলটি প্রতিটি অনুরোধের জন্য লোড করা হবে এবং কোন ক্রমে। যদি প্রথম টেমপ্লেটটি হায়ারার্কিতে না থাকে তবে ওয়ার্ডপ্রেস পরেরটি লোড করার চেষ্টা করবে এবং যতক্ষণ না আপনি index.php এ শেষ হচ্ছেন।

টেমপ্লেট শ্রেণিবিন্যাসের বিস্তারিত বর্ণনা করার জন্য সর্বোত্তম উপায় অবশ্যই সম্পূর্ণ কাঠামো সহ একটি চিত্র ব্যবহার করা:

Archive Page



Singular Page

Single Post Page

Static Page



Site Front Page

front-page.php



category-books.php শুধুমাত্র স্লাগ বই সহ বিভাগের জন্য ব্যবহার করা হবে। আরেকটি উদাহরণ হল page-\$id.php যেটি শুধুমাত্র একটি নির্দিষ্ট আইডি সহ একটি পৃষ্ঠাকে টার্গেট করে, যেমন page-41.php শুধুমাত্র আইডি 41 সহ পৃষ্ঠাটিকে লক্ষ্য করে।

টেমপ্লেটের পরে যা নির্দিষ্ট ধরণের বা পোস্টকে লক্ষ্য করে আমরা জেনেরিক টাইপ টেমপ্লেট পাই, যেমন সমস্ত সংরক্ষণাগার পৃষ্ঠাগুলির জন্য archive.php বা সমস্ত পৃষ্ঠাগুলির জন্য page.php। কিন্তু মনে রাখবেন, সেগুলি শুধুমাত্র তখনই ব্যবহার করা হবে যদি বর্তমান পৃষ্ঠাটি অনুক্রমের উচ্চতর টেমপ্লেটগুলির সাথে মেলে না।

সবশেষে, যদি ওয়ার্ডপ্রেস টেমপ্লেট ডিরেক্টরিতে কোনো মিল টেমপ্লেট খুঁজে না পায়, তাহলে শেষ ফলব্যাচটি সর্বদা হয় index.php যা একটি ওয়ার্ডপ্রেস থিমের একমাত্র প্রয়োজনীয় টেমপ্লেট ফাইল।

## ডিবাগিং

হায়ার্কি ডিবাগ করার সময় হারিয়ে যাওয়া সহজ। আপনি PHP এর বিল্ট ইন কমান্ড ব্যবহার করতে পারেন

ডিবাগ\_ব্যাকট্রেস।

আপনি ডিবাগ করতে চান এমন যেকোনো টেমপ্লেটের ভিতরে পরবর্তী স্নিপেট রাখুন এবং জেনারেট হওয়া পৃষ্ঠাটি দেখুন:

```
<!--  
<?php print_r( debug_backtrace() ) ?>  
-->
```

## অধ্যায় 59: টেমপ্লেট\_ইনক্লুড

### পরামিতি

পরামিটার ব্যাখ্যা	
\$টেমপ্লেট	ফিল্টারে একটি পরামিটার পাস করে, \$টেমপ্লেট হল সক্রিয় চাইল্ড থিম বা প্যারেন্ট থিমে পাওয়া পোস্ট টাইপের জন্য উপযুক্ত ফাইলের বর্তমান পাথ (যদি কোনো চাইল্ড থিম জায়গায় বা চাইল্ড থিমের টেমপ্লেট কম না থাকে। আরো বিস্তারিত জানার জন্য ওয়ার্ডপ্রেস টেমপ্লেট অনুক্রম দেখুন)।

### মন্তব্য

সংশোধন না করলেও আপনাকে অবশ্যই \$template ফেরত দিতে হবে। যদি এটি আপনাকে বিভ্রান্ত করে তবে উদাহরণগুলি দেখুন যেখানে apply\_filter() কোডে ব্যবহার করা হয়েছে

পরে ব্যবহারের জন্য আপনার এখানে ভেরিয়েবল সেট আপ করা উচিত নয়, এর জন্য আরও ভাল হুক রয়েছে।

এই ফিল্টারের জন্য একটি দরকারী প্রোগ্রাম প্রবাহ হল:

1. \$টেমপ্লেট চেক করুন আমাদের কাস্টম পোস্ট টাইপ নাম --> টেমপ্লেট শ্রেণিবিন্যাস অন্তর্ভুক্ত করে!! 2. যদি না হয়, উপযুক্ত ফাইলগুলির জন্য আমাদের প্লাগইন অনুসন্ধান করুন --> ফাইলগুলির জন্য ফোল্ডারগুলির মাধ্যমে অনুসন্ধান করার পরিবর্তে নির্দিষ্ট ফাইলগুলিতে নির্দেশ করা ভাল। আরো দক্ষ. কিন্তু সম্পূর্ণরূপে বিকাশকারীর উপর।
3. টেমপ্লেট ফেরত দিন।

### উদাহরণ

#### সহজ উদাহরণ

এই ফিল্টার খুব দরকারী। ডেভেলপারদের জন্য একটি সাধারণ সমস্যা হল কীভাবে তাদের বিকাশ করা প্লাগইনগুলিতে টেমপ্লেটগুলি অন্তর্ভুক্ত করা যায়।

ওয়ার্ডপ্রেস wp শ্রেণিবিন্যাস ব্যবহার করে সক্রিয় শিশু/অভিভাবক থিমে উপযুক্ত টেমপ্লেট সনাক্ত করার সাথে সাথে ফিল্টারটি প্রয়োগ করা হয়।

আপনি যখন টেমপ্লেট পাথ পরিবর্তন করতে চান তখন সংজ্ঞায়িত করতে সতর্ক থাকুন। নীচের উদাহরণে, বর্তমান পৃষ্ঠাটি আমাদের কাস্টম পোস্ট টাইপ cpt-এর একক দৃশ্য কিনা তা দেখতে কোডটি পরীক্ষা করে।

শুরু করার জন্য যথেষ্ট সহজ উদাহরণ!

```
add_filter('টেমপ্লেট_ইনক্লুড', 'কাস্টম_ফাংশন');

ফাংশন custom_function($template){

    //একটি পোস্ট টেমপ্লেট পরিবর্তন করুন...
```

```

if( is_singular('cpt') ){ $template= 'path/to/
    another/template_file';
}

ফেরত $টেমপ্লেট;

}

```

## আরো Adv উদাহরণ

```

add_filter('টেমপ্লেট_ইনক্লুড', 'কাস্টম_ফাংশন');

ফাংশন custom_function($template){

    /*
    *     এই উদাহরণটি একটু বেশি উন্নত।
    *     $template-এ আমাদের পোস্ট-টাইপ পাথে আছে কিনা তা পরীক্ষা করে দেখবে।
    *     যদি এটি হয়, থিমে একটি উচ্চ স্তরের টেমপ্লেট রয়েছে যেমন single-cpt.php না থাকলে আমরা ফাইলটির জন্য প্লাগইন প্যারেন্ট ফোল্ডারে তাকাই। যেমন একক-
    *     cpt.php
    */

    //পোস্টের ধরন ফাইলের নাম (উচ্চ অগ্রাধিকার ফাইল) আছে কিনা তা দেখতে পরীক্ষা করুন //রিটার্ন টেমপ্লেট যদি তা হয়!

    বিশ্বব্যাপী $পোস্ট;

    if( strpos($template, 'single-'. $post->post_type.'.php') !== false && strpos($template, 'archive-'. $post->post_type.'.php') !== মিথ্যা ){ $template ফেরত দিন;

    }

    $plugin_path = 'var/etc/wp-content/plugins/plugin'; //এখানে নিজস্ব যুক্তি অন্তর্ভুক্ত করুন...

    if( is_singular($post->post_type) && file_exists($plugin_path.'/single-'. $post-
    >পোস্ট_টাইপ!'.php)){
        $template= $plugin_path.'/single-'. $post->post_type.'.php';
    } elseif ( is_archive($post->post_type) && file_exists($plugin_path.'/archive-'. $post- >post_type.'.php') ) $template= $plugin_path.'/archive-'. $post- >পোস্ট_টাইপ!'.php';
        {

    }

    ফেরত $টেমপ্লেট;

}

```

## অধ্যায় 60: \$wpdb অবজেক্ট

### মন্তব্য

\$wpdb অবজেক্ট রেফারেন্স করার দুটি উপায় আছে। প্রথমটি হল বস্তুর গ্লোবাল ইনস্ট্যান্সে কাজ করার জন্য PHP কীওয়ার্ড গ্লোবাল ব্যবহার করা।

```
বিশ্বব্যাপী $wpdb; echo  
$wpdb->উপসর্গ;  
// ডাটাবেসের জন্য উপসর্গ আউটপুট করে
```

\$wpdb অবজেক্ট ব্যবহার করার দ্বিতীয় উপায় হল PHP এর \$GLOBALS সুপার গ্লোবাল ভেরিয়েবল রেফারেন্স করা।

```
প্রতিধ্বনি $GLOBALS['wpdb']->উপসর্গ;  
// এটি ডাটাবেসের জন্য উপসর্গও আউটপুট করবে
```

দ্বিতীয় উপায়টি নিরুৎসাহিত করা হয় কারণ এটি সেরা অনুশীলন হিসাবে বিবেচিত নাও হতে পারে।

