

# 第1章 初识Spring Boot，开发社区首页

---

## 1.1 课程介绍

---

### 1-课程介绍

大家好。

欢迎来到牛客网张华高级工程师系列课程，我是课程的主讲老师李红鹤，我们推出这套课的目的是想让你真正的掌握 Java web开发相关的技术，我们会从理论、实践面试这三个环节来帮助你，让你成长为一名合格的 Java高级工程师。

当然了想要达成这个目标并不会很容易，因为现在的互联网公司他的用人标准都蛮高的，为了让你能够达到互联网公司的标准，成功的拿到满意的offer，我们的课程会非常的侧重于实践，你看这是第一章的标题，叫初始sprint开发社区首页，也就是说第一章我们就要完成游客社区的首页的功能，那就要开始我们的实践了。

然后后面还有7章，我们会在它的基础上不断的加深实践，最终你会开发出一个完整的游客社区的项目。好了，在开始我们的时间之前，我想我有必要更详细的给大家介绍一下我们的课程，包括我们需要学哪些东西，我们需要做出哪些东西，以及我们需要在什么样的开发环境下来完成这样的内容。

大家可以看一下，我们这个课主要有三个目标，第一是教你学会主流的外部开发技术和框架。第二是想让你积累一个真实的外部项目的开发经验。

第三个是让你掌握热点的面试题的答题策略，好顺利的通过面试。

那么关于第一个目标，你需要知道主流的外部开发技术框架都有哪些，然后他们互相之间是什么关系，我们需要按照什么样的路径来学习，并且你要知道那学会以后我能利用这个技术做什么样的功能，这些内容我会在下一页PPT进行详细的介绍。

关于第二个目标，我想强调的是是一个真实的项目是非常重要的，因为你在实现真实的业务时，才能够透彻地理解一项技术它能够解决什么样的问题，而且在开发的过程中，你才能够遇到足够多的坑，然后在你趟过这些坑以后，你才会积累到足够多的解决实际问题的经验，企业最在于什么？他最在意的就是你能不能解决实际的问题，对不对？所以这个课程我为大家讲解演示的就是牛客网的核心代码核心功能。

那么第三个目标是在前两个目标都达成的前提下才能去实现的。所以我把他安排到了最后一章，你需要注意的是面试题他不能死记硬背，所谓的面试策略，它一定是建立在你对框架的透彻理解，以及深度的应用的前提之下，这样的话面试官无论怎么去问你，你都可以灵活应变，甚至可以举一反三，这样的话你拿到offer就是一个顺理成章的事了。

下面我们一起来看一下我们课程中要开发的项目的，你看这就是项目的精神页面，那么这个项目叫牛客社区，实际上就是牛客网的讨论区的功能，我把它从牛客网中抽离出来给大家进行讲解，那么牛客网实际上是有很多功能的，那是我们技术团队经过多年的努力，然后才开发到现在这种规模的。

所以你是不可能一个人在短时间内把整个网站都做出来。

那么我之所以选择讨论区来讲解，是因为讨论区的它的业务是一套比较典型比较通用的业务，你看大家所熟知的微博、知乎、豆瓣，其实他们的很多功能和我们的讨论区是一样的，那么解决问题的思路技术方案也是类似的，所以你如果说在简历上写上这样的项目经历，比较容易让面试官产生兴趣面试的时候，你们可以聊到投机一点。

那么这套静态页面是为了录制这个课单独做的，是在讨论区的基础上做了一定的精简，因为这个课是Java工程师的课，而Java开发它是侧重于服务端的，重点不是在前端牛客网是有专业的，前端页面可以做得很丰富，也很复杂，但是如果我们直接使用原版页面来做，那就需要你具备丰富的前端的知识，最后我们的课程定位就矛盾了，那么简化以后的页面就简单多了，只要你熟悉 html常用标签，了解一点 css

和GS，那么就可以上手开发了。

下面我们详细的看一下这套页面，首先我们看到的是社区首页，它实际上就是一个帖子列表，那么可以按照最新或者是最热门的程度来排序，然后如果你点这个按钮，那就可以发一个帖子，为了防止你发布不健康的帖子，那么网站的服务端需要做敏感词过滤，然后把非法的词隐藏掉，采用什么样的算法做敏感词过滤，这个才是我们课程中要重点探讨的话题。

另外如果你没有登录，你只能看这个帖子是不能发布的，而且你也不能查看消息，也不能看到个人主页账号设置，退出登录这样的功能。

因为当前我们是静态页面，这些就都显示出来了，实际开发的时候我们是要进行权限控制，然后动态的让这些功能或显示或隐藏的，然后当你点击某一个帖子的时候，你就能够进入帖子的详情页面，在这里你们可以看到帖子的详细的信息，当然还有回帖的内容，然后如果你是登陆的，你就可以给帖子点赞，或者是在下面给帖子进行评论，当然你也可以对别人的回复加以评论。

然后还要注意咱们这个页面的右上方有三按钮，那么这三个按钮不是普通用户可以用的，实际上它只有管理员和版主才能看到才能点。这里也需要做权限控制。

这里是注册功能，当你点了注册按钮以后，服务端就会给你发一封邮件，那么邮箱里会包含一个激活链接，你需要点链接将账号激活以后才能用，这个就是登录功能，这里的验证码是服务端的随机生成的，每次刷新页面都会变，然后当你点登录的时候，服务端需要对验证码账号密码作出判断，全都正确才能登录成功，然后登录成功以后，服务端需要记住用户的登录状态，这样你下次再来访问网站就不用重复登录了，然后登录以后你就可以查看个人主页，看看你关注了多少人，有多少人关注了你，以及你获得过多少个赞，然后这两个地方是可以点进去的，比如说你点这可以看到你关注的人，然后一点这儿可以看到有多少人关注了你，然后在这里你还可以看到你曾经发过的帖子，在这儿你可以看到你曾经做过的回复，另外别人也可以点击你的头像，进到你的主页，然后他可以点这个按钮来关注你。

这里面要注意的是这个点赞和关注，它是非常高频的一个操作，我们需要重点考虑性能的问题，采用什么样的方案才能提高系统的性能，然后在网站流量的高峰时段，不让服务器挂掉，这是我们课程中的重点要探讨的，也是面试官的非常关心的话题。然后个人主页的下面还有一个账号设置功能，在这里面你可以上传头像，然后可以修改密码做这样的操作。

