

Welcome to Technology arena

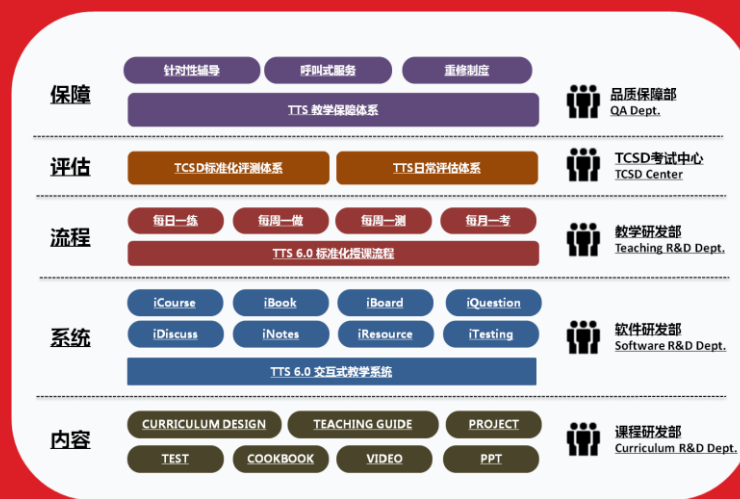
达内精品课程



集团教学研发部
2013-12

TTS6.0

Online



讲师简介



李翊 达内集团教研副总裁

曾任IBM中国研发中心系统架构师，WBI高级项目经理。

曾经主持研发：

- 某省级电信计费系统
- 国务院办公厅信息系统
- ePod电子政务基础平台
- 多个基于WBI的系统集成方案
- 国有软件基础平台集成测试系统
- 基础目录体系交换系统

等大型项目。

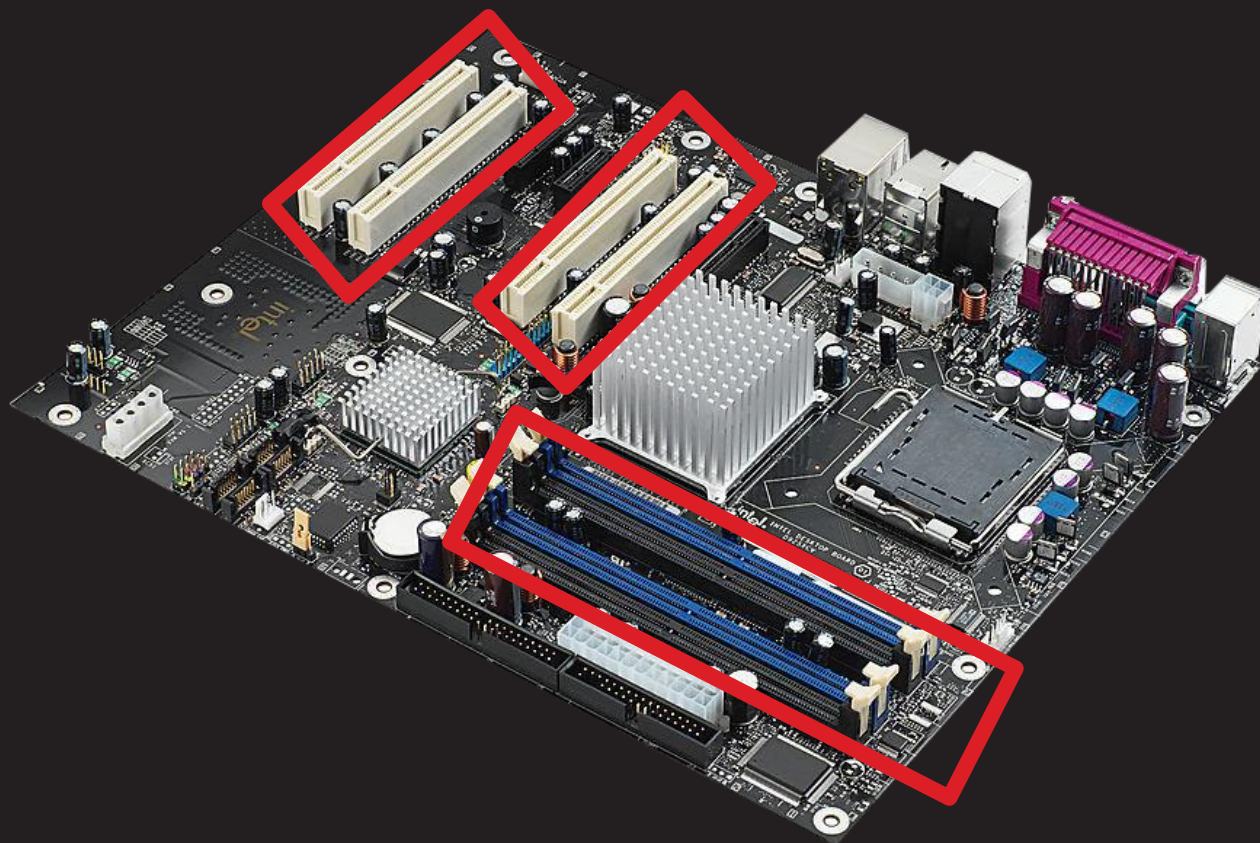
面向对象原理(3)

