

HIZLI AÇIKLAMA KLAVUZU

-- : Azalt operatörü

' : Biçimli dizgede özel dolgu kullanılacağını belirtmek için kullanılır. Muhakkak genişlik belirticileri ile birlikte kullanılması gerekir. Varsayılan değeri boşluktur. (printf() veya sprintf() metodu ile)

- : Çıkart operatörü

- : Değerin sola doğru genişliğinin kaç karakter olması gerektiğini belirtir. Muhakkak genişlik belirticileri ile birlikte kullanılması gerekir. (printf() veya sprintf() metodu ile)

- : Negatif ol operatörü

! : Değil operatörü

!= : Eşit değil (MySQL'de)

!= : Eşit değil ise operatörü

!== : Denk değil ise operatörü

: Gizli açıklama satırı tanımlamak için kullanılır.

\$: Karakter diziliminin sonu ile düzenli ifade arasında bir eşleşme aramak için kullanılır. (RegExp ile)

\$: Karakter diziliminin sonu ile ifade arasında bir eşleşme aramak için kullanılır. (.htaccess ile)

\$_COOKIE : Herhangi bir kullanıcının browser'ına yerleştirilmiş olan verilerine ulaşılabilme imkanı tanır.

\$_FILES : Http server'ına herhangi bir dosya yükleme işlemi esnasında, ilgili dosya verilerine ulaşılabilme imkanı tanır.

\$_GET : URL üzerinden gelen verilere ulaşılabilme imkanı tanır. Genellikle URL üzerinden gelen istekler ve form işleme işlemleri için kullanılmaktadır.

\$_POST : İstek gönderim tag'ları aracılığı ile gelen verilere ulaşılabilme imkanı tanır. Genellikle kapalı yöntemle gelen istekler ve form işleme işlemleri için kullanılmaktadır.

\$_REQUEST : URL üzerinden veya istek gönderim tag'ları aracılığı ile gelen verilere ulaşılabilme imkanı tanır. Genellikle URL üzerinden gelen istekler, kapalı yöntemle gelen istekler ve form işleme işlemleri için kullanılmaktadır.

\$_SERVER : Http server'ı tarafından oluşturulan, server ve işletme ortamı bilgilerine ulaşılabilme imkanı tanır. Fakat her Http server'ı bu değerleri oluşturacak diye bir garanti yoktur. Ayrıca her Http server'ı farklı değerler üretebilir.

\$_SESSION : Herhangi bir kullanıcı için session tanımlama imkanı tanır. Ayrıca daha önceden tanımlanmış olan session verilerine de ulaşılabilme imkanı tanır.

\$GLOBALS : Atanacak olan değişken için kapsama / etki alanlarının değiştirilmesini sağlar. Bu sayede global alanda tanımlanmış bir değişkene local alandan, local alanda tanımlanmış bir değişkene global alandan ulaşılabilme imkanı tanır.

\$this : Bu sınıfta şeklinde sınıfı işaret etmek için kullanılır.

% : Biçimli dizge istek talebinin tanımlanmasını sağlar. Dizge istek talebi belirtilmezse hata kodu dönmeye neden olur. Fakat string veri türünde % işareti kullanılması gerekecek olur ise iki tane yan yan % işareti kullanılmalıdır. (printf() veya sprintf() metodu ile)

% : Konum (MySQL'de)

% : Modülüs (kalan) operatörü

%% : % karakteri değerini döndürür. (strftime() metodu ile)

%= : Modülüs (kalan) ataması operatörü

%A : Günün adı değerini döndürür. (Yerel) (strftime() metodu ile)

%a : Günün üç haneli olacak şekilde adı değerini döndürür. (Yerel) (strftime() metodu ile)

%B : Ay adı değerini döndürür. (Yerel) (strftime() metodu ile)

%b : Ayın üç haneli olacak şekilde adı değerini döndürür. (Yerel) (strftime() metodu ile)

%c : Tam tarih ve saat değerini döndürür. (Yerel) (strftime() metodu ile)

%C : Yüzyılın iki haneli olacak şekilde sayısal değerini döndürür. (Yerel) (strftime() metodu ile)

%D : Ay, gün ve yıl sıralamasını esas alarak tarih değerini döndürür. (Yerel) (strftime() metodu ile)

%d : Gün farkı değerini döndürür. (date_diff() metodu ile)

%d : Günün iki haneli olacak şekilde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (Yerel) (strftime() metodu ile)

%e : Günün sayısal değerini döndürür. (Yerel) (strftime() metodu ile)

%F : Yıl, ay, gün sıralamasını esas alarak tarih değerini döndürür. (Yerel) (strftime() metodu ile)

%G : Yılın dört haneli olacak şekilde sayısal değerini döndürür. (Yerel) (strftime() metodu ile)

%g : Yılın iki haneli olacak şekilde sayısal değerini döndürür. (Yerel) (strftime() metodu ile)

%H : 24 saatlik sistemde saatin iki haneli olacak şekilde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (Yerel) (strftime() metodu ile)

%h : Ayın üç haneli olacak şekilde adı değerini döndürür. (Yerel) (strftime() metodu ile)

%h : Saat farkı değerini döndürür. (date_diff() metodu ile)

%I : 12 saatlik sistemde saatin iki haneli olacak şekilde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (Yerel) (Büyük ı harfi) (strftime() metodu ile)

%i : Dakika farkı değerini döndürür. (date_diff() metodu ile)

%j : Yılın gününün sayısal değerini döndürür. 1'den 366'ya (artık gün) kadar. (Yerel) (strftime() metodu ile)

%m : Ay farkı değerini döndürür. (date_diff() metodu ile)

%m : Ayın iki haneli olacak şekilde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler (Yerel) (strftime() metodu ile)

%M : Dakikanın iki haneli olacak şekilde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (Yerel) (strftime() metodu ile)

%n : New line karakteri değerini döndürür. (strftime() metodu ile)

%p : 12 saatlik sistemde saatin sayısal değeri için iki karakterli ingilizce AM (öğleden önce) ve PM (öğleden sonra) takısı değerini döndürür. (Yerel) (strftime() metodu ile)

%r : 12 saatlik sistemde ikişer haneli olacak şekilde saat, dakika ve saniyenin AM (öğleden önce) ve PM (öğleden sonra) takılı sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (Yerel) (strftime() metodu ile)

%R : Pozitif ve negatif imlecini çıkartmak için kullanılır. (date_diff() metodu ile)

%R : Saat ve dakika sıralamasını esas alarak saat değerini döndürür. (Yerel) (strftime() metodu ile)

%s : Saniye farkı değerini döndürür. (date_diff() metodu ile)

%S : Saniyenin iki haneli olacak şeklinde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (Yerel) (strftime() metodu ile)

%t : Tab karakteri değerini döndürür. (strftime() metodu ile)

%T : Tam saat değerini döndürür. (Yerel) (strftime() metodu ile)

%u : ISO-8601 standartlarına uygun olacak şekilde haftanın gününün sayısal değerini döndürür. 1'dan (pazartesi) 7'ya (pazar) kadar. (Yerel) (strftime() metodu ile)

%U : Yılın haftasının sayısal değerini döndürür. 0'dan (1. hafta) 51'e (52. hafta) kadar. (Hafta başlangıcı pazar) (Yerel) (strftime() metodu ile)

%V : ISO-8601 standartlarına uygun olacak şekilde yılın haftasının sayısal değerini döndürür. (Yerel) (strftime() metodu ile)

%w : Haftanın gününün sayısal değerini döndürür. 0'dan (pazar) 6'ya (cumartesi) kadar. (Yerel) (strftime() metodu ile)

%W : Yılın haftasının sayısal değerini döndürür. 0'dan (1. hafta) 51'e (52. hafta) kadar. (Hafta başlangıcı pazartesi) (Yerel) (strftime() metodu ile)

%x : Ay, gün ve yıl sıralamasını esas alarak tarih değerini döndürür. (Yerel) (strftime() metodu ile)

%X : Tam saat değerini döndürür. (Yerel) (strftime() metodu ile)

%Y : ISO-8601 standartlarına uygun olacak şekilde yılın dört haneli olacak şeklinde sayısal değerini döndürür. (Yerel) (strftime() metodu ile)

%y : ISO-8601 standartlarına uygun olacak şekilde yılın iki haneli olacak şeklinde sayısal değerini döndürür. (Yerel) (strftime() metodu ile)

%y : Yıl farkı değerini döndürür. (date_diff() metodu ile)

%Z : Tüm tarih ve saat işlevleri tarafından kullanılan varsayılan geçerli zaman dilimi değerini döndürür. (strftime() metodu ile)

& : Bitsel ve operatörü

&& : Ve (MySQL'de)

&& : Ve operatörü

&= : Bitsel ve ataması operatörü

() : Düzenli ifadelerde grup tanımlamak için kullanılır. (RegExp ile)

() : Grup (.htaccess ile)

(?:) : Düzenli ifadeler içerisinde alt grup oluşturmak için kullanılır. (RegExp ile)

(array) : Dizi veri türüne dönüştürme.

(bool) : Mantıksal veri türüne dönüştürme.

(boolean) : Mantıksal veri türüne dönüştürme.

(double) : Ondalıklı sayı veri türüne dönüştürme.

(float) : Ondalıklı sayı veri türüne dönüştürme.

(int) : Tam sayı veri türüne dönüştürme.

(integer) : Tam sayı veri türüne dönüştürme.

(object) : Nesne veri türüne dönüştürme.

(real) : Ondalıklı sayı veri türüne dönüştürme.

(string) : Alfabetik (a-z A-Z 0-9) veri türüne dönüştürme.

(unset) : null veri türüne dönüştürme.

(x|y) : Karakter diziliminde bulunan değerin, belirtilen x yada z karakterlerinden herhangi birisi ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

***** : Çarp operatörü

***** : Karakter diziliminde bulunan değerin 0 defa yada daha fazla tekrarlanmış olması gerektiğini belirtmek için kullanılır. (.htaccess ile)

***** : Karakter diziliminde bulunan değerin 0 defa yada daha fazla tekrarlanmış olması gerektiğini belirtmek için kullanılır. (RegExp ile)

***=** : Çarpma ataması operatörü

, : Virgül / ekleme operatörü

. : Birleştirme operatörü

. : Herhangi bir karakter (.htaccess ile)

. : Karakter diziliminde bulunan değerin, yeni satır karakterleri hariç herhangi birisi ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

. : Numerik değerlerde ondalık basamak sayısını veya string değerlerde maksimum genişliğinin kaç karakter olacağını belirtir. (printf() veya sprintf() metodu ile)

.= : Birleştirme ataması operatörü

/ : Böl operatörü

/* ... */ : Gizli açıklama satır blokları tanımlamak için kullanılır.

// : Gizli açıklama satırı tanımlamak için kullanılır.

/= : Bölme ataması operatörü

? : Karakter diziliminde bulunan değerin 0 defa yada 1 defa tekrarlanmış olması gerektiğini belirtmek için kullanılır. (.htaccess ile)

? : Karakter diziliminde bulunan değerin 0 defa yada 1 defa tekrarlanmış olması gerektiğini belirtmek için kullanılır. (RegExp ile)

?! : Karakter dizilimi içerisindeki belirtilen bir referans değer ile takip edilmeyen değerleri düzenli ifade ile eşleşebilecek değerleri aramak için kullanılır. (RegExp ile)

?: : Ternary operatörü (Üçlü operatör)

?? : Null koşul ifadesi

?= : Karakter dizilimi içerisindeki belirtilen bir referans değerin önünde düzenli ifade ile eşleşebilecek değerleri aramak için kullanılır. (RegExp ile)

@ : Hata denetim operatörü

[] : Aranan (.htaccess ile)

[^abc] : Karakter diziliminde bulunan değerin, belirtilen karakterlerin dışındaki karakterlerden herhangi birisi ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

[^a-z] : Karakter diziliminde bulunan değerin, belirtilen karakterler aralığı dışındaki karakterlerden herhangi birisi ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

[abc] : Karakter diziliminde bulunan değerin, belirtilen karakterlerden herhangi birisi ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

[a-z] : Karakter diziliminde bulunan değerin, belirtilen karakterler aralığından herhangi birisi ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

**** : Düzenli ifadelerde ardından gelecek olan karakterin özel bir karakter olup olmadığını belirtmek için kullanılır. (RegExp ile)

**** : Kaçırma operatörü

\b : Düzenli ifadeye uygun olabilecek metin parçalarını, karakter diziliminin içerisindeki kelimelerin başında yada sonunda aramak için kullanılır. (RegExp ile)

\B : Düzenli ifadeye uygun olabilecek metin parçalarını, karakter diziliminin içerisindeki kelimelerin içerisinde aramak için kullanılır. (RegExp ile)

\D : Karakter diziliminde bulunan değerin, rakam karakterleri dışındaki karakterlerden herhangi birisi ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

\d : Karakter diziliminde bulunan değerin, sadece rakam karakterlerinden herhangi birisi ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

\S : Karakter diziliminde bulunan değerin, boşluk karakteri dışındaki karakterlerden herhangi birisi ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

\s : Karakter diziliminde bulunan değerin, sadece boşluk karakteri ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

\W : Karakter diziliminde bulunan değerin, harf ve rakamlardan veya _ karakterler dışındaki karakterlerden herhangi birisi ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

\w : Karakter diziliminde bulunan değerin, harf ve rakamlardan veya _ karakterlerinden herhangi birisi ile eşleşmesi gerektiğini belirtmek için kullanılır. (RegExp ile)

^ : Bitsel ayrıcalıklı veya operatörü

^ : Karakter diziliminin başlangıcı ile düzenli ifade arasında bir eşleşme aramak için kullanılır. (RegExp ile)

^ : Karakter diziliminin başlangıcı ile ifade arasında bir eşleşme aramak için kullanılır. (.htaccess ile)

^= : Bitsel ayrıcalıklı veya ataması operatörü

__CLASS__ : PHP dosyası içerisinde bulunduğu sınıfın, sınıf adı bilgisi değerini döndürür.

__construct() : Belirtilecek olan sınıfa, yapıcı metod tanımlamak için kullanılır.

__destruct() : Belirtilecek olan sınıfa, yıkıcı metod tanımlamak için kullanılır.

__DIR__ : Çalışmakta olan PHP dosyasının bulunduğu dizin bilgisi değerini döndürür. PHP dosyası adını içermez.

__FILE__ : Çalışmakta olan PHP dosyasının bulunduğu dizin ve dosya adı bilgisi değerini döndürür.

__FUNCTION__ : PHP dosyası içerisinde bulunduğu fonksiyonun, fonksiyon adı bilgisi değerini döndürür.

__LINE__ : PHP dosyası içerisinde bulunduğu satırın, satır numarası bilgisi değerini döndürür.

__METHOD__ : PHP dosyası içerisinde bulunduğu sınıf dahilinde kullanılacak olan metodun, metod adı bilgisi değerini döndürür.

__NAMESPACE__ : PHP dosyası içerisinde tanımlanmış olan namespace'ın, namespace adı bilgisi değerini döndürür.

__TRAIT__ : PHP dosyası içerisinde bulunduğu sınıf dahilinde kullanılacak olan özelliğin, özellik adı bilgisi değerini döndürür.

` : Çalıştırma operatörü

{ } : Özel tanım (.htaccess ile)

{x,} : Karakter diziliminde bulunan değerin x defa yada daha fazla tekrarlanmış olması gerektiğini belirtmek için kullanılır. (RegExp ile)

{x,y} : Karakter diziliminde bulunan değerin en az x defa yada en fazla y defa tekrarlanmış olması gerektiğini belirtmek için kullanılır. (RegExp ile)

{x} : Karakter diziliminde bulunan değerin x defa tekrarlanmış olması gerektiğini belirtmek için kullanılır. (RegExp ile)

| : Bitsel veya operatörü

| : Veya (.htaccess ile)

|| : Veya (MySQL'de)

|| : Veya operatörü

|= : Bitsel veya ataması operatörü

~ : Bitsel değil operatörü

+ : Birleştir operatörü

+ : Karakter diziliminde bulunan değerin 1 defa yada daha fazla tekrarlanmış olması gerektiğini belirtmek için kullanılır. (.htaccess ile)

+ : Karakter diziliminde bulunan değerin 1 defa yada daha fazla tekrarlanmış olması gerektiğini belirtmek için kullanılır. (RegExp ile)

+ : Numerik değerlerin önüne + ve – olmak üzere pozitiflik ve negatiflik işareti ekler. Belirtilmezse – işareti negatif sayıların önünde görüntülenirken, + işareti pozitif sayıların önünde görüntülenmez. (printf() veya sprintf() metodu ile)

+ : Pozitif ol operatörü

+ : Topla operatörü

++ : Arttır operatörü

+= : Toplama ataması operatörü

< : Küçük (MySQL'de)

< : Küçük ise operatörü

<< : Bitsel sola kaydırma operatörü

<<= : Bitset sola kaydırma ataması operatörü

<= : Küçük veya eşit (MySQL'de)

<= : Küçük veya eşit ise operatörü

<> : Eşit değil ise operatörü

-= : Çıkartma ataması operatörü

= : Eşit (MySQL'de)

= : Eşittir operatörü

== : Eşit ise operatörü

=== : Denk ise operatörü

> : Büyük (MySQL'de)

> : Büyük ise operatörü

>= : Büyük veya eşit (MySQL'de)

>= : Büyük veya eşit ise operatörü

>> : Bitset sağa kaydırma operatörü

>>= : Bitset sağa kaydırma ataması operatörü

0 : Default (curl_setopt() veya curl_setopt_array() ve CURLOPT_SSLVERSION ifadesi metodu ile)

0 : Ortak bir isim özelliği değeri aranmaz. (curl_setopt() veya curl_setopt_array() ve CURLOPT_SSL_VERIFYHOST ifadesi metodu ile)

0 : Seviye hataları kapatması. (error_reporting() metodu ile)

0, 1, 2, 3, ... : Genişlik belirticisidir. Değerin kaç karakter olacağını belirler. (printf() veya sprintf() metodu ile)

1 : Africa (timezone_identifiers_list() metodu ile)

1 : SSL eş sertifikasında ortak bir ismin varlığına bakılır. (curl_setopt() veya curl_setopt_array() ve CURLOPT_SSL_VERIFYHOST ifadesi metodu ile)

1 : TLSv1 (curl_setopt() veya curl_setopt_array() ve CURLOPT_SSLVERSION ifadesi metodu ile)

128 : Europe (timezone_identifiers_list() metodu ile)

16 : Asia (timezone_identifiers_list() metodu ile)

2 : America (timezone_identifiers_list() metodu ile)

2 : SSL eş sertifikasında ortak bir ismin varlığı ile birlikte server'ın ana bilgisayar adının da eşleşmesine bakılır. (curl_setopt() veya curl_setopt_array() ve CURLOPT_SSL_VERIFYHOST ifadesi metodu ile)

2 : SSLv2 (curl_setopt() veya curl_setopt_array() ve CURLOPT_SSLVERSION ifadesi metodu ile)

2047 : All (timezone_identifiers_list() metodu ile)

256 : Indian (timezone_identifiers_list() metodu ile)

3 : SSLv3 (curl_setopt() veya curl_setopt_array() ve CURLOPT_SSLVERSION ifadesi metodu ile)

32 : Atlantic (timezone_identifiers_list() metodu ile)

4 : Antarctica (timezone_identifiers_list() metodu ile)

4 : TLSv1_0 (curl_setopt() veya curl_setopt_array() ve CURLOPT_SSLVERSION ifadesi metodu ile)

5 : TLSv1_1 (curl_setopt() veya curl_setopt_array() ve CURLOPT_SSLVERSION ifadesi metodu ile)

512 : Pacific (timezone_identifiers_list() metodu ile)

6 : TLSv1_2 (curl_setopt() veya curl_setopt_array() ve CURLOPT_SSLVERSION ifadesi metodu ile)

64 : Australia (timezone_identifiers_list() metodu ile)

8 : Arctic (timezone_identifiers_list() metodu ile)

a : 12 saatlik sistemde saatin sayısal değeri için iki karakterli ingilizce am (öğleden önce) ve pm (öğleden sonra) takısı değerini döndürür. (date() metodu ile)

a : Yazma amaçlı açar. Dosya verisine dokunmadan sonuna ek yaparak yazar. Dosya mevcut değilse oluşturulur. (fopen() metodu ile)

a+ : Hem okuma hemde yazma amaçlı açar. Dosya verisine dokunmadan sonuna ek yaparak yazar. Dosya mevcut değilse oluşturulur. (fopen() metodu ile)

abs() : Herhangi bir sayının mutlak değerini bularak, bulduğu değeri geriye döndürür.

abstract : Belirtilecek olan herhangi bir sınıfı, soyut sınıf haline dönüştürmek için kullanılır.

acos() : -1 ila 1 arasındaki herhangi bir sayının ark kosinüsünü radyan değeri cinsinden bularak, bulduğu değeri geriye döndürür.

acosh() : Herhangi bir sayının hiperbolik ark kosinüsünü bularak, bulduğu değeri geriye döndürür.