### উদাহরণ

একটি পরিবর্তনশীল নির্বাচন

সবচেয়ে মৌলিক আকারে, একটি এসকিউএল কোয়েরিতে অবজেক্টের get\_var মেথড পাঁসিং কল করে একটি টেবিল থেকে একটি একক ভেরিয়েবল নির্বাচন করা সম্ভব।

```
বিশ্বব্যাপী $wpdb; $user  
= $wpdb->get_var( "$wpdb->ব্যবহারকারীরা যেখানে user_email='foo@bar.com'" থেকে আইডি নির্বাচন করুন");
```

এটা মনে রাখা খুবই গুরুত্বপূর্ণ যে আক্রমণ থেকে রক্ষা করার জন্য প্রশ্নগুলিতে ব্যবহৃত যেকোন অবিশ্বস্ত মান অবশ্যই এড়িয়ে যেতে হবে। এটি বস্তুর প্রস্তুত পদ্ধতি ব্যবহার করে করা যেতে পারে।

```
বিশ্বব্যাপী $wpdb;  
$email = $_POST['ইমেইল']; $user = $wpdb->  
>get_var( $wpdb->প্রস্তুত করুন ( "$wpdb  
থেকে আইডি নির্বাচন করুন->ব্যবহারকারীরা যেখানে ব্যবহারকারী_ইমেইল=%s", $ইমেইল)  
);  
if( !is_null( $user )  
echo $user; } else  
{ echo 'User  
not found';  
}
```

একাধিক সারি নির্বাচন করা হচ্ছে

আপনি ডাটাবেস থেকে একাধিক সারি পেতে get\_results ব্যবহার করতে পারেন।

```
বিশ্বব্যাপী $wpdb;
```

```
$userTable = $wpdb->উপসর্গ!"users";
```

```
$selectUser = $wpdb->get_results("$userTable থেকে *নির্বাচন করুন");
```

এটি অ্যাারেতে সমস্ত ব্যবহারকারীর তালিকা দেখাবে।

# অধ্যায় 61: অ্যাডমিন বার (ওরফে "দ্য টুলবার")

## মন্তব্য

ওয়ার্ডপ্রেস অ্যাডমিন টুলবারটি 3.1 সংস্করণে যোগ করা হয়েছে এবং এতে সাধারণ লিঙ্ক রয়েছে

প্রশাসনিক কাজের পাশাপাশি ব্যবহারকারীর প্রোফাইল এবং অন্যান্য ওয়ার্ডপ্রেস তথ্যের লিঙ্ক।

যাইহোক, অনেক সাইটের মালিকরা ডিফল্টভাবে সমস্ত লগ-ইন করা ব্যবহারকারীদের টুলবার দেখানো অপছন্দ করেন এবং/অথবা এতে তাদের নিজস্ব বিকল্প যোগ করতে চান।

## উদাহরণ

প্রশাসক ছাড়া সকলের কাছ থেকে অ্যাডমিন টুলবার সরানো হচ্ছে

অ্যাডমিনিস্ট্রেটর ইউজার লেভেল ব্যতীত অন্য সবার থেকে এটি অপসারণ করতে functions.php- এ নিম্নলিখিত কোড যোগ করুন :

```
add_action('after_setup_theme', 'no_admin_bar');

function no_admin_bar() {
    যদি (!current_user_can('প্রশাসক') && !is_admin()) {
        show_admin_bar(false);
    }
}
```

ফিল্টার ব্যবহার করে অ্যাডমিন টুলবার সরানো হচ্ছে

অ্যাডমিন বার লুকানোর আরেকটি উপায় হল যোগ করা

```
যদি (! বর্তমান_ব্যবহারকারী_ক্যান ('ম্যানেজ_অপশন') ) {
    add_filter('শো_অ্যাডমিন_বার', '_রিটার্ন_ফলস', 1000);
}
```

যে ব্যবহারকারীদের সেটিংস পৃষ্ঠা অ্যাক্সেস করার বিশেষাধিকার নেই, তারা অ্যাডমিন বার দেখতে সক্ষম হবে না।

অ্যাডমিন বার থেকে কিভাবে ওয়ার্ডপ্রেস লোগো সরান

ওয়ার্ডপ্রেস অ্যাডমিন বার বা টুলবার থেকে আইটেমগুলি সরাতে বিকাশকারীরা অ্যাডমিন\_বার\_মেনু অ্যাকশন ব্যবহার করতে পারে।

```
add_action('admin_bar_menu', 'remove_wp_logo_from_admin_bar', 999); function
remove_wp_logo_from_admin_bar( $wp_admin_bar ) $wp_admin_bar->remove_node('wp-logo'); {
}
```

উপরের কোড অ্যাডমিন বার থেকে ওয়ার্ডপ্রেস লোগো সরিয়ে দেয়। আপনাকে যা করতে হবে তা হল কোডটি পেস্ট করুন

আপনার functions.php ফাইলের ভিতরে।

রিমুভ\_নোড পদ্ধতিতে পাস করা প্যারামিটার হল নোডের আইডি যা আপনি সরাসরি চান। ওয়ার্ডপ্রেস পৃষ্ঠার HTML সোর্স কোডে একটি টুলবার সহ আইডি পাওয়া যাবে। উদাহরণস্বরূপ, টুলবারে বাম দিকে ওয়ার্ডপ্রেস লোগোর জন্য li উপাদানের আইডি হল "wp-admin-bar-wp-logo":

```
<li id="wp-admin-bar-wp-logo" class="menupop"> ... </li>
```

নোডের আইডি পেতে li এর আইডি থেকে "wp-admin-bar-" সরান। এই উদাহরণ থেকে নোড আইডি হল "wp-logo"।

আপনি আপনার অ্যাডমিন বারে বিভিন্ন আইটেম বা নোডের নোড আইডি খুঁজে বের করতে ব্রাউজার ইন্সপেক্টর টুল ব্যবহার করতে পারেন।

## অ্যাডমিন লগইন পৃষ্ঠায় আপনার কাস্টম লোগো এবং কাস্টম লিঙ্ক যোগ করুন

আপনি আপনার নিজস্ব লোগো এবং ডিফল্ট ওয়ার্ডপ্রেস লোগো প্রতিস্থাপন করার জন্য লিঙ্ক যোগ করতে নীচের হুক যোগ করতে পারেন।

### কাস্টম লোগো যোগ করতে

```
ফাংশন custom_login_logo() { echo '<style type="text/
css"> h1 a { background-image:
url(.'.get_bloginfo('template_directory').'/images/custom-logo.png) !গুরুত্বপূর্ণ; পটভূমির আকার : 100% !গুরুত্বপূর্ণ; প্রস্থ: 300px !গুরুত্বপূর্ণ; উচ্চতা : 100px !গুরুত্বপূর্ণ;} </
style>'; } অ্যাড_অ্যাকশন ('লগইন_হেড', 'কাস্টম_লগইন_লোগো');
```

### কাস্টম লোগো লিঙ্ক যোগ করতে

```
add_filter('login_headerurl', 'custom_loginlogo_url'); ফাংশন custom_loginlogo_url($url) {
    home_url();
}
```

## অধ্যায় 62: লুপ (প্রধান ওয়ার্ডপ্রেস লুপ)

### উদাহরণ

বেসিক ওয়ার্ডপ্রেস লুপ স্ট্রাকচার

প্রতিবার ওয়ার্ডপ্রেস পৃষ্ঠাটি লোড করলে, এটি প্রধান লুপ চালাবে।

লুপ হল আপনি বর্তমানে যে পৃষ্ঠায় আছেন তার সাথে সম্পর্কিত সমস্ত উপাদানের উপর পুনরাবৃত্তি করার উপায়।

প্রধান লুপ একটি বিষয়বাহী WP\_Query অবজেক্টে কাজ করবে। কোয়ারীটির একটি বিষয়বাহী পদ্ধতি আছে have\_posts(), যা আমাদের সমস্ত ফলাফল লুপ করতে দেয়। অবশেষে লুপের ভিতরে আপনি the\_post() পদ্ধতিকে কল করতে পারেন (এছাড়াও একটি গ্লোবাল ফাংশন হিসাবে), যা লুপের ভিতরে বর্তমান পোস্টে গ্লোবাল পোস্ট অবজেক্ট সেট করে এবং পোস্টডেটা বর্তমান পোস্টে সেট করে। এর জন্য ধন্যবাদ আপনি সরাসরি লুপের ভিতরে the\_title, the\_content, the\_author (টেমপ্লেট ট্যাগ) এর মতো ফাংশনগুলিকে কল করতে পারেন।

উদাহরণস্বরূপ আপনি যদি পোস্টের তালিকায় থাকেন তবে প্রধান লুপে সমস্ত পোস্ট সহ একটি কোয়ারী অবজেক্ট থাকবে।

আপনি যদি একক পোস্টে (অথবা পৃষ্ঠা) থাকেন তবে এটিতে আপনি বর্তমানে যে একক পোস্ট (পৃষ্ঠা) আছেন তার সাথে একটি প্রশ্ন থাকবে। চালু।

```
যদি ( have_posts() ) যখন :
    ( have_posts() ) :
        পোস্টটি();
        var_dump($post); endwhile;
যদি শেষ;
```

