

PHP File Cache Sınıfım

14 Nisan Salı 15

<http://selcuk.in/php-file-cache-sinifim.html>

Merhaba arkadaşlar, En son yazımı 8 Ekim 2014 de yazmışım. Biraz daha dursam neredeyse bir yıl olacakmış. Fazla boş bırakmayayım dedim ve bunu da yararlı bir konu ile yapayım. Bugün bir projemde file cache gerekiyordu. Bende onun için basit bir sınıf yazdım ve bunu da paylaşmayı düşündüm. İleride ki zamanlarda da bunun geliştirmesini yapacağım. Kısaca kullanımından bahsedecek olursak;

sCache Sınıf Kullanımı

Cache Ayarları (Opsiyonel) Bu ayarlar opsiyoneldir, boş bırakılırsa default değerler geçerli sayılacaktır. Comment şeklinde default değerler verilmiştir.

```
<?php
/* settings();
  sCache::settings(array(
    '_path' => 'cache',
    '_name' => 'default',
    '_extension' => 'scache'
  ));
*/

// örnek
sCache::settings(array(
  '_path' => 'cache',
  '_name' => 'cache',
  '_extension' => '.selco'
));
?>
```

Cache Tanımlama Bu fonksiyon ile bir değer altında belleğe alınacak değeri belirliyoruz.

```
<?php
/* setCache();
  sCache::setCache($key, $data, $writing = FALSE);
  NOT: Daha önce tanımlanmış bir cache üstüne yazmak için $writing değeri TRUE yapınız.
*/

// örnek
try {

  $boolean = sCache::setCache('test', 'test içerik');
  echo ($boolean ? 'cache tanımlandı.' : 'cache tanımlanamadı.');

} catch (Exception $e )
{
  echo $e->getMessage(); // Hata oluşursa ekrana yaz.
}
?>
```

Tanımlanan Cache Ulaşma Bu fonksiyon ile tanımlanmış bir cache değerine ulaşip onun değerini alıyoruz. Verilecek değer **setCache()**; de tanımlanan değer olmalıdır.

```
<?php
/* getCache();
  sCache::getCache($key);
*/

// örnek
try {

  $string = sCache::getCache('test');
  echo $string;

} catch (Exception $e )
{
  echo $e->getMessage(); // Hata oluşursa ekrana yaz.
}
?>
```

```
}  
?>
```

Cache Yoluna Ulaşma Bazı durumlarda cache yapılan dosyaya ulaşılmak istenirse diye cache yolunu alma fonksiyonumuz.

```
<?php  
/* getCachedDir();  
sCache::getCachedDir($key);  
*/  
  
// Örnek  
try {  
  
    $string = sCache::getCachedDir('test');  
    echo $string;  
  
} catch (Exception $e )  
{  
    echo $e->getMessage(); // Hata oluşursa ekrana yaz.  
}  
?>
```

Cache Kontrol Set edilen bir cache istemsiz bir durumda silinmiş olabilir. Bu gibi durumlar için cache kontrol fonksiyonumuz.

```
<?php  
/* isCached();  
sCache::isCached($key);  
*/  
  
// Örnek  
try {  
  
    $boolean = sCache::isCached('test');  
    echo ($boolean) ? 'Cache var' : 'Cache yok';  
  
} catch (Exception $e )  
{  
    echo $e->getMessage(); // Hata oluşursa ekrana yaz.  
}  
?>
```

Cache Silme Bir değere ait cache kayıtlarını silme fonksiyonumuz.

```
<?php  
/* destroyCache();  
sCache::destroyCache($key);  
*/  
  
// Örnek  
try {  
  
    $boolean = sCache::destroyCache('test');  
    echo ($boolean) ? 'Cache silindi' : 'Cache silinemedi.';  
  
} catch (Exception $e )  
{  
    echo $e->getMessage(); // Hata oluşursa ekrana yaz.  
}  
?>
```

Tüm Cache Dosyalarının Silinmesi Tüm cache kayıtlarının tamamen silinme fonksiyonumuz.

```
<?php  
/* destroyAllCaches();  
sCache::destroyAllCaches();  
*/  
  
// Örnek  
try {  
  
    $boolean = sCache::destroyAllCaches();  
    echo ($boolean) ? 'Tüm kayıtlar silindi.' : 'Kayıtlar silinemedi.';  
  
} catch (Exception $e )  
{  
    echo $e->getMessage(); // Hata oluşursa ekrana yaz.  
}  
}
```

?>

İsteyenler konunun ek dosyalar kısmından sınıfı indirebilir ya da github [link](#)inden indirebilir.

Bu yazı <http://selcuk.in/php-file-cache-sinifim.html> adresinden indirilmiştir