

## 19. 数据库的事务和获取器

学习要点：

1. 事务处理
2. 获取器

本节课我们要学习查询中的事务处理，和数据库获取器的使用方法。

### 一. 事务处理

1. 数据库的表引擎需要是 InnoDB 才可以使用，如果不是调整即可；
2. 事务处理，需要执行多个 SQL 查询，数据是关联恒定的；
3. 如果成功一条查询，改变了数据，而后一条失败，则前面的数据回滚；
4. 比如：蜡笔小新给路飞 3 块钱，自己-3，对方+3，这时需要事务处理；
5. 系统提供了两种事务处理的方式，第一种是自动处理，出错自动回滚；

```
Db::transaction(function () {  
    Db::name('user')->where('id', 19)->save(['price'=>Db::raw('price - 3')]);  
    Db::name('user1')->where('id', 20)->save(['price'=>Db::raw('price + 3')]);  
});
```

6. 手动处理，基本和原生处理类似，可以自行输出错误信息；

```
//启动事务  
Db::startTrans();  
try {  
    Db::name('user')->where('id', 19)->save(['price'=>Db::raw('price - 3')]);  
    Db::name('user1')->where('id', 20)->save(['price'=>Db::raw('price + 3')]);  
    //提交事务  
    Db::commit();  
} catch (\Exception $e) {  
    echo '执行 SQL 失败！';  
    //回滚  
    Db::rollback();  
}
```

### 二. 获取器

1. 获取器的意思就是：将数据的字段进行转换处理再进行操作；
2. 比如在获取数据列表的时候，将获取到的邮箱字段全部大写；

```
$user = Db::name('user')->withAttr('email', function ($value, $data) {  
    return strtoupper($value);  
})->select();  
  
return json($user);
```