### বিকল্প লুপ সিনট্যাক্স

আপনি এই মত কোঁকড়া বন্ধনী সহ লুপ ব্যবহার করতে পারেন:

```
যদি ( have_posts() ) { যখন
    ( have_posts() ) {
        পোস্টটি();
        var_dump($post);
    }
}
```

### লুপে কোন আইটেম হ্যান্ডলিং

আপনি যদি এই ধরনের পরিস্থিতি পরিচালনা করতে চান তবে একটি if/else বিবৃতি যোগ করুন।

```
যদি ( আছে_পোস্ট() ) : যখন ( have_posts() ) :
    পোস্টটি();
    var_dump($post);
```

endwhile; অন্য:

\_\_('এই প্রস্নের কোনো ফলাফল নেই');

যদি শেষ;

# অধ্যায় 63: the\_title()

## ভূমিকা

এই ফাংশনটি বর্তমান পোস্টের শিরোনাম প্রদান করে।

## বাক্য গঠন

- the\_title( \$আগে, \$পরে, \$echo);

## পরামিতি

পরামিতি বিবরণ	
\$ আগে	(স্ট্রিং) (ঐচ্ছিক) শিরোনামের আগে লেখা পাঠ্য।
\$পরে	(স্ট্রিং) (ঐচ্ছিক) শিরোনামের পরে লেখা পাঠ্য।
\$প্রতিধ্বনি	(বুলিয়ান) (ঐচ্ছিক) শিরোনামটি প্রদর্শন করুন বা পিএইচপি-তে ব্যবহারের জন্য ফেরত দিন

## মন্তব্য

- \$echo প্যারামিটারের জন্য, শিরোনাম প্রদর্শনের জন্য সত্য ব্যবহার করুন এবং পিএইচপি-তে ব্যবহারের জন্য ফেরত দিতে মিথ্যা ব্যবহার করুন
- অনুগ্রহ করে মনে রাখবেন যে\_টাইটেলটি শুধুমাত্র লুপগুলিতে ব্যবহার করা যেতে পারে, আপনি যদি এটি লুপের বাইরে ব্যবহার করতে চান তবে অনুগ্রহ করে get\_the\_title ব্যবহার করুন

## উদাহরণ

### শিরোনামের সহজ ব্যবহার

#### কোড

```
শিরোনাম( );
```

#### আউটপুট

বর্তমান পোস্ট বা পৃষ্ঠার শিরোনাম

আগে এবং পরে কোড সহ শিরোনাম প্রিন্ট করা

#### কোড

```
the_title( '<h1>', '</h1>');
```

আউটপুট

h1 ট্যাগে বর্তমান পোস্ট বা পৃষ্ঠার শিরোনাম

# অধ্যায় 64: থিম

## ভূমিকা

ওয়ার্ডপ্রেস থিম আপনার ওয়েবসাইটের প্রথম প্রান্ত। তারা সাইট ডিজিট যখন মানুষ কি তারা দেখতে। বেছে নেওয়ার জন্য হাজার হাজার থিম রয়েছে, প্রদত্ত এবং বিনামূল্যের সংস্করণ। এমনকি আপনি কয়েকটি প্রয়োজনীয় ফাইল দিয়ে আপনার নিজস্ব কাস্টম থিম তৈরি করতে পারেন।

## উদাহরণ

### ওয়ার্ডপ্রেস থিম

## কিভাবে একটি থিম চয়ন করুন

প্রতিটি ওয়ার্ডপ্রেস ইন্সটল একটি প্রি-ইনস্টল করা থিমের সাথে আসে। আপনি ড্যাশবোর্ড থেকে আপনার থিমগুলি পরিচালনা করেন। থিম ইনস্টল, পূর্বরূপ, মুছে, সক্রিয় এবং আপডেট করতে উপস্থিতি > থিমগুলিতে যান। বর্তমান থিমটি এই মেনুর উপরের বাম কোণে পাওয়া যায়।

থিম ইমেজের উপর ঘোরানো একটি 'থিম বিশদ' বোতাম প্রকাশ করে। এই বোতামটি থিম সম্পর্কে তথ্য প্রদান করে, যেমন সংস্করণ এবং বিবরণ। বর্তমান থিম ছবিতে ক্লিক করলে আপনি শিরোনামের মতো নির্দিষ্ট থিম সেটিংস কাস্টমাইজ করার অ্যাক্সেস পাবেন।

## আপডেট উপলব্ধ

ইনস্টল করা থিমগুলির জন্য আপডেটগুলি উপলব্ধ থাকলে, আপনি একটি বার্তা পাবেন যা আপনাকে জানায় যে একটি নতুন সংস্করণ উপলব্ধ। আপনি নতুন সংস্করণের বিবরণ দেখতে বা এখনই আপডেট করতে সক্ষম হবেন।

- সংস্করণ বিবরণ দেখুন

সংস্করণ বিবরণ লিঙ্কে ক্লিক করলে আপনি ওয়ার্ডপ্রেস থিম ডিরেক্টরি থেকে একটি পৃষ্ঠায় নেভিগেট করবেন। এখানে আপনি আপগ্রেড সংস্করণের বিশদ বিবরণ পাবেন।

- এখন হালনাগাদ করুন

আপডেট এখন লিঙ্কে ক্লিক করলে থিম আপগ্রেড ইনস্টল হবে। অ্যাডমিনিস্ট্রেশন > ড্যাশবোর্ড > আপডেট স্ক্রিন থেকেও থিমগুলি আপগ্রেড করা যেতে পারে।

আপনার বর্তমান থিম ছাড়াও, থিমগুলি পরিচালনা করুন স্ক্রীনটি অন্যান্য থিমগুলিও দেখায় যা ইনস্টল করা আছে কিন্তু বর্তমানে নিষ্ক্রিয়। প্রতিটি থিম একটি ছোট স্ক্রিনশট দ্বারা প্রতিনিধিত্ব করা হয়। এই ইমেজগুলির উপর ঘোরানো 'থিম বিশদ', 'অ্যাক্টিভেট' এবং 'লাইভ প্রিভিউ' বোতামগুলি প্রকাশ করে। এছাড়াও আপনি এই পৃষ্ঠা থেকে নিষ্ক্রিয় থিমগুলি আপগ্রেড করতে বা মুছতে সক্ষম হবেন। এই স্ক্রিনের প্রতিটি পৃষ্ঠা একবারে 15টি থিমের স্ক্রিনশট প্রদর্শন করবে।

- সক্রিয় করুন

এই লিঙ্কে ক্লিক করলে এটি বর্তমান থিম হয়ে যায়।

- সরাসরি সম্প্রচার

এই লিঙ্কে ক্লিক করলে এই নির্দিষ্ট থিম সংস্করণ সহ ব্লগের একটি পূর্বরূপ দেখায়।

- মুছে ফেলা

এই লিঙ্কে ক্লিক করলে এই থিমটি সম্পূর্ণরূপে মুছে যাবে, সমস্ত থিম ফাইল এবং ফোল্ডার অন্তর্ভুক্ত করুন। ব্যাক আপ না করা কিছু চিরতরে হারিয়ে যাবে।

- আপডেট উপলব্ধ

উপরের আপডেট উপলব্ধ বিভাগে পড়ুন।

## থিম ইনস্টল করুন

থিম ইনস্টল করার বিভিন্ন উপায় নীচে তালিকাভুক্ত করা হয়েছে:

- স্বয়ংক্রিয় থিম ইনস্টলার

এটি ওয়ার্ডপ্রেস থিম ডিরেক্টরি থেকে থিম ইনস্টল করতে ব্যবহার করা যেতে পারে। উপস্থিতি থিম স্ক্রীন খুঁজতে প্রশাসন > উপস্থিতি > থিম-এ যান। Add New বাটনে ক্লিক করুন। এখান থেকে আপনি ব্যবহার করার জন্য থিমগুলি পাবেন যা পরিবর্তনমুক্ত। এই স্ক্রিনের শীর্ষে একটি নতুন থিম খুঁজে পেতে তিনটি উপলব্ধ পদ্ধতি সহ অনুসন্ধান বৈশিষ্ট্য রয়েছে; ফিল্টার, কীওয়ার্ড এবং অ্যাট্রিবিউট অনুসন্ধান।

- আপলোড পদ্ধতি ব্যবহার করে

আপলোড পদ্ধতি একটি জিপ ফাইলের মাধ্যমে একটি থিম ইনস্টল করে। ওয়ার্ডপ্রেস থিম ডিরেক্টরির সমস্ত থিম এইভাবে ইনস্টল করা যেতে পারে। জিপ ফাইল ডাউনলোড করার পরে, অ্যাডমিনিস্ট্রেশন > চেহারা > থিমগুলিতে যান এবং নতুন যোগ করুন বোতামে ক্লিক করুন। এরপরে, আপলোড থিম লিঙ্কে ক্লিক করুন।

জিপ ফাইলের জন্য ব্রাউজ করুন এবং এখন ইনস্টল করুন ক্লিক করুন। এটি বর্তমান থিম তৈরি করা শেষ করতে সক্রিয় লিঙ্কে ক্লিক করুন।

- FTP পদ্ধতি ব্যবহার করে

FTP পদ্ধতিতে একটি থিম ইনস্টল করতে আপনাকে প্রথমে আপনার স্থানীয় কম্পিউটারে থিম ফাইলগুলি ডাউনলোড করতে হবে।