接口的作用——“契约”

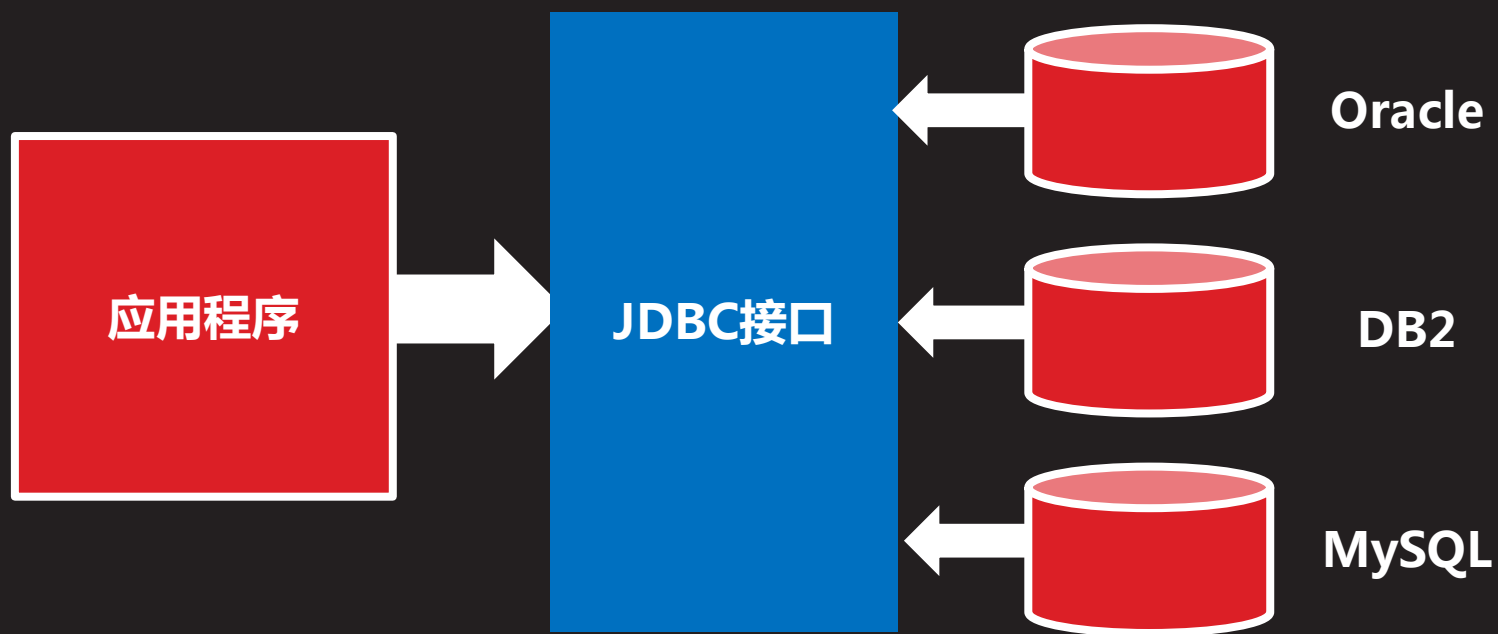
接口的作用——回调

接口的作用——面向接口编程

“契约”从硬件结构谈起



“契约” —— JDBC的成功经验



“契约” —— JDBC的成功经验

```
public class EmpDAO {  
    public Emp[] findByName(String name) {  
        ... ..  
        Connection con = ....  
        Statement stmt = conn.createStatement();  
        stmt.execute( "select * from emp" );  
        ... ..  
    }  
}
```



