



	REGULAR EXPRESSION SYNTAX	DATE FORMATTING	Regex
Arrays array_intersect array_merge array_pop array_push array_reverse array_walk count in_array	^ Start of string \$ End of string . Any single character (a b) a or b (...) Group section [abc] Item in range (a or b or c) [^abc] Not in range (not a or b or c)	Y 4 digit year 2005 y 2 digit year 05 F Long month January M Short month Jan m Month (leading zeros) 01 to 12 n Month 1 to 12 D Short day name Mon l Long day name Monday d Day (leading zeros) 01 to 31 j Day 1 to 31	ereg eregi ereg_replace eregi_replace split spliti preg_match preg_match_all preg_replace preg_split
Date and Time date mktime strtotime time	\s White space a? Zero or one of a a* Zero or more of a a+ One or more of a	h 12 Hour (leading zeros) 01 to 12 g 12 Hour 1 to 12 H 24 Hour (leading zeros) 00 to 23 G 24 Hour 0 to 23 i Minutes (leading zeros) 00 to 59 s Seconds (leading zeros) 00 to 59	Sessions session_destroy session_id session_start
Filesystem clearstatcache copy fclose fgets file filemtime filesize file_exists fopen fread fwrite is_dir is_file readfile	a{3} Exactly 3 of a a{3,} 3 or more of a a{3,6} Between 3 and 6 of a \ Escape character [:punct:] Any punctuation symbol [:space:] Any space character [:blank:] Space or tab	w Day of week ¹ 0 to 6 z Day of year 0 to 365 W Week of year ² 1 to 53 t Days in month 28 to 31	Strings addslashes crypt echo explode implode md5 nl2br sprintf strip_tags stripslashes stristr strpos strrev strstr strtolower strtoupper str_replace substr trim
Headers header headers_sent setcookie	PCRE MODIFIERS i Case-insensitive s Period matches newline m ^ and \$ match lines U Ungreedy matching e Evaluate replacement x Pattern over several lines	a am or pm A AM or PM B Swatch Internet Time 000 to 999 S Ordinal Suffix st, nd, td, th	URLs rawurldecode rawurlencode urldecode urlencode
Mail mail		T Timezone of machine GMT Z Timezone offset (seconds) O Difference to GMT (hours) +0200 I Daylight savings 1 or 0 L Leap year 1 or 0	Variables empty is_array is_int is_null is_numeric is_object isset serialize unserialize
Numbers ceil floor number_format round	FUNCTION ARGUMENT ORDERS array_push (array, element) in_array (needle, haystack) explode (separator, string) split (pattern, string) preg_split (pattern, string) ereg (pattern, string) strpos (haystack, needle) str_replace (search, replace, string)	U Seconds since Epoch ³ c ISO 8601 (PHP 5) r RFC 2822	SuperGlobals \$_ENV \$_GET \$_POST \$_COOKIE \$_SESSION \$_SERVER \$_FILES \$_REQUEST \$GLOBALS
Miscellaneous define eval exit		1. date("w"): 0 is Sunday, 6 is Saturday. 2. Week that overlaps two years belongs to year that contains most days of that week. Hence week number for 1st January of a given year can return 53rd week if week belongs to previous year. date("W", mktime(0, 0, 0, 12, 28, \$year)) always gives correct number of weeks in \$year. 3. The Epoch was the 1st January 1970.	
Network fsockopen			
PHP Options ini_get ini_restore ini_set phpinfo	fopen() MODES r Read r+ Read and write, prepend w Write, truncate w+ Read and write, truncate a Write, append a+ Read and write, append		
Execution exec			