那么我们可以把头像传到自己的应用服务器，也可以传到第三方的云服务器，那么这两种方案我在课程中都会给大家做演示。

最后账号设置的下面还有一个推出的登录的功能，那么可以注销你的登录状态，这里会显示你的未读消息数量，然后点进去以后你可以看到你和朋友之间的私信，以及系统给你发的一些通知，然后你可以点这个按钮给某个人发私信也可以点这里，然后看私信的详情，或者是点按钮给某人的回信，这是关于私信的操作。

再看一下这个系统通知，当别人评论了你的帖子，或者是给你点个赞，或者是他关注了你的时候，这个时候系统就会自动的发消息通知你一下，然后你可以点进来看一下通知的详细内容。

那么很明显网站每天都会产生大量的消息，所以为了降低应用服务器的压力，发布消息需要使用专业的消息队列服务器。

那么为了深刻的理解消息队列，你需要加强多线程方面的知识，需要理解生产者与消费者模式，在这里你可以进行全站的搜索，那么搜索的结果会把你输入的关键词做高亮显示，实际上就是标记为红色，全站搜索它并不是说从数据库中查一下就完了，是需要通过专业的搜索引擎来实现的，那么这也是出于性能的考虑，那么我们的数据库和搜索引擎之间是如何同步数据的，是如何支持中文分词的，如何做高能显示的，这些都是我们重点探讨的话题。

除了上面的功能之外，还有一个隐含的功能，它只有管理员能够使用，访问路径也不对外暴露，我需要手动的敲一下，让你看看在这里管理员可以统计某一段时间之内，网站的UV或者是某一段时间之内网站的活跃用户数，那么网站的用户是非常多的，用户访问网站的次数就更多了，所以统计与用户行为有关数据的时候，那么怎么样做才能够节约存储空间，才能够拥有更高的统计效率，这是我们将来要探讨的

话题。

好了，以上我们课程中大致要讲解和演示的功能，那么从界面上看它并不复杂，但是你要想把这些功能做到安全高效易用的程度，我们是要在服务端花很多心思去处理的，所以服务端开发它可不是简单的将业务实现就可以了，如何提高系统性能，如何让系统更安全，它永远是我们追求的目标，这个也是企业关注的焦点。

那么在这些方面我在课程里都会给大家做详细的讲解，让你学会专业的服务端开发技术。那么功能介绍完了，我们再回到PPT，接下来我再给大家介绍一下咱们这个项目的技术架构，让大家清楚的了解自己要学会哪些技能。那么大家应该都知道Java语言是暂推出的，而Java外部的开发标准其实也是由Sam制定的，就是Java ee而Java ee的核心是e接b我想可能很多人都没听过这个词儿，那是因为 e阶b设计的太复杂，太难用了，它已经被淘汰了。

目前市场上基于Java援助外部开发的主流框架就是spring，基本上大家都用spring，可以说 spring它已经是事实上的行业标准了，但是spring它不是一个框架，而是一套框架的组合，在开发过程中我们所遇到的常见的需求，它都能够给你提供很好的解决方案。当然有些功能是spring它自身支持的，有些功能是spring整合了第三方的框架支持的，总之无论是项目中有什么需求，通过spring都能够找到解决方案，所以这就是大家钟爱spring的原因了。目前最流行的做法不是直接使用spring，而是使用 spring boot。

那么在spring boot的帮助之下，spring它会变得更简单，就是spring boot它就是为了进一步简化spring而生的，然后在他的支持下，我们可以采用spring来管理项目中的对象，然后并且对其他的技术进行整合，所以 spring它才是整个项目的基石，是我们要学习的重点的内容。那么对于服务端程序来说，我们首要解决的就是如何处理浏览器的请求，在这方面是不是mvc是最流行的技术。

其次服务端程序要解决的是如何访问数据库，在这方面 my bets是最流行的技术。当然我们不会单独的使用iPad4，而是采用的思路来整合它，然后让它变得更简单。

那么这三个框架，spring、mvc以及Matisse，他们加起来其实就是我们通常所说的sm框架了。

这套组合是目前招外部开发当中使用最多的技术方案，在我们的项目中，所有的基本的功能都是由sm来实现的，比如注册、登录、发帖、评论、私信等等ssm是我们整个项目的基础，是我们时时刻刻都得使用的技术，所以前三章我们主要就是学ssm并且利用 sm将项目中的常规功能都给它开发完，而这个readiness它是no circle数据库，默认是将数据存在内存里，然后在适当的时候它再把内存中的数据存到硬盘上，那么由于它是操作内存的数据库，所以它的性能非常的好，在我们的项目中对于那些性能要求很高的功能，我们可以把这样的数据存到release里，来提高服务器的处理能力。

卡夫卡它是目前性能最好，也是最流行的消息，队列的服务器，我们的项目中所有的消息都是由卡夫卡来发布，然后 electric它是目前性能最好最流行的搜索引擎，我们项目中的搜索功能就是它来实现的，reds卡夫卡一来txt，他们都能够某一个层面上提高应用的性能，所以它是我们重点要掌握的内容，也是面试官的非常关心的话题，我们在使用的時候也会通过spring来整合他们，让这些技术变得更方便。

我们课程当中的第四章、第五章、第六章，就是分别学这三种框架，spring security可以管理系统的权限，让每个功能被正确的人访问，可以提高系统的安全性。

而。

spring act所以在我们系统上线以后，对系统进行全面的一个监控，然后让运维的人员随时掌握系统的运行状况。

因为很多中小公司它是让程序员兼顾运维工作的，所以说关于这方面大家最好也做一个了解好。

上面我所列举的是我们的项目中的最核心的技术框架，除此以外还有很多零碎的技术，比如说怎么发邮件，怎么生成验证码，怎么去过滤敏感词，怎么去定时启动一个任务，怎么生成长图，怎么把代码存到云服务器上等等，这些内容其实也很重要，我也会在课程里面给大家做讲解，做详细的演示。

最后我再为大家介绍一下我们要使用的开发环境，然后后面的课程里我会逐个的给大家演示这些工具怎么安装怎么使用。

首先我们需要构建项目，那么构建项目包括对项目的创建、编译、测试、打包、生成文档等等好多个环节，我们选择maven来做这样的事情，因为它是目前最流行的项目构建工具。

