jsp

基本原理，servlet 把jsp翻译成servlert类，编译成class文件

脚本部分 <% %>

标准动作以及jstl+el

items 項目 数据项

是一个集合， 是很多数据

var 所定义的数据也是放在作用域中

函数标签库

前面两个都是通过标签来调用

函数标签，不是标签的使用 fn:调用定义的函数

左包右开

jstl+el jsp程序

servlet

目前原理，通讯技术，使用它实现开发模式 mvc

# mvc架构模式

模式==套路

开发流程变得更加便捷，容易修改，容易维护等

请求->处理->显示

显示 视图 view

处理 控制 controller

数据 模型 modal

MVC

分层概念

没分层也可以但是很麻烦

servlet也可以实现mvc 页面和逻辑都用java代码书写

jsp 也可以实现mvc 页面和逻辑都用jsp书写

整合

servlet+jsp+el+jstl mvc模式

3种

第一种 没有分层

第二种 分层 servlert或者jsp

第三种 分层 servlert+jsp

结果都是一样的

第二阶段

j2ee 开发基础

web开发 静态页面

html css js

java基础

web开发 动态网页开发

xml

servlert

jsp

el jstl ajax mvc 等等

# redis

内存版的数据库

关系型的数据库

只有key-value

map 结构

c语言设计的

10万次/s

使用数据库 操作的硬盘

相对于内存，cpu 处理速度是很慢的

连接数据库 操作数据库 瓶颈 高并发

放在内存中，map list 都可以

redis出来

把数据库中的数据，备份到redis中 直接操作redis

再把redis中的数据存入到数据库 永久保存

内存 ，重启 关机 数据就会丢失

直接操作redis

内存 容量 小于硬盘

基础

缓存

消息队列

mvc 开发模式

很多框架

struts1

struts2

springmvc

webwork

等等

ORM 模式

hibernate

mybatis

jpa

MQ 消息队列

基础原理搞懂、

AM

KK

RMQ

等等

key-value

字符串

list

map->key-value

对象

set

等

跟着我 操作

一些命令 跟着我操作

默认的配置文件使用的是代码中固定的配置

如果需要自己的配置文件，启动的时候，指定配置文件、

常用命令

keys \* 查看所有的key

set a A 设置key 为a 的。值为A

get a 获取key为 a 的值

del a 删除key为a