ADD / ADD COLUMN : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya yeni bir sütun eklemek için kullanılır. (ALTER TABLE ifadesi ile)

addAttribute() : Belirtilecek olan ve SimpleXMLElement() metodu kullanılarak oluşturulmuş olan XML kodlama yapılarına ait elemana / elemente özellik eklemek için kullanılır.

addChild() : Belirtilecek olan ve SimpleXMLElement() metodu kullanılarak oluşturulmuş olan XML kodlama yapılarına yeni eleman / element eklemek için kullanılır.

addslashes() : Belirtilecek olan içerikte bulunan, belirtilecek olan özel karakterlerin önüne \ yani kaçırma operatörü ekler ve sade bir metin içeriği oluşturarak, oluşturduğu değeri geriye döndürür.

addslashes() : Belirtilecek olan içerikte bulunan ' ve " işaretlerinin önüne \ yani kaçırma operatörü ekler ve sade bir metin içeriği oluşturarak, oluşturduğu değeri geriye döndürür.

affected_rows : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya nesnesel yapı ile bağlanılarak işlem yapıldığında etkilenen kayıt sayısını bulmak için kullanılır.

AFTER : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya eklenecek olan yeni sütunun, belirtilecek olan sütundan bir sonra eklenmesi gerektiğini belirtmek için kullanılır. (ALTER TABLE ifadesi ile)

ALTER TABLE : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya yeni bir sütun ekleneceğini, herhangi bir sütunun silineceğini veya herhangi bir sütun adının yada yapısının değiştirileceğini belirtmek için kullanılır.

ANALYZE : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun çözümlenmesi / analiz edilmesi için kullanılır.

AND : Ve (MySQL'de)

and : Ve operatörü

array() : Dizi oluşturmak için kullanılır.

array_change_key_case() : Dizi içerisinde bulunan anahtar veya anahtarların büyük harf küçük harf dönüştürme işlemleri için kullanılır.

array_chunk() : Dizin belli bölümlerini alıp parçalayarak yeni bir çok boyutlu dizi oluşturmak için kullanılır.

array_column() : Çok boyutlu bir dizi içerisinde bulunan elemanı / elemanları belirtilecek olan anahtar koşullarına göre biçimlendirerek yeni bir dizi oluşturmak için kullanılır.

array_combine() : İki farklı diziyi ilişkisel olarak birleştirerek yeni bir dizi oluşturmak için kullanılır.

array_count_values() : Dizi içerisinde bulunan eleman veya elemanların kaçar defa tekrarlandığını bulmak için kullanılır.

array_diff() : Birden fazla dizi içerisinde bulunan ve farklı eleman değerlerine sahip elemanlardan yeni bir dizi oluşturmak için kullanılır.

array_diff_assoc() : Birden fazla dizi içerisinde bulunan ve farklı anahtar isimlerine ve farklı eleman değerlerine sahip elemanlardan yeni bir dizi oluşturmak için kullanılır.

array_diff_key() : Birden fazla dizi içerisinde bulunan ve farklı anahtar isimlerine sahip elemanlardan yeni bir dizi oluşturmak için kullanılır.

array_fill() : Belirtilecek olan değerler doğrultusunda doldurma işlemi yaparak yeni bir dizi oluşturmak için kullanılır.

array_fill_keys() : Dizi içerisinde bulunan eleman veya elemanları anahtar gibi görerek, belirtilecek olan değerler doğrultusunda doldurma işlemi yaparak yeni bir dizi oluşturmak için kullanılır.

array_filter() : Dizin içerisinde bulunan boş eleman veya elemanları filtre ederek yeni bir dizi oluşturmak için kullanılır.

array_flip() : Dizi içerisinde bulunan anahtarların ve elemanlarının yer değişimi yapması için kullanılır.

array_intersect() : Birden fazla dizi içerisinde bulunan ve aynı eleman değerlerine sahip elemanlardan yeni bi dizi oluşturmak için kullanılır.

array_intersect_assoc() : Birden fazla dizi içerisinde bulunan ve aynı anahtar isimlerine ve aynı eleman değerlerine sahip elemanlardan yeni bi dizi oluşturmak için kullanılır.

array_intersect_key() : Birden fazla dizi içerisinde bulunan ve aynı anahtar isimlerine sahip elemanlardan yeni bi dizi oluşturmak için kullanılır.

array_key_exists() : Dizi içerisinde bulunan anahtar veya anahtarlar dahilinde, belirtilecek olan anahtarın olup olmadığını kontrol etmek için kullanılır.

array_keys() : Dizin anahtarlarını alıp yeni bir dizi oluşturmak için kullanılır.

array_merge() : Birden fazla diziyi birleştirerek yeni bir dizi oluşturmak için kullanılır.

array_merge_recursive() : Birden fazla diziyi birleştirerek yeni bir dizi oluşturmak için kullanılır.

array_multisort() : Bir veya birden fazla dizinin elemanlarını gelişmiş olarak çok yönlü sıralamak için kullanılır.

array_pad() : Dizin eleman veya elemanlar sayısını baz alarak, belirtilecek olan değerler doğrultusunda baştan veya sondan doldurma işlemi yaparak yeni bir dizi oluşturmak için kullanılır.

array_pop() : Dizi içerisinde bulunan son elemanı silmek için kullanılır. Aynı zamanda dizi içeriğinden sildiği elemanın değerini geriye döndürür.

array_product() : Dizi içerisinde bulunan eleman veya elemanların rakamsal değerler içermesi durumunda tüm elemanların çarpımını almak için kullanılır.

array_push() : Dizin en sonuna yeni eleman veya elemanlar eklemek için kullanılır. Aynı zamanda eklenecek olan eleman veya elemanların diziye dahil edilmesi ile, dizi içeriğinin toplam eleman sayısı değerini geriye döndürür.

array_rand() : Dizi içerisinde bulunan elemanlardan, belirtilecek eleman sayısı kadar rastgele değer döndürmek için kullanılır.

array_replace() : Birden fazla dizi içerisinde bulunan eleman veya elemanların, ilk dizi içerisinde olanları baz alınarak, kendisinden sonra gelecek olan dizi veya diziler içerisinde bulunan eleman veya elemanlarla güncelleyerek yeni bir dizi oluşturmak için kullanılır.

array_replace_recursive() : Birden fazla dizi içerisinde bulunan eleman veya elemanların, ilk dizi içerisinde olanları baz alınarak, kendisinden sonra gelecek olan dizi veya diziler içerisinde bulunan eleman veya elemanlarla güncelleyerek yeni bir dizi oluşturmak için kullanılır.

array_reverse() : Dizi içerisinde bulunan eleman veya elemanların tersten sıralanmasını sağlayarak yeni bir dizi oluşturmak için kullanılır.

array_search() : Dizi içerisinde bulunan eleman veya elemanlar dahilinde, belirtilecek olan elemanı aramak ve ilgili elemanın anahtarını bulmak için kullanılır.

array_shift() : Dizi içerisinde bulunan ilk elemanı silmek için kullanılır. Aynı zamanda dizi içeriğinden sildiği elemanın değerini geriye döndürür.

array_slice() : Dizin belli bölümlerini alarak yeni bir dizi oluşturmak için kullanılır.

array_splice() : Dizin belli bölümlerini alıp, belli bölümlerini silip, eğer belirtilirse yeni elemanlarda ekleyerek yeni bir dizi oluşturmak için kullanılır.

array_sum() : Dizi içerisinde bulunan eleman veya elemanların rakamsal değerler içermesi durumunda tüm elemanların toplamını almak için kullanılır.

array_unique() : Dizin içerisinde bulunan ve tekrarlanan elemanların fazla olanlarını silerek yeni bir dizi oluşturmak için kullanılır.

array_unshift() : Dizin en başına yeni eleman veya elemanlar eklemek için kullanılır. Aynı zamanda eklenecek olan eleman veya elemanların diziye dahil edilmesi ile, dizi içeriğinin toplam eleman sayısı değerini geriye döndürür.

array_values() : Dizin elemanlarını alıp yeni bir dizi oluşturmak için kullanılır.

arsort() : Dizin elemanlarını büyük harf küçük harf duyarlı olacak ve anahtarlarını bozmayacak şekilde z'den a'ya yada büyükten küçüğe sıralamak için kullanılır.

AS : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun, belirtilecek olan herhangi bir sütununa alias atamak için kullanılır.

as : Sınıf için tanımlanmış olan özellik ve metod gruplarında, metoda takma isim belirleme veya görünürlük atama işlemleri için kullanılır.

asin() : -1 ila 1 arasındaki herhangi bir sayının ark sinüsünü radyan değeri cinsinden bularak, bulunduğu değeri geriye döndürür.

asinh() : Herhangi bir sayının hiperbolik ark sinüsünü bularak, bulunduğu değeri geriye döndürür.

asort() : Dizin elemanlarını büyük harf küçük harf duyarlı olacak ve anahtarlarını bozmayacak şekilde a'dan z'ye yada küçükten büyüğe sıralamak için kullanılır.

astronomical_twilight_begin : Astronomik alacakaranlık başlangıcı (date_sun_info() metodu ile)

astronomical_twilight_end : Astronomik alacakaranlık bitişi (date_sun_info() metodu ile)

asXML() : Belirtilecek olan ve SimpleXMLElement() metodu kullanılarak oluşturulmuş olan XML kodlama yapılarının çıktılanmasını sağlamak için kullanılır.

atan() : Herhangi bir sayının ark tanjantını radyan değeri cinsinden bularak, bulunduğu değeri geriye döndürür.

atan2() : Herhangi iki sayının ark tanjantını radyan değeri cinsinden bularak, bulunduğu değeri geriye döndürür. (x, y)

atanh() : -1 ila 1 arasındaki herhangi bir sayının hiperbolik ark tanjantını bularak, bulunduğu değeri geriye döndürür.

autocommit() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya nesnel yapı ile işlem yapılacağı anda otomatik işleyen geçici işlem yapılarını açmak veya kapatmak için kullanılır.

AVG : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun belirtilecek olan sütununun tüm kayıt değerlerinin ortalamasını bulmak için kullanılır.

b : Blob (bind_param()) metodu ile)

b : Değeri tam sayı şeklinde ele alarak, değerin binary sayı sisteminde görüntülenmesini sağlar. (printf() veya sprintf() metodu ile)

base_convert() : Herhangi bir sayının, binary, octal, decimal, hexadecimal sayı sistemleri karşılığını bularak, bulunduğu değeri geriye döndürür.

base64_decode() : Belirtilecek olan ve base64 algoritması kullanılarak kodlanmış olan içeriği geri çözerek, çözdüğü değeri geriye döndürür.

base64_encode() : Belirtilecek olan içeriğin base64 algoritması kodlamasını bularak, bulunduğu değeri geriye döndürür.

basename() : Belirtilecek olan değer doğrultusunda, kaynağın sadece dosya adını bularak, bulunduğu değeri geriye döndürür.

beginTransaction() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya işlem yapılacağı anda otomatik işleyen geçici işlem yapılarını açmak veya kapatmak için kullanılır.

BETWEEN : Aralığında (MySQL'de)

bind_param() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya yapılacak işleme göre prepare() metodu kullanılarak hazırlanmış olan query'yi derlemek (parametre veti türleri ve verileri) için kullanılır.

bind_result() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun veri okuma işlemi için prepare() metodu kullanılarak hazırlanmış ve execute() metodu kullanılarak çalıştırılmış olan query'nin sonuçlarını almak için kullanılır.

bindec() : Herhangi bir binary sayının, decimal sayı sistemi karşılığını bularak, bulunduğu değeri geriye döndürür.

bindParam() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya yapılacak işleme göre prepare() metodu kullanılarak hazırlanmış olan query'yi derlemek (parametre veti türleri ve verileri) için kullanılır.

bindValue() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya yapılacak işleme göre prepare() metodu kullanılarak hazırlanmış olan query'yi derlemek (parametre veti türleri ve verileri) için kullanılır. (bindParam metodunun benzeri)

boolval() : Herhangi bir değişken içeriğinin boolean veri türü değerini geriye döndürür.

break : Döngünün çalışmasını sonlandırmak için kullanılır.

break : Koşul kontrolünü sonlandır.

c : Değeri tam sayı ve ASCII karakter seti türü gibi ele alarak, ilgili değerin karşılığı olan karakterin görüntülenmesini sağlar. (printf() veya sprintf() metodu ile)

c : ISO-8601 standartlarına uygun olacak şekilde tarih ve saat değerini döndürür. (date()) metodu ile)

case : Koşul kontrol et.

CASE_LOWER : Küçük harf yapar. (array_change_key_case()) metodu ile)

CASE_UPPER : Büyük harf yapar. (array_change_key_case() metodu ile)

catch : Hata oluşursa çalıştırılacak kodlar için kullanılır.

ceil() : Herhangi bir ondalıklı sayıyı yukarıya yuvarlayarak, yuvarlama sonucunda oluşan değeri geriye döndürür.

CHANGE / CHANGE COLUMN : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablodaki herhangi bir sütunun adını yada yapısını değiştirmek için kullanılır. (ALTER TABLE ifadesi ile)

chdir() : Belirtilecek olan dizin bilgisine göre geçerli çalışma dizinini değiştirmek için kullanılır.

CHECK : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun kontrol edilmesi için kullanılır.

checkdate() : Belirtilecek olan zamani Gregoryen takvimine göre doğrulamasını yaparak, doğrulama sonucunu boolean veri türünde döndürmek için kullanılır.

children() : Belirtilecek olan XML kodlama değeri içerisinde bulunan, belirtilecek olan tag adı değerine bağlı olan bir alt tüm çocuk elemanları / elementleri işaret etmek ve seçmek için kullanılır.

chmod() : Belirtilecek olan değer ve izin doğrultusunda, ilgili kaynağın dosya / dizin erişim izinlerini ayarlamak için kullanılır. Ayrıca istenirse işlem sonucunu boolean veri türünde döndürmek için kullanılır.

chop() : Belirtilecek olan içeriğin, varsa sonundaki / sağındaki tüm boşlukları, sekmeleri ve yeni satır karakterlerini silerek, oluşturduğu değeri geriye döndürür. Ayrıca belirtilecek olan içeriğin, varsa sonundaki / sağındaki, belirtilecek olan karakterleride silerek, oluşturduğu değeri geriye döndürür.

chr() : Belirtilecek olan ASCII değerinin standart karakter karşılığını bularak, bulduğu değeri geriye döndürür.

civil_twilight_begin : Kara alacakaranlık başlangıcı (date_sun_info() metodu ile)

civil_twilight_end : Kara alacakaranlık bitişi (date_sun_info() metodu ile)

class : Herhangi bir programlamada belirli bir veya daha fazla işi gerçekleştirmek üzere önceden hazırlanan nesnelere verilen isimdir.

class : Sınıf tanımlamak için kullanılır.

class_exists() : Belirtilecek olan class'ın varlığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

clearstatcache() : stat() veya fstat() metodu kullanılarak elde edilen ve ön belleğe alınan verileri temizlemek için kullanılır.

close() : Daha önceden nesnesel yapı ile açılmış olan MySQL bağlantısını kapatmak / sonlandırmak için kullanılır.

close() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablo ile alakalı nesnesel yapı ile istenilen işlem / işlemler bittiğinde hafızayı boşaltmak için kullanılır.

closedir() : opendir() metodu ile açılmış ve belleğe alınmış olan dizini kapatarak, belleği boşaltır.

commit() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya işlem yapılacağı anda daha önceden beginTransaction() metodu kullanılarak açılmış olan otomatik işleyen geçici işlem yapıları dahilinde işleme alınmış olan query'nin geçicilikten çıkarak kalıcı hale döndürülmesi için kullanılır.

commit() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya nesnesel yapı ile işlem yapılacağı anda daha önceden autocommit() metodu kullanılarak açılmış olan otomatik işleyen geçici işlem yapıları dahilinde işleme alınmış olan query'nin geçicilikten çıkarak kalıcı hale döndürülmesi için kullanılır.

compact() : Dizi içerisinde bulunan elamanı / elemanları anahtar gibi görerek, daha önceden aynı isimle tanımlanmış olan değişkenlerden ilgili anahtarlara değer üretmek için kullanılır.

connect_errno : MySQL sunucusuna yeni bir bağlantı açma işlemi sırasında olası bir hata ile karşılaşması durumunda hata kodu değerini bularak, bulunduğu değeri geriye döndürür.

connect_error : MySQL sunucusuna yeni bir bağlantı açma işlemi sırasında olası bir hata ile karşılaşması durumunda hata açıklama değerini bularak, bulunduğu değeri geriye döndürür.

const : Sabit oluşturmak için kullanılır.

continue : Döngünün belli bir kısmının atlanarak, çalışmakta olan döngünün bir sonraki işleme devam etmesi için kullanılır.

convert_uuencode() : Belirtilecek olan ve uuencode algoritması kullanılarak kodlanmış olan içeriği geri çözerek, çözdüğü değeri geriye döndürür.

convert_uuencode() : Belirtilecek olan içeriği uuencode algoritması kodlamasını bularak, bulunduğu değeri geriye döndürür.

copy() : Belirtilecek olan değer doğrultusunda, dosya kopyalamak için kullanılır. Ayrıca istenirse işlem sonucunu boolean veri türünde döndürmek içinde kullanılır.

cos() : Herhangi bir sayının kosinüsünü bularak, bulunduğu değeri geriye döndürür.

cosh() : Herhangi bir sayının hiperbolik kosinüsünü bularak, bulunduğu değeri geriye döndürür.

COUNT : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun belirtilecek olan sütununun toplam kayıt sayısını bulmak için kullanılır.

count() : Dizi içerisindeki eleman sayısını bulmak için kullanılır.

COUNT_RECURSIVE : Çok boyutlu dizilerde, tüm boyutlar içerisindeki elemanlarda sayma işlemine dahil edilir. (count() veya sizeof() metodu ile)

crc32() : Belirtilecek olan içeriğin crc32 temsilini hesaplayıp bularak, bulunduğu değeri geriye döndürür.

CREATE DATABASE : MySQL sunucusuna yeni bir database ekleme / oluşturma için kullanılır.

CREATE TABLE : MySQL sunucusundaki database içerisine yeni bir tablo ekleme / oluşturma için kullanılır. Ayrıca istenirse herhangi bir tablo içeriklerini başka bir tabloya kopyalamak için de kullanılabilir.

crypt() : Belirtilecek olan içeriğin tek yönlü gelişmiş bir şekilde şifrelemesini yaparak, şifrelediği değeri geriye döndürür.

curl_close() : Kendisine parametre olarak verilen değer doğrultusunda başlatılmış olan bir CURL oturumunu sonlandırmak / kapatmak için kullanılır.

curl_error() : Kendisine parametre olarak verilen değer doğrultusunda başlatılmış olan bir CURL oturumunun olası bir hata ile karşılaşması durumunda hata değerini bularak, bulunduğu değeri geriye döndürür.

curl_exec() : Kendisine parametre olarak verilen değer doğrultusunda başlatılmış olan bir CURL oturumunu çalıştırmak için kullanılır.

curl_getinfo() : Kendisine parametre olarak verilen değer doğrultusunda başlatılmış olan bir CURL oturumunun tüm bağlantı bilgilerini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

curl_init() : Yeni bir CURL oturumu başlatmak / tanımlamak için kullanılır.

curl_setopt() : Kendisine parametre olarak verilen değer doğrultusunda başlatılmış bir CURL oturumuna seçenek eklemek için kullanılır.

curl_setopt_array() : Kendisine parametre olarak verilen değer doğrultusunda başlatılmış bir CURL oturumuna dizi halinde seçenek / seçenekler eklemek için kullanılır.

curl_version() : Kullanılmakta olan server'da çalışmakta olan libcurl kütüphanesinin tüm bilgilerini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

CURLFile() : Başlatılmış olan bir CURL oturumunda istenilen URL adresine HTTP POST işlemi tanımlandıktan sonra, CURLOPT_POSTFIELDS dahilinde karşı server'a dosya yüklemek için kullanılır.