其次咱们这个项目代码量挺大的，所以我们最好选一个好用的集成开发工具，它可以大大提高我们的编码的速度。

目前在Java领域引开的 idea是最流行的集成开发工具了，当然它分为社区版和收费版，我们选择免费的社区版就可以了，而关系型数据库我们选择my circle，然后浓缩的数据库，我们选择reds，还有外部项目需要部署在应用服务器上才能运行起来。

在这方面我们选择开源的阿帕奇他们开的，最后你代码开发完了，我们需要上传到版本控制服务器二一个是为了便于团队成员之间的互相合作，这方面我们选择最流行的 beat，好，上面所列举的是我们项目中最基本的开发工具，大部分的企业都会用到这些工具。

除此以外还有很多工具也会用到，比如说my circle的客户端，my circle，workplace模拟浏览器发布请求的 pose麦，进行性能测试的解麦特等等。那么关于这些工具，我也会在课程里给大家演示他们怎么用。好了，牛克彰化高级工程师这门课程我就为大家介绍到这里，那下节课我会教大家搭建自己的开发环境，然后我们的实践就从下次课开始了，咱们下次课再见。

## 1.2 搭建开发环境

---

### 2-搭建开发环境

这次课我们来搭建开发环境，那么在搭建环境之前，我想先强调几点，第一点，作为一个开发者，你应该养成一个好的习惯，那么你安装的所有的开发工具都应该把它放到英文的并且没有空格的路径之下，因为有些开发工具放到这样的路径下会有问题，所以大家养成这样的一个习惯，规避这样的问题。

第二个我们是做账号开发的，那么我们安装的很多工具它是依赖于jdk的，所以在我安装这些工具之前，需要把jdk安装好。

对于DK你安装的时候最低的版本要求是吧，必须大于等于我安装的是12，那么因为我们课程中要用的 spring boot，他对jdk的版本有这样的要求，再一个我们这次课并不是说把所有的工具都装好，只是先装两个，一个是没问，另外一个idea，然后能够让我们普通的基本的程序能够跑起来，先达到这样的目标，后面的课程中我们到哪个阶段再安装对应的工具。

好了，那么首先我们先来安装maven，那么在安装maven之前我再对maven做一个简单的介绍，没问它对我们来讲主要有两个价值，第一个它能帮我们构建项目，包括创建项目编译项目，测试打包，生成文档等等，我们时时刻刻都离不开这样的构建工具。

第二个它能够帮我们管理项目中的查包，我统计了一下，当我把游客社区项目开发完以后，那么他一共引入了160多个包，如果说没有一个统一的工具去管理这些包，没有一个统一的位置去下载这些包，你到处去找这些包的话，很显然是非常麻烦的一件事，那么能够很好的帮我们管理这些扎包是一个很好用的工具。

那么关于maven我们需要先了解它的一个核心的概念叫maven仓库，那就是存放构建的一个位置，什么是构件？没问它所依赖的资源，在构建项目时，它所依赖的插件查包这都叫构件，而没问仓库，它分为两种，一种是本地仓库，一种是远程仓库，那么本地仓库就是你本机你自己电脑上的一个目录，对于Windows系统而言，那么它的目录是其中到了，不是不到了，这个波浪它代表的是加目录，那就我这个电脑而言，我的电脑的名字叫奥特曼尼斯特，那么加目录它指的是我的系统盘，你看一下我的系统盘是c盘，然后用户 administrator这个就是加目录，当然我这个目录之下现在还没有什么点m二，那么我需要把妹妹安装好，然后能够跑起来，这个时候它会自动创建这个点m二，默认没有，然后远程仓库是网站，那么我们可以从网站的下载包，然后它又分为中央仓库、镜像仓库和伺服仓库，中央仓库就是没文的官网，官方的镜像仓库是第三方，别的网站也提供了对中央仓库的拷贝，因为如果全球所有人都访问man中央仓库，那么速度很慢，一定会很慢，那么我们可以访问离我们近的镜像仓库，这样的话下载速

度会快，我们国内访问阿里云的镜像仓库就比较方便，所以一会我安装维稳以后会把远程仓库改为镜像仓库，而不用默认的中央仓库。

第三个是私服仓库，就是你去一家比较大的公司，他可能会自己搭建一个经营仓库，那么他自己用就叫私服仓库，那么作为我们这个学生来说，我们在学习阶段肯定是大4分，没有必要访问中间仓库又慢，所以镜像仓库是我们最好的选择。好了，那么再补充一点，那么本地仓库和远程仓库它有什么联系？就是当妹妹想下载包，想下载构件的时候，那么他会先看一下你本地仓库里有没有这个东西，如果有就用，如果没有他会从远程仓库下载到本地，然后再用，它是这样的一个顺序。

好，那么微文介绍完以后，接下来我给大家演示一下微文怎么去安装，怎么去使用。这个是微文的官网，我点击进去，因为我们要下载是吧，所以我就点这个登录的。然后它下面应该有一个表格，这个表格有列举了几个版本，其中 binary开头的是不带源码的版本， Sars开头的是带源码的版本，因为我们只是用所以说我们下载 banana就可以了。

然后 banana又分为两个，一个是它一个是GIP，这个是针对Linux或者是UNIX系统而言的，这个是针对windows，因为我是windows，所以我点下载，当然我本金已经提前下载好了，我就不点了，你们自己点进行下载。

好我打开我本机上下载好的资源就是然后安装就很简单了，只要你把它解压缩到某一个目录下就可以，我就来解压。

解压文件我把它解压缩到 d盘work目录下就可以了，好，然后我去找一下d盘 work，这是解压缩好的密封工具，那么接下来我刚才说了，它默认会访问，访问的远程仓库是没问到中央仓库有点慢，我要把它改成阿里云的，在哪改在这？

Conf是config配置的缩写，打开以后，那么有一个settings点xml那么我们打开配置文件去修改，然后你可以从头往下看，它这里大部分都是注释，都是一些说明，那么如果要改镜像仓库的话，在这个位置 mirrors merit就是镜像的意思，但是merit是复数，表示里边可以包含多个镜像，然后他给了一个例子，但是注掉了，你改的话就参考这个例子来改就可以，我把它copy一下，粘贴过来，然后需要加以修改。