জিপ ফাইলের বিষয়বস্তু বের করুন, ফাইলের কাঠামো সংরক্ষণ করুন এবং একটি নতুন ফোল্ডারে যুক্ত করুন। যদি থিম লেখক থেকে নির্দেশাবলী থাকে তবে সেগুলি অনুসরণ করতে ভুলবেন না।

আপনার সাইটের ওয়েব সার্ভার অ্যাক্সেস করতে একটি FTP ক্লায়েন্ট ব্যবহার করুন। ওয়ার্ডপ্রেস দ্বারা প্রদত্ত আপনার wp-content/themes ডিরেক্টরিতে আপলোড করা থিম ফাইল যোগ করুন। প্রয়োজন হলে, wp-content/themes ডিরেক্টরির মধ্যে আপনার নতুন থিম রাখার জন্য একটি ফোল্ডার তৈরি করুন। এর একটি উদাহরণ হল, যদি আপনার থিমের নাম টেস্ট করা হয় তবে এটি wp-content/themes/test এ থাকা উচিত।

Administration > Appearance > Themes-এ যান এবং Activate লিঙ্কে ক্লিক করুন আপনার বর্তমান থিম হিসেবে থিমটি নির্বাচন করুন।

- cPanel দিয়ে ইনস্টল করা হচ্ছে

cPanel কন্ট্রোল প্যানেলগুলি জিপ বা GZ ফাইলগুলির সাথে থিমগুলি ইনস্টল করার জন্য আরেকটি পদ্ধতি অফার করে। cPanel ম্যানেজারে, ওয়ার্ডপ্রেস ইনস্টল করা থাকলে, আপনার থিম ফোল্ডারে যান। পথটি 'public\_html/wp-content/themes'-এর মতো দেখতে হবে। আপলোড ফাইলে ক্লিক করুন এবং জিপ ফাইল আপলোড করুন। cPanel-এ ZIP ফাইলটি নির্বাচন করুন এবং সেই ফাইলটিকে আনকমপ্রেস করতে ডানদিকে প্যানেলে Extract File Contents-এ ক্লিক করুন।

Administration > Appearance > Themes-এ যান এবং Activate লিঙ্কে ক্লিক করুন আপনার বর্তমান থিম হিসেবে থিমটি নির্বাচন করুন।

উপরে তালিকাভুক্ত সমস্ত তথ্য ওয়ার্ডপ্রেস কোডেক্স অনুযায়ী। সংক্ষিপ্ততার জন্য এটি সংক্ষিপ্ত করা হয়েছে। মূল উৎস উপাদান এখানে পাওয়া যাবে. অথবা আরও তথ্যের জন্য [codex.wordpress.org](http://codex.wordpress.org) দেখুন।

## একটি কাস্টম থিম তৈরি করা

এই নির্দেশাবলী একটি খুব মৌলিক, ন্যূনতম মান সম্মত ওয়ার্ডপ্রেস থিম তৈরি করে।

প্রথম ধাপ হল আপনার ওয়ার্ডপ্রেস থিম ডিরেক্টরির ভিতরে একটি নতুন থিম ফোল্ডার তৈরি করা। সঠিক পথটি হবে: > wp-content > থিম > একটি বৈধ থিম তৈরি করতে, ওয়ার্ডপ্রেস থিমগুলিতে কমপক্ষে এই দুটি ফাইল থাকতে হবে:

- index.php
- style.css

আপনার স্টাইলশীটে একটি মন্তব্য থাকা উচিত যা ওয়ার্ডপ্রেসকে সতর্ক করে যে এখানে একটি থিম বিদ্যমান।

```
/*  
থিমের নাম: <theme name>  
লেখক: <author name>  
বর্ণনা: <description এখানে যায়>  
সংস্করণ: <থিম সংস্করণ #>  
ট্যাগ: <ট্যাগ টু আইডি থিম>  
*/
```

আপনার থিম এখন তৈরি করা হয়েছে. ওয়ার্ডপ্রেস ড্যাশবোর্ডে যান এবং এটি সক্রিয় করতে উপস্থিতি > থিমগুলিতে ক্লিক করুন।

## অধ্যায় 65: ম্যানুয়ালি ওয়ার্ডপ্রেস আপডেট করুন

### উদাহরণ

এফটিপি মাধ্যমে

আপনার স্থানীয় কম্পিউটারে [www.wordpress.org](http://www.wordpress.org) থেকে ওয়ার্ডপ্রেসের পছন্দসই সংস্করণ ডাউনলোড করুন 1. এবং ফাইলটি আনজিপ করুন।

- এছাড়াও আপনার বর্তমান সংস্করণের একটি ব্যাকআপ রাখুন... ঠিক সেক্ষেত্রে।
2. আপনার পছন্দের FTP ক্লায়েন্টের সাথে আপনার ওয়েবসাইটের সাথে সংযোগ করুন (FileZilla জনপ্রিয় এবং সহজ, কিন্তু যেকোনো FTP ক্লায়েন্ট ঠিক থাকবে)।
    - এর জন্য নির্দেশাবলী ওয়ার্ডপ্রেসের সুযোগের বাইরে, তবে প্রস্তাবিত FTP বিষয়ে ভবিষ্যতের তারিখে পাওয়া যেতে পারে।
  3. আপনার সার্ভারে "wp-admin" এবং "wp-includes" শিরোনামের ফোল্ডারগুলি (এবং তাদের বিষয়বস্তু) আপলোড করুন। বর্তমান ফোল্ডারগুলিকে ওভাররাইট করতে ভুলবেন না।
    - আপনি "wp-content" ফোল্ডার আপলোড করা বাদ দিতে পারেন যদি না আপনি অন্তর্ভুক্ত থিমগুলির একটি ব্যবহার করতে চান। আপনি যদি আপনার নির্বাচিত সংস্করণের সাথে অন্তর্ভুক্ত ডিফল্ট থিমগুলি আপডেট/আপলোড করতে চান তবে আপনাকে এই ফোল্ডারটিও আপলোড করতে হবে।
  4. হোম ফোল্ডারে পৃথক ফাইল আপলোড করুন (index.php, wp-\*.php, ইত্যাদি)।
    - আপনি "license.txt" এবং "readme.html" শিরোনামের ফাইলগুলিকে বাদ দিতে পারেন কারণ তাদের কাজ করার প্রয়োজন নেই এবং সেগুলি নিরাপত্তা কাজে লাগানোর জন্য আপনার WP সংস্করণ নির্ধারণের পদ্ধতি হিসাবে ব্যবহার করা যেতে পারে।
  5. যেকোনো প্রয়োজনীয় ডাটাবেস আপডেট করার জন্য আপনার ওয়েবসাইটে যান এবং লগ ইন করুন।
    - সব WP আপডেটে DB পরিবর্তন হয় না, কিন্তু কিছু হয়।

দ্রষ্টব্য: এই পদ্ধতিটি এতিম ফাইল তৈরি করবে যা সময়ের সাথে সাথে তৈরি হতে পারে এবং নিরাপত্তা ঝুঁকি উপস্থাপন করতে পারে।

সমাপ্তির পরে একটি ফাইল তুলনা করতে ভুলবেন না এবং আগের WP সংস্করণগুলি থেকে আপনার সার্ভার থেকে পুরানো ফাইলগুলি মুছে ফেলুন যা আর ব্যবহারে নেই।

# অধ্যায় 66: ওয়ার্ডপ্রেস প্লাগইন তৈরি

## ভূমিকা

ওয়ার্ডপ্রেস প্লাগইনগুলির আপনার ওয়েবসাইট অ্যাপ্লিকেশনের সার্ভার লজিক এবং/অথবা অ্যাডমিন অংশগুলিতে ফোকাস থাকা উচিত। ভাল প্লাগইনগুলি ভাল অ্যাপের মতো, তারা একটি জিনিস সত্যিই ভাল করে। এগুলি একটি মডুলার উপায়ে CMS এর অংশগুলিকে উন্নত এবং স্বয়ংক্রিয় করার উদ্দেশ্যে, যেহেতু আপনি সেগুলি সক্রিয় এবং নিষ্ক্রিয় করতে পারেন। ভাল প্লাগইনগুলি ওয়ার্ডপ্রেস কোর অ্যাকশন, ফিল্টার এবং বিদ্যমান জাভাস্ক্রিপ্ট এবং সিএসএস ফ্রেমওয়ার্ক ব্যবহার করে।

## উদাহরণ

একটি প্লাগইন ফোল্ডার এবং ফাইলের ন্যূনতম সেটআপ

প্লাগইন তৈরির প্রথম ধাপ হল ফোল্ডার এবং ফাইল তৈরি করা যা থেকে প্লাগইন লোড হবে।

প্লাগইনগুলি /wp-content/plugins/- এ অবস্থিত।

ওয়ার্ডপ্রেস স্ট্যান্ডার্ড হল একটি ফোল্ডার এবং ফাইলের নাম তৈরি করা যা একে অপরকে এভাবে প্রতিফলিত করে:

```
/wp-content/plugins/myplugin/ /wp-content/plugins/  
myplugin/myplugin.php
```