CURLOPT_COOKIE : Başlatılmış olan bir CURL oturumunda istenilen URL adresine HTTP isteğinde kullanılacak olan yeni cookie / cookie'ler oluşturmak için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_COOKIEFILE : Başlatılmış olan bir CURL oturumunda istenilen URL adresinde kullanılacak olan cookie'lerin tutulduğu dosyayı belirtmek ve içeriğindeki değerleri karşı server'a iletmek için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_COOKIEJAR : Başlatılmış olan bir CURL oturumu sonlandırıldığında / kapatıldığında daha önceden oluşmuş olan cookie'lerin kaydedileceği dosyayı tanımlamak için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_FILE : Başlatılmış olan bir CURL oturumunda istenilen URL adresinden elde edilen içeriğin aktarılacağı dosyayı tanımlamak için kullanılır.

CURLOPT_FOLLOWLOCATION : Başlatılmış bir CURL oturumunda istenilen URL adresine ait server'ın HTTP başlığının bir parçası olarak gönderdiği Location başlığını izleyebilmek için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_HEADER : Başlatılmış olan bir CURL oturumunda istenilen URL adresinden elde edilen içerik ile birlikte HTTP üst bilgisinde bulunması / alınması gerektiğini belirtmek için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_HTTPHEADER : Başlatılmış olan bir CURL oturumunda istenilen URL adresine HTTP üst bilgisi göndermek / tanımlamak için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_NOBODY : Başlatılmış olan bir CURL oturumunda istenilen URL adresinden elde edilen içeriğin body alanının alınmasını veya çıktılanmasını durdurmak için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_POST : Başlatılmış olan bir CURL oturumunda istenilen URL adresine HTTP POST işlemi tanımlamak için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_POSTFIELDS : Başlatılmış olan bir CURL oturumunda istenilen URL adresine HTTP POST işleminde gönderilecek olan veri / verileri tanımlamak için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_REFERER : Başlatılmış olan bir CURL oturumunda istenilen URL adresine HTTP isteğinde kullanılacak olan referans / gönderen site URL adresini belirtmek için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_RETURNTRANSFER : Başlatılmış bir CURL oturumunda istenilen URL'den elde edilen içeriğin doğrudan ekrana çıktılanmasını durdurarak belirtilecek olan değişkene aktarılmasını sağlamak için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_SSL_VERIFYHOST : Başlatılmış olan bir CURL oturumunda istenilen URL ile veri alışverişi sırasında SSL eş sertifikasını tanımlamak için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_SSL_VERIFYPEER : Başlatılmış olan bir CURL oturumunda istenilen URL ile veri alışverişi sırasında SSL eş sertifikasını doğrulamak veya doğrulamayı iptal etmek için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_SSLVERSION : Başlatılmış olan bir CURL oturumunda istenilen URL ile veri alışverişi sırasında SSL eş sertifikasının sürümünü belirtmek için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_TIMEOUT : Başlatılmış bir CURL oturumunda maksimum çalıştırılabilirlik süresini saniye cinsinden belirtmek için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_TIMEOUT_MS : Başlatılmış bir CURL oturumunda maksimum çalıştırılabilirlik süresini milisaniye cinsinden belirtmek için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_URL : Başlatılmış bir CURL oturumunda içeriğine ulaşılacak istenilen URL adresini belirtmek için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

CURLOPT_USERAGENT : Başlatılmış olan bir CURL oturumunda istenilen URL adresine HTTP isteğinde kullanılacak olan web browser'ının tanımlanmasını ve ilgili browser'dan erişim sağlandığını belirtmek için kullanılır. (curl_setopt() veya curl_setopt_array() metodu ile)

current() : Dizinin göstericisi konumundaki elemanı bulmak için kullanılır.

d : Değeri tam sayı şeklinde ele alarak, değerin işaretli decimal sayı sisteminde görüntülenmesini sağlar. (printf() veya sprintf() metodu ile)

d : Double (bind_param()) metodu ile)

d : Günün iki haneli olacak şekilde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (date() metodu ile)

D : Günün üç haneli olacak şekilde adı değerini döndürür. (date() metodu ile)

date() : Tarih ve saat bilgilerini bularak, bulunduğu değeri geriye döndürür. Ayrıca tarih ve saat bilgilerini isteğe bağlı olarak biçimlendirmek içinde kullanılır.

date_add() : Oluşturulmuş olan yeni bir tarih nesnesine, belirtilecek olan tarih değerini eklemek için kullanılır.

date_create() : Belirtilecek olan tarih değerlerine göre, yeni bir tarih nesnesi dizisi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

date_create_from_format() : Belirtilecek olan formata göre biçimlendirilmiş tarih değerlerine göre, yeni bir tarih nesnesi dizisi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

date_date_set() : Oluşturulmuş olan yeni bir tarih nesnesine, belirtilecek olan yeni bir tarihi atamak için kullanılır.

date_default_timezone_get() : Tüm tarih ve saat işlevleri tarafından kullanılan varsayılan geçerli zaman dilimini bularak, bulunduğu değeri geriye döndürür.

date_default_timezone_set() : Tüm tarih ve saat işlevleri tarafından kullanılan varsayılan geçerli zaman dilimini belirtmek için kullanılır.

date_diff() : Oluşturulmuş olan yeni iki farklı tarih nesnesi arasındaki farkı hesaplayarak yeni bir tarih sonuç nesnesi dizisi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

date_format() : Oluşturulmuş olan yeni bir tarih nesnesini, belirtilecek olan bir formata göre biçimlendirerek, biçimlendirdiği değeri geriye döndürür.

date_interval_create_from_date_string() : date_add() metodu ile tarih değeri ekleme işlemlerinde, eklenecek olan değeri tanımlayabilmek için kullanılır.

date_modify() : Oluşturulmuş olan yeni bir tarih nesnesine, belirtilecek olan tarih değerini eklemek için kullanılır.

date_sun_info() : Belirtilecek olan tarih ve lokasyon değerlerine göre, ilgili yerin gün doğumu, gün batımı, öğlen, alacakaranlık başlangıç ve alacakaranlık bitiş değerlerinden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

date_sunrise() : Belirtilecek olan tarih ve lokasyon değerlerine göre, ilgili yerin gün doğumu değerini bularak, bulunduğu değeri geriye döndürür.

date_sunset() : Belirtilecek olan tarih ve lokasyon değerlerine göre, ilgili yerin gün batımı değerini bularak, bulunduğu değeri geriye döndürür.

date_time_set() : Oluşturulmuş olan yeni bir tarih nesnesine, belirtilecek olan yeni bir saati atamak için kullanılır.

date_timestamp_get() : Oluşturulmuş olan yeni bir tarih nesnesinin, Unix zaman damgası değerini bularak, bulunduğu değeri geriye döndürür.

date_timestamp_set() : Oluşturulmuş olan yeni bir tarih nesnesine, belirtilecek olan Unix zaman damgası değerini atar ve yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

decbin() : Herhangi bir decimal sayının, binary sayı sistemi karşılığını bularak, bulunduğu değeri geriye döndürür.

dechex() : Herhangi bir decimal sayının, hexadecimal sayı sistemi karşılığını bularak, bulunduğu değeri geriye döndürür.

declare(strict_types=1) : Kodlama dosyası içerisindeki tüm kodlamaların kural yapısını belirler.

decoct() : Herhangi bir decimal sayının, octal sayı sistemi karşılığını bularak, bulunduğu değeri geriye döndürür.

default : Koşul olumsuzlukları sonucu.

define() : Sabit oluşturmak için kullanılır.

defined() : Belirtilecek olan sabitin varlığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

deg2rad() : Derece değeri cinsinden herhangi bir sayının, radyan değeri cinsinden karşılığını bularak, bulunduğu değeri geriye döndürür.

Değişken : Girdiğimiz değerleri alan veya programın çalışması ile bazı değerlerin atandığı veri tutuculardır.

DELETE : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun herhangi bir kaydını içeren satırı / satırları silmek için kullanılır.

DISTINCT : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun verilerini okuma işlemi sırasında belirtilecek olan sütuna / sütunlara ait tekrar eden verilerin sadece ilkini işleme dahil ederek diğerlerinin işlem dışında bırakılmasını sağlamak için kullanılır.

die() : Kullanıldığı satırdan sonraki tüm kodlamaların çalışmasını iptal etmek / sonlandırmak için kullanılır. Ayrıca istenirse ek olarak belirtilecek olan değeri geriye döndürür.

dirname() : Belirtilecek olan değer doğrultusunda, kaynağın içerisinde bulunduğu dizin / dizinlerin adını bularak, bulunduğu değeri geriye döndürür.

disk_free_space() : Kullanılmakta olan server'ın, belirtilecek olan sürücüsü doğrultusunda kalan kullanılabilir disk kapasitesinin değerini byte cinsinden bularak, bulunduğu değeri geriye döndürür.

disk_total_space() : Kullanılmakta olan server'ın, belirtilecek olan sürücüsü doğrultusunda toplam disk kapasitesinin değerini byte cinsinden bularak, bulunduğu değeri geriye döndürür.

Dizi : Girdiğimiz sıralı değerleri alan veya programın çalışması ile bazı sıralı değerlerin atandığı veri tutuculardır.

do while : Döngü işlemi başlatır.

DOCUMENT_ROOT : Server'ın ayar dosyasında tanımlandığı gibi, erişilen geçerli PHP dosyasının çalıştığı dizini bilgisi değerini döndürür. İlgili değer Apache server'ı içerisinde "DocumentRoot" yönergesinde tanımlıdır. (\$_SERVER süper globali ile)

DOCUMENT_ROOT : Server'ın ayar dosyasında tanımlandığı gibi, erişilen geçerli PHP dosyasının çalıştığı root dizini bilgisi değerini döndürür. İlgili değer Apache server'ı içerisinde "DocumentRoot" yönergesinde tanımlıdır. (getenv() metodu ile)

DOMDocument : Kullanıldığı dokümanı bir DOMDocument nesne örneğine yükleyerek işleme hazır hale getirmek için kullanılır.

doubleval() : Herhangi bir değişken içeriğinin double veri türü değerini geriye döndürür.

DROP / DROP COLUMN : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablodaki herhangi bir sütunu silmek için kullanılır. (ALTER TABLE ifadesi ile)

DROP DATABASE : MySQL sunucusundaki içerisinde bulunan herhangi bir database'i silmek için kullanılır.

DROP TABLE : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloyu silmek için kullanılır.

e : Değeri bilimsel gösterim şeklinde ele alarak görüntülenmesini sağlar. (printf() veya sprintf() metodu ile)

e : Tüm tarih ve saat işlevleri tarafından kullanılan varsayılan geçerli zaman dilimi değerini döndürür. (date() metodu ile)

E_ALL : Tüm seviye hataları. (Diğer yazım şekli : -1) (error_reporting() metodu ile)

E_ERROR : Önemli çalışma zamanı hataları. (error_reporting() metodu ile)

E_NOTICE : Çalışma zamanı bildirimleri. (error_reporting() metodu ile)

E_PARSE : Derleme zamanı ayrıştırma hataları. (error_reporting() metodu ile)

E_WARNING : Önemli olmayan çalışma zamanı uyarıları. (error_reporting() metodu ile)

echo : Bir veya daha fazla içeriğin ekran çıktılanmasını sağlamak için kullanılır.

echo() : Bir veya daha fazla içeriğin ekran çıktılanmasını sağlamak için kullanılır.

else : Değilse.

elseif : Değilse eğer.

empty() : Belirtilecek olan değişkenin veya dizi içerisinde belirtilecek olan anahtarın daha önceden tanımlanmamış olduğunu sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

encoding : DOMDocument metodu ile oluşturulmuş ve loadXML() metodu ile nesne örneğine yüklenmiş olan içeriğin belge türü belirtiminde kullanılacak olan karakter kodlamasını belirtmek için kullanılır. (DOMDocument metodu ile)

end() : Dizi göstericisinin son konumundaki elemanı bulmak için kullanılır.

ENT_COMPAT : html_entity_decode metodu kullanılarak geri dönüştürülecek olan metin için ' işaretlerini dönüşümden çıkartmak / iptal etmek için kullanılır. (html_entity_decode() metodu ile)

ENT_COMPAT : htmlentities metodu kullanılarak dönüştürülecek olan metin için ek olarak " işaretlerini de dönüştürmek için kullanılır. (htmlentities() metodu ile)

ENT_COMPAT : htmlspecialchars metodu kullanılarak dönüştürülecek olan metin için ek olarak " işaretlerini de dönüştürmek için kullanılır. (htmlspecialchars() metodu ile)

ENT_COMPAT : htmlspecialchars_decode metodu kullanılarak geri dönüştürülecek olan metin için ' işaretlerini dönüşümden çıkartmak / iptal etmek için kullanılır. (htmlspecialchars_decode() metodu ile)

ENT_NOQUOTES : html_entity_decode metodu kullanılarak geri dönüştürülecek olan metin için ' ve " işaretlerini dönüşümden çıkartmak / iptal etmek için kullanılır. (html_entity_decode() metodu ile)

ENT_NOQUOTES : htmlentities metodu kullanılarak dönüştürülecek olan metin için ' ve " işaretlerini dönüşümden çıkartmak / iptal etmek için kullanılır. (htmlentities() metodu ile)

ENT_NOQUOTES : htmlspecialchars metodu kullanılarak dönüştürülecek olan metin için ' ve " işaretlerini dönüşümden çıkartmak / iptal etmek için kullanılır. (htmlspecialchars() metodu ile)

ENT_NOQUOTES : htmlspecialchars_decode metodu kullanılarak geri dönüştürülecek olan metin için ' ve " işaretlerini dönüşümden çıkartmak / iptal etmek için kullanılır. (htmlspecialchars_decode() metodu ile)

ENT_QUOTES : html_entity_decode metodu kullanılarak geri dönüştürülecek olan metin için ek olarak ' ve " işaretlerini de geri dönüştürmek için kullanılır. (html_entity_decode() metodu ile)

ENT_QUOTES : htmlentities metodu kullanılarak dönüştürülecek olan metin için ek olarak ' ve " işaretlerini de dönüştürmek için kullanılır. (htmlentities() metodu ile)

ENT_QUOTES : htmlspecialchars metodu kullanılarak dönüştürülecek olan metin için ek olarak ' ve " işaretlerini de dönüştürmek için kullanılır. (htmlspecialchars() metodu ile)

ENT_QUOTES : htmlspecialchars_decode metodu kullanılarak geri dönüştürülecek olan metin için ek olarak ' ve " işaretlerini de geri dönüştürmek için kullanılır. (htmlspecialchars_decode() metodu ile)

error_reporting() : Hata raporlama işlemleri dahilinde bulunan tüm hata seviyelerinin ayarlama işlemleri için kullanılır.

eval() : Belirtilecek olan içeriğin, PHP kodu olarak algılanmasını sağlar ve kodu çalıştırarak, oluşan değeri geriye döndürür.

Exception : Tüm istisnalar için önceden tanımlı olan temel sınıfı çağırmak için kullanılır.

exec() : MySQL sunucusuna PDO ile açılmış olan bağlantı dahilinde yeni bir sorgu işlemi yapmak için kullanılır. Ayrıca istenirse sorgu işleminde etkilenen kayıt sayısını da bulmak için kullanılır.

exec() : PHP dosyası dahilinde belirtilecek olan komutu çalıştırır ve işlem sonucunda bulduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

execute() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya yapılacak işleme göre prepare() metodu kullanılarak hazırlanmış olan query'yi çalıştırmak için kullanılır.

exit() : Kullanıldığı satırdan sonraki tüm kodlamaların çalışmasını iptal etmek / sonlandırmak için kullanılır. Ayrıca istenirse ek olarak belirtilecek olan değeri geriye döndürür.

exp() : Euler sayısının belirtilecek değer kadar üssünü bularak, bulduğu değeri geriye döndürür.

explode() : Düz bir metni belirtilecek olan ayrıca göre parçalayarak yeni bir dizi oluşturmak için kullanılır.

expm1() : Euler sayısının belirtilecek değer kadar üssünü bularak ve bir eksilterek, bulduğu değeri geriye döndürür.

extends : Bir sınıfı başka bir sınıftan türetmek istediğimiz zaman kullanılır.

extension_loaded() : Belirtilecek olan PHP eklentisinin yüklü olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

F : Ay adı değerini döndürür. (date() metodu ile)

f : Değeri gerçek sayı şeklinde ele alarak, değer yerele bağlı gerçek sayı olarak görüntülenmesini sağlar. (printf() veya sprintf() metodu ile)

F : Değeri gerçek sayı şeklinde ele alarak, değerin yerele bağlı olmaksızın gerçek sayı olarak görüntülenmesini sağlar. (printf() veya sprintf() metodu ile)

fclose() : fopen() metodu ile açılmış olan belirtilecek olan dosyayı veya URL'i kapatmak için kullanılır. Ayrıca istenirse işlem sonucunu boolean veri türünde döndürmek için kullanılır.

feof() : Belirtilecek olan dosya veya URL içeriğinin sonuna kadar okunup okunmadığını tesbit etmek için kullanılır.

fetch() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun tekil veri okuma işlemi için prepare() metodu kullanılarak hazırlanmış, execute() metodu kullanılarak çalıştırılmış ve sonuçları alınmış olan query'nin verilerini okumak için kullanılır. Ayrıca istenirse verilere erişim şeklide belirtilebilir.

fetch() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun tekil veri okuma işlemi için query() metodu kullanılarak hazırlanmış olan query'nin verilerini okumak için kullanılır. Ayrıca istenirse verilere erişim şeklide belirtilebilir.

fetch() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun veri okuma işlemi için prepare() metodu kullanılarak hazırlanmış, execute() metodu kullanılarak çalıştırılmış ve bind_result() metodu kullanılarak sonuçları alınmış olan query'nin verilerini okumak için kullanılır.

fetch_array() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun nesnesel yapı ile tüm verilerini düzenli bir dizi halinde okumak için kullanılır.

FETCH_ASSOC : Tablonun verilerini okuma işlemi sırasında oluşturulacak olan dizide ilgili tablonun sütun isimleri dizinin anahtarları olarak tanımlanır ve ilgili verilere sütun isimleri ile erişilir. (fetch() veya fetchAll() metodu ile)

FETCH_ASSOC : Tablonun verilerini okuma işlemi sırasında oluşturulacak olan dizide ilgili tablonun sütun isimleri dizinin anahtarları olarak tanımlanır ve ilgili verilere sütun isimleri ile erişilir. (query() metodu ile)

fetch_assoc() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun nesnesel yapı ile tüm verilerini ilişkisel bir dizi halinde okumak için kullanılır.

FETCH_BOTH : Tablonun verilerini okuma işlemi sırasında oluşturulacak olan dizide dizinin anahtarları hem ilgili tablonun sütun isimleri ile hemde PHP tarafından otomatik olarak tanımlanır ve ilgili verilere ister sütun isimleri ile ister sütun sıra numaraları ile erişilebilir. (query() metodu ile)

FETCH_BOTH : Tablonun verilerini okuma işlemi sırasında oluşturulacak olan dizide dizinin anahtarları hem ilgili tablonun sütun isimleri ile hemde PHP tarafından otomatik olarak tanımlanır ve ilgili verilere ister sütun isimleri ile ister sütun sıra numaraları ile erişilebilir. (fetch() veya fetchAll() metodu ile)

FETCH_NUM : Tablonun verilerini okuma işlemi sırasında oluşturulacak olan dizide dizinin anahtarları PHP tarafından otomatik olarak tanımlanır ve ilgili verilere sütun sıra numaraları ile erişilir. (fetch() veya fetchAll() metodu ile)

FETCH_NUM : Tablonun verilerini okuma işlemi sırasında oluşturulacak olan dizide dizinin anahtarları PHP tarafından otomatik olarak tanımlanır ve ilgili verilere sütun sıra numaraları ile erişilir. (query() metodu ile)

FETCH_OBJ : Tablonun verilerini okuma işlemi sırasında oluşturulacak olan nesnede ilgili tablonun sütun isimleri nesnenin özellikleri olarak tanımlanır ve ilgili verilere sütun isimleri ile erişilir. (fetch() veya fetchAll() metodu ile)

FETCH_OBJ : Tablonun verilerini okuma işlemi sırasında oluşturulacak olan nesnede ilgili tablonun sütun isimleri nesnenin özellikleri olarak tanımlanır ve ilgili verilere sütun isimleri ile erişilir. (query() metodu ile)

fetch_object() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun nesnesel yapı ile tüm verilerini nesne halinde okumak için kullanılır.

fetch_row() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun nesnesel yapı ile tüm verilerini numaralandırılmış bir dizi halinde okumak için kullanılır.

fetchAll() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun çoğul veri okuma işlemi için prepare() metodu kullanılarak hazırlanmış, execute() metodu kullanılarak çalıştırılmış ve sonuçları alınmış olan query'nin verilerini okumak için kullanılır. Ayrıca istenirse verilere erişim şeklide belirtilebilir.

fetchAll() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun çoğul veri okuma işlemi için query() metodu kullanılarak hazırlanmış olan query'nin verilerini okumak için kullanılır. Ayrıca istenirse verilere erişim şeklide belirtilebilir.

fgetc() : Belirtilecek olan dosya veya URL içeriğini kontrol ederek ve bilgileri karakter karakter bularak, bulunduğu değeri geriye döndürür. Ayrıca tüm içerikte bulunan karakter verileri herhangi bir döngü ile elde edilebilir.

fgetcsv() : Belirtilecek olan csv dosyası içeriğini kontrol ederek ve bilgileri satır satır bularak, bulunduğu değeri geriye döndürür. Ayrıca tüm dosya içeriğinde bulunan satır verileri herhangi bir döngü yardımıyla elde edilebilir.

fgets() : Belirtilecek olan dosya veya URL içeriğini kontrol ederek ve bilgileri satır satır bularak, bulunduğu değeri geriye döndürür. Ayrıca tüm içerikte bulunan satır verileri herhangi bir döngü ile elde edilebilir.