那么配置仓库其实主要就是配一个网址，阿里云的仓库的网址是多少，咱们可以搜一下，其实很好记搜一下。它叫没问点阿里云.com，然后其实它有很多镜像仓库，其中三串中央仓库的景象，我们要用的就是路径就是可以复制一下。

当然当你点这个仓库以后，它底下会显示出仓库的资源来，就是一个文件夹在阿里云上，然后你可以展开，它里面有很多子目录，里边包含了很多炸包，我们下载这个包就从这下载好，我们就不细看了。

然后我回到刚才的配置文件，主要是把路径改成阿里云的路径copy的，然后 ID name也改一改，那么我叫ID，我叫阿里，没问题。

然后 mayor of就是说他是谁的，它是中央库的镜像仓库，所以说写清楚一点，以免有歧义，写上 center。

还有这个是仓库的一个名字，你给它取个名字，我取名叫阿里云，没问这样的话，镜像仓库就配置好了，配置好以后，那么我们就可以用了。

妹妹我们需要默认是要采用命令行的方式去访问它的，那么要用它的命令行工具，你要使用的是并目录下的，没问.come on，你可以打开命令行，然后 CD的目录下来用。

工具，当然如果为了方便，我们也可以把变路径配到环境变量里，这样的话我们随时随地不用CD就可以访问了。我配置一下环境变量这样方便一点。然后在这个环境变量pass里一新建加一条这目录就可以了。好这样的话我只要打开命令行工具，我在任意的路径下都可以访问那个工具。好，我们试一下看看命令的安装的好不好用，敲一个最简单的命令检查没文的版本， Mv没文杠word，然后他打出了版本是 3.6.1，没有问题就安装好了。

安装好以后，下面我给大家演示一下微文常用的命令就演示几个，因为微文的命令很多，然后其他的没有演示的，一个是我们用的相对少一点，再一个的话其实我们在这个工作当中一般也不直接敲命令，我们后面会用这个idea来集成 maven，到时候有了那个工具我们就可以点按钮了，但是这个命令你最好有个了解，这是对maven的一个本质的认识，这还是有必要的。

关于没文的命令，我们可以从刚才官网上找到它的一些介绍，非常详细，我们再去看一下官网，然后我们看左侧它有一个目录叫 user center，就是用户中心，我们点进去看，然后这里包含什么？Maybe in five minutes就是5分钟，你就能学会基本的使用，然后那么getting started get然后就开始正式的使用，没有更详细的介绍。

后面的话还有一些专项的讲解，包括仓库包括配置，反正这个内容挺多的，我就不挨个给大家看了。我们快速入门，我们看 in fact many。好了，你看他这有介绍，安装完以后我们怎么去看版本，刚才我敲的命令输出的内容是这样的，格式他也给你列出来了。

好，然后我刚才说了，没问的他能帮我们创建项目、编译项目，测试项目等等，我就把怎么创建怎么编译，怎么测试给大家演示一下，至于其他的什么生成文档等等，类似的命令就是一个单词，那么那些如果你有兴趣自己看一看手册。

好，那么首先创建一个项目，就这句话比较长，但是好我们在这能 Copy一下，然后我再回到命令行。我在执行命令之前我需要先CD到某一个目录下，你想把创建的项目放到就CD到哪，这样我想把这个项目放到放到d盘 work space的下面我要把项目放到这里来，好我就CD到目录下，我先去d盘，然后 CD d盘 work work space好，然后我把刚才的命令拷贝过来，右键就可以了，这个命令就是大部分地方都不动，但是有一部分的地方需要我们去改，我来给你讲解一下。

首先 maven命令都是以mvn开头，没什么好说的，然后 rt type generate意思是我要以模板原型的方式去生成一个文件，然后杠d group ID就是组织ID，那么通常我们写公司域名的倒叙，这里我要改一下，我要把它改成 come点老抠点，然后他建议后面写上你的项目的名，闹code，因为我们现在做的是一个小的事例，我就不写正式的项目名了，我随便取个名叫没问。Demo一，然后后面是杠d第三个ID，后面这是项目的ID，我也得重新改一下，没问demo一，再往后是刚d rk type二的fidid就是说你要使用的生成项目的模板的ID是多少？

这里写的是maven rp type，quick start是这个模板，然后-rk type，word版本是1.4，这是官方的推荐的，我们不用动。

最后还有一个参数是 interact mode等于force。

这个是问你是否启用交互模式，我们写false不启用，如果启用交互模式的话，它每执行一步就会问你是和否，这有点麻烦，直接取消了我们这些参数，因为都已经设置好了，就不用交互了，那么就可以执行了，回车执行以后，那么没文工具就会去检查你的本地仓库，看有没有东西，一看肯定没有，然后它就会访问远程仓库，去下载资源，你看他访问的都是阿里云对吧？

这说明我们刚才配置的镜像仓库是有效的。

然后你第一次使用慰问的话，这个会比较慢，因为他要去远程仓库下载一大堆的东西，你需要耐心，等一会儿不要着急，好，我就等一会儿结束了，最后它提示你build success成功了。

好我们看一下，你看work space的下面确实有这样一个项目，这个项目叫maven demo，一进去看有一个泡沫文件，将来我们要很频繁的使用它，但现在我先不细说了，然后有一个s二c的目录进去看，有 main test，main下放的是正式的代码，test下放的是测试的代码，在开发时，那么我们把正式代码测试代码分开写好，将来我们部署项目上线项目的时候，测试代码就会被抛弃。

然后命一下有Java com，Nokia没什么demo，一他默认给我带了一个类类型，非常简单看一下，就一个输出一个hello word非常就是初级的代码不说了，然后回退到src目录下，还有 test下有Java com到 coder，没什么demo，一包和刚才面一下一样，然后里边有一个类APP test，也看一下，它是基于解忧类的测试的，然后默认带了一个方法，这个方法是用断言的方式去判断参数是否为数，如果是真的话，