আপনার প্লাগইন ফাইল তৈরি করার পরে আপনাকে একটি প্লাগইন হেডার দিয়ে আপনার প্লাগইন শুরু করতে হবে। এটি ওয়ার্ডপ্রেসকে আপনার প্লাগইন ফাইল স্ক্যান করতে এবং প্লাগইন সম্পর্কে মেটা ডেটা সঞ্চয় করার অনুমতি দেয় এবং ব্যবহারকারীদের এটি ব্যবহার করতে এবং তারা আপনার প্লাগইন সক্রিয় বা নিষ্ক্রিয় করতে চায় কিনা তা নির্ধারণ করতে দেয়। এই টেমপ্লেটটি আপনার তৈরি আপনার প্রধান প্লাগইন ফাইলের শীর্ষে অনুলিপি করুন এবং প্রয়োজন অনুসারে এটি সংশোধন করুন:

```
<?php  
/**  
 * প্লাগইন নাম: PLUGIN-NAME  
 * প্লাগইন ইউআরআই: HTTP-লিঙ্ক-টু-ওয়েবসাইট-প্লাগইন-পৃষ্ঠা-বা-রেপো  
 * বর্ণনা: সংক্ষিপ্ত বিবরণ - এটি সংক্ষিপ্ত রাখুন  
 * লেখক: ওয়ার্ডপ্রেস-ডেভ-ওআরজি-ইউজারনেম  
 * সংস্করণ: 0.0.1  
 * লেখক URI: HTTP-লিঙ্ক-টু-রক্ষণাবেক্ষণকারী  
 * লাইসেন্স: GNU জেনারেল পাবলিক লাইসেন্স v2 বা তার পরে  
 * লাইসেন্স ইউআরআই: http://www.gnu.org/licenses/gpl-2.0.html * টেক্সট ডোমেন: short_prefix  
 */  
  
// কাস্টম পিএইচপি ওয়ার্ডপ্রেস প্লাগইন কাজ শুরু করুন
```

মনে রাখবেন যে ওয়ার্ডপ্রেস প্লাগইনগুলি সাধারণত GPL হিসাবে লাইসেন্স করা উচিত। যদিও এই বিষয়ের অংশ হিসেবে লাইসেন্সিং নিয়ে আলোচনা করা উচিত নয়।

এই মুহুর্তে, আপনি ইতিমধ্যেই ওয়ার্ডপ্রেস অ্যাডমিন এলাকায় আপনার নতুন প্লাগইন দেখতে সক্ষম হওয়া উচিত। একটি স্ট্যান্ডার্ড সেটআপে আপনি এই এলাকাটি /wp-admin/plugins.php এ সনাক্ত করবেন। এগিয়ে যান এবং আপনার সক্রিয়

প্লাগইন, এবং আপনি আপনার প্লাগইন তৈরির পরবর্তী ধাপে যেতে প্রস্তুত!

শুধু কর্মযোগ্য কিছুতে আমাদের উদাহরণ শেষ করার জন্য, আপনি এখন আপনার প্লাগইন ফাইলের নীচে নিম্নলিখিত যোগ করতে পারেন:

```
die('আমার কাস্টম প্লাগইন লোড হয়েছে :! __FILE__');
```

এই পরিবর্তনের পরে আপনার সাইট রিফ্রেশ করার ফলে সমস্ত পৃষ্ঠাগুলি এই পাঠ্যটি মুদ্রণ করবে। প্রোডাকশন (লাইভ) সাইটগুলিতে এটি কখনই করবেন না এবং চালিয়ে যাওয়ার আগে সর্বদা এটি ফিরিয়ে নেওয়ার কথা মনে রাখবেন।

# অধ্যায় 67: ওয়ার্ডপ্রেস থিম এবং চাইল্ড থিম বিকাশ

## ভূমিকা

ওয়ার্ডপ্রেস হল একটি বহুল ব্যবহৃত সিএমএস যা সাধারণ তথ্যের ওয়েবসাইট তৈরি করার জন্য কিন্তু আরও পরিশীলিত ওয়েবসাইট এবং এমনকি ছোট ওয়েবশপ তৈরির জন্যও।

ওয়ার্ডপ্রেস থিম ব্যবহার করে। এই থিমগুলি একটি Wordpress ওয়েবসাইটের লে-আউট এবং বিষয়বস্তু কার্যকারিতা তৈরি করতে ব্যবহৃত হয়। থিমগুলো ইন্টারনেটে পাওয়া যাবে।

প্রতিটি থিমের নিজস্ব অনন্য কার্যকারিতা এবং লে-আউট রয়েছে তবে কখনও কখনও একটি ওয়েবসাইটের জন্য সঠিক থিম খুঁজে পাওয়া কঠিন। ভাগ্যক্রমে আমরা আমাদের নিজস্ব থিম তৈরি করতে সক্ষম।

## উদাহরণ

### আপনার নিজস্ব থিম বিকাশ

একটি ওয়ার্ডপ্রেস থিমে দুই ধরনের ফাইল থাকে। প্রতিটি থিমের মৌলিক ফাইল এবং যে ফাইলগুলি থিমের বিন্যাস এবং কার্যকারিতা নির্ধারণ করে। এই দ্বিতীয় গ্রুপ আমি থিম নির্দিষ্ট ফাইল কল করতে যাচ্ছি।

#### মৌলিক থিম ফাইলগুলি মৌলিক

থিম ফাইলগুলি হল সেই ফাইলগুলি যা একটি থিম সেটআপ এবং নিবন্ধন করতে ব্যবহৃত হয়। নীচের তালিকায় আমি শীঘ্রই প্রতিটি ফাইল এবং এর ব্যবহার বর্ণনা করব। পরে আমি আপনার নিজের ওয়ার্ডপ্রেস থিম সেট আপ করার জন্য প্রয়োজনীয় সবচেয়ে মৌলিক উদাহরণ ফাইলগুলি যোগ করব।

- `functions.php`: `functions.php` ফাইলটি সমস্ত ফাংশন, সাইডবার, স্ক্রিপ্ট এবং থিমের অন্তর্ভুক্ত নিবন্ধন করতে ব্যবহৃত হয়। এই ফাইলটিতে আপনি, উদাহরণস্বরূপ, CSS ফাইল, JS ফাইল ইত্যাদি অন্তর্ভুক্ত করতে পারবেন।
- `হেডার এবং ফুটার`: `হেডার এবং ফুটার` ফাইলগুলি (`header.php` এবং `footer.php`) হল সেই ফাইলগুলি যা `হেডার এবং ফুটার` কল করতে ব্যবহৃত হয়। `হেডার এবং ফুটার` ফাইল উদাহরণস্বরূপ ওয়ার্ডপ্রেস ব্যাক-এন্ড সিস্টেমের লিঙ্কটি ধরে রাখে। `index.php`: `index.php` ফাইল হল সেই ফাইল যা ডিফল্ট-পৃষ্ঠা
- `টেমপ্লেট তৈরি করে`। এই ফাইলটিতে আপনি এই ডিফল্ট-টেমপ্লেট লে-আউটের টুকরো দেখতে, সম্পাদনা করতে এবং সরাসরি পারেন। `single.php`: `single.php` ফাইল হল সেই ফাইল যা একক পোস্ট টেমপ্লেট পৃষ্ঠা
- `তৈরি করে`। পৃষ্ঠাগুলির জন্য ডিফল্ট-টেমপ্লেটের মতো কিন্তু এখন একক পোস্ট পৃষ্ঠাগুলির জন্য। `format.php` `format.php` ফাইল হল সেই ফাইল যা একটি পৃষ্ঠা থেকে বিষয়বস্তু-টেমপ্লেট টেমপ্লেট তৈরি করে। তাই যদি
- আপনার একটি হোম পেজ থাকে এবং আপনি একটি টেমপ্লেট যোগ করে ব্যাক-এন্ড থেকে এটি সম্পাদনা করবেন।

এই ফাইলটি এই টেমপ্লেটের স্ট্যান্ডার্ড মার্কআপ তৈরি করে। `404.php`

- `404.php` ফাইলটি `404` টেমপ্লেট তৈরি করে। এই ফাইলটি এই পৃষ্ঠার মৌলিক লে-আউট নিয়ে গঠিত।

- `archive.php` `archive.php` ফাইলটি আর্কাইভ পৃষ্ঠার লে-আউট তৈরি করে। `style.css` মৌলিক
- `স্টাইলশীট` ফাইল।

তাই এই তালিকায় আপনি আপনার নিজস্ব ওয়ার্ডপ্রেস থিম সেট আপ করার জন্য প্রয়োজনীয় সমস্ত ফাইল দেখতে পারেন। এখন কিছু ফাইল দেখে নেওয়া যাক যা আপনি চাইলে তৈরি করতে পারবেন কিন্তু ওয়ার্ডপ্রেস থিমের জন্য প্রয়োজনীয় ফাইল নয়। এই ফাইলগুলি বেশিরভাগই টেমপ্লেট ফাইল এবং অন্যান্য কার্যকরী এক্সটেনশন।