FILE_APPEND : Eski içerikler silinmeden yeni içerikler sona eklenir. (file_put_contents() metodu ile)

FILTER_FLAG_HOST_REQUIRED : Belirtilecek olan URL adreside host adının olup olmadığını sınamak için kullanılır. İşlem sonucu başarılı olursa değeri geriye döndürür. İşlem sonucu başarısız olursa herhangi bir değer döndürmez. (filter_var() metodu ile)

FILTER_FLAG_IPV4 : Belirtilecek olan IP adresinin IPV4 uyumlu olup olmadığını sınamak için kullanılır. İşlem sonucu başarılı olursa değeri geriye döndürür. İşlem sonucu başarısız olursa herhangi bir değer döndürmez. (filter_var() metodu ile)

FILTER_FLAG_IPV6 : Belirtilecek olan IP adresinin IPV6 uyumlu olup olmadığını sınamak için kullanılır. İşlem sonucu başarılı olursa değeri geriye döndürür. İşlem sonucu başarısız olursa herhangi bir değer döndürmez. (filter_var() metodu ile)

FILTER_FLAG_PATH_REQUIRED : Belirtilecek olan URL adresinin path değerinin olup olmadığını sınamak için kullanılır. İşlem sonucu başarılı olursa değeri geriye döndürür. İşlem sonucu başarısız olursa herhangi bir değer döndürmez. (filter_var() metodu ile)

FILTER_FLAG_QUERY_REQUIRED : Belirtilecek olan URL adresinin query değerinin olup olmadığını sınamak için kullanılır. İşlem sonucu başarılı olursa değeri geriye döndürür. İşlem sonucu başarısız olursa herhangi bir değer döndürmez. (filter_var() metodu ile)

FILTER_FLAG_SCHEME_REQUIRED : Belirtilecek olan URL adresinin RFC uyumlu olup olmadığını sınamak için kullanılır. İşlem sonucu başarılı olursa değeri geriye döndürür. İşlem sonucu başarısız olursa herhangi bir değer döndürmez. (filter_var() metodu ile)

FILTER_VALIDATE_BOOLEAN : Belirtilecek olan içeriğin boolean veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner. (filter_var() metodu ile)

FILTER_VALIDATE_EMAIL : Belirtilecek olan içeriğin E-Mail adresi türünde olup olmadığını sınamak için kullanılır. İşlem sonucu başarılı olursa değeri geriye döndürür. İşlem sonucu başarısız olursa herhangi bir değer döndürmez. (filter_var() metodu ile)

FILTER_VALIDATE_INT : Belirtilecek olan içeriğin integer veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner. (filter_var() metodu ile)

FILTER_VALIDATE_IP : Belirtilecek olan içeriğin IP adresi türünde olup olmadığını sınamak için kullanılır. İşlem sonucu başarılı olursa değeri geriye döndürür. İşlem sonucu başarısız olursa herhangi bir değer döndürmez. (filter_var() metodu ile)

FILTER_VALIDATE_URL : Belirtilecek olan içeriğin URL adresi türünde olup olmadığını sınamak için kullanılır. İşlem sonucu başarılı olursa değeri geriye döndürür. İşlem sonucu başarısız olursa herhangi bir değer döndürmez. (filter_var() metodu ile)

FIND_IN_SET : Ayrılmış halde içeren (MySQL'de)

FIRST : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya eklenecek olan yeni sütunun, diğer tüm sütunların en başına eklenmesi gerektiğini belirtmek için kullanılır. (ALTER TABLE ifadesi ile)

file() : Belirtilecek olan dosya veya URL içeriğini kontrol ederek ve bilgileri satır satır bularak, bulduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

file_exists() : Dosyanın belirtilecek olan path'e bağlı olarak varlığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

file_get_contents() : Belirtilecek olan dosya veya URL içeriğini kontrol ederek ve bilgileri tek bir defada bularak, bulduğu değeri geriye döndürür.

file_put_contents() : Belirtilecek olan dosyaya, belirtilecek olan verileri yazmak için kullanılır. Ayrıca istenirse işlem sonucunda yazılan karakter sayısı değerini bularak, bulduğu değeri geriye döndürür.

fileatime() : Belirtilecek olan değer doğrultusunda, ilgili kaynağın Unix zaman damgası türünde dosya son erişim tarihi değerini bularak, bulduğu değeri geriye döndürür.

filectime() : Belirtilecek olan değer doğrultusunda, ilgili kaynağın Unix zaman damgası türünde dosya oluşturulma tarihi değerini bularak, bulduğu değeri geriye döndürür.

filegroup() : Belirtilecek olan değer doğrultusunda, ilgili kaynağın dosya sahibi grubu değerini bularak, bulduğu değeri geriye döndürür.

fileinode() : Belirtilecek olan değer doğrultusunda, ilgili kaynağın dosya düğüm numarası değerini bularak, bulduğu değeri geriye döndürür.

filemtime() : Belirtilecek olan değer doğrultusunda, ilgili kaynağın Unix zaman damgası türünde dosya değiştirilme tarihi değerini bularak, bulduğu değeri geriye döndürür.

fileowner() : Belirtilecek olan değer doğrultusunda, ilgili kaynağın dosya sahibi değerini bularak, bulduğu değeri geriye döndürür.

fileperms() : Belirtilecek olan değer doğrultusunda, ilgili kaynağın dosya erişim izni değerini bularak, bulduğu değeri geriye döndürür.

filesize() : Belirtilecek olan değer doğrultusunda, ilgili kaynağın dosya boyutu değerini bularak, bulunduğu değeri geriye döndürür.

filetype() : Belirtilecek olan değer doğrultusunda, ilgili kaynağın dosya mı yada dizin mi olduğunu bularak, bulunduğu değeri geriye döndürür.

filter_var() : Kendisine parametre olarak verilen değer doğrultusunda içeriği filtreleyerek, içeriğin filtrelemeye uygun olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

final : En son kullanılan sınıftır ve türeyen son sınıf anlamını taşır. Ayrıca metodlar üzerinde de kullanılabilir ve final tanımlanırsa bağlı sınıflar içerisinde aynı metod adı kullanılamaz.

finally : Hata oluşsada oluşmasada çalıştırılacak kodlar için kullanılır.

floatval() : Herhangi bir değişken içeriğinin double veri türü değerini geriye döndürür.

floor() : Herhangi bir ondalıklı sayıyı aşağıya yuvarlayarak, yuvarlama sonucunda oluşan değeri geriye döndürür.

fmod() : Herhangi bir sayının belirtilecek olan değere göre modülüsünü (kalanını) bularak, bulunduğu değeri geriye döndürür.

Fonksiyon : Herhangi bir programlamada belirli bir veya daha fazla işi gerçekleştirmek üzere önceden hazırlanan küçük program parçacıklarına verilen isimdir.

fopen() : Belirtilecek olan dosyayı veya URL'i, belirtilecek olan modda açmak için kullanılır.

for : Döngü işlemi başlatır.

foreach : Diziler için döngü işlemi başlatır.

format() : date_diff() metodu ile hesaplanan iki tarih arasındaki farkı belirtilecek olan bir formata göre biçimlendirerek, biçimlendirdiği değeri geriye döndürür.

formatOutput : DOMDocument metodu ile oluşturulmuş ve loadXML() metodu ile nesne örneğine yüklenmiş olan içerikte sekme / girinti özelliğinin kullanılıp kullanılmayacağını belirtmek için kullanılır. (DOMDocument metodu ile)

fpassthru() : Belirtilecek olan dosya veya URL içeriğini kontrol ederek ve tüm içeriği bir defada bularak, bulunduğu değeri geriye döndürür.

fputcsv() : Belirtilecek olan csv dosyasına, belirtilecek olan verileri yazmak için kullanılır. Ayrıca istenirse işlem sonucunda yazılan karakter sayısı değerini bularak, bulunduğu değeri geriye döndürür.

fputs() : Belirtilecek olan dosyaya, belirtilecek olan verileri yazmak için kullanılır. Ayrıca istenirse işlem sonucunda yazılan karakter sayısı değerini bularak, bulunduğu değeri geriye döndürür.

fread() : Belirtilecek olan dosya veya URL içeriğini kontrol ederek ve belirtilecek karakter değerlerine bölünmüş içerikleri bularak, bulunduğu değeri geriye döndürür. Ayrıca tüm içerikte bulunan ve belirtilecek karakter değerlerine bölünmüş verileri herhangi bir döngü ile elde edilebilir.

free() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablo ile alakalı nesnesel yapı ile istenilen işlem / işlemler bittiğinde hafızayı boşaltmak için kullanılır.

free_result() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablo ile alakalı nesnesel yapı ile istenilen işlem / işlemler bittiğinde hafızayı boşaltmak için kullanılır.

fseek() : Belirtilecek olan dosya veya URL içeriğinde okumaya hazır bulunan konumu değiştirerek, değiştirdiği değeri geriye döndürür.

fstat() : fopen() metodu kullanılarak açılmış olan ve belirtilecek olan değer doğrultusunda, ilgili kaynağın dosya hakkındaki tüm bilgilerini bularak, bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu değeri geriye döndürür.

ftell() : Belirtilecek olan dosya veya URL içeriğinde okumaya hazır bulunan konumu bularak, bulunduğu değeri geriye döndürür.

ftruncate() : Belirtilecek olan dosya verisinin, belirtilecek olan karakter numarasından sonraki tüm karakterleri silmek için kullanılır. Ayrıca istenirse işlem sonucunu boolean veri türünde döndürmek için kullanılır.

func_get_arg() : Fonksiyona gönderilen parametre değerlerini bir dizi olarak kabul ederek, belirtilecek olan anahtara ait elemanın değerini döndürmek için kullanılır.

func_get_args() : Fonksiyona gönderilen parametre değerlerini alarak yeni bir dizi oluşturmak için kullanılır.

func_num_args() : Fonksiyona gönderilen parametre sayısını bulmak için kullanılır.

function : Fonksiyon tanımlamak için kullanılır.

function_exists() : Belirtilecek olan fonksiyonun varlığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

fwrite() : Belirtilecek olan dosyaya, belirtilecek olan verileri yazmak için kullanılır. Ayrıca istenirse işlem sonucunda yazılan karakter sayısı değerini bularak, bulunduğu değeri geriye döndürür.

g : 12 saatlik sistemde saatin sayısal değerini döndürür. (date() metodu ile)

G : 24 saatlik sistemde saatin sayısal değerini döndürür. (date() metodu ile)

g : Değeri kısa gösterim şeklinde ele alarak görüntülenmesini sağlar. (printf() veya sprintf() metodu ile)

GATEWAY_INTERFACE : PHP dosyasının çalıştığı server'ın desteklediği belirtiminin sürüm bilgisi değerini döndürür. (\$_SERVER süper globali ile)

GATEWAY_INTERFACE : PHP dosyasının çalıştığı server'ın desteklediği CGI belirtiminin sürüm bilgisi değerini döndürür. (getenv() metodu ile)

get_class_methods() : Kendisine parametre olarak verilen değer doğrultusunda sınıf içeriğini kontrol ederek, sınıf dahilinde bulunan metod isimlerini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür. İşlemde public ve static tanımlı metodlar değerlendirilir.

get_class_vars() : Kendisine parametre olarak verilen değer doğrultusunda sınıf içeriğini kontrol ederek, sınıf dahilinde bulunan değişken ve özellik isimleri ile birlikte değerlerini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür. İşlemde public ve static tanımlı özellikler değerlendirilir.

get_declared_classes() : PHP yazılımı içerisinde kullanılabilecek ve sisteme tanımlı olan ve aynı zamanda yazılımcı tarafından da oluşturulmuş olan tüm sınıfların listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_declared_interfaces() : PHP yazılımı içerisinde kullanılabilecek ve sisteme tanımlı olan ve aynı zamanda yazılımcı tarafından da oluşturulmuş olan tüm arayüzlerin listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_declared_traits() : PHP yazılımı içerisinde kullanılabilecek ve sisteme tanımlı olan ve aynı zamanda yazılımcı tarafından da oluşturulmuş olan tüm niteliklerin listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_defined_constants() : PHP yazılımı içerisinde kullanılabilecek ve sisteme tanımlı olan ve aynı zamanda yazılımcı tarafından da oluşturulmuş olan tüm sabitlerin listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_defined_functions() : PHP yazılımı içerisinde kullanılabilecek ve sisteme tanımlı olan ve aynı zamanda yazılımcı tarafından da oluşturulmuş olan tüm fonksiyonların listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_defined_vars() : PHP yazılımı içerisinde kullanılabilecek ve sisteme tanımlı olan ve aynı zamanda yazılımcı tarafından da oluşturulmuş olan tüm değişkenlerin listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_extension_funcs() : PHP yazılımı içerisinde derlenmiş ve yüklenmiş sisteme tanımlı ve belirtilecek olan modülün tüm fonksiyonlarının listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_headers() : Belirtilecek olan URL doğrultusunda HTML isteğine karşı sunucunun verdiği tüm yanıtları kontrol eder ve aldığı değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_html_translation_table() : PHP yazılımı içerisinde kullanılabilecek olan sisteme tanımlı tüm htmlspecialchars() ve htmlentities() dönüştürülebilir karakterlerinin listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_included_files() : Kullanılacak olan PHP dosyasına, dahil edilmiş / eklenmiş tüm PHP dosyalarının path'ini ve isimlerini bularak, bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_loaded_extensions() : PHP yazılımı içerisinde derlenmiş ve yüklenmiş sisteme tanımlı olan tüm modüllerin listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_meta_tags() : Belirtilecek olan URL doğrultusunda HTML dosyasında bulunan ve name değerine sahip tüm meta etiketlerini kontrol eder ve aldığı değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

get_object_vars() : Kendisine parametre olarak verilen değer doğrultusunda sınıf içeriğini kontrol ederek, sınıf dahilinde bulunan değişken ve özellik isimleri ile birlikte değerlerini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür. İşlemden yalnızca public tanımlı özellikler değerlendirilir.

get_required_files() : Kullanılacak olan PHP dosyasına, dahil edilmiş / eklenmiş tüm PHP dosyalarının path'ini ve isimlerini bularak, bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

getAvailableDrivers() : PHP yazılımı içerisinde sisteme tanımlı olan ve kullanılabilir PDO sürücülerini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

getcwd() : Geçerli çalışma dizini bilgisini bularak, bulunduğu değeri geriye döndürür.

getdate() : Geçerli zaman bilgileri değerlerinden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

getenv() : PHP yazılımı içerisinde kullanılabilecek ve sisteme tanımlı olan bir ortam değişkeninin değerini bularak, bulunduğu değeri geriye döndürür.

getMessage() : MySQL sunucusuna yeni bir bağlantı açma işlemi sırasında olası bir hata ile karşılaşması durumunda hata açıklama değerini bularak, bulunduğu değeri geriye döndürür.

getMessage() : Tüm istisnalar için oluşturulacak olan iletilerin atandığı veri tutuculardır.

getrandmax() : rand() metodu kullanılarak üretilebilecek en büyük sayının değerini bularak, bulunduğu değeri geriye döndürür.

gettimeofday() : Geçerli zamanın Unix zaman damgası, mikro saniye, yaz saati uygulaması ve Greenwich batısı değerlerinden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür. Ayrıca istenirse string olarak da geçerli zamanın Unix zaman damgası ve mikro saniye değerini bularak, bulunduğu değeri geriye döndürür.

gettype() : Herhangi bir değişken içeriğinin veri türünü bularak, bulunduğu değeri geriye döndürür.

glob() : Belirtilecek olan dizin içeriğini, belirtilecek olan kalıba kontrol ederek, ilgili dizin içerisinde bulunan ve kalıpla eşleşen tüm alt dizin ve dosya bilgilerinden (isimler ve varsa uzantılar) yeni bir dizi oluşturarak, oluşturduğu değeri geriye döndürür.

GLOB_BRACE : Belirtilecek olan genişletilmiş arama kalıbına göre dizin içerisindeki tüm dizinleri ve dosyaları, genişletilmiş arama kalıbı ile eşleştirerek listeler. (glob() metodu ile)

GLOB_ERR : Belirtilecek olan arama kalıbına göre dizin içerisindeki tüm dizinleri ve dosyaları, arama kalıbı ile eşleştirerek listeler. Ayrıca listeleme işlemi sırasında herhangi bir hatayla karşılaşılacak olur ise işlem durdurulur. (glob() metodu ile)

GLOB_MARK : Dizin içerisindeki tüm dizinleri ve dosyaları listeler. Ayrıca her dizin adının sonuna bir \ işareti ekler. (glob() metodu ile)

GLOB_NOCHECK : Belirtilecek olan arama kalıbına göre izin içerisindeki tüm izinleri ve dosyaları, arama kalıbı ile eşleştirerek listeler. Ayrıca arama kalıbı ile hiçbir eşleşme sağlanmıyorsa, arama kalıbı değerini geriye döndürür. (glob() metodu ile)

GLOB_NOSORT : Dizin içerisindeki tüm izinleri ve dosyaları listeler. Ayrıca izinlere ve dosyalara sıralama yapmaz. (glob() metodu ile)

GLOB_ONLYDIR : Belirtilecek olan arama kalıbına göre izin içerisindeki tüm izinleri, arama kalıbı ile eşleştirerek listeler. (glob() metodu ile)

goto : Kodlama dosyası içerisinde belirtilecek olan imlenmiş / hedeflenmiş satıra atlama koşulunu sağlamak için kullanılır.

GROUP BY : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun verilerini okuma işlemi sırasında belirtilecek olan sütuna / sütunlara ait tekrar eden verilerin gruplandırılmasını sağlamak için kullanılır.

h : 12 saatlik sistemde saatin iki haneli olacak şekilde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (date() metodu ile)

H : 24 saatlik sistemde saatin iki haneli olacak şekilde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (date() metodu ile)

hash() : Belirtilecek olan içeriğin hash özetini üreterek, ürettiği değeri geriye döndürür.

hash_algos() : PHP yazılımı içerisinde kullanılabilecek olan ve sisteme tanımlı tüm algoritmaların listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu değeri geriye döndürür.

hash_copy() : hash özeti üretme işlemlerini başlatmak ve algoritma türünü belirlemek için daha önceden oluşturulmuş olan hash_init() metodunu kopyalamak için kullanılır.

hash_file() : Belirtilecek olan dosyanın hash özetini üreterek, ürettiği değeri geriye döndürür.

hash_final() : Daha önceden hash özeti üretme işlemi başlatılarak içeriğide dahil edilmiş bir işlemin hash özetini üreterek, ürettiği değeri geriye döndürür.

hash_hmac() : Belirtilecek olan içeriğin HMAC yöntemi ile hash özetini üreterek, ürettiği değeri geriye döndürür.

hash_hmac_algos() : PHP yazılımı içerisinde anahtarlı olarak HMAC yöntemi ile kullanılabilecek olan sisteme tanımlı tüm algoritmaların listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

hash_hmac_file() : Belirtilecek olan dosyanın HMAC yöntemi ile hash özetini üreterek, ürettiği değeri geriye döndürür.

hash_init() : hash özeti üretme işlemlerini başlatmak ve algoritma türünü belirtmek için kullanılır.

hash_update() : Belirtilecek olan içeriği, daha önceden hash özeti üretme işlemi başlatılmış bir işleme dahil etmek için kullanılır.