那么他不会做任何的就没有任何的事情发生，如果是force他就会抛异常，这显然就不会抛异常，就是没有问题。

好，看完以后说明我们这个项目就是生成的没有问题。然后我再给你看一下我那个价目录，加目录我的c盘用户automatic data，这回你看到了 m二，然后 repository仓库，本地仓库，这里边就有一大堆东西，那就是 maven自动从远程仓库下载的资源在这里。

好了，那么我们再看官方的手册，他后面也给列出来了，说这个项目结构应该是什么样子，你看这个也可以好了，那么接下来项目建好了以后，我再给你演示其他的命令，我就不看手册了，你课后自己慢慢看，因为比较长，接下来给你演示一下我怎么把这个项目编译，因为你刚才看了那个项目里它只有java文件，就两个，一个正式一个测，一个是测试的类，但是没有class文件，你要想运行还运行不起来，所以我们需要对项目进行编译。

那么你要编译哪个项目，你要CD到那个项目的目录下，CD到这儿来，就CD到带有泡沫点xml这个文件的目录下，好我就CD到maven demo一下面，然后我要想编译项目非常简单一个命令 maven，然后怎么派回车，需要一点点时间，因为在编译的过程中它也需要一些插件，它又需要一些包又会去下载，因为本地仓库还是没有，不过第二次就好了，因为第二次他已经下载过了，这已经成功了，我们看一下目录多了一个它盖的编译的结果都会放到它盖的下，这里面我们主要是看 class里边就是编译好的，有人会想怎么没有 test class，这两个不用看这两个文件夹是它编译的过程中自身依赖的资源，这个classes是我们需要的东西，那么因为我们只是编译，并没有要求测试，所以他没有去把测试类进行编译，是这样的。

好了，接下来比如说我要想重新编译一下，我可以这样写，没问我先可令一下，把刚才的内容删掉，可令是删除清除，你看他盖的就没了，然后我再重新编译。好，再看就又有有了，那么可令命令它可以和编译命令就是混合在一起用，你可以这样写没问题。可令compare意思是我要先清理再编译好，再执行一遍，也是ok的。超概念还在。好，比如说现在我不只是要编译，我还要进行一个测试，就是编译好以后，让它自动跑一下，测试的代码你可以这样没问可列清理完以后 test测试。

好，那么他又需要下载一些包，因为做测试这件事需要依赖一些资源，好，我们稍微等待一下。还挺快的，成功了他说了乱了，一个方法失败了，0出错了，0忽略了0你就成功了，乱的是谁？是这个APP test。

那么大家注意我刚才执行的命令是test，其实 test命令它是包含了compare，因为你想我是先清理了，他get就没了，然后我test他一定得是先编译好再测试对吧？所以其实test它是包含了品牌，所以它可能面临的是有一个包含的关系。

好，我们看一下目录，target那里边就多了一个test classes，这里面就是它编译好的测试的程序。好，那么这个maven它基本的使用就是这样，当然它还有其他的命令，那么大家可以自己参考官方的手册去做更多的了解。那么没门装好以后，下面我们再去看一下集成开发工具隐态的 Idea，还是先说两句。

那么intel的 idea它是目前最流行的，开发java的集成开发工具，基本上大部分的互联网公司都是用工具的安裝就比较容易了一个软件。那么我不说废话，直接给你演示怎么去装，那么这个网址是ID的官网，我点进去看，好稍等一下，略微有一点点慢。

好来到官网以后我要下载，所以说点这个download，然后它有两个版本，一个是专业版，一个是社区版，专业版你可以免费试用，但是最终是要收费的，是按年收费好像不便宜，那么我们用社区版免费的就可以免费开源的，这个也足够我们用了，那么你点下载就可以了，我已经下载好了我就不点了。

然后我给你看一下我下载好的程序，就是它exe文件。

好，下面我就给大家装一下白点。下一步换一个安装目录你不换反正也行，我换一下我换到d盘，还是放到work之下。好，然后下一步稍微有点卡顿稍微等一下有点慢，这个界面你需要选一下多少位？我的系统是64位，所以我选64。那么其他的就不需要动，然后下一步直接点安装就可以了，好，这个过程稍微有点慢，咱们等一会儿，好终于安装完了，我就点击完成就可以了，然后我打开一下idea，咱们看一下，它会生成一个图标在桌面上双击打开。

好，打开的时候，那么他会问你要不要做一下配置什么的，好我就不做，不要导入配置，我就不导入了，然后选这个然后 Ok ok以后它会让你选择皮肤是白的还是经典的，其实一般我们在工作的时候大家都愿意用黑的，就是你一天工作时间久了，黑色还是比相对来说比较护眼，我就用我们常用的方式，然后下一步是否选择安装插件我就不安装了，好然后开始使用idea就行了，然后它就会进入到这样一个界面，在这个界面你可以选择创建一个新项目，或者是导入一个项目，或者是打开一个项目，导入和打开的区别是什么？

如果我是用之前用idea创建了过一个项目，我这个idea就会直接打开它。

如果我已经有了一个用Eclipse创建的项目，我需要导入把它转成idea的项目，是这样的一个关系，然后这里还可以做一些比如设置这个字号设置没问我先做一个设置就点 settings，我要设置一下字号，不然的话字太小，你们看不清楚，设置字号在艾特下面，然后有一个phone，当然你也可以在这搜 phone，这个字体默认我不太喜欢，我换成我喜欢的字体，然后字号调成16稍微大一点，你们自己根据自己的情况去选。

另外我还要设置一下，配置一下没问，因为我们在开发使用idea去写代码，然后写完代码以后怎么去编译对吧？怎么去测试，其实也是用idea的不用去命令行求命令，idea可以帮我们去调用 main的命令，前提是你告诉他密文装在哪。好，那我就选一下 build构建的工具， build tools。没问第一个，然后我们需要做的是你要选择你安装的没问的路径，我装在d盘work阿帕奇maven。

Ok另外你要指定你自定义的妹妹配置文件，因为配置文件里我们不是改了阿里云吗？对吧？不然的话他就会用默认的，我们还是要指定就这块 Override用户的配置文件，我们选一下我们自己的地盘， work。阿里阿帕奇没问。 Can figure settings。这个是本地仓库的位置，默认就是点m二没有问题。好，不赖。Ok。好那么 idea我就配置好了，配置好以后，接下来我们就可以利用它创建一个新的项目，点 create new project，然后左侧选择项目类型，我们还是要创建美本项目，所以我选择美本。然后右侧的上方你要选择jdk，它是默认选好了。