কাস্টম পৃষ্ঠা টেমপ্লেট পৃষ্ঠা-<আপনার

নিজের নাম>.php: একটি ওয়ার্ডপ্রেস থিমে আপনি একাধিক পৃষ্ঠা টেমপ্লেট তৈরি করতে সক্ষম হন। নতুন পৃষ্ঠা টেমপ্লেট ফাইল তৈরি করে। একটি স্ট্যান্ডার্ড পৃষ্ঠা টেমপ্লেট ফাইল নিম্নলিখিত নামের বৈশিষ্ট্যগুলি নিয়ে গঠিত। টেমপ্লেটের পৃষ্ঠার নাম এবং .php যদি আপনি আপনার ব্লগ পৃষ্ঠার জন্য একটি নতুন পৃষ্ঠা টেমপ্লেট তৈরি করতে চান তবে আপনি এটিকে page-blog.php বলতে পারেন Wordpress স্বয়ংক্রিয়ভাবে ফাইলটি পড়ে এবং ফাইলটিকে চয়ন টেমপ্লেট মেনুতে যুক্ত করে। নিশ্চিত করুন যে আপনি অন্তত get\_header() এবং get\_footer() ফাংশন অন্তর্ভুক্ত করেছেন। এছাড়াও নিশ্চিত করুন যে আপনি নিম্নলিখিত উদাহরণ যোগ করে ফাইলের শীর্ষে একটি মন্তব্যে আপনার টেমপ্লেটের নাম দিয়েছেন।

```
<?php /*
 *
 * টেমপ্লেটের নাম: হোমপেজ টেমপ্লেট */ get_header();

?>
```

কাস্টম একক পোস্ট পৃষ্ঠা টেমপ্লেট একক-<your

own name>.php: উপরে বর্ণিত পৃষ্ঠা টেমপ্লেটের মতো একটি Wordpress থিমে আপনি আপনার নিজস্ব একক পোস্ট পৃষ্ঠা টেমপ্লেট তৈরি করতে সক্ষম। পৃষ্ঠার টেমপ্লেটের মতোই ফাইলটিতে তিনটি অংশ রয়েছে যা ঘোষণা করার জন্য এটি একটি একক পোস্ট পৃষ্ঠা <আপনার টেমপ্লেটের নাম> এবং ফাইল এক্সটেনশন .php। ঠিক পৃষ্ঠার টেমপ্লেটের মতো ন্যূনতম প্রয়োজনীয়তা নিশ্চিত করতে ওয়ার্ডপ্রেস নতুন টেমপ্লেটটি পড়তে পারে তা নিশ্চিত করতে get\_header() এবং get\_footer() ফাংশন যোগ করছে। এবং অবশ্যই নীচের উদাহরণের মত আপনার টেমপ্লেট নাম যোগ করুন

```
<?php

/*
 * টেমপ্লেটের নাম: পোস্ট পোর্টফোলিও
 * টেমপ্লেট পোস্টের ধরন: পোস্ট, পৃষ্ঠা */

?>
```

আমরা টেমপ্লেট পোস্টের ধরনটিও নির্দেশ করি: যা এই ক্ষেত্রে পোস্ট এবং পৃষ্ঠার ধরনের টেমপ্লেটকে বোঝায়।

কাস্টম পোস্ট টেক্সট টেমপ্লেট ফরম্যাট

-<আপনার নিজের নাম>.php: একটি ওয়ার্ডপ্রেস থিমে আপনি পোস্ট আউটপুট টেমপ্লেট তৈরি করতে পারবেন। এই বিন্যাস টেমপ্লেটগুলি একটি পোস্টের লে-আউট এবং বিষয়বস্তু। উদাহরণস্বরূপ, যদি কিছু ক্ষেত্রে আপনি পোস্টটি শুধুমাত্র বিষয়বস্তু বা পোস্টের শিরোনাম প্রদর্শন করতে চান তবে আপনি এই ধরনের সমন্বয় তৈরি করতে এই টেমপ্লেটগুলি ব্যবহার করতে পারেন। যেহেতু এই ধরনের টেমপ্লেটগুলি শুধুমাত্র পোস্টের ব্যাক-এন্ড কন্টেন্ট ফর্ম্যাট করে যা একজন ব্যবহারকারীর দ্বারা তৈরি করা হয়েছে আমাদের get\_header() এবং get\_footer() অন্তর্ভুক্ত করার দরকার নেই কারণ এইগুলি ইতিমধ্যেই পেজ টেমপ্লেটগুলিতে সংজ্ঞায়িত করা আছে। নিশ্চিত করুন যে আপনার টেমপ্লেট নিম্নলিখিত মৌলিক উদাহরণ ব্যবহার করে একটি পোস্ট চিনতে সক্ষম।

```
<div>
```

```
<আর্টিকেল id="post-<?php the_ID(); ?>" <?php post_class(); ?>> </আর্টিকেল> </div>
```

তাই এখন যেহেতু আমরা মৌলিক ফাইল এবং অনেক টেমপ্লেট নির্দিষ্ট ফাইল সম্পর্কে কিছু জানি এখন সাইডবার এবং উইজেট সম্পর্কে কথা বলা শুরু করার সময়। ভবিষ্যতে এটি একটি নিজস্ব ওয়ার্ডপ্রেস থিম তৈরির ধাপ থেকে ধাপে টিউটোরিয়ালের শুরুর সাথে যোগ করা হবে।

# অধ্যায় 68: wp\_get\_current\_user()

বাক্য গঠন

- wp\_get\_current\_user()

## উদাহরণ

বর্তমান ব্যবহারকারী পেয়ে

পূর্বনির্ধারিত ফাংশন ব্যবহার করে ওয়ার্ডপ্রেসে বর্তমান ব্যবহারকারীর সমস্ত তথ্য পাওয়া  
wp\_get\_current\_user();

```
<?php

$current_user = wp_get_current_user();

প্রতিধ্বনি "ব্যবহারকারীর নাম :"$current_user->user_login; প্রতিধ্বনি "ব্যবহারকারীর
নাম :"$current_user->ID; প্রতিধ্বনি "ব্যবহারকারীর
নাম :"$current_user->user_pass; প্রতিধ্বনি "ব্যবহারকারীর নাম :"$current_user-
>user_nicename; প্রতিধ্বনি "ব্যবহারকারীর নাম :"$current_user->user_email;
প্রতিধ্বনি "ব্যবহারকারীর নাম :"$current_user->user_url; প্রতিধ্বনি "ব্যবহারকারীর
নাম :"$current_user->user_registered; প্রতিধ্বনি "ব্যবহারকারীর
নাম :"$current_user->user_activation_key; প্রতিধ্বনি "ব্যবহারকারীর নাম :"$current_user-
>user_status; প্রতিধ্বনি "ব্যবহারকারীর নাম :"$current_user->display_name;

?>
```

wp\_get\_current\_user() থেকে ব্যবহারকারীর তথ্য পেতে foreach লুপ ব্যবহার করুন

```
$user = wp_get_current_user();

foreach($user->ডাটা হিসেবে $key=>$user_data){
    যদি($key == 'user_pass' || $key == 'user_activation_key' || $key=='user_status'){} else{ $nice_key = ucfirst(str_replace('_', '', $key));

    if($key == 'user_registered'){ $user_data =
        date_i18n(get_option('date_format'), strtotime($user_data));
    }

    প্রতিধ্বনি $nice_key. ' : ' . $user_data | <br />;
}
}
```

# অধ্যায় 69: wp\_get\_current\_user()

## উদাহরণ

বর্তমান লগইন ব্যবহারকারী তথ্য পান

বর্তমানে লগ ইন করা ব্যবহারকারীর সাথে সম্পর্কিত তথ্য পুনরুদ্ধার করে এবং এটিকে বিশ্বব্যাপী স্থাপন করে পরিবর্তনশীল `$current_user`

এই ফাংশন কোনো পরামিতি গ্রহণ করে না।

ব্যবহার:

```
<?php wp_get_current_user(); ?>
```

উদাহরণ:

```
<?php
$current_user = wp_get_current_user();

ইকো 'ব্যবহারকারীর নাম:'          . $current_user->user_login! "\n";
প্রতিধ্বনি 'ব্যবহারকারীর ইমেল:'    . $current_user->user_email! "\n";
ইকো 'ব্যবহারকারীর স্তর:'          . $current_user->user_level "\n";
প্রতিধ্বনি 'ব্যবহারকারীর প্রথম নাম:' . $current_user->user_firstname "\n";
প্রতিধ্বনি 'ব্যবহারকারীর শেষ নাম:' . $current_user->user_lastname "\n";
প্রতিধ্বনি 'ব্যবহারকারীর প্রদর্শন নাম:' . $current_user->display_name! "\n";
ইকো 'ইউজার আইডি:' $current_user->আইডি! "\n";

?>
```

একজন ভিজিটর লগ ইন করেছে কি না তা নির্ধারণ করতে, আপনি আগে `is_user_logged_in()` ব্যবহার করতে পারেন, তারপর পান বর্তমান ব্যবহারকারীর তথ্য যদি ভিজিটর লগ ইন করে থাকে:

```
<?php
যদি ( is_user_logged_in() ) $current_user = {
    wp_get_current_user();

    প্রতিধ্বনি 'স্বাগত,' ) অন্য {          . $current_user->user_login! '!';

    প্রতিধ্বনি 'স্বাগত, দর্শক!';
}
?>
```

---

## অধ্যায় 70: WP\_Query() লুপ

### ভূমিকা

পোস্ট, পৃষ্ঠা এবং কাস্টম পোস্ট প্রকারের জন্য প্রশ্ন করার জন্য WP\_Query। আপনি নির্দিষ্ট পোস্ট এবং পেজ বা কাস্টম পোস্ট প্রকারের জন্য তালিকা পাবেন। WP\_Query আপনাকে আপনার মানদণ্ড অনুযায়ী ডাটাবেস থেকে পোস্ট টানতে দেয়।

