

基于数据的可视化展示开发培训

一、前后端数据交互

- Ajax
 - 通过POST或GET请求来访问后端controller层接口最终访问数据库的数据，响应并传给前端。

二、UI框架

- LayUI
- element-ui
 - vue组件
 - 在html结构中的任意地方引入想要的组件和组件对应的层叠样式表css即可实现效果
- Echarts
 - 可视化图标库
 - 引入echarts.js库，在js中粘贴所需图表的代码，并设置一个div容器来接收它，即可在页面显示。

三、后端

1. 多态

polymorphic : 方法或对象具有多种形态。是面向对象的第三大特征，多态建立在封装和继承的基础上。

2. 注解

- 注解：说明程序的。给计算机看的
- 注释：用文字描述程序的。给程序员看的
- 定义：从JDK5开始,Java增加对元数据的支持，也就是注解，注解与注释是有一定区别的，可以把注解理解为代码里的特殊标记，这些标记可以在编译，类加载，运行时被读取，并执行相应的处理。通过注解开发人员可以在不改变原有代码和逻辑的情况下在源代码中嵌入补充信息。
- 概念描述
 - JDK1.5之后的新特性
 - 说明程序的
 - 使用：@注解名称
- 作用分类

1. 编写文档：通过代码里标识的注解生成文档【生成doc文档】
2. 代码分析：通过代码里标识的注解对代码进行分析【使用反射】
3. 编译检查：通过代码里标识的注解让编译器能够实现基本的编译检查【Override】

3. 反射

- 在不操作源码的情况下，在运行时操作程序，符合ocp原则。

4. 接口

对接口的理解：

假如有三个程序员ABC，需要分别实现对MYSQL、ORACLE、DB2这三个数据库，连接和关闭的操作。

A程序员：在A-MYSQL这个类中用f1()和f2()两个方法实现了上诉操作。

B程序员：在B-ORACLE这个类中用con()和close()两个方法实现了上诉操作。

C程序员：在A-DB2这个类中用connect()和shutdown()两个方法实现了上诉操作。

当用户需要访问上面三个数据库进行相同的连接和关闭操作时，因为每个数据库提供进行操作的方法名不规范和不统一，所以实例化不同数据库对象时，调用相同操作的方法时的写法也不统一。

所以就有了接口，事先规定连接数据库操作为 `connect()`，关闭数据库操作为 `close()`。ABC三个程序员只需实现(`implements`)这个接口规定的方法。这样当用户实例化不同数据库的对象，连接和关闭数据库的代码

是统一的。比如：MYSQL对象.connect() 和 ORACLE对象.connect()。 `接口就是一种规范`。