如果你没有默认选好的话，你就点这个按钮在弹出框里选择你GD Kay安装的位置，然后点 create from rk time，我要用这个模板来创建明文项，还是利用刚才梅文的官方的模板就是它好，然后下一步这里要填group ID，公司域名的倒叙come down not cold，后面最好带上你的应用名，我这个项目叫没问demo二，下面写的是项目ID明文demo二，然后是版本号默认1.0就可以，下一步这是让你确认是不是用的是没问配置文件，以及这些配置信息这些没有问题。

再下一步这儿我要填的是这个项目的名称，它默认就是跟ID一样也没有问题，然后让你选择这个项目存放的位置，我要重新选一下，我要把它放到 work space的下面，然后建一个文件夹和项目同名，叫没问demo二，好到这就可以了，点击完成好，然后你最好点一下 enable output import允许它自动引入一些资源，然后把它最大化一下。

好，那么这个项目就创建好了，创建好以后我们可以左侧展开这个项目，看一下它的结构，根目录下也是有 pom点xml然后 src的下面也有main和test，main和test下面也有包以及自带的类和我们刚才直接使用没问的工具命令行创建项目，这是一样的。

好，那么在 idea工具下我们怎么去编译这个项目呢？因为你看现在他还没有to get，那也很简单，右侧有一个美文的这么一个标签你展开，然后把这个展开，这里边有他全部的命令，比如说你要清理点命令，比如说你要编译就点 compare双击一下，好很快的就编译好了，他get就有了。

当然如果你要测试你就点 test，它会重新的进行编译，然后并且进行一个测试。

好，你看这是测试结果，它打印到了控制台也没有问题，那么其实我在实际的开发室一般不太愿意点这个地方，因为稍微有点麻烦，还得展开。还有一个更简便的方式，你可以点 Build构建，然后 build project构建项目。那么点完以后它就会重新的进行编译，当然它的快捷键是CTRL f9，所以说我们平时写代码的时候要重新编译，就按一下快捷键就比较方便。



好，那么如果说你想单独执行某一个类的方法也是可以的，比如说我打开 APP类，我要运行 main方法，那就和我们学学基本的Java没什么区别，右键软面方法，输出结果，hello word没有问题，APP test，比如说它这里会有很多测试方法，我只想执行这一个，那么你就选中这一个方法，右键，那么结果是正确的，它勾就表示正确的，没有报错，就表示测试通过了。

好了，这就是这个idea它最基本的使用的方式。

好，我们再回到PPT，看下一项内容，那么接下来我再给大家演示一下怎么去使用 spring in this letter来创建一个spring部的项目，有人可能会奇怪，为什么要用这个工具呢？不是说好了没问。是这样的，就是说没本虽然说已经能帮我们解决很多问题了，但是他还有优化的空间，或者说它还有能更方便的一个空间，就是说我们用没问去管理包，尽管很方便，但是包太多的时候也会有麻烦，因为你想我们最终项目100包160多个包，如果说你都用maven去管理的话，你都需要通过maven的官方去搜索这些包，去下载这些包，我们通过慰问怎么去搜这个包，我给你演示一下。

那么我们搜没问的包，其实你可以通过阿里云的仓库，这儿有这个社区，比如说你要做 My circle的开发，你可以搜my circle回车，然后它这里有很多结果，但是好哪个不好哪个常不常用看不出来，这是阿里云的搜索的一个缺点，我更喜欢用的是另外的一个网站，maven reporter的com。

好，那么我们也可以在这个网站去搜索微盟仓库里的资源，比如说我搜买circle回车，那么他会有建议，它是按照使用的数量排名的，显然用的是最多，所以说我们现在它应该还是比较合适，所以基于这种情况我们会用来搜，但是你想一个项目有那么多个包，如果说你挨个包去这么搜一下，再去把它下载过来多麻烦，当然如果你要下载这个包怎么下载你就点进去。

然后选择一个版本最新的，然后这段代码拷贝一下，然后复制到我们的项目的泡沫文件里，然后把它放到得攀登CS之下，比如说这个位置就可以了。你只要一保存，那么maybe根据配置就会自动的下载包，但是包太多了，你这样做还是很麻烦。

好，再回到我们的正题，那么我们使用 spring in the slider工具就能很好地解决这个问题，他是怎么解决的？他是把这些包把我们平时开发的这些包做了一个整合，然后按照功能按照需求进行了划分归类，比如说我要做外部开发，我就输入web，或者说我搜索一下web，那么我引入web相关的包，它是引入一批包利用这个工具，总之我要做哪个方面的开发，我只要把相关的功能搜一下，我就能把这一方面相关的几十个包一下都引进来，所以说这是他对包做的整合，当然它的底层还是基于没问。

好了，我接下来给大家演示一下，我们利用这个工具可以通过这个网站来使用，点一下这个网址。

好，你看首先它的项目默认就是明文项目，语言就是Java，然后项目的版本默认选2.1.5，2.1.5是正式的版本，像其他的带有这种标记的，它是非正式的版本，所以我们就用2.1.5，然后你要输入项目的骨肉和ID，com点老q点点，这回我取一个正式的名字，我们后面的开发就基于这个项目，com点到q的点，community就是社区的意思，游客社区好，然后项目的ID community还有更多的配置，项目名肯定没问题。

项目的描述我就写一下老客户点，他没问题的留客社区，然后包名com点老客户点，他没有那天一个就够了，就这样。

好，然后是这儿还是那么按理来说我们开发外部项目应该是外包，但是 spring部的工具它就这个框架比较神奇，我们选择炸包也可以运行起来。

好，至于说选择外包怎么做，将来我们在后期讲就项目开发的差不多了，我们去部署的时候，我也会给你演示，但我们现在用这种简单的方式，最后选择Java语言的版本，我最新的是12我就选12，你根据自己的实际情况去进行选择就可以，比如说你用的是9，在这里没有你就选8就可以。

好，然后我想开发这个项目，我想开发哪些功能，我在这儿去搜。当然了这需要你基于对spring有一个了解，你目前不了解，那就照着我这个操作去做。

我们在开发的过程中需要用到 Spring 的一个很核心的功能，aop 我们后面会重点讲，所以搜 aop 就是它点一下表示说我一会重建项目时会带上功能相关的包，然后我要做 web 开发，做 web，我还要用到模板引擎来生成页面，搜一下 time leave，用它最后再来一个 dev tools，这是开发者工具。