HAVING : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun verilerini okuma işlemi sırasında belirtilecek olan sütuna / sütunlara ait tekrar eden verilerin GROUP BY ifadesi ile gruplandırılması sırasında koşul / koşullar tanımlamak / belirtmek için kullanılır.

header() : Belirtilecek olan içerik dahilinde, istemciye işlenmiş HTTP üst bilgisi iletmek için kullanılır.

hexdec() : Herhangi bir hexadecimal sayının, decimal sayı sistemi karşılığını bularak, bulunduğu değeri geriye döndürür.

highlight_string() : Belirtilecek olan herhangi bir PHP kodlamasını, PHP'nin yerleşik söz dizimi vurgulayıcısında tanımlanmış olan renkleri kullanarak, vurgulu bir halde ekran çıktılanmasını sağlamak için kullanılır.

host_info : Sisteme tanımlı olan MySQL sunucusunun nesnesel yapı ile isim ve bağlantı protokol türü bilgisini bularak, bulunduğu değeri geriye döndürür.

HTML_ENTITIES : get_html_translation_table metodu kullanılarak htmlentities metodu için tanımlanmış ve dönüştürülecek olan karakter listesini elde etmek için kullanılır. (get_html_translation_table() metodu ile)

html_entity_decode() : Belirtilecek olan ve htmlentities metodu kullanılarak düz metin halinde sadece okunabilir HTML metnine dönüştürülmüş olan tüm özel karakterleri, orjinal karakter haline geri dönüştürerek, dönüştürdüğü değeri geriye döndürür.

HTML_SPECIALCHARS : get_html_translation_table metodu kullanılarak htmlspecialchars metodu için tanımlanmış ve dönüştürülecek olan karakter listesini elde etmek için kullanılır. (get_html_translation_table() metodu ile)

htmlentities() : Belirtilecek olan içerikte bulunabilecek olan tüm özel karakterleri düz metin halinde okunabilir HTML metnine dönüştürerek, dönüştürdüğü değeri geriye döndürür.

htmlspecialchars() : Belirtilecek olan içerikte bulunabilecek olan özel karakterleri düz metin halinde okunabilir HTML metnine dönüştürerek, dönüştürdüğü değeri geriye döndürür.

htmlspecialchars_decode() : Belirtilecek olan ve htmlspecialchars metodu kullanılarak düz metin halinde sadece okunabilir HTML metnine dönüştürülmüş olan özel karakterleri, orjinal karakter haline geri dönüştürerek, dönüştürdüğü değeri geriye döndürür.

HTTP_ACCEPT : PHP dosyasına erişim isteği için kabul edilen başlık içeriği bilgisi değerini döndürür. (\$_SERVER süper globali ile)

HTTP_ACCEPT : PHP dosyasına erişim isteği için kabul edilen başlık içeriği bilgisi değerini döndürür. (getenv() metodu ile)

HTTP_ACCEPT_ENCODING : PHP dosyasına erişim isteği için kabul edilen kodlama başlık içeriği bilgisi değerini döndürür. (\$_SERVER süper globali ile)

HTTP_ACCEPT_ENCODING : PHP dosyasına erişim isteği için kabul edilen kodlama başlık içeriği bilgisi değerini döndürür. (getenv() metodu ile)

HTTP_ACCEPT_LANGUAGE : PHP dosyasına erişim isteği için kabul edilen lisan başlık içeriği bilgisi değerini döndürür. (\$_SERVER süper globali ile)

HTTP_ACCEPT_LANGUAGE : PHP dosyasına erişim isteği için kabul edilen lisan başlık içeriği bilgisi değerini döndürür. (getenv() metodu ile)

http_build_query() : Belirtilecek olan dizi içerisindeki tüm anahtar ve değerleri bularak, bulduğu değerleri URL kodlamalı bir sorgu dizisine dönüştürerek, dönüştürdüğü değeri geriye döndürür.

HTTP_CONNECTION : PHP dosyasına erişim isteği için kabul edilen bağlantı başlık içeriği bilgisi değerini döndürür. (\$_SERVER süper globali ile)

HTTP_CONNECTION : PHP dosyasına erişim isteği için kabul edilen bağlantı başlık içeriği bilgisi değerini döndürür. (getenv() metodu ile)

HTTP_HOST : PHP dosyasının çalıştığı host başlığı bilgisi değerini döndürür. (\$_SERVER süper globali ile)

HTTP_HOST : PHP dosyasının çalıştığı host başlığı bilgisi değerini döndürür. (getenv() metodu ile)

HTTP_REFERER : Kullanıcının browser'ını geçerli PHP dosyasına gönderen (referans olan) sayfanın URL bilgisi değerini döndürür. Bazen değer elde edilemeyebilir. (getenv() metodu ile)

HTTP_REFERER : Kullanıcının browser'ını geçerli PHP dosyasına gönderen / referans olan sayfanın bilgisi değerini döndürür. Bazen değer elde edilemeyebilir. (\$_SERVER süper globali ile)

HTTP_USER_AGENT : PHP dosyasına erişim isteğinde bulunan kullanıcı browser'ı sistem bilgisi değerini döndürür. (\$_SERVER süper globali ile)

HTTP_USER_AGENT : PHP dosyasına erişim isteğinde bulunan kullanıcı browser'ı sistem bilgisi değerini döndürür. (getenv() metodu ile)

HTTPS : PHP dosyasına erişim isteği için Htts protokolünün kullanılıp kullanılmadığı bilgisi değerini döndürür. Htts protokolü yoksa değer boş döner. (\$_SERVER süper globali ile)

HTTPS : PHP dosyasına erişim isteği için Htts protokolünün kullanılıp kullanılmadığı bilgisi değerini döndürür. Htts protokolü yoksa değer boş döner. (getenv() metodu ile)

hypot() : Belirtilecek olan uzunluk değerlerine sahip bir dik üçgenin hipotenüsünü hesaplayarak, bulduğu değeri geriye döndürür.

I : Tüm tarih ve saat işlevleri tarafından kullanılan yaz saati uygulamasının olup olmadığı değerini döndürür. Yaz saati uygulaması var ise 1, yaz saati uygulaması yok ise 0. (Büyük I harfi) (date() metodu ile)

IGNORE : Belirtilecek olan içerikte gösterilmeyen bir karaktere / karakterlere rastlanacak olursa, ilgili karakter / karakterler yok sayılır veya ilgili karakterin / karakterlerin yerine kural dışı karakter değeri yerleştirmek için kullanılır. (iconv() metodu ile)

IN : İçeren (MySQL'de)

INFO_ALL : Tüm bilgiler (phpinfo() metodu ile)

INFO_CONFIGURATION : Konfigurasyon bilgileri (phpinfo() metodu ile)

INFO_CREDITS : Katlıda bulunanların bilgileri (phpinfo() metodu ile)

INFO_ENVIRONMENT : Ortam bilgileri (phpinfo() metodu ile)

INFO_GENERAL : Genel bilgiler (phpinfo() metodu ile)

INFO_LICENSE : Lisans bilgileri (phpinfo() metodu ile)

INFO_MODULES : Modül bilgileri (phpinfo() metodu ile)

INFO_VARIABLES : \$_SERVER süper globali / öntanımlı değişken bilgileri (phpinfo() metodu ile)

INNER JOIN : MySQL sunucusundaki database içerisinde bulunan herhangi iki veya daha fazla tablonun eşleşen değerlerinin birbirleri ile ilişkilendirilmesi için kullanılır.

INSERT INTO : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya yeni bir kayıt satırı ile birlikte verisini de / verilerini de eklemek için kullanılır.

i : Dakikanın iki haneli olacak şekilde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (date() metodu ile)

i : Integer (bind_param() metodu ile)

i : Karakter diziliminin karşılaştırılmasında büyük harf / küçük harf ayrımının yapılıp yapılmayacağının tanımlamasında kullanılır. (RegExp ile)

iconv() : Belirtilecek olan içeriği, belirtilecek olan karakter kodlaması dahilinde dönüştürerek, karakter kodlaması dönüştürülmüş olan değeri geriye döndürür.

iconv_get_encoding() : PHP sayfası dahilinde kullanılmakta olan karakter kodlama bilgileri listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür. Ayrıca belirtilecek olan değer doğrultusunda tüm karakter kodlama bilgisi yerine sadece ilgili karakter kodlaması bilgisi değerinde döndürebilir.

iconv_set_encoding() : PHP sayfası dahilinde kullanılmakta olan karakter kodlama türüne, belirtilecek olan karakter kodlama kümesini atamak için kullanılır.

if : Eğer

implements : Daha önceden oluşturulmuş olan herhangi bir arayüz sınıfını, ilgili sınıfa tanımlamak / uygulamak için kullanılır.

implode() : Diziyi birleştirerek düz bir metin içeriği oluşturmak için kullanılır.

in_array() : Dizi içerisinde bulunan eleman veya elemanlar dahilinde, belirtilecek olan elemanın olup olmadığını kontrol etmek için kullanılır.

include() : PHP dosyasına, belirtilecek olan yeni bir PHP dosyasını dahil etmek / eklemek için kullanılır. (Yeni dosyanın daha önceden eklenip eklenmediği kontrol edilmez. Herhangi bir hata kodu dönecek olur ise kodlar okunmaya devam eder.)

include_once() : PHP dosyasına, belirtilecek olan yeni bir PHP dosyasını dahil etmek / eklemek için kullanılır. (Yeni dosyanın daha önceden eklenip eklenmediği kontrol edilir. Herhangi bir hata kodu dönecek olur ise kodlar okunmaya devam eder.)

ini_get() : PHP yazılımı içerisinde kullanılabilir ve sisteme tanımlı olan bir yapılandırma yönergesinin değerini bularak, bulunduğu değeri geriye döndürür.

ini_get_all() : PHP yazılımı içerisinde kullanılabilir ve sisteme tanımlı olan tüm yapılandırma yönergeleri listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür. Ayrıca belirtilecek olan değer doğrultusunda tüm yapılandırma yönergeleri yerine sadece ilgili yapılandırma yönergesi değerinde döndürebilir.

ini_restore() : PHP yazılımı içerisinde kullanılabilir ve sisteme tanımlı olan bir yapılandırma yönergesinin, ini_set() metodu kullanılarak değiştirilmiş olan değerini varsayılan haline geri döndürmek için kullanılır.

ini_set() : PHP yazılımı içerisinde kullanılabilir ve sisteme tanımlı olan bir yapılandırma yönergesine, değer atamak için kullanılır.

insert_id() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya nesnesel yapı ile yeni bir kayıt eklendiğinde, eklenen son kaydın id değerini / verisini bulmak için kullanılır.

instanceof : Örneği operatörü

insteadof : Sınıf için tanımlanmış olan özellik ve metod gruplarında tercih / seçim işlemleri için kullanılır.

intdiv() : Herhangi bir sayının belirtilecek olan sayı değerine tam olarak kaç defa bölünebildiğini bularak, bulunduğu değeri geriye döndürür.

interface : Belirtilecek olan herhangi bir sınıfı, soyut arayüz sınıfı haline dönüştürmek için kullanılır.

interface_exists() : Belirtilecek olan class'a ait, belirtilecek olan interface'in varlığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

intval() : Herhangi bir değişken içeriğinin integer veri türü değerini geriye döndürür.

is_a() : Belirtilecek olan class'ın nesnel örneğini varlığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_array() : Belirtilecek olan değişken içeriğinin array veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_bool() : Belirtilecek olan değişken içeriğinin boolean veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_callable() : Belirtilecek olan değişken içeriğinin callable veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_dir() : İçeriğin belirtilecek olan path'e bağlı olarak dizin olduğunu sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_double() : Belirtilecek olan değişken içeriğinin double / float veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_executable() : Belirtilecek olan path'e bağlı olarak dosyanın çalıştırılabilir olduğunu sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_file() : İçeriğin belirtilecek olan path'e bağlı olarak dosya olduğunu sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_float() : Belirtilecek olan değişken içeriğinin double / float veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_int() : Belirtilecek olan değişken içeriğinin integer veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_integer() : Belirtilecek olan değişken içeriğinin integer veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_long() : Belirtilecek olan değişken içeriğinin integer veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_null() : Belirtilecek olan değişken içeriğinin null veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_numeric() : Belirtilecek olan değişken içeriğinin numeric veri türlerinden biri olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_object() : Belirtilecek olan değişken içeriğinin object veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_readable() : Belirtilecek olan path'e bağlı olarak dosyanın okunabilir olduğunu sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_real() : Belirtilecek olan değişken içeriğinin double / float veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_resource() : Belirtilecek olan değişken içeriğinin resource veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_string() : Belirtilecek olan değişken içeriğinin string veri türünde olup olmadığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_subclass_of() : Belirtilecek olan class'dan türeyen, belirtilecek olan sub class'ın varlığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_uploaded_file() : Belirtilecek olan path'e bağlı olarak dosyanın yüklenebilir olduğunu sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_writable() : Belirtilecek olan path'e bağlı olarak dosyanın yazılabilir olduğunu sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

is_writeable() : Belirtilecek olan path'e bağlı olarak dosyanın yazılabilir olduğunu sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

isset() : Belirtilecek olan değişkenin veya dizi içerisinde belirtilecek olan anahtarın varlığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

j : Günün sayısal değerini döndürür. (date() metodu ile)

json_decode() : Belirtilecek olan JSON değerinin standart kodlama karşılığını bularak, bulunduğu değeri geriye döndürür.

json_encode() : Belirtilecek olan içeriğin, JSON karşılığını bularak, bulunduğu değeri geriye döndürür. (json_encode() metodu ile)

JSON_FORCE_OBJECT : Belirtilecek olan içeriği nesne dönüşüm yapısına göre zorlamak için kullanılır. (json_encode() metodu ile)

JSON_HEX_AMP : Belirtilecek olan içerikte bulunan & karakterlerini hexadecimal türünde dönüştürmek için kullanılır. (json_encode() metodu ile)

JSON_HEX_APOS : Belirtilecek olan içerikte bulunan ' karakterlerini hexadecimal türünde dönüştürmek için kullanılır. (json_encode() metodu ile)

JSON_HEX_QUOT : Belirtilecek olan içerikte bulunan " karakterlerini hexadecimal türünde dönüştürmek için kullanılır. (json_encode() metodu ile)

JSON_HEX_TAG : Belirtilecek olan içerikte bulunan tag başlangıç ve bitiş karakterlerini hexadecimal türünde dönüştürmek için kullanılır. (json_encode() metodu ile)

key() : Dizinin göstericisi konumundaki elemanın anahtarını bulmak için kullanılır.

krsort() : Dizinin anahtarlarını büyük harf küçük harf duyarlı olacak şekilde z'den a'ya yada büyükten küçüğe sıralamak için kullanılır.

ksort() : Dizinin anahtarlarını büyük harf küçük harf duyarlı olacak şekilde a'dan z'ye yada küçükten büyüğe sıralamak için kullanılır.

l : Günün adı değerini döndürür. (Küçük L harfi) (date() metodu ile)

L : Tanımlanmadan önce ifade edilen kurallar sonraki kurallara geçmemesini demektir. (.htaccess ile)

L : Yılın artık bir yıl olup olmadığı değerini döndürür. Yıl artık yıl ise 1, artık bir yıl değil ise 0. (date() metodu ile)

lastInsertId() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya yeni bir kayıt eklendiğinde, eklenen son kaydın id değerini / verisini bulmak için kullanılır.

lcfirst() : Belirtilecek olan içeriğin ilk harfini küçük harf olacak şekilde değiştirerek, değiştirilmiş olan değeri geriye döndürür.

lcg_value() : 0 ila 1 arasında rastgele bir onluklu sayı üreterek, ürettiği değeri geriye döndürür.

LEFT JOIN : MySQL sunucusundaki database içerisinde bulunan herhangi iki veya daha fazla tablonun soldaki tabloya göre birbirleri ile ilişkilendirilmesi için kullanılır.

lengths() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun herhangi bir kaydını içeren satırındaki tüm sütunlarda bulunan verilerin nesnesel yapı ile ayrı ayrı toplam karakter sayısını bulmak için kullanılır.

levenshtein() : Belirtilecek olan iki değer arasındaki levenshtein mesafesini bularak, bulunduğu değeri geriye döndürür.

LIBXML_NOCDATA : Belirtilecek olan XML kodlama değeri içerisinde bulunan CDATA değeri / değerlerini metin düğümleri ile birleştirmek için kullanılır. (simplexml_load_file() metodu ile)

LIKE : İçerisinde (MySQL'de)

LIMIT : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun işlem esnasında tüm verilerini işlemek yerine sadece belirtilen adet veya aralıklar kadar işlemek istenildiğini belirtmek için kullanılır.

list() : Dizi içerisindeki eleman / elemanları değişkenlere atamak için kullanılır.

loadXML() : DOMDocument metodu ile oluşturulmuş olan DOMDocument nesne örneğine belirtilecek olan herhangi bir içeriği yüklemek için kullanılır. (DOMDocument metodu ile)

localtime() : Yerel zaman bilgileri değerlerinden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

log() : Herhangi bir sayının doğal logaritmasını bularak, bulunduğu değeri geriye döndürür.

log10() : Herhangi bir sayının 10'luk tabana göre doğal logaritmasını bularak, bulunduğu değeri geriye döndürür.

ltrim() : Belirtilecek olan içeriğin, varsa başındaki / solundaki tüm boşlukları, sekmeleri ve yeni satır karakterlerini silerek, oluşturduğu değeri geriye döndürür. Ayrıca belirtilecek olan içeriğin, varsa başındaki / solundaki, belirtilecek olan karakterleride silerek, oluşturduğu değeri geriye döndürür.

m : Ayın iki haneli olacak şekilde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (date() metodu ile)

M : Ayın üç haneli olacak şekilde adı değerini döndürür. (date() metodu ile)

M_1_PI : 1/PI sayısının değerini geriye döndürür.

M_2_PI : 2/PI sayısının değerini geriye döndürür.

M_2_SQRTPI : 2/PI sayısının karekökünü bularak, bulunduğu değeri geriye döndürür.

M_E : Euler sayısı değerini geriye döndürür.

M_EULER : Euler sabiti değerini geriye döndürür.

M_LN10 : 10 Sayısının doğal logaritması değerini elde etmek için kullanılır.

M_LN2 : 2 Sayısının doğal logaritması değerini elde etmek için kullanılır.

M_LNPI : PI sayının doğal logaritması değerini geriye döndürür.

M_LOG10E : Euler sayısının 10 tabanına göre logaritmik değerini geriye döndürür.

M_LOG2E : Euler sayısının 2 tabanına göre logaritmik değerini geriye döndürür.

M_PI : PI sayısı değerini geriye döndürür.

M_PI_2 : PI/2 sayısının değerini geriye döndürür.

M_PI_4 : PI/4 sayısının değeri geriye döndürür.

M_SQRT1_2 : 1/2 Sayısının karekök değeri geriye döndürür.

M_SQRT2 : 2 Sayısının karekök değeri geriye döndürür.

M_SQRT3 : 3 Sayısının karekök değeri geriye döndürür.

M_SQRTPI : PI sayısının karekökünü bularak, bulunduğu değeri geriye döndürür.

MAX : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun belirtilecek olan sütununun en büyük kayıt değeri bulmak için kullanılır.

max() : Değişken içeriklerinin veya dizi elemanlarının rakamsal değerler içermesi durumunda en büyük rakamsal değeri bularak, bulunduğu değeri geriye döndürür.

max() : Dizi içerisinde bulunan eleman veya elemanların rakamsal değerler içermesi durumunda en büyük rakamsal değer içeren elemanı bulmak için kullanılır.