### উদাহরণ

সর্বশেষ 10টি পোস্ট পুনরুদ্ধার করুন

```
$args = অ্যারে( 'post_type'  
=> 'পোস্ট', 'posts_per_page' =>  
'10'  
);  
$latest_posts_query = নতুন WP_Query($args); if($latest_posts_query-  
>have_posts()); যখন ($latest_posts_query-> have_posts());  
$latest_posts_query->the_post();  
//পোস্টের বিস্তারিত এখানে শেষ পর্যন্ত পান;  
যদি শেষ;
```

# অধ্যায় 71: WP-CLI

## ভূমিকা

WP-CLI হল ওয়ার্ডপ্রেস ইনস্টলেশন পরিচালনার জন্য কমান্ড-লাইন টুলের একটি সেট। আপনি ওয়েব ব্রাউজার ব্যবহার না করে প্লাগইন আপডেট করতে, মাল্টিসাইট ইনস্টল কনফিগার করতে এবং আরও অনেক কিছু করতে পারেন।

## উদাহরণ

থিম পরিচালনা করুন

থিম একটি তালিকা পান.

```
$ wp থিম তালিকা
```

wordpress.org থেকে সর্বশেষ সংস্করণটি ইনস্টল করুন এবং সক্রিয় করুন

```
$wp থিম ইনস্টল টুয়েন্টিসিক্সটিন --অ্যাক্টিভেট
```

একটি স্থানীয় জিপ ফাইল থেকে ইনস্টল করুন

```
$wp থিম ইনস্টল করুন ../my-theme.zip
```

একটি দূরবর্তী জিপ ফাইল থেকে ইনস্টল করুন

```
$wp থিম ইনস্টল http://s3.amazonaws.com/bucketname/my theme.zip?  
AWSAccessKeyId=123&Expires=456&Signature=abcdef
```

একটি ইনস্টল করা থিমের বিশদ বিবরণ পান

```
$ wp থিম বাইশ-ফিল্ডস = নাম, শিরোনাম, সংস্করণ পান
```

থিমের স্ট্যাটাস পান

```
$wp থিম স্ট্যাটাস টুয়েন্টিসিক্সটিন
```

প্লাগইন পরিচালনা করুন

প্লাগইনগুলির একটি তালিকা পান

```
$ wp প্লাগইন তালিকা
```

সাইটে সক্রিয় প্লাগইন তালিকা.

```
$ wp প্লাগইন তালিকা --status=active --format=json
```

একটি নেটওয়ার্কে প্রতিটি সাইটে প্লাগইন তালিকাভুক্ত করুন।

```
$ wp সাইটের তালিকা --field=url | xargs -I % wp প্লাগইন তালিকা --url=%
```

প্লাগইন সক্রিয় করুন

```
$wp প্লাগইন হ্যালো-ডলি সক্রিয় করুন
```

প্লাগইন নিষ্ক্রিয় করুন

```
$wp প্লাগইন হ্যালো-ডলি নিষ্ক্রিয় করুন
```

প্লাগইন মুছুন

```
$wp প্লাগইন হ্যালো-ডলি মুছে দিন
```

wordpress.org থেকে সর্বশেষ সংস্করণটি ইনস্টল করুন এবং সক্রিয় করুন

```
$wp প্লাগইন bbpress --activate ইনস্টল করুন
```

WP-CLI নিজেই পরিচালনা করুন

বর্তমানে ইনস্টল করা সংস্করণ প্রদর্শন করুন।

```
$ wp cli সংস্করণ
```

WP-CLI-তে আপডেটের জন্য চেক করুন।

```
$wp cli চেক-আপডেট
```

সর্বশেষ স্থিতিশীল রিলিজে WP-CLI আপডেট করুন।

```
$wp cli আপডেট
```

সমস্ত উপলব্ধ উপনামের তালিকা করুন।

```
$wp cli উপনাম
```

WP-CLI পরিবেশ সম্পর্কে বিভিন্ন বিবরণ প্রিন্ট করুন।

```
$ wp cli ভথ্য
```

JSON হিসাবে ইনস্টল করা কমান্ডের তালিকা ডাম্প করুন।

```
$wp cli cmd-ডাম্প
```

একটি ওয়ার্ডপ্রেস ইনস্টল ডাউনলোড, ইনস্টল, আপডেট এবং পরিচালনা করুন।

ওয়ার্ডপ্রেস কোর ডাউনলোড করুন

```
$ wp কোর ডাউনলোড --locale=nl_NL
```

ওয়ার্ডপ্রেস ইন্সটল করুন

```
$ wp core install --url=example.com --title=Example --admin_user=supervisor -- admin_password=strongpassword --  
admin_email=info@example.com
```

ওয়ার্ডপ্রেস সংস্করণ প্রদর্শন করুন

```
$ wp মূল সংস্করণ
```

একটি একক-সাইট ইনস্টলকে একটি ওয়ার্ডপ্রেস মাল্টিসাইট ইনস্টলে রূপান্তর করুন।

```
$ wp core multisite-convert
```

স্ক্যাচ থেকে ওয়ার্ডপ্রেস মাল্টিসাইট ইনস্টল করুন।

```
$wp core multisite-install
```

ব্যবহারকারীদের ম্যানেজ করুন

ব্যবহারকারীর আইডি তালিকা করুন

```
$ wp ব্যবহারকারী তালিকা --field=ID
```

একটি নতুন ব্যবহারকারী তৈরি করুন.

```
$ wp ব্যবহারকারী তৈরি করুন bob@example.com --role=author
```

একটি বিদ্যমান ব্যবহারকারী আপডেট করুন.

```
$ wp ব্যবহারকারী আপডেট 123 --display_name=Mary --user_pass=marypass
```

ব্যবহারকারী 123 মুছুন এবং 567 ব্যবহারকারীকে পোস্টগুলি পুনরায় বরাদ্দ করুন

```
$ wp ব্যবহারকারী মুছে ফেলুন 123 --reassign=567
```

wp-config.php-এ সংরক্ষিত শংসাপত্র ব্যবহার করে মৌলিক ডাটাবেস অপারেশনগুলি সম্পাদন করুন

একটি নতুন ডাটাবেস তৈরি করুন।

```
$ wp db তৈরি করুন
```

একটি বিদ্যমান ডাটাবেস ফেলে দিন।

```
$ wp db ড্রপ -- হ্যাঁ
```

বর্তমান ডাটাবেস রিসেট করুন।

```
$ wp db রিসেট -- হ্যাঁ
```

একটি ফাইলে সংরক্ষিত একটি এসকিউএল কোয়েরি চালান।

```
$ wp db ক্যোয়ারী < debug.sql
```

# অধ্যায় 72: WP-Cron

## উদাহরণ

### wp\_schedule\_event() উদাহরণ

```
// রেজিস্টার অ্যাক্টিভেশন হুক
register_activation_hook( __FILE__, 'example_activation' );

// অ্যাক্টিভেশন হুক ফাংশনের জন্য ফাংশন example_activation()
{
    // নির্ধারিত হুক বিদ্যমান কিনা তা পরীক্ষা করুন যদি (!
    wp_next_scheduled( 'my_event' ) ) { // একটি হুক নির্ধারণ করে // time() -
        একটি ইভেন্টের প্রথমবার চালানোর
        জন্য ( UNIX টাইমস্ট্যাম্প বিন্যাস ) // 'ঘন্টা' - পুনরাবৃত্তি ( 'ঘন্টা', 'দুইবার', 'প্রতিদিন' ) // 'my_event' - চালানোর জন্য একটি অ্যাকশন হকের নাম।
        wp_schedule_event( time(), 'hourly', 'my_event' );
    }
}

add_action( 'my_event', 'do_this_hourly' );

// আপনার ঘন্টায় ইভেন্ট ফাংশনের কোড do_this_hourly() { //
আপনার কোড এখানে রাখুন

}

// রেজিস্টার নিষ্ক্রিয়করণ হুক
register_deactivation_hook( __FILE__, 'example_deactivation' );

// নিষ্ক্রিয়করণ হুক ফাংশনের জন্য ফাংশন example_deactivation()
{
    // পরিষ্কার নির্ধারিত হুক
    wp_clear_scheduled_hook( 'my_event' );
}
}
```

গুরুত্বপূর্ণ: ওয়ার্ডপ্রেস ক্রন তখনই চলে যখন আপনার ওয়েবসাইটের কিছু পৃষ্ঠা হিট হয়। সুতরাং, কম ট্রাফিক সহ ওয়েবসাইটের জন্য আপনাকে আপনার হোস্টিং-এ পৃষ্ঠাগুলি হিট করার জন্য ক্রোন সেটআপ করতে হবে।

### wp\_schedule\_event() এ কাস্টম পুনরাবৃত্তি ব্যবধান

```
// এই ফাংশনটি $schedules ফাংশনে কাস্টম ব্যবধান (5 মিনিট) যোগ করে Five_minutes_interval( $schedules )
$schedules['five_minutes'] = অ্যারে(
    {
        'interval' => 60*5, 'display' => '5 মিনিট';
    }
);
ফেরত $শিডিউল;
}

// একটি কাস্টম ব্যবধান ফিল্টার যোগ করুন
add_filter( 'cron_schedules', 'five_minutes_interval' );
```

```
// একটি হক শিডিউল করে
```

```
wp_schedule_event( time(), 'five_minutes', 'my_event' );
```