那么我们在开发的过程中有这个工具就会很方便，比如说我代码改了，我需要重新启动服务器，那么有了这个工具，我们就用手点去重新启动服务器，它会自动重启。

当然我们项目中不只是用这几个包，还有别的几类，那么后面我们用到时再说我们先导这几类。你看我只要输入 4 个单词，其实它背后就是几十个包就有了。

好，那么输入完以后我就点生成项目，generally the project 好，然后我打找到它所存放的位置，那么得到了生成了项目以后，那么我就把它解压缩到某一个目录下，解压缩一下，那么我把它放到 d 盘 work space 下面确定。

好我去看一下，有了，那么我就回到我的 idea 开发工具，我要用开发工具去开发它，所以说我要打开那个项目，这个项目我们只是刚才的一个简单的演示就不再使用了，所以我就点这个 file，然后点 open，打开就可以。

然后 d 盘 work space community。

然后他问你是在一个新的界面打开，还是在当前窗口打开，那就得在新界面打开，当前的还保留，在当前窗口打开它会被覆盖掉，因为这个不用了我就覆盖掉。那打开以后它会去下载一些炸包一些资源，这需要一点时间，然后这个项目才能够变得完整，我们稍微等待一下。

好了，这个包已经下载好了，那么这个项目也已经生成了，那么你注意这个项目和刚才的项目就有区别了，因为它不是一个最基本的普通的项目，它可是一个外部项目，所以说它的结构略有不同，你看它默认有这个账号目录，里边有我们建的包，这个包下有自带的一个类，这个类里边的代码，就和我们刚才 demo 就不一样。

然后还有一个 resource，文件夹，这个文件夹之下存放的是配置文件以及一些静态资源，包括我们要用的网页模板，包括我们网页所依赖的图片样式，js 都可以放到这里面来，这以后会详细的讲。然后这个是配置文件以后也会详细的说明，然后 test 目录下也有一个类生成，然后他带了一个空的测试方法。

好，当然它还有一些注解，那么这些细节我们这节课这次课先不详细说了，因为我们这次课主要是把环境安装好，然后把视力调通，表示我环境可用，那么具体的说这句话什么意思？注解是怎么回事，那么我们会在后面的课程里面进行详细的讲解。

好，那么这是一个外部项目，我怎么去运行它呢？其实和刚才我们运行 demo 一样，就运行 main 方法我就运行一下是吧？它会进行编译，然后启动这回运行和刚才的结果不一样，刚才我们运行的是 maven demo，然后它只是打印了简单的一句话，但你这次运行你看它打印的内容很多，包括一个 spring 的图标，这个叫 banner，包括一些信息，这个信息你仔细看有一个很关键的提示他说 Tom cat started on port 8080。

就是说 Tom cat 以 8080 端口启动成功了，而且这个程序没有终止，你看这是停止的意思就关掉程序的意思，stop 的意思，但它没有变灰，表示说它一直在运行着，所以说我们运行这个方法它底层做了什么，它底层自动帮我们启动了 Tom cat 应用服务器，好，有人说你没装应用服务器，你注意 spring boot 它有一个好处是他带的包他的包里扎包里会带有一个 tomcat，spring boot 它是以炸包的方式内嵌了一个汤盖，我们启动就是内嵌的汤盖，那少装一个工具，这样不是更方便吗？如果你想知道汤盖在哪，你可以在这看一下他最终所依赖的，或者说他所下载的所有的包你看有几十个，可我们引入的时候只写了几个单词而已，对吧？

所以说这就是死人不倒的优势了，很方便，然后这里边你去找你看 Tom cat 什么就这几个包内嵌的汤盖好了，那么服务器启动了，我们通过浏览器就可以访问，因为服务器就是为浏览器提供这个服务的好，我打开浏览器访问一下，这个服务服务器它的端口是 8080，好，我就输入 Local host 冒号 80 80 回车，然后给我这样一个页面，white label error page，就是一个空白的错误页面。

那么为什么是这样子一个结果？因为你不用细想也知道，我们的项目中默认什么都没有办法给浏览器提供任何的服务，没有办法给浏览器返回任何的网页对吧？所以说他不知道能给你返回什么，因此返回一个错误的提示，啥都没有，只要我们写一些代码，能够为了哪些返回数据，浏览器就能得到正确的结果。

但是当你看到这个页面的时候，表明说我们这个服务器确实在工作，如果说服务器没工作，它是不会返回这样的文字的网页。

好了，所以说我们这一步利用spring in this letter创建一个spring的项目，这个环节其实就已经成功了。好，那么我们再进入下一个环节，也是我们在的最后一个环节。我们开发一个最简单的入门实例Spring，but入门实例好，让我们这个程序能够正常的跑起来。什么事例？就是说当我访问服务器的某一个路径的时候，它能够给我返回一句话就行。

Hello spring boot就可以表示说我这个东西确实通了。

好，那么在写这个例子之前，我先对spring再做一个简单的介绍，那么spring boot它的核心的价值就在于什么？三点，第一个它能够给我们的项目提供起步的依赖，什么叫起步的依赖？刚才我们创建项目是不是搜索了web Aop对吧？

搜索的东西就叫起步的依赖，就是少量的依赖，其实就能够依赖很多个包，这个依赖体现在哪里？我们打开这个项目看一下，home点xml在Xml的体现，在得攀登CS之下，你看他依赖的这就是我们搜索之后它依赖的包 spring boot starter。Lp。Starter time live。Starter wine是 spring boot。Dv tools。好，它都叫什么？Starter，这个就叫起步依赖。这个依赖它依赖的不是一个包是一堆包。

你比如说以 web为例，你可以CTRL点它进去，然后这是他底层的文件，我们看他实际的dependence有很多个，就每一个dependency，每一个dependency就是依赖的一个包，你看它依赖的其实很多个包，而且有的包也叫什么start也是一个组合，所以它是通过这样的方式让我们只需要写出这么一段话，就依赖了一大堆包，其实它是对一堆包的组合，这个就叫起步依赖，让我们构建这个项目依赖包方便好。

