面向对象编程

1. Golang是基于结构体实现面向对象编程的，通过接口进行关联。
2. 面向对象三大特性：封装、继承、多态
   1. 封装（结构体封装事物属性）
   2. 继承（内嵌共有字段的结构体）
   3. 多态（通过接口实现）
3. **工厂模式**
4. **MVC分别表示什么，有什么作用（结合项目经历来讲）**

Model：模型，用来封装对象，查询数据库等完成具体的业务操作。

View：视图，对数据进行表示。

Controller：控制器，获取View的请求，调用模型进行操作，然后将结果进行展示

在我做过的一个前后端分离的清单小项目中，对MVC有一个比较深的体会。假设分为多个小组来实现它，一个小组负责前端，一个小组负责后端，他们的使用的语言甚至不一样代码可以完全分离开，这实现了代码低耦合的思想。在后端，我会设置一个model文件夹，存放和数据库进行数据迁移的模型以及可以操作这些数据实现具体业务的方法，将它们封装起来作为一个模型。然后再设置一个控制器controller文件夹，里面只存放着业务调用的逻辑。这样一来前端View通过选择控制器controller提供服务就可以让控制器对模型进行操作，实现具体的业务功能，之后将结果返回给前端Vie让他展现出来。这样就实现了高内聚低耦合的思想。这样去开发可以将软件用户界面和业务逻辑分离，使代码可扩展性，可复用性和可维护性都加强，真正实现高内聚，低耦合，而这也正是MVC的宗旨。而且MVC更容易去理解，所以现在的很多项目都会选择MVC这个设计模式。