

71. 缓存功能

学习要点：

1. 缓存功能

本节课我们来学习一下系统提供的缓存功能，实现缓存的写入和读取。

一. 缓存功能

1. 系统内置了很多类型的缓存，除了 **File**，其它均需要结合相关产品；
2. 我们这里主要演示 **File** 文本缓存，其它的需要学习相关产品；
3. 配置文件 **cache.php** 进行缓存配置，默认生成在 **runtime/cache** 目录；
4. **::set()**方法，可以设置一个缓存，参数三为过期时间；

```
Cache::set('user', 'Mr.Lee');  
Cache::set('user', 'Mr.Lee', 3600);
```
5. **::has()**方法，判断缓存是否存在，返回布尔值；

```
Cache::has('user');
```
6. **::get()**方法，从缓存中获取到相应的数据，无数据返回 **null**；

```
Cache::get('user');
```
7. **::inc()**和**::dec()**实现缓存数据的自增和自减操作；

```
Cache::inc('num');  
Cache::inc('price', 3);  
Cache::dec('num');  
Cache::dec('price', 3);
```
8. **::push()**实现缓存的数组数据追加的功能；

```
Cache::set('arr', [1,2,3]);  
Cache::push('arr', 4);
```
9. **::delete()**方法，可以删除指定的缓存文件；

```
Cache::delete('user');
```
10. **::pull()**方法，先获取缓存值，然后再删除掉这个缓存，无数据返回 **null**；

```
Cache::pull('user');
```
11. **::remember()**方法，如果数据不存在，则写入数据，可以依赖注入；

```
Cache::remember('start_time', time());  
Cache::remember('start_time', function (Request $request) {})
```

12. `::clear()`方法，可以清除所有缓存；

```
Cache::clear();
```

13. `::tag()`标签，可以将多个缓存归类到标签中，方便统一管理，比如清除；

```
Cache::tag('tag')->set('user', 'Mr.Lee');
```

```
Cache::tag('tag')->set('age', 20);
```

```
Cache::tag('tag')->clear();
```

14. 助手函数的使用：`cache()`；

```
//设置缓存
```

```
cache('user', 'Mr.Lee', 3600);
```

```
//输出缓存
```

```
echo cache('user');
```

```
//删除指定缓存
```

```
cache('user', null);
```