এস. না	অধ্যায়	অবদানকারী
1	দিয়ে শুরু করা ওয়ার্ডপ্রেস	4444, উঃ রাজা, অ্যাল্ডু, শক্রতা অনুপম, ক্রিস ফ্লেচার, সিপ্রিয়ান, সম্প্রদায়, ফ্লোরিডা, জেমস জোল, জোনাসসিজেড, লিও চ, মার্ক সেন্ট রেমন্ড, মায়াক্স গুপ্তা, মিলাপ, নুস, পান্ডা, র্যাপ-2-ঘণ্টা, শেঠ সি., শুভম, ট্রেভর ক্লার্ক, বজ্রসার
2	অ্যাকশন এবং ফিল্টার	ডেভিড, ইহর ভোরটনভ, মৃগাল হক, নিজেকে চেষ্টা করছি
3	শর্টকোড যোগ করুন	purvik7373, রমেনশেফ
4	পরিচিতি যোগ করুন/সরান সঙ্গে ব্যবহারকারীদের জন্য তথ্য user_contact methods ফিল্টার হক	পেটার পপোভিচ, রমেনশেফ
5	যোগ_ক্রিয়া()	অ্যাবেল মেলকিয়াদেস ক্যালাজো, ওয়াকাস বুখারী
6	add_editor_style()	গ্যাব্রিয়েল চি হং লি
7	যোগ_মেনু_পৃষ্ঠা()	ব্রাসোফিলো, গ্যাব্রিয়েল চি হং লি
8	add_submenu_page()	গ্যাব্রিয়েল চি হং লি, তাত্ত্বিক
9	add_theme_support()	গ্যাব্রিয়েল চি হং লি
10	অ্যাডমিন ড্যাশবোর্ড উইজেট	তাত্ত্বিক
11	AJAX	অ্যান্ডি, দিগ্বিজয়দ, গৌরব শ্রীবাস্তব, গ্রেটরেক্স, নিসর্গ প্যাটেল, রুসলান মুরারভ, কাঁটা
12	বিকল্প প্রধান লুপ (pre_get_posts ফিল্টার)	দাউদ আরবানস্কি, পেটার পপোভিচ
13	চাইল্ড থিম বেসিক	আন্দ্রেই, রাজভান ওনোফ্রেই, ব্লাদ ওলারু
14	একটি পোস্ট তৈরি করুন প্রোগ্রামগতভাবে	রমেনশেফ, রোয়েল ম্যাগডালেনো, আররিকেশ
15	এর জন্য টেমপ্লেট তৈরি করুন কাস্টম পোস্টের ধরন	অশোক জি, এগনারো, জো ডুলি, মনরোনহা
16	একটি রীতি তৈরি করা টেমপ্লেট	পেটার পপোভিচ

17	সঙ্গে কাস্টম excerpts excerpt_length এবং excerpt_more	inkista, পেটার পপোভিচ, রমেনশেফ
18	কাস্টম পোস্ট প্রকার	কাইও ফেলিপ পেরেইরা, ড্যান ডিভাইন, জেডি, জানউ, jgraup, কুশল শাহ, ওমর খৈয়াম, রানুকা, তাত্ত্বিক
19	কাস্টমাইজার বেসিক (প্যানেল, বিভাগ যোগ করুন, সেটিং, নিয়ন্ত্রণ)	4444, আহমদ আওয়াইস, রমেনশেফ
20	কাস্টমাইজার হ্যালো বিশ্ব	ড্যান গ্রিন-লিপসিগার
21	ডিবাগিং	বারবোক, ডিঙ্গো_ডি, jgraup
22	সারিবদ্ধ স্ক্রিপ্ট	ডিঙ্গো_ডি, হর্ষাল লিমায়ে, জেডি, এমব্যাকন40, মন্ট্রিলিস্ট, Pelmered, পেটার পপোভিচ
23	সারিবদ্ধ শৈলী	ডিঙ্গো_ডি, হর্ষাল লিমায়ে, লক্ষণ, মনরোনহা, মন্ট্রিলিস্ট, পেটার পপোভিচ, রমেনশেফ, রুসলান মুরারভ, ভার্চুয়াল লাস্ট
24	ফাংশন: wp_trim_words()	হর্ষাল লিমায়ে
25	ফাংশন: add_action()	bosco, রমেনশেফ
26	get_bloginfo()	অ্যাবেল মেলকিয়াদেস ক্যালেজো, হর্ষাল লিমায়ে, হেই ক্যামেরন, কেনবি , নেট বিয়ার, রমেনশেফ, টম জে নওয়েল, ভার্চুয়াল শেষ, ওয়েস মোবারলি
27	get_home_path()	ইহর ভোরটনভ
28	get_option()	গ্যাব্রিয়েল চি হং লি
29	get_permalink()	গ্যাব্রিয়েল চি হং লি
30	get_template_part()	ড্যান ডিভাইন, কুশল শাহ
31	get_the_category()	গ্যাব্রিয়েল চি হং লি
32	টি_শিরোনাম ()	গ্যাব্রিয়েল চি হং লি
33	home_url()	ডিঙ্গো_ডি, কুশল শাহ, ম্যাথু, মিস্টার ডেভেলপার
34	কিভাবে আমি একত্রিত করতে পারেন সঙ্গে মার্কডাউন সম্পাদক অগ্রিম কাস্টম ফিল্ডের রিপিটার অ্যাড চালু.	ফ্যাটবিট

35	init	অ্যাবেল মেলকিয়াদেস ক্যালাজো, বারবোক
36	ইনস্টলেশন এবং কনফিগারেশন	কেনিয়ন, মার্কো রোমানো, পিং.চেন, এসএল বার্থ, stig-js, তাত্ত্বিক, ইউয়ান লুং লুও
37	নেটওয়ার্ক তৈরি করা HTTP এর সাথে অনুরোধ API	জর্ডান, মজংদা, রাস্ট
38	মেটা বক্স	অস্টিন উইনস্টানলি
39	বিকল্প API	হর্ভাল লিমায়ে, প্যাট জে, রমেনশেফ
40	প্লাগইন ডেভেলপমেন্ট	কোণ.ৱ, পিং.চেন
41	পোস্ট ফরম্যাট	শশাঙ্ক আগরওয়াল
42	পোস্ট জিজ্ঞাসা	dingo_d
43	অটো লাইন সরান বিষয়বস্তু থেকে বিরতি এবং উদ্ধৃতি	অস্টিন উইনস্টানলি
44	থেকে সংস্করণ সরান ওয়ার্ডপ্রেস এবং স্টাইলশীট	জয় জীবনী, মনরোনহা, তাত্ত্বিক
45	REST API	পিকার্ড
46	ওয়ার্ডপ্রেস স্থানীয় চালান XAMPP সহ	পিয়ের.ভিয়েল, তাত্ত্বিক
47	আপনার নিরাপদ স্থাপন	জেমস জোল
48	ওয়ার্ডপ্রেসে নিরাপত্তা - পালানো	লক্ষ্মণ, 4কিলোমিটার
49	ওয়ার্ডপ্রেসে নিরাপত্তা - স্যানিটাইজেশন	লক্ষ্মণ
50	শর্টকোড	অ্যাড উইকস, অ্যাডাম জেনশ্যাফ্ট, ব্রাসোফিলো, জন স্নেজারস, কিলার, শশাঙ্ক আগরওয়াল
51	সঙ্গে শর্টকোড বৈশিষ্ট্য	দিগ্বিজয়দ, আণ্ডনের কুয়াশা, রমেনশেফ
52	শর্টকোড	Pelmered
53	সাইডবার	ডিজো_ডি, কুশল শাহ, তাত্ত্বিক

54	সাইট মাইগ্রেশন	অস্টিন উইনস্টানলি
55	শ্রেণীবিন্যাস	adifatz, কুশল শাহ, purvik7373
56	টেমপ্লেট অনুক্রম	jgraup, মারজাব, Pelmered, তাত্ত্বিক
57	টেমপ্লেট_অন্তর্ভুক্ত	অ্যাবেল মেলকিয়াদেস ক্যালোজো, ডেভিড, রমেনশেফ
58	\$wpdb অবজেক্ট	কুশল শাহ, mcon, কাঁটা
59	অ্যাডমিন বার (ওরফে "দ্যা টুলবার")	ডিজো_ডি, হর্ষাল লিমায়ে, JCL1178, কুশল শাহ
60	লুপ (প্রধান ওয়ার্ডপ্রেস লুপ)	anik4e, ডেভিড আরবানস্কি
61	the_title()	গ্যাব্রিয়েল চি হং লি
62	থিম	জেফ
63	ওয়ার্ডপ্রেস আপডেট করুন ম্যানুয়ালি	নাইটহক
64	ওয়ার্ডপ্রেস প্লাগইন সৃষ্টি	শেঠ সি.
65	ওয়ার্ডপ্রেস থিম এবং শিশু-থিম উন্নয়ন	মুতুঝাড়
66	wp_get_current_user()	Abel Melquiades Callejo, বেনোটি, মুহাম্মদ ফররুখ ফয়েজী
67	WP_Query() লুপ	ব্রজেশ
68	WP-CLI	jgraup
69	WP-Cron	কাঁটা