MB_CASE_LOWER : Belirtilecek olan içeriğin tüm harflerini küçük harf olacak şekilde değiştirmek için kullanılır. (mb_convert_case() metodu ile)

MB_CASE_TITLE : Belirtilecek olan içeriğin tüm kelimelerinin baş harflerini harflerini büyük harf olacak şekilde değiştirmek için kullanılır. (mb_convert_case() metodu ile)

MB_CASE_UPPER : Belirtilecek olan içeriğin tüm harflerini büyük harf olacak şekilde değiştirmek için kullanılır. (mb_convert_case() metodu ile)

mb_convert_case() : Belirtilecek olan içeriğin, belirtilecek olan dönüşüm türünde ve karakter kodlaması dahilinde gelişmiş olarak tüm harflerini değiştirerek, değiştirilmiş olan değeri geriye döndürür.

mb_convert_encoding() : Belirtilecek olan içeriği, belirtilecek olan karakter kodlaması dahilinde dönüştürerek, karakter kodlaması dönüştürülmüş olan değeri geriye döndürür.

mb_detect_encoding() : Belirtilecek olan içeriğin karakter kodlamasını algılayarak, bulunduğu değeri geriye döndürür.

mb_list_encodings() : PHP yazılımı içerisinde kullanılabilecek olan sisteme tanımlı tüm karakter kodlaması listesini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

mb_parse_str() : Belirtilecek olan URL değeri içerisindeki değişken değerlerini, kendi içerisinde belirtilecek olan yeni bir değişkene çözümleyerek, çözümlediği değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

mb_split() : Belirtilecek olan içeriği, belirtilecek olan düzenli ifade dahilinde bölerek, bölümleme işlemi sonucunda oluşan değerlerden yeni bir dizi oluşturarak, oluşturduğu değeri geriye döndürür.

mb_strpos() : Belirtilecek olan içerikte, belirtilecek olan herhangi bir değer büyük harf / küçük harf ayrımı olmaksızın, belirtilecek olan karakter kodlaması dahilinde gelişmiş olarak ilk eşleşme sonucundaki konumunu / sırasını bularak, bulunduğu değeri geriye döndürür.

mb_strlen() : Belirtilecek olan içeriğin, belirtilecek olan karakter kodlaması dahilinde gelişmiş olarak karakter sayısını bularak, bulunduğu değeri geriye döndürür.

mb_strpos() : Belirtilecek olan içerikte, belirtilecek olan herhangi bir değer, belirtilecek olan karakter kodlaması dahilinde gelişmiş olarak ilk eşleşme sonucundaki konumunu / sırasını bularak, bulunduğu değeri geriye döndürür.

mb_strrpos() : Belirtilecek olan içerikte, belirtilecek olan herhangi bir değer büyük harf / küçük harf ayrımı olmaksızın, belirtilecek olan karakter kodlaması dahilinde gelişmiş olarak son eşleşme sonucundaki konumunu / sırasını bularak, bulunduğu değeri geriye döndürür.

mb_strrpos() : Belirtilecek olan içerikte, belirtilecek olan herhangi bir değer, belirtilecek olan karakter kodlaması dahilinde gelişmiş olarak son eşleşme sonucundaki konumunu / sırasını bularak, bulunduğu değeri geriye döndürür.

mb_strtolower() : Belirtilecek olan içeriğin, belirtilecek olan karakter kodlaması dahilinde gelişmiş olarak tüm harflerini küçük harf olacak şekilde değiştirerek, değiştirilmiş olan değeri geriye döndürür.

mb_strtoupper() : Belirtilecek olan içeriğin, belirtilecek olan karakter kodlaması dahilinde gelişmiş olarak tüm harflerini büyük harf olacak şekilde değiştirerek, değiştirilmiş olan değeri geriye döndürür.

mb_substr() : Belirtilecek olan içerikte, belirtilecek olan karakter sıra numarası aralıklarına göre, belirtilecek olan karakter kodlaması dahilinde gelişmiş olarak istenilen değeri bularak, bulunduğu değeri geriye döndürür.

mb_substr_count() : Belirtilecek olan herhangi bir değer, belirtilecek olan içerikte, belirtilecek olan karakter kodlaması dahilinde gelişmiş olarak kaç tane olduğunu bularak, bulunduğu değeri geriye döndürür.

md5() : Belirtilecek olan içeriğin md5 özetini hesaplayıp bularak, bulunduğu değeri geriye döndürür.

md5_file() : Belirtilecek olan dosyanın md5 özetini hesaplayıp bularak, bulunduğu değeri geriye döndürür.

memory_get_usage() : Çalışmakta olan PHP dosyası için sistemden ayrılan memory miktarı değerini byte cinsinden bularak, bulunduğu değeri geriye döndürür.

method_exists() : Belirtilecek olan class içerisinde, belirtilecek olan metodun varlığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

MIN : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun belirtilecek olan sütununun en küçük kayıt değerini bulmak için kullanılır.

microtime() : Geçerli zamanın Unix zaman damgası ve mikro saniye değerini bularak, bulunduğu değeri geriye döndürür.

min() : Değişken içeriklerinin veya dizi elemanlarının rakamsal değerler içermesi durumunda en küçük rakamsal değeri bularak, bulunduğu değeri geriye döndürür.

min() : Dizi içerisinde bulunan eleman veya elemanların rakamsal değerler içermesi durumunda en küçük rakamsal değer içeren elemanı bulmak için kullanılır.

mkdir() : Belirtilecek olan değer ve izinler doğrultusunda yeni bir dizin oluşturmak için kullanılır. Ayrıca istenirse işlem sonucunu boolean veri türünde döndürmek için kullanılır.

mktime() : Belirtilecek olan zamanın Unix zaman damgası değerini bularak, bulunduğu değeri geriye döndürür.

MODIFY / MODIFY COLUMN : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablodaki herhangi bir sütunun yapısını değiştirmek için kullanılır. (ALTER TABLE ifadesi ile)

move_uploaded_file() : Herhangi bir dosyayı belirtilecek olan dizine yüklemek için kullanılır.

mt_getrandmax() : mt_rand() metodu ile Mersenne twister algoritması kullanılarak üretilebilecek olan en büyük sayı değerini bularak, bulunduğu değeri geriye döndürür.

mt_rand() : Mersenne twister algoritması kullanılarak rastgele bir tam sayı üreterek, ürettiği değeri geriye döndürür.

MYSQLI_ASSOC : Tablonun verilerini okuma işlemi sırasında oluşturulacak olan dizide ilgili tablonun sütun isimleri dizinin anahtarları olarak tanımlanır ve ilgili verilere sütun isimleri ile erişilir. (mysqli_fetch_array() veya fetch_array() metodu ile)

MYSQLI_BOTH : Tablonun verilerini okuma işlemi sırasında oluşturulacak olan dizide dizinin anahtarları hem ilgili tablonun sütun isimleri ile hemde PHP tarafından otomatik olarak tanımlanır ve ilgili verilere ister sütun isimleri ile ister sütun sıra numaraları ile erişilebilir. (mysqli_fetch_array() veya fetch_array() metodu ile)

MYSQLI_NUM : Tablonun verilerini okuma işlemi sırasında oluşturulacak olan dizide dizinin anahtarları PHP tarafından otomatik olarak tanımlanır ve ilgili verilere sütun sıra numaraları ile erişilir. (mysqli_fetch_array() veya fetch_array() metodu ile)

mysqli_affected_rows() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya bağlanılarak işlem yapıldığında etkilenen kayıt sayısını bulmak için kullanılır.

mysqli_autocommit() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya işlem yapılacağı anda otomatik işleyen geçici işlem yapılarını açmak veya kapatmak için kullanılır.

mysqli_close() : Kendisine parametre olarak verilen değer doğrultusunda, daha önceden açılmış olan MySQL bağlantısını kapatmak / sonlandırmak için kullanılır.

mysqli_commit() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya işlem yapılacağı anda daha önceden mysqli_autocommit() metodu kullanılarak açılmış olan otomatik işleyen geçici işlem yapıları dahilinde işleme alınmış olan query'nin geçicilikten çıkarak kalıcı hale döndürülmesi için kullanılır.

mysqli_connect() : Kendisine parametre olarak verilen değerler doğrultusunda, MySQL sunucusuna yeni bir bağlantı açmak için kullanılır. Ayrıca istenirse açılacak olan MySQL sunucusunun tüm bağlantı bilgilerini bulur ve bulduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

mysqli_connect_errno() : MySQL sunucusuna yeni bir bağlantı açma işlemi sırasında olası bir hata ile karşılaşması durumunda hata kodu değerini bularak, bulduğu değeri geriye döndürür.

mysqli_connect_error() : MySQL sunucusuna yeni bir bağlantı açma işlemi sırasında olası bir hata ile karşılaşması durumunda hata açıklama değerini bularak, bulduğu değeri geriye döndürür.

mysqli_fetch_array() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun tüm verilerini düzenli bir dizi halinde okumak için kullanılır.

mysqli_fetch_assoc() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun tüm verilerini ilişkisel bir dizi halinde okumak için kullanılır.

mysqli_fetch_lengths() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun herhangi bir kaydını içeren satırındaki tüm sütunlarda bulunan verilerin ayrı ayrı toplam karakter sayısını bulmak için kullanılır.

mysqli_fetch_object() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun tüm verilerini nesne halinde okumak için kullanılır.

mysqli_fetch_row() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun tüm verilerini numaralandırılmış bir dizi halinde okumak için kullanılır.

mysqli_free_result() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablo ile alakalı istenilen işlem / işlemler bittiğinde hafızayı boşaltmak için kullanılır.

mysqli_get_client_info() : PHP yazılımı içerisinde sisteme tanımlı olan MySQL istemcisi kütüphanesinin (libmysql) detay bilgisini bularak, bulduğu değeri geriye döndürür.

mysqli_get_client_version() : PHP yazılımı içerisinde sisteme tanımlı olan MySQL istemcisi kütüphanesinin (libmysql) sürüm bilgisini bularak, bulduğu değeri geriye döndürür.

mysqli_get_host_info() : Sisteme tanımlı olan MySQL sunucusunun isim ve bağlantı protokol türü bilgisini bularak, bulduğu değeri geriye döndürür.

mysqli_get_proto_info() : Sisteme tanımlı olan MySQL sunucusunun bağlantı protokolü sürüm bilgisini bularak, bulduğu değeri geriye döndürür.

mysqli_get_server_info() : Sisteme tanımlı olan MySQL sunucusunun detay bilgisini bularak, bulduğu değeri geriye döndürür.

mysqli_get_server_version() : Sisteme tanımlı olan MySQL sunucusunun sürüm bilgisini bularak, bulduğu değeri geriye döndürür.

mysqli_init() : MySQL sunucusuna yeni bir bağlantı açmak için kullanılır.

mysqli_insert_id() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya yeni bir kayıt eklendiğinde, eklenen son kaydın id değerini / verisini bulmak için kullanılır.

mysqli_num_rows() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun toplam kayıt sayısını bulmak için kullanılır.

mysqli_query() : MySQL sunucusuna açılmış olan bağlantı dahilinde yeni bir sorgu işlemi yapmak için kullanılır. Ayrıca istenirse sorgu işlemi ile alakalı tüm bilgileri bulur ve bulduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

mysqli_real_connect() : mysqli_init() metodu kullanılarak açılmış olan yeni MySQL bağlantısına, bağlantı bilgilerini atamak / tanımlamak için kullanılır.

mysqli_real_escape_string() : Belirtilecek olan içerikte bulunabilecek olan tüm özel karakterlerin önüne \ işareti yani kaçırma operatörü ekler ve sade bir metin içeriği oluşturarak, oluşturduğu değeri geriye döndürür.

mysqli_rollback() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya işlem yapılacağı anda daha önceden mysqli_autocommit() metodu kullanılarak açılmış olan otomatik işleyen geçici işlem yapıları dahilinde işleme alınmış olan query'nin geçicilikten çıkarak işlemin geri alınması / iptal edilmesi için kullanılır.

mysqli_select_db() : Kendisine parametre olarak verilen değerler doğrultusunda, daha önceden açılmış olan MySQL bağlantısı üzerinden istenilen veritabanı tablosunu seçmek için kullanılır.

mysqli_set_charset() : Kendisine parametre olarak verilen değerler doğrultusunda, daha önceden açılmış olan MySQL bağlantısına varsayılan istemci karakter setini atamak / tanımlamak için kullanılır.

n : Ayın sayısal değerini döndürür. (date() metodu ile)

N : ISO-8601 standartlarına uygun olacak şekilde haftanın gününün sayısal değerini döndürür. 1'dan (pazartesi) 7'ya (pazar) kadar. (date() metodu ile)

namespace : Sınıflar için özel isim alanları / küme tanımlamak için kullanılır.

natcasesort() : Dizinin elemanlarını doğal sıralama şeklinde a'dan z'ye yada küçükten büyüğe sıralamak için kullanılır.

natsort() : Dizinin elemanlarını büyük harf küçük harf duyarlı olacak ve doğal sıralama şeklinde a'dan z'ye yada küçükten büyüğe sıralamak için kullanılır.

NATURAL JOIN : MySQL sunucusundaki database içerisinde bulunan herhangi iki veya daha fazla tablonun eşleşen değerlerinin birbirleri ile otomatik ilişkilendirilmesi için kullanılır.

nautical_twilight_begin : Deniz alacakaranlık başlangıcı (date_sun_info() metodu ile)

nautical_twilight_end : Deniz alacakaranlık bitişi (date_sun_info() metodu ile)

NC : Büyük harf / küçük harf duyarlılığını kaldırır. (.htaccess ile)

new : Daha önceden tanımlanmış olan herhangi bir sınıfı çağırmak için kullanılır.

new : Yeni operatörü

next() : Dizinin göstericisi konumundaki anahtarı veya elemanı bir ileri almak için kullanılır.

nl2br() : Belirtilecek olan içerikte bulunabilecek olan tüm özel satır sonlandırma karakterlerini (\n) HTML satır sonlandırma karakterlerine (
) değiştirerek, değiştirilmiş olan değeri geriye döndürür.

NOT : Olumsuzluk (MySQL'de)

NOT BETWEEN : Aralığında değil (MySQL'de)

NOT FIND_IN_SET : Ayrılmış halde içermeyen (MySQL'de)

NOT IN : İçermeyen (MySQL'de)

NOT LIKE : İçerisinde değil (MySQL'de)

num_rows : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun nesnesel yapı ile toplam kayıt sayısını bulmak için kullanılır.

number_format() : Herhangi bir sayıyı belirtilecek olan ayraçlar doğrultusunda biçimlendirerek, biçimlendirdiği değeri geriye döndürür.

o : Deęeri tam sayı řeklinde ele alarak, deęerin iřaretli octal sayı sisteminde grntlenmesini saęlar. (printf() veya sprintf() metodu ile)

o : ISO-8601 standartlarına uygun olacak řekilde yılın drt haneli olacak řeklinde sayısal deęerini dndrr. (date() metodu ile)

O : Tm tarih ve saat iřlevlerindeki GMT zaman farkı deęerini dndrr. (date() metodu ile)

ob_end_clean() : PHP sayfası dahilinde bařlatılmıř olan ve veri depolayan ıktı tamponlamasını siler ve ıktı tamponlamasını kapatır.

ob_end_flush() : PHP sayfası dahilinde bařlatılmıř olan ve veri depolayan ıktı tamponlamasını bořaltır ve ıktı tamponlamasını kapatır.

ob_get_contents() : PHP sayfası dahilinde bařlatılmıř olan ve veri depolayan ıktı tamponlaması ierięini bularak, bulduęu deęeri geriye dndrr.

ob_start() : PHP sayfası dahilinde ıktı tamponlamasını bařlatır.

octdec() : Herhangi bir octal sayının, decimal sayı sistemi karřılıęını bularak, bulduęu deęeri geriye dndrr.

ON : MySQL sunucusundaki database ierisinde bulunan herhangi iki veya daha fazla tablonun birbirleri ile iliřkilendirilmesi sırasında stn kořulunu tanımlamak / belirtmek iin kullanılır. (JOIN ifadeleri ile)

opendir() : Belirtilecek olan dizini aarak, atıęı dizini belleęe almak iin kullanılır.

OPTIMIZE : MySQL sunucusundaki database ierisinde bulunan herhangi bir tablonun optimize edilmesi iin kullanılır.

OR : Veya (MySQL'de)

or : Veya operatr

ord() : Belirtilecek olan standart bir karakter deęerinin ASCII karřılıęını bularak, bulduęu deęeri geriye dndrr.

ORDER BY : MySQL sunucusundaki database ierisinde bulunan herhangi bir tablonun verilerine iřlem esnasında hangi sıra / sıralar dahilinde eriřilmek istenildięini belirtmek iin kullanılır.

P : Tm tarih ve saat iřlevlerindeki saat ve dakika olarak GMT zaman farkı deęerini dndrr. (date() metodu ile)

parent:: : Ebeveyn sınıfımda řeklinde sınıfı iřaret etmek iin kullanılır.

parse_ini_file() : PHP yazılımı ierisinde kullanılabilecek ve yazılımcı tarafından oluřturulmuř olan tm yapılandırma ynergeleri listesini bulur ve bulduęu deęerlerden yeni bir dizi oluřturarak, oluřturduęu dizi deęerini geriye dndrr.

parse_str() : Belirtilecek olan URL deęeri ierisindeki deęiřken deęerlerini czmleyerek, czmledięi deęiřkenin deęerini geriye dndrr.

parse_url() : Belirtilecek olan URL deęerini paralarına ayırřtırarak iřlem sonucunda bulduęu deęerlerden yeni bir dizi oluřturarak, oluřturduęu deęeri geriye dndrr.

PATH_INFO : alıřmakta olan PHP dosyası URL adresinin path bilgisi deęerini dndrr. (getenv() metodu ile)

pathinfo() : Belirtilecek olan deęer doęrultusunda, kaynaęın ierisinde bulunduęu dizin / dizinlerin adını ve dosya adını bularak, bulduęu deęerlerden yeni bir dizi oluřturarak, oluřturduęu dizi deęerini geriye dndrr.

PDO() : Kendisine parametre olarak verilen deęerler doęrultusunda, MySQL sunucusuna yeni bir baęlantı amak iin kullanılır.

PDO::PARAM_BOOL : Boolean (bindParam() veya bindValue() metodu ile)

PDO::PARAM_INT : Integer (bindParam() veya bindValue() metodu ile)

PDO::PARAM_NULL : Null (bindParam() veya bindValue() metodu ile)

PDO::PARAM_STR : String (bindParam() veya bindValue() metodu ile)

PDOException : MySQL sunucusuna yeni bir bağlantı açma işlemi sırasında olası bir hata ile karşılaşması durumunda hatayı yakalamak için kullanılır.

php_ini_loaded_file() : PHP yapılandırma yönergelerinin bulunduğu php.ini dosyasının içerisinde bulunduğu dizin / dizinlerin adını / adlarını bularak, bulunduğu değeri geriye döndürür.

