01月17日 日报

1. 已完成的内容

复习加昨天的作业讲解：

1. ng-repeat渲染无监听的，渲染在异步请求之前，所以会造成请求数据以后能赋值改变repeat的数据源，但页面不会加载
2. 异步加载结果会放到最后
3. 使请求变为同步
4. {}=={}，两个对象值完全相同也不相等
5. Ng-repeat的嵌套使用
6. jQuery ajax的使用
7. 数据创建格式

**首要目标：掌握什么是服务，怎么用；**

**服务（依赖注入）**：专门用来操作某个功能或存储数据的一个方法；依赖注入是一种设计思想，主要用来解耦，提高开发效率。

**耦合度**：表明一个程序各个功能之间的联系相关大小。

**解耦**：降低耦合度；

**服务包含**：

1. Value：声明一个变量 要求：了解
2. Service：声明服务 要求：掌握 其次
3. Factory 声明工厂服务，要求：掌握 最常用
4. Provider:声明服务，包含service和factory 要求：熟悉
5. Constant：声明一个常量 要求：了解

Service和factory的区别：

Service以自身作为对象返回

Factory是返回声明对象

**怎么使用：**

声明moudule时，[]内是注入依赖模块的

Controller的（）中是注入服务的

Ng-app=“注入服务”

要是用，首先要有<声明>---1.内置服务 2.自定义服务（指五种服务）然后<注入>到需要使用的函数，服务，模块中（即谁给谁用）

**重点**：1.记住使用逻辑顺序 2.记住格式

**服务声明方式：**

均和控制器声明方式类似

1. 模块名.Value（“名称”，值）
2. 模块名.service（“名称”，fun(){}）
3. 模块名.service（“名称”，fun(){}）
4. 模块名.constant(“名称”，值)
5. 模块名.provider（“名称”，fun（）{}）

其中provider是模块，config方法中$provide服务下的一个方法

Factory(“名称1”，fun(){

两种书写方式：

1. 直接return一个对象，如：

Return{

A:1

Fun：fun（）{

}

Arr:[]

}

1. 在回调函数内部书写逻辑，赋值到变量，最后返回该变量，（注意：该变量数据格式为对象），如：

Var fac={}

Fac.a=1;

Fac.fun（）{}

Fac.arr=[];

Return fac

})

.service(“名称”，fun(需要注入的服务，名称1){

名称1.a 名称1就是fac  
this.a=1;

This.fun=fun(){

Var b=2;

Return b

};

This.arr=[];

})

模块.config(fun($provide){

$provide.provider(“名称”，fun（）{

This.$get=fun(){

该方法和factory完全相同

}

})

})

注入：

1. 模块间注入，相当于两个公司董事长的交易
2. 模块内方法内部注入服务，相当于员工之间的合作

通过半天的复习，昨天的知识还算是掌握了，今天的内容

有点复杂，最后大概是弄明白了一些，但是自己写的话可能还是会出问题，可能还是模块的声明格式上的问题。

教师评语：