

Data Caching

Each cache type has its own class of the framework. All these classes are inherited from common abstract class **ClickBlocks\Cache\Cache** and therefore have single interface of interaction with cache.

General operations

The main operations of the cache are: checking of the availability of a specified type of cache, write to the cache, read from the cache, deleting data from the cache, the cache clean up.

To determine the availability of the given cache type you can use static method **isAvailable()**:

```
// Checks availability of the file cache
// (always returns TRUE).
echo (int)\Cache\File::isAvailable();
// Checks availability of the cache APC.
echo (int)\Cache\APC::isAvailable();
// Checks availability of the cache Memcache.
echo (int)\Cache\Memory::isAvailable();
// Checks availability of the cache Redis.
echo (int)\Cache\Redis::isAvailable();
```

Adding data in the cache, reading and removing data from the cache, and verification of the cached data occur as follows:

```
// Adds data in the cache for a hour.
// Identifier of the cached data is "foo".
$cache->set('foo', 'some data', 3600);
// Reads data from the cache.
echo $cache->get('foo');
// Checks whether the cached data is not expired.
if (!$cache->isExpired('foo'))
{
    // If the cached data is not expired then we remove them.
    $cache->remove('foo');
}
// Cleans all cached data.
$cache->clean();
```

For the file cache method **get()** always returns the previously cached data even if they are expired. To remove expired data you should use method **gc()**.

Groups

It is possible to associate the data passed to the cache with some identifier (in addition to the unique data identifier) is called a group. Many heterogeneous data having a common group can be stored in the cache.

We can read and remove group data:

```
// Adds numbers from 1 to 10 in group "test" for a minute.
foreach (range(1, 10) as $n) $cache->set('var' . $n, $n, 60, 'test');
// Gets all data of group "test" as an associative array.
print_r($cache->getByGroup('test'));
// Removes all data from group "test".
$cache->cleanByGroup('test');
```

To get group list and identifiers of group data it is enough to call the appropriate method:

```
print_r($cache->getVault());
```

When you remove some cache data that associated with a group, the appropriate cache key won't be automatically removed from the vault. In this case you should invoke method **normalizeVault()**. (Method **gc()** also supports the vault normalization).