PHP_SELF : Çalışmakta olan PHP dosyasının yol tanımlama bilgisi değerini döndürür. Web sitesinin alan adını içermez. (\$_SERVER süper globali ile)

PHP_URL_FRAGMENT : Belirtilecek olan URL adresinin # sonrası değerini elde etmek için kullanılır. (parse_url() metodu ile)

PHP_URL_HOST : Belirtilecek olan URL adresinin domain bilgisi değerini elde etmek için kullanılır. (parse_url() metodu ile)

PHP_URL_PASS : Belirtilecek olan URL adresinin kullanıcı şifresi bilgisi değerini elde etmek için kullanılır. (parse_url() metodu ile)

PHP_URL_PATH : Belirtilecek olan URL adresinin path bilgisi değerini elde etmek için kullanılır. (parse_url() metodu ile)

PHP_URL_PORT : Belirtilecek olan URL adresinin port bilgisi değerini elde etmek için kullanılır. (parse_url() metodu ile)

PHP_URL_QUERY : Belirtilecek olan URL adresinin query bilgisi değerini elde etmek için kullanılır. (parse_url() metodu ile)

PHP_URL_SCHEME : Belirtilecek olan URL adresinin protokol bilgisi değerini elde etmek için kullanılır. (parse_url() metodu ile)

PHP_URL_USER : Belirtilecek olan URL adresinin kullanıcı adı bilgisi değerini elde etmek için kullanılır. (parse_url() metodu ile)

phpinfo() : PHP'nin o anki durumu hakkındaki bilgilerin ekran çıktılanmasını sağlamak için kullanılır.

phpversion() : Kullanılmakta olan server'da çalışmakta olan PHP sisteminin sürüm numarası değerini bularak, bulunduğu değeri geriye döndürür.

pi() : Pi sayısı değerini geriye döndürür.

pos() : Dizinin göstericisi konumundaki elemanı bulmak için kullanılır.

pow() : Herhangi bir sayının belirtilecek değer kadar üssünü bularak, bulunduğu değeri geriye döndürür.

preg_filter() : Kendisine parametre olarak verilen düzenli ifadeli değer / değerler doğrultusunda içeriği kontrol eder ve aranan değer / değerlerin eşleşmesi durumunda, eşleşen değer / değerlerin değişimini sağlayarak yeni bir içerik veya dizi oluşturarak, oluşturduğu içeriği veya diziyi geriye döndürür.

preg_grep() : Kendisine parametre olarak verilen düzenli ifadeli değer doğrultusunda dizi içeriğini kontrol eder ve aranan değer / değerlerin eşleşmesi durumunda, eşleşmeyi sağlayan içeriğin değer / değerlerinden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

preg_match() : Kendisine parametre olarak verilen düzenli ifadeli değer doğrultusunda içeriği kontrol eder ve aranan değer eşleşmesi durumunda, eşleşmeyi sağlayan içeriğin değerinden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

preg_match_all() : Kendisine parametre olarak verilen düzenli ifadeli değer doğrultusunda içeriği gelişmiş olarak kontrol eder ve aranan değer eşleşmesi durumunda, eşleşmeyi sağlayan içeriğin değerinden / değerlerinden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

preg_quote() : Kendisine parametre olarak verilen içeriği kontrol eder ve içerik dahilinde özel karakter / karakterlerin bulunması durumunda, bulunan özel karakter / karakterlerin önüne \ ekleyerek yeni bir içerik oluşturur ve oluşturduğu içeriği geriye döndürür.

preg_replace() : Kendisine parametre olarak verilen düzenli ifadeli değer / değerler doğrultusunda içeriği kontrol eder ve aranan değer / değerlerin eşleşmesi durumunda, eşleşen değer / değerlerin değişimini sağlayarak yeni bir içerik veya dizi oluşturarak, oluşturduğu içeriği veya diziyi geriye döndürür.

preg_split() : Kendisine paranetre olarak verilen düzenli ifadeli değer doğrultusunda içeriği kontrol eder ve eşleşmeyi sağlayan değer / değerlerden içeriği bölümleyerek yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

prepare() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya yapılacak işleme göre query hazırlamak için kullanılır.

preserveWhiteSpace : DOMDocument metodu ile oluşturulmuş ve loadXML() metodu ile nesne örneğine yüklenmiş olan içerikte WhiteSpace özelliğinin kullanılıp kullanılmayacağını belirtmek için kullanılır. (DOMDocument metodu ile)

prev() : Dizinin göstericisi konumundaki anahtarı veya elemanı bir geri almak için kullanılır.

print : Bir veya daha fazla içeriğin ekran çıktılanmasını sağlamak için kullanılır.

print() : Bir veya daha fazla içeriğin ekran çıktılanmasını sağlamak için kullanılır.

print_r() : Bir dizinin insanlarca okunabilecek gösteriminin ekran çıktılanmasını sağlamak için kullanılır.

printf() : Belirtilecek olan içeriğin, biçimli bir yapıya göre ekran çıktılanmasını sağlamak için kullanılır.

private : Özel. Yalnızca sınıf içerisinde erişilebilir.

property_exists() : Belirtilecek olan class içerisinde, belirtilecek olan özelliğin varlığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

protected : Korumalı. Sınıf içerisinde ve türetilen sınıflardan erişilebilir.

protocol_version : Sisteme tanımlı olan MySQL sunucusunun nesnesel yapı ile bağlantı protokolü sürüm bilgisini bularak, bulunduğu değeri geriye döndürür.

public : Genel. Her yerden erişilebilir.

putenv() : PHP yazılımı içerisinde kullanılabilecek ve sisteme tanımlı olan bir ortam değişkenine, belirtilecek olan değeri atamak için kullanılır.

QSA : GET metodu ile yönlendirme yapıldığında, yönlendirilen yere sonuna eklenen GET parametrelerini de dahil et demektir. (.htaccess ile)

query() : MySQL sunucusuna nesnesel yapı ile açılmış olan bağlantı dahilinde yeni bir sorgu işlemi yapmak için kullanılır. Ayrıca istenirse sorgu işlemi ile alakalı tüm bilgileri bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

query() : MySQL sunucusuna PDO ile açılmış olan bağlantı dahilinde yeni bir sorgu işlemi yapmak için kullanılır. Ayrıca istenirse sorgu işlemi sonucunda dönen tüm veri değerlerini bulur ve bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

QUERY_STRING : PHP dosyasına erişim esnasında kullanılan sorgu dizesi bilgisi değerini döndürür. Web sitesinin alan adını ve yol tanımlama bilgisini içermez. (\$_SERVER süper globali ile)

QUERY_STRING : PHP dosyasına erişim esnasında kullanılan sorgu dizesi bilgisi değerini döndürür. Web sitesinin alan adını ve yol tanımlama bilgisini içermez. (getenv() metodu ile)

quotemeta() : Belirtilecek olan içerikte bulunan, özel karakterlerin önüne \ yani kaçırma operatörü ekler ve sade bir metin içeriği oluşturarak, oluşturduğu değeri geriye döndürür.

R : Kullanılırsa browser'daki (tarayıcıdaki) URL değişir, kullanılmazsa yönlendirme arkaplanda yapılır fakat kullanıcının girdiği URL aynı kalır. (.htaccess ile)

r : Okuma amaçlı açar. (fopen() metodu ile)

r : RFC 2822 standartlarına uygun olacak şekilde tarih ve saat değerini döndürür. (date() metodu ile)

r+ : Hem okuma hemde yazma amaçlı açar. (fopen() metodu ile)

rad2deg() : Radyan değeri cinsinden herhangi bir sayının, derece değeri cinsinden karşılığını bularak, bulduğu değeri geriye döndürür.

rand() : Rastgele bir tam sayı üreterek, ürettiği değeri geriye döndürür.

range() : Belirtilecek olan değer aralıkları doğrultusunda doldurma işlemi yaparak yeni bir dizi oluşturmak için kullanılır.

rawurldecode() : Belirtilecek olan ve URL Encoding karakter kodlaması kullanılarak kodlanmış olan içeriği geri çözerek, çözdüğü değeri geriye döndürür.

rawurlencode() : Belirtilecek olan içeriğin URL Encoding karakter kodlamasını bularak, bulduğu değeri geriye döndürür.

readdir() : opendir() metodu ile açılmış ve belleğe alınmış olan dizini kontrol ederek, ilgili dizin içerisinde bulunan tüm alt dizin ve dosya bilgilerini (isimler ve varsa uzantılar) okuyarak, okuduğu tüm bilgileri depolar. Ayrıca depoladığı tüm bilgiler döngü yardımıyla elde edilebilir.

readfile() : Belirtilecek olan dosya veya URL içeriğini kontrol ederek ve bilgileri tek bir defada bularak, bulduğu değeri geriye döndürür. Ayrıca dosyalara download veya transfer özelliği atamak için kullanılır.

real_connect() : mysqli_init() metodu kullanılarak nesnesel yapı ile açılmış olan yeni MySQL bağlantısına, bağlantı bilgilerini atamak / tanımlamak için kullanılır.

real_escape_string() : Belirtilecek olan içerikte bulunabilecek olan tüm özel karakterlerin önüne nesnesel yapı ile \ işareti yani kaçırma operatörü ekler ve sade bir metin içeriği oluşturarak, oluşturduğu değeri geriye döndürür.

realpath() : Belirtilecek olan değer doğrultusunda, kaynağın mutlak yolunu bularak, bulduğu değeri geriye döndürür.

REMOTE_ADDR : Çalışmakta olan PHP dosyasını görüntüleyen kullanıcının IP adresi bilgisi değerini döndürür. (\$_SERVER süper globali ile)

REMOTE_ADDR : Çalışmakta olan PHP dosyasını görüntüleyen kullanıcının IP adresi bilgisi değerini döndürür. (getenv() metodu ile)

REMOTE_HOST : Çalışmakta olan PHP dosyasını görüntüleyen kullanıcının host başlığı bilgisi değerini döndürür. İlgili değer oluşturulabilmesi için Apache server'ı içerisinde "HostnameLookups On" tanımı muhakkak kullanılmalıdır. (\$_SERVER süper globali ile)

REMOTE_HOST : Çalışmakta olan PHP dosyasını görüntüleyen kullanıcının host başlığı bilgisi değerini döndürür. İlgili değer oluşturulabilmesi için Apache server'ı içerisinde "HostnameLookups On" tanımı muhakkak kullanılmalıdır. (getenv() metodu ile)

REMOTE_PORT : Çalışmakta olan PHP dosyasını görüntüleyen kullanıcı cihazının, web server ile haberleşmek için kullandığı port bilgisi değerini döndürür. (\$_SERVER süper globali ile)

REMOTE_PORT : Çalışmakta olan PHP dosyasını görüntüleyen kullanıcı cihazının, web server ile haberleşmek için kullandığı port bilgisi değerini döndürür. (getenv() metodu ile)

RENAME : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun adını değiştirmek / güncellemek için kullanılır. Ayrıca istenirse herhangi bir tabloyu içerisinde bulunduğu database'den başka bir database'e taşımak için de kullanılabilir.

rename() : Belirtilecek olan değer doğrultusunda, dosyayı yada dizini yeniden adlandırmak veya taşımak için kullanılır. Ayrıca istenirse işlem sonucunu boolean veri türünde döndürmek içinde kullanılır.

REPAIR : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun onarılması için kullanılır.

REPLACE : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun herhangi bir kaydını içeren satırındaki / satırlarındaki sütuna ait veriyi tam veya kısmi olarak değiştirmek için kullanılır.

REQUEST_METHOD : PHP dosyasına erişim için kullanılan istek yöntemi bilgisi değerini döndürür. (\$_SERVER süper globali ile)

REQUEST_METHOD : PHP dosyasına erişim için kullanılan istek yöntemi bilgisi değerini döndürür. (getenv() metodu ile)

REQUEST_TIME : PHP dosyasına erişim için başlatılan isteğin başlangıç zaman damgası bilgisi değerini döndürür. (\$_SERVER süper globali ile)

REQUEST_TIME_FLOAT : PHP dosyasına erişim için başlatılan isteğin mikro saniye değeri de dahil olmak üzere başlangıç zaman damgası bilgisi değerini döndürür. (\$_SERVER süper globali ile)

REQUEST_URI : Çalışmakta olan PHP dosyasının URI yol tanımlama bilgisi değerini döndürür. Web sitesinin alan adını ve PHP dosyası adını içermez. (\$_SERVER süper globali ile)

REQUEST_URI : Çalışmakta olan PHP dosyasının URI yol tanımlama bilgisi değerini döndürür. Web sitesinin alan adını ve PHP dosyası adını içermez. (getenv() metodu ile)

require() : PHP dosyasına, belirtilecek olan yeni bir PHP dosyasını dahil etmek / eklemek için kullanılır. (Yeni dosyanın daha önceden eklenip eklenmediği kontrol edilmez. Herhangi bir hata kodu dönecek olur ise kodlar okunmaya devam edilmez.)

require_once() : PHP dosyasına, belirtilecek olan yeni bir PHP dosyasını dahil etmek / eklemek için kullanılır. (Yeni dosyanın daha önceden eklenip eklenmediği kontrol edilir. Herhangi bir hata kodu dönecek olur ise kodlar okunmaya devam edilmez.)

reset() : Dizinin göstericisi konumundaki anahtarı veya elemanı varsayılan konuma (ilk) geri döndürmek için kullanılır.

return : Fonksiyondan değer döndürmek için kullanılır.

rewind() : Belirtilecek olan dosya veya URL içeriğinde değiştirilmiş olan okumaya hazır bulunan konumu sıfırlamak / varsayılan haline döndürmek için kullanılır.

RIGHT JOIN : MySQL sunucusundaki database içerisinde bulunan herhangi iki veya daha fazla tablonun sağdaki tabloya göre birbirleri ile ilişkilendirilmesi için kullanılır.

rmdir() : Belirtilecek olan değer doğrultusunda dizin silmek için kullanılır. Ayrıca istenirse işlem sonucunu boolean veri türünde döndürmek için kullanılır.

rollback() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya işlem yapılacağı anda daha önceden beginTransaction() metodu kullanılarak açılmış olan otomatik işleyen geçici işlem yapıları dahilinde işleme alınmış olan query'nin geçicilikten çıkarak işlemin geri alınması / iptal edilmesi için kullanılır.

rollback() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tabloya nesnesel yapı ile işlem yapılacağı anda daha önceden autocommit() metodu kullanılarak açılmış olan otomatik işleyen geçici işlem yapıları dahilinde işleme alınmış olan query'nin geçicilikten çıkarak işlemin geri alınması / iptal edilmesi için kullanılır.

round() : Herhangi bir ondalıklı sayıyı kendisine en yakın tam sayıyı kontrol ederek yukarıya veya aşağıya yuvarlayarak, yuvarlama sonucunda oluşan değeri geriye döndürür.

rowCount() : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun toplam kayıt sayısını bulmak veya işlem yapıldığında etkilenen kayıt sayısını bulmak için kullanılır.

rsort() : Dizinin elemanlarını büyük harf küçük harf duyarlı olacak şekilde z'den a'ya yada büyükten küçüğe sıralamak için kullanılır.

rtrim() : Belirtilecek olan içeriğin, varsa sonundaki / sağındaki tüm boşlukları, sekmeleri ve yeni satır karakterlerini silerek, oluşturduğu değeri geriye döndürür. Ayrıca belirtilecek olan içeriğin, varsa sonundaki / sağındaki, belirtilecek olan karakterleride silerek, oluşturduğu değeri geriye döndürür.

s : Değeri string veri türü şeklinde ele alarak, değer string veri türünde görüntülenmesini sağlar. (printf() veya sprintf() metodu ile)

S : Günün sayısal değeri için iki karakterli ingilizce sıralama takısı değerini döndürür. (j ile uyumlu çalışmaktadır.) (date() metodu ile)

s : Karakter diziliminin birden fazla satırı varsa tüm satırlarının karşılaştırmada kontrol edilip edilmeyeceğini tanımlamada kullanılır. (RegExp ile)

s : Saniyenin iki haneli olacak şekilde sayısal değerini döndürür. Tek haneli değerlerin başına sıfır ekler. (date() metodu ile)

s : string (bind_param() metodu ile)

Sabit : Girdiğimiz değerleri alan veya programın çalışması ile bazı değerlerin atandığı veri tutuculardır.

save() : DOMDocument metodu ile oluşturulmuş ve loadXML() metodu ile nesne örneğine yüklenmiş olan içeriğin belirtilecek olan dosyaya kaydedilmesini sağlamak için kullanılır. (DOMDocument metodu ile)

saveXML() : DOMDocument metodu ile oluşturulmuş ve loadXML() metodu ile nesne örneğine yüklenmiş olan içeriğin çıktılanmasını sağlamak için kullanılır. (DOMDocument metodu ile)

scandir() : Belirtilecek olan dizin içeriğini kontrol ederek, ilgili dizin içerisinde bulunan tüm alt dizin ve dosya bilgilerinden (isimler ve varsa uzantılar) yeni bir dizi oluşturarak, oluşturduğu değeri geriye döndürür.

SCRIPT_FILENAME : Çalışmakta olan PHP dosyasının server üzerinde bulunduğu mutlak yol tanımlama bilgisi değerini döndürür. (\$_SERVER süper globali ile)

SCRIPT_FILENAME : Çalışmakta olan PHP dosyasının server üzerinde bulunduğu mutlak yol tanımlama bilgisi değerini döndürür. (getenv() metodu ile)

SCRIPT_NAME : Çalışmakta olan PHP dosyasının server üzerinde bulunduğu yol tanımlama bilgisi değerini döndürür. (\$_SERVER süper globali ile)

SCRIPT_NAME : Çalışmakta olan PHP dosyasının server üzerinde bulunduğu yol tanımlama bilgisi değerini döndürür. (getenv() metodu ile)

select_db() : Kendisine parametre olarak verilen değer doğrultusunda, daha önceden nesnesel yapı ile açılmış olan MySQL bağlantısı üzerinden istenilen veritabanı tablosunu seçmek için kullanılır.

self:: : Kendi sınıfımda şeklinde sınıfı işaret etmek için kullanılır.

serialize() : Herhangi bir değeri saklanabilir veri türüne dönüştürerek, dönüştürdüğü değeri geriye döndürür.

SERVER_ADDR : PHP dosyasının çalıştığı server'ın IP adresi bilgisi değerini döndürür. (\$_SERVER süper globali ile)

SERVER_ADDR : PHP dosyasının çalıştığı server'ın IP adresi bilgisi değerini döndürür. (getenv() metodu ile)

SERVER_ADMIN : PHP dosyasının çalıştığı server'ın yönetici e-mail adresi bilgisi değerini döndürür. İlgili değer Apache server'ı içerisinde "ServerAdmin" yönergesinde tanımlıdır. (\$_SERVER süper globali ile)

SERVER_ADMIN : PHP dosyasının çalıştığı server'ın yönetici e-mail adresi bilgisi değerini döndürür. İlgili değer Apache server'ı içerisinde "ServerAdmin" yönergesinde tanımlıdır. (getenv() metodu ile)

server_info : Sisteme tanımlı olan MySQL sunucusunun nesnesel yapı ile detay bilgisini bularak, bulduğu değeri geriye döndürür.

SERVER_NAME : PHP dosyasının çalıştığı server'ın isim bilgisi değerini döndürür. İlgili değer Apache server'ı içerisinde "ServerName" yönergesinde tanımlıdır. (\$_SERVER süper globali ile)

SERVER_NAME : PHP dosyasının çalıştığı server'ın isim bilgisi değerini döndürür. İlgili değer Apache server'ı içerisinde "ServerName" yönergesinde tanımlıdır. (getenv() metodu ile)

SERVER_PORT : PHP dosyasının çalıştığı server'ın iletişim amacıyla kullanılan server port bilgisi değerini döndürür. İlgili değer Apache server'ı içerisinde "Listen" yönergesinde tanımlıdır. (\$_SERVER süper globali ile)

SERVER_PORT : PHP dosyasının çalıştığı server'ın iletişim amacıyla kullanılan server port bilgisi değerini döndürür. İlgili değer Apache server'ı içerisinde "Listen" yönergesinde tanımlıdır. (getenv() metodu ile)

SERVER_PROTOCOL : PHP dosyasına isteğinin yapıldığı protokolün isim ve sürüm bilgisi değerini döndürür. (\$_SERVER süper globali ile)

SERVER_PROTOCOL : PHP dosyasına isteğinin yapıldığı protokolün isim ve sürüm bilgisi değerini döndürür. (getenv() metodu ile)

SERVER_SIGNATURE : PHP dosyasının çalıştığı server'ın, server yazılımı ve diğer yazılımların sürümünü, sanal host adını ve server port bilgisi değerini döndürür. Bu bilgiler kısaca server imzası olarak adlandırılır. (\$_SERVER süper globali ile)

SERVER_SIGNATURE : PHP dosyasının çalıştığı server'ın, server yazılımı ve diğer yazılımların sürümünü, sanal host adını ve server port bilgisi değerini döndürür. Bu bilgiler kısaca server imzası olarak adlandırılır. (Eğer etkinse.) (getenv() metodu ile)

SERVER_SOFTWARE : PHP dosyasının çalıştığı server'ın yazılım bilgisi değerini döndürür. (\$_SERVER süper globali ile)

SERVER_SOFTWARE : PHP dosyasının çalıştığı server'ın yazılım bilgisi değerini döndürür. (getenv() metodu ile)

server_version : Sisteme tanımlı olan MySQL sunucusunun nesnesel yapı ile sürüm bilgisini bularak, bulduğu değeri geriye döndürür.

session_destroy() : Daha önceden oluşturulmuş olan session'ları silmek için kullanılır.

session_start() : session başlatmak veya session'ları takip etmek için kullanılır.

set_charset() : Kendisine parametre olarak verilen değer doğrultusunda, daha önceden nesnesel yapı ile açılmış olan MySQL bağlantısına varsayılan istemci karakter setini atamak / tanımlamak için kullanılır.

set_time_limit() : Çalışmakta olan PHP dosyası için belirtilecek olan değer doğrultusunda azami / maksimum çalışma süresi değeri atamak için kullanılır.

setcookie() : Belirtilecek olan değerler doğrultusunda cookie oluşturmak veya silmek için kullanılır.

setlocale() : Sistemin yerel ayarlarını belirtmek için kullanılır.

settype() : Herhangi bir değişken içeriğinin veri türünü belirlemek / değiştirmek için kullanılır.

sha1() : Belirtilecek olan içeriğin sha1 özetini hesaplayıp bularak, bulduğu değeri geriye döndürür.

sha1_file() : Belirtilecek olan dosyanın sha1 özetini hesaplayıp bularak, bulduğu değeri geriye döndürür.

