62. Session

学习要点：

1.Session

2.二维和助手函数

本节课我们来学习一下系统提供的 Session 存储功能，以及二维和助手函数。

一. Session

1. Session 第一次调用时，会按照 config/session.php 进行初始化；

```
return [
    // SESSION 前缀
    'prefix'         => 'think',
    // 是否自动开启 SESSION
    'auto_start'     => true,
];
```

2. 从上面截取配置内容可以了解到，名称前缀定义了 think，并且自动开启；
3. Session 是服务器的内容存储，而相对的客户端的是 Cookie，基础知识不再赘述；
4. 系统也支持通过方法调用的形式进行初始化，比如::init()方法；

```
Session::init([
    'prefix'         => 'tp',
    'auto_start'     => true,
]);
```

5. 所有配置参数，如下表：

参数	描述
type	session类型
expire	session过期时间
prefix	session前缀
auto_start	是否自动开启
use_trans_sid	是否使用use_trans_sid
var_session_id	请求session_id变量名
id	session_id
name	session_name
path	session保存路径
domain	session cookie_domain
use_cookies	是否使用cookie
cache_limiter	session_cache_limiter
cache_expire	session_cache_expire
secure	安全选项
httponly	使用httponly

6. 直接使用::set()和::get()方法去设置 Session 的存取；

```
Session::set('user', 'Mr.Lee');
echo Session::get('user');
dump(Request::session());
echo $_SESSION['think']['user'];
```

7. ::set()赋值中，第三个参数可以设置前缀，即 Session 作用域；

```
Session::set('user', 'Mr.Lee', 'tp');
```

8. ::get()取值时，第二参数可强调作用域，不写则默认当前作用域；

```
Session::get('user', 'tp');
```

9. ::has()判断当前作用域 session 是否赋值，第二参数可以指定作用域；

```
Session::has('user');
Session::has('user', 'tp');
```

10. ::delete()删除，默认删除当前作用域的值，第二参数可以指定作用域；

```
Session::delete('user');
Session::delete('user', 'tp');
```

11. ::prefix()方法，可以指定当前作用域；

```
Session::prefix('tp');
```

12. ::pull()方法，取出当前的值，并删除掉这个 session，不存在返回 null；

```
echo Session::pull('user');
```

13. ::clear()方法，清除当前作用域的 session，可指定作用域；

```
Session::clear('');
Session::clear('think');
```

14. ::flash()方法，设置闪存数据，只请求一次有效的情况，再请求会失效；

```
Session::flash('user','Mr.Lee');
```

15. ::flush()方法，可以清理闪存数据的有效 session；

```
Session::flush();
```

二. 二维和助手函数

1. 二维操作，就是对象和数组的调用方式，如下：

```
// 赋值（当前作用域）
Session::set('obj.user','Mr.Lee');
// 判断（当前作用域）是否赋值
Session::has('obj.user');
```

```
// 取值（当前作用域）
Session::get('obj.user');
// 删除（当前作用域）
Session::delete('obj.user');
```

2. 助手函数，更加方便操作，如下：

```
//赋值
session('user', 'Mr.Wang');
//带作用域赋值
session('user', 'Mr.Lee', 'tp');
//has 判断
session('?user');
//delete 删除
session('user', null);
//清理
session(null);
//带作用域清理
session(null, 'tp');
//带作用域输出
echo session('user', '', 'tp');
//输出
echo session('user');
```