回调(Callback)——传递的是代码

```
/**  
 * 将指定的消息传递给对方...  
 */  
public void send(String msg) {...}  
// 参数传递的是数据
```

```
/**  
 * 将指定的消息按照指定的算法加密以后再传递给对方...  
 */  
public void send(String msg , ???) {...}  
// 如果要传递代码呢？
```



回调(Callback)——传递的是代码

定义一个加密器接口

```
public interface Encryptor {  
    public String encrypt(String msg);  
}
```



回调(Callback)——传递的是代码

send方法用这个接口作为参数，代码可以这样写...

```
public void send(String msg , Encryptor en) {
```

```
    // encryptMsg 为加密后的信息
```

```
    String encryptMsg = en.encrypt(msg);
```

```
    // 然后再将其发送.
```

```
    ... ..
```

```
}
```



回调(Callback)——传递的是代码

方法的调用者可以按照不同的方式实现这个接口

```
public class FoolEncryptor  
    implements Encryptor {  
  
    public String encrypt(String msg) {  
        return msg.toUpperCase();  
    }  
  
}
```



回调(Callback)——传递的是代码

是这样调用的...

```
FoolEncryptor en = new FoolEncryptor();  
...send( "i love u" , en ) ;
```



回调(Callback)——传递的是代码

也可以用不同的加密是算法

```
public class BloodyFoolEncryptor  
    implements Encryptor {  
  
    public String encrypt(String msg) {  
        return msg.toLowerCase();  
    }  
  
}
```



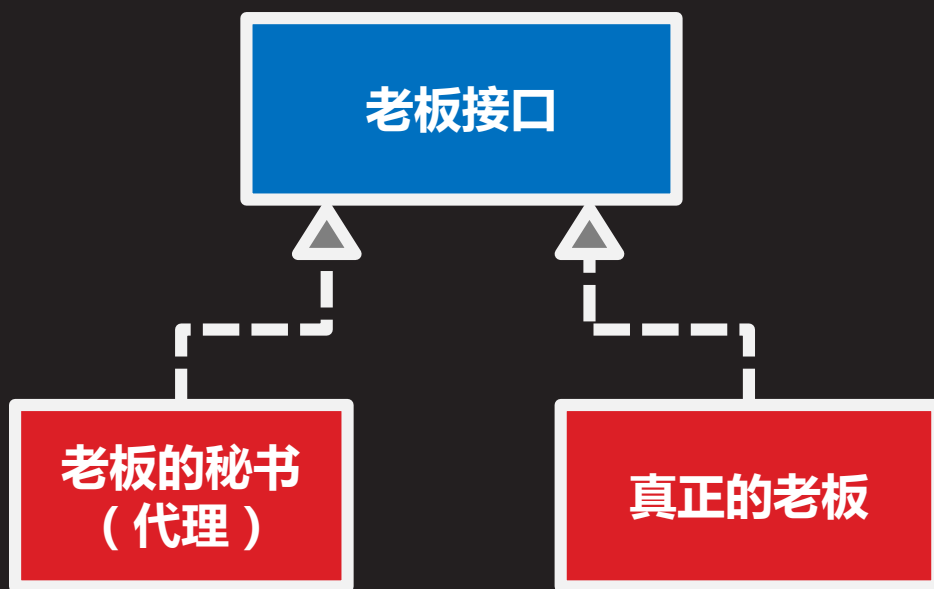
回调(Callback)——传递的是代码

更多的是用内部匿名类

```
...send( "i love u" , new Encryptor() {  
  
    public String encrypt(String msg) {  
        return msg.toLowerCase();  
    }  
  
}) ;
```



面向接口的编程——老板的例子



集团教学研发部

谢谢各位！

2013年12月