SHOW COLUMNS : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun tüm sütunlarının listesini bulmak için kullanılır.

SHOW TABLES : MySQL sunucusundaki database içerisinde bulunan tüm tabloların listesini bulmak için kullanılır.

shuffle() : Dizi içerisinde bulunan elemanların sıralarının karıştırılması için kullanılır.

similar_text() : Belirtilecek olan iki değer arasındaki benzerliği yüzdesel olarak bularaki bulduğu değeri geriye döndürür.

simplexml_load_file() : Belirtilecek olan XML dosyası içeriği kodlama değerinin standart kodlama karşılığını bularak, bulduğu değeri geriye döndürür.

simplexml_load_string() : Belirtilecek olan XML kodlama değerinin standart kodlama karşılığını bularak, bulduğu değeri geriye döndürür.

SimpleXMLElement() : Belirtilecek olan içerikler dahilinde XML kodlama yapıları oluşturmak için kullanılır.

sin() : Herhangi bir sayının sinüsünü bularak, bulduğu değeri geriye döndürür.

sinh() : Herhangi bir sayının hiperbolik sinüsünü bularak, bulduğu değeri geriye döndürür.

sizeof() : Dizi içerisindeki eleman sayısını bulmak için kullanılır.

sleep() : Kullanıldığı satırdan sonraki tüm kodlamanın belirtilecek olan saniye kadar bekleterek, süre tamamlandığında çalıştırmak için kullanılır.

sort() : Dizinin elemanlarını büyük harf küçük harf duyarlı olacak şekilde a'dan z'ye yada küçükten büyüğe sıralamak için kullanılır.

SORT_ASC : A'dan Z'ye / Küçükten büyüğe (array_multisort() metodu ile)

SORT_DESC : Z'den A'ya / Büyükten küçüğe (array_multisort() metodu ile)

SORT_NATURAL : Alfabetik sıralama (String Doğal) (array_multisort() metodu ile)

SORT_NUMERIC : Rakamsal sıralama (array_multisort() metodu ile)

SORT_REGULAR : Standart sıralama (array_multisort() metodu ile)

SORT_STRING : Alfabetik sıralama (String) (array_multisort() metodu ile)

spl_autoload_register() : Döküman içerisinde belirtilecek olan sınıfın bulunmaması durumunda, belirtilecek olan parametreye göre bir sınıf dökümanını otomatik olarak belgeye dahil etmek / yüklemek için kullanılır.

sprintf() : Belirtilecek olan içeriğin, biçimli bir yapıya göre yeni bir değişken içerisine depolanmasını sağlamak için kullanılır.

sqrt() : Herhangi bir sayının karekökünü bularak, bulunduğu değeri geriye döndürür.

stat() : Belirtilecek olan değer doğrultusunda, ilgili kaynağın dosya hakkındaki tüm bilgilerini bularak, bulunduğu değerlerden yeni bir dizi oluşturarak, oluşturduğu değeri geriye döndürür.

static : Sabit. Sınıf içerisindeki herhangi bir özellik veya metoda sınıf çağırılmadan erişilebilir.

str_ireplace() : Belirtilecek olan içerikte, belirtilecek olan değer / değerleri büyük harf / küçük harf ayrımı olmaksızın arayarak, eşleşen değer / değerler olması durumunda, belirtilecek olan değer / değerlerle değiştirerek, değiştirilmiş olan değeri geriye döndürür. Ayrıca belirtilecek olan ve eşleşen tüm değer / değerlerin sayısının da geriye döndürebilir.

str_pad() : Belirtilecek olan içeriğin, belirtilecek olan değer / değerler doğrultusunda, başına, sonuna veya hem başına hemde sonuna değer ekleyerek, eklenmiş olan değeri geriye döndürür.

STR_PAD_BOTH : Belirtilecek olan içeriğin hem başına / soluna, hemde sonuna / sağına değer eklemek için kullanılır. (str_pad() metodu ile)

STR_PAD_LEFT : Belirtilecek olan içeriğin başına / soluna değer eklemek için kullanılır. (str_pad() metodu ile)

STR_PAD_RIGHT : Belirtilecek olan içeriğin sonuna / sağına değer eklemek için kullanılır. (str_pad() metodu ile)

str_repeat() : Belirtilecek olan içeriği, belirtilecek olan sayı değeri kadar tekrarlayarak, oluşturulmuş olan değeri geriye döndürür.

str_replace() : Belirtilecek olan içerikte, belirtilecek olan değer / değerleri arayarak, eşleşen değer / değerler olması durumunda, belirtilecek olan değer / değerlerle değiştirerek, değiştirilmiş olan değeri geriye döndürür. Ayrıca belirtilecek olan ve eşleşen tüm değer / değerlerin sayısının da geriye döndürebilir.

str_rot13() : Belirtilecek olan içeriğin ROT13 kodlaması değerini bularak, bulunduğu değeri geriye döndürür. Ayrıca ROT13 kodlaması kullanılarak kodlanmış olan içeriği geri çözerek, çözdüğü değeri geriye döndürür.

str_shuffle() : Belirtilecek olan içeriğin tüm karakterlerini rastgele karıştırarak, karıştırılmış olan değeri geriye döndürür.

str_split() : Belirtilecek olan içeriği, belirtilecek olan karakter sayısına göre bölerek, bölümlleme işlemi sonucunda oluşan değerlerden yeni bir dizi oluşturarak, oluşturduğu değeri geriye döndürür.

str_word_count() : Belirtilecek olan içeriğin kelime sayısını bularak, bulunduğu değeri geriye döndürür.

strchr() : Belirtilecek olan içerikte, belirtilecek olan değer doğrultusunda arama yaparak, eşleşen değer olması durumunda, ilk eşleşen değer öncesini veya sonrasını bularak, bulunduğu değeri geriye döndürür.

strftime() : Yerel tarih ve saat bilgilerini belirtilecek olan formata göre biçimlendirerek, biçimlendirdiği değeri geriye döndürür.

strip_tags() : Belirtilecek olan içerikte bulunabilecek olan tüm HTML kodlarını temizleyerek, temizlenmiş olan değeri geriye döndürür.

stripslashes() : Belirtilecek olan içerikte bulunan ve önüne \ yani kaçırma operatörü eklenmiş olan tüm közel arakterlerin önünde bulunan \ yani kaçırma operatörlerini kaldırır ve aktif bir metin içeriği oluşturarak, oluşturduğu değeri geriye döndürür.

stripos() : Belirtilecek olan içerikte, belirtilecek olan herhangi bir değer büyük harf / küçük harf ayrımı olmaksızın ilk eşleşme sonucundaki konumunu / sırasını bularak, bulunduğu değeri geriye döndürür.

stripslashes() : Belirtilecek olan içerikte bulunan ve önüne \ yani kaçırma operatörü eklenmiş olan tüm karakterlerin önünde bulunan \ yani kaçırma operatörlerini kaldırır ve aktif bir metin içeriği oluşturarak, oluşturduğu değeri geriye döndürür.

stristr() : Belirtilecek olan içerikte, belirtilecek olan değer doğrultusunda herhangi bir değer büyük harf / küçük harf ayrımı olmaksızın arama yaparak, eşleşen değer olması durumunda, ilk eşleşen değer öncesini veya sonrasını bularak, bulunduğu değeri geriye döndürür.

strlen() : Belirtilecek olan içeriğin karakter sayısını bularak, bulunduğu değeri geriye döndürür.

strpbrk() : Belirtilecek olan içeriği, belirtilecek olan ayraç doğrultusunda kontrol ederek, eşleşen ayraç değeri olması durumunda, ilk eşleşen ayraç sonrasında bulunan değeri bularak, bulunduğu değeri geriye döndürür.

strpos() : Belirtilecek olan içerikte, belirtilecek olan herhangi bir değer ilk eşleşme sonucundaki konumunu / sırasını bularak, bulunduğu değeri geriye döndürür.

strrchr() : Belirtilecek olan içerikte, belirtilecek olan değer doğrultusunda arama yaparak, eşleşen değer olması durumunda, son eşleşen değer sonrasını bularak, bulunduğu değeri geriye döndürür.

strrev() : Belirtilecek olan içeriği sondan başa olmak üzere ters çevirerek, çevrilmiş olan değeri geriye döndürür.

stripos() : Belirtilecek olan içerikte, belirtilecek olan herhangi bir değer büyük harf / küçük harf ayrımı olmaksızın son eşleşme sonucundaki konumunu / sırasını bularak, bulunduğu değeri geriye döndürür.

strrpos() : Belirtilecek olan içerikte, belirtilecek olan herhangi bir değer son eşleşme sonucundaki konumunu / sırasını bularak, bulunduğu değeri geriye döndürür.

strstr() : Belirtilecek olan içerikte, belirtilecek olan değer doğrultusunda arama yaparak, eşleşen değer olması durumunda, ilk eşleşen değer öncesini veya sonrasını bularak, bulunduğu değeri geriye döndürür.

strtok() : Belirtilecek olan içeriği, belirtilecek olan ayraç doğrultusunda kontrol ederek, eşleşen ayraç değeri olması durumunda, ilk eşleşen ayraç öncesinde bulunan değeri bularak, bulunduğu değeri geriye döndürür. Ayrıca eşleşen tüm ayraçların öncesinde bulunan değerlerde herhangi bir döngü yardımı ile tek tek elde edilebilir.

strtolower() : Belirtilecek olan içeriğin tüm harflerini küçük harf olacak şekilde değiştirerek, değiştirilmiş olan değeri geriye döndürür.

strtotime() : İngilizce metin içerikli bir zamanın Unix zaman damgası değerini bularak, bulunduğu değeri geriye döndürür.

strtoupper() : Belirtilecek olan içeriğin tüm harflerini büyük harf olacak şekilde değiştirerek, değiştirilmiş olan değeri geriye döndürür.

strtr() : Belirtilecek olan içerikte, belirtilecek olan dizi türündeki değer / değerleri arayarak, eşleşen değer / değerler olması durumunda, belirtilecek olan değer / değerlerle değiştirerek, değiştirilmiş olan değeri geriye döndürür.

substr() : Belirtilecek olan içerikte, belirtilecek olan karakter sıra numarası aralıklarına göre istenilen değeri bularak, bulunduğu değeri geriye döndürür.

substr_count() : Belirtilecek olan herhangi bir değerin, belirtilecek olan içerikte kaç tane olduğunu bularak, bulunduğu değeri geriye döndürür.

substr_replace() : Belirtilecek olan içerikte, belirtilecek olan karakter sıra numarası aralıklarına göre, istenilen değer / değerleri ekleyerek / silerek / değiştirerek, yeni oluşturulmuş olan değeri geriye döndürür.

SUM : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun belirtilecek olan sütununun tüm kayıt değerlerinin toplamı bulmak için kullanılır.

SUNFUNCS_RET_DOUBLE : Sonuç double veri türünde değer döndürür. (date_sunrise() veya date_sunset() metodu ile)

SUNFUNCS_RET_STRING : Sonuç string veri türünde değer döndürür. (date_sunrise() veya date_sunset() metodu ile)

SUNFUNCS_RET_TIMESTAMP : Sonuç zaman damgası olarak döner. (date_sunrise() veya date_sunset() metodu ile)

sunrise : Gün doğumu (date_sun_info() metodu ile)

sunset : Gün batımı (date_sun_info() metodu ile)

switch : Koşul başlat.

sys_get_temp_dir() : Kullanılmakta olan server'da çalışmakta olan PHP sisteminde kullanılan temp dizinin path değerini bularak, bulunduğu değeri geriye döndürür.

t : Ayın toplam gün sayısı değerini döndürür. (date() metodu ile)

T : Tüm tarih ve saat işlevleri tarafından kullanılan varsayılan geçerli zaman dilimi kısaltması değerini döndürür. Türkiye'nin zaman dilimi değeri : Central European Summer Time (date() metodu ile)

tan() : Herhangi bir sayının tanjantını bularak, bulunduğu değeri geriye döndürür.

tanh() : Herhangi bir sayının hiperbolik tanjantını bularak, bulunduğu değeri geriye döndürür.

this : Bu operatörü

throw : Hata oluşturabilecek kodlar için istisnai durumlarda kullanılır.

time() : Geçerli zamanın Unix zaman damgası değerini bularak, bulunduğu değeri geriye döndürür.

time_nanosleep() : Kullanıldığı satırdan sonraki tüm kodlamanın belirtilecek olan saniye ve nanosaniye kadar bekleterek, süre tamamlandığında çalıştırmak için kullanılır.

time_sleep_until() : Kullanıldığı satırdan sonraki tüm kodlamanın belirtilecek olan Unix zaman damgası geçerli zamanına kadar bekleterek, süre tamamlandığında çalıştırmak için kullanılır.

timezone_identifiers_list() : Tüm zaman dilimi tanımlıyıcıları değerlerinden yeni bir dizi oluşturarak, oluşturduğu dizi değerini geriye döndürür.

touch() : Belirtilecek olan değer doğrultusunda, yeni bir dosya oluşturmak için kullanılır. Ayrıca istenirse işlem sonucunu boolean veri türünde döndürmek içinde kullanılır.

trait : Sınıflar için özellik ve metod gruplarının oluşturulması için kullanılır.

trait_exists() : Belirtilecek olan class'a ait, belirtilecek olan trait'in varlığını sınamak için kullanılır. İşlem sonucu daima boolean veri türünde döner.

transit : Öğle (date_sun_info() metodu ile)

TRANSLIT : Belirtilecek olan içerikte gösterilmeyen bir karaktere / karakterlere rastlanacak olursa, ilgili karakterin / karakterlerin yerine benzer bir karakter / kural dışı karakter değeri yerleştirmek için kullanılır. (iconv() metodu ile)

trim() : Belirtilecek olan içeriğin, varsa başındaki ve sonundaki tüm boşlukları, sekmeleri ve yeni satır karakterlerini silerek, oluşturduğu değeri geriye döndürür. Ayrıca belirtilecek olan içeriğin, varsa başındaki ve sonundaki, belirtilecek olan karakterleride silerek, oluşturduğu değeri geriye döndürür.

TRUNCATE TABLE : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun tüm içeriklerini boşaltmak / sıfırlamak için kullanılır.

try : Hata oluşturabilecek kodlar için kullanılır.

U : Geçerli zamanın Unix zaman damgası değerini döndürür. (date() metodu ile)

u : Karakter diziliminin karşılaştırılmasında Unicode tanımlaması kullanılır. (RegExp ile)

ucfirst() : Belirtilecek olan içeriğin ilk harfini büyük harf olacak şekilde değiştirerek, değiştirilmiş olan değeri geriye döndürür.

ucwords() : Belirtilecek olan içerikte bulunan tüm kelimelerin ilk harflerini büyük harf olacak şekilde değiştirerek, değiştirilmiş olan değeri geriye döndürür.

uniqid() : Mikrosaniye cinsinden geçerli zamana dayalı olarak belirtilen örnek ile benzersiz / eşsiz bir değer üreterek / ürettiği değeri geriye döndürür.

unlink() : Belirtilecek olan değer doğrultusunda, dosya silmek için kullanılır. Ayrıca istenirse işlem sonucunu boolean veri türünde döndürmek içinde kullanılır.

unserialize() : serialize() metodu kullanılarak üretilmiş olan herhangi bir saklanabilir veri türündeki değeri eski orjinal haline dönüştürerek, dönüştürdüğü değeri geriye döndürür.

unset() : Belirtilecek olan değişkeni, dizi anahtarını ve elemanını veya session'ı silmek / yok etmek için kullanılır.

UPDATE : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun herhangi bir kaydını içeren satırındaki sütuna / sütunlara ait veriyi / verileri güncellemek için kullanılır.

urldecode() : Belirtilecek olan ve URL Encoding karakter kodlaması kullanılarak kodlanmış olan içeriği geri çözerek, çözdüğü değeri geriye döndürür.

urlencode() : Belirtilecek olan içeriğin URL Encoding karakter kodlamasını bularak, bulunduğu değeri geriye döndürür.

use : Daha önceden tanımlanmış olan herhangi bir özellik veya metod grubunu, ilgili sınıfa tanımlamak / uygulamak için kullanılır.

USING : MySQL sunucusundaki database içerisinde bulunan herhangi iki veya daha fazla tablonun birbirleri ile ilişkilendirilmesi sırasında ortak sütun koşulunu tanımlamak / belirtmek için kullanılır. (JOIN ifadeleri ile)

usleep() : Kullanıldığı satırdan sonraki tüm kodlamanın belirtilecek olan mikrosaniye kadar bekleterek, süre tamamlandığında çalıştırmak için kullanılır.

var : Herhangi bir sınıf içerisinde standart değişken tanımlamak için kullanılır.

var_dump() : Herhangi bir değişkene ait tüm yapıya ve bilgilere ulaşmak için kullanılır.

var_export() : Belirtilecek olan herhangi bir değişkenin çözümlenebilir gösterimini bularak, bulunduğu gösterimin ekran çıktılanmasını sağlamak için kullanılır.

w : Haftanın gününün sayısal değerini döndürür. 0'dan (pazar) 6'ya (cumartesi) kadar. (date() metodu ile)

W : ISO-8601 standartlarına uygun olacak şekilde yılın haftasının sayısal değerini döndürür. (Hafta başlangıcı pazartesi) (date() metodu ile)

w : Yazma amaçlı açar. Dosya verisini boşaltarak yeniden yazar. Dosya mevcut değilse oluşturulur. (fopen() metodu ile)

w+ : Hem okuma hemde yazma amaçlı açar. Dosya verisini boşaltarak yeniden yazar. Dosya mevcut değilse oluşturulur. (fopen() metodu ile)

WHERE : MySQL sunucusundaki database içerisinde bulunan herhangi bir tablonun işlem esnasında tüm verilerini işlemek yerine sadece koşula bağlı verilerin işlemek istenildiğini belirtmek için kullanılır.

while : Döngü işlemi başlatır.

wordwrap() : Belirtilecek olan içeriğe, belirtilecek olan karakter sıra numarası aralıklarına göre, belirtilecek olan değer / değerleri ekleyerek, eklenmiş olan değeri geriye döndürür.

X : Değeri tam sayı şeklinde ele alarak, değerin büyük harfli hexadecimal sayı sisteminde görüntülenmesini sağlar. (printf() veya sprintf() metodu ile)

x : Değeri tam sayı şeklinde ele alarak, değerin küçük harfli hexadecimal sayı sisteminde görüntülenmesini sağlar. (printf() veya sprintf() metodu ile)

x : Karakter diziliminin karşılaştırılmasında desende bulunan boşluğu etkisiz hale getirmek için kullanılır. (RegExp ile)

x : Yazma amaçlı açar. Dosya mevcut değilse fopen() metodu false değeri döndürerek hata üretir. (fopen() metodu ile)

x+ : Hem okuma hemde yazma amaçlı açar. Dosya mevcut değilse fopen() metodu false değeri döndürerek hata üretir. (fopen() metodu ile)

xmlVersion : DOMDocument metodu ile oluşturulmuş ve loadXML() metodu ile nesne örneğine yüklenmiş olan içeriğin belge türü belirtiminde kullanılacak olan XML sürümünü / versiyonunu belirtmek için kullanılır. (DOMDocument metodu ile)

xor : Ayrıcıklı veya operatörü

xpath() : Belirtilecek olan XML kodlama değeri içerisinde bulunan, belirtilecek olan değer / değerler doğrultusunda eşleşen tüm elemanları / elementleri işaret etmek ve seçmek için kullanılır.

Y : Yılın dört haneli olacak şekilde sayısal değerini döndürür. (date() metodu ile)

y : Yılın iki haneli olacak şekilde sayısal değerini döndürür. (date() metodu ile)

yield : Fonksiyonlardan değer döndürmek için kullanılır.

z : Yılın gününün sayısal değerini döndürür. 0'dan (1. gün) 365'e (366. gün / artık gün) kadar. (date() metodu ile)

zend_version() : Kullanılmakta olan server'da çalışmakta olan zend motorunun sürüm numarası değerini bularak, bulunduğu değeri geriye döndürür.