然后第二个它的核心的作用是在于它能够做自动的配置，如果你不用spring boot，那么你需要配置很多的内容，很多的配置文件，你要想启动外部项目，你要想使用spring框架，你要做很多配置，但是我们利用spring几乎不用做配置，我们就可以把服务启动起来。

好，第三个作用是端点监控，就是说当我们项目上线以后，我们项目运行的状况是否良好，我们可以对它做一些监视，那么它是不能够对这个功能提供支持。

好的，端点监控是我们以后要探讨的话题，因为这是项目开发完以后上线以后的事，我们在最后一章再进行一个详细的讲解，先不说了好了，那么大致了解了spring boot的核心价值，也了解了我们要做什么。

下面我就给你演示一下这个例子，因为是第一次写思政部的项目的例子，所以说我们设计的比较简单，也就十几行代码，好，那么我要现在要想提供一个功能，在服务端提供一个功能，可以被浏览器访问到，能够给浏览器返回一个简短的问候，怎么办？这样做。

首先我先建个包，这个包 Ctrl代表什么含义，以后再说CTRL之下新建一个类，那么这个类只是一个事例，不是我们正式项目的代码，但是我也放到这个项目下，只是一个小例子，我给它取名叫阿尔法CTRL，然后类上需要写一个注解，controller还要再写一个注解，那么这两个注解它是spring mvc的注解，我们将来学spring mvc我还会进行详细的介绍，我们现在先这么写，我写的斜线阿尔法是个啥意思？

我是给类取一个访问的，那么因为你要给浏览器提供一个服务，浏览器能够访问你，能够访问这个类里面的某个方法，它怎么访问这个类呢？

通过这个名字来访问这个类，所以需要取给它取一个访问名，然后在这个类当中我还要写一个能够给浏览器提供服务的方法，好，或者说能够处理浏览器的请求的方法，这个方法我们先这样写，他返回一个字符串，咱们先返回一点简单的数据，方法名我叫c hello，然后我直接就推一个字符串，hello，spring就完了。

那么这个方法能够被浏览器访问到的前提是它前面也要写这样的注解，给这个方法也取一个访问路径，我叫hello。另外这个方法，如果你这样写，浏览器会认为它会返回一个是网页，但我们返回的不是网页，那只是返回普通的字符串，还需要加一个注解进行声明，当然注解也是以后详细讲，先写到这种程度。

好，写完以后咱们再试一下。

那么因为刚才我他们开的已经启动了，那么因为我加了新的代码，我需要重新编译，重新启动，你可以点这个按钮重新启动，在启动之前它会重新编译，或者你也可以直接按 build project CTRL f9，那么它就会自动重新编译，你看它已经重新编译了，然后自动重启了，自动编译以后自动重启，这是 DV tools 的作用。

那么然后我就打开浏览器，我再访问一下，8080我要访问出来那个类，它的访问路径是阿尔法，然后访问这个类之下的c hello方法，它的路径是hello成功了。

你看我浏览器访问的服务器，服务器给我返回了我想要的一句话，所以说你看我们用spring boot去构建项目，去开发一个前后端能够交互的功能，居然是这样的简单，所以说这是死不认不的项目的优势非常的方便。

好了，那么到这我还想多说两句，当你开发完以后，说我想不用这个服务了，那么你要把服务关掉，不用的时候把它关掉，点按钮就是关闭tomcat，然后 tomcat启动的时候，它默认的端口号是8080，那么有些同学可能自己电脑上事先安装过别的软件，端口恰恰是80 80产生了冲突，这个时候你要么去改软件的端口，要么是改tomcat的端口。

我们就介绍一下如何改Tom cat的端口。那么在superb下我们想修改服务器端口非常简单，去改 application，点process配置文件。

好，当然配置文件就是为什么这样写，我以后也会详细讲。

咱先这样写着，我要改服务器端口就写server点part所有的端口等于8080，当然你可以改成任意的808180828088就随便我还是把它改成8080，然后通常我们可能还会给这个项目取一个访问路径，我给它取一个12瓦点surveys点context pas等于那么我希望这个项目的访问路径是这样的，community那么配置完以后，我再重新启动服务点按钮就行了，当然你也可以在这儿右键软，或者是在这儿选中某一个类点软。

好，那么就启动好了，启动好以后它的端口还是80 80，我没有具体的变，你可以把它改成任意的值，这都无所谓。

然后打开浏览器，你这回不能这样访问了，因为我给项目加了一级路径，所以你要80 80之后你要访问community，项目之下再访问control以及它的方法，回车，这个结果依然是正确的。

好了，那么通过这个例子，我们就初步的体会了sprint，我们是怎么利用它去开发项目的这种感觉，有了点感觉。然后通过这小例子，你也能知道你的环境搭建的是ok的，我们这次课的目的就达到了。好了，这次课我们就进行到这里，咱们下次课程再见。

## 1.3 Spring入门

---

### 3-Spring入门.mp4

这次课我们来学习spring，那么spring之前我们也提到过，它不是一个框架，是一套框架。我们项目中基本上所有的需求 spring都能帮我们解决，它能够给我们提供了非常全面的解决方案。么spring这套框架把他们加在一起，我们通常称其为spring全家桶。那么spring全家桶它大体上分为4个方面，第一个方面就是space，框架它是基石，它是核心。

第二个是spring，boot我们可以利用它的构建项目，然后在这个项目的基础上去做spring开发，会更容易更方便，上次课我们也做了演示。

第三个是spring, cloud就是做微服务的, 如果说我们有一个很大的项目维护起来比较困难, 我们可以利用 Screen cloud把它拆分成若干个子项目, 然后把他们再集中在一起, 每一个子项目它的规模比较小, 就便于维护了, 但同时开发的难度会有所提高, 是这样的。

第四个是spring cloud data flow, 就是spring做数据集成的这么一个功能, 比如说你的应用有很多客户端, 包括移动设备传感器, 还有可穿戴设备, 甚至是汽车等等, 有很多客户端, 那么这些客户端所采集到的数据形态各异, 怎么把它们集成在一起, 整合在一起, 然后形成更有价值的数据是他所做的事情。

当然这4部分功能我们不会都用上, 因为我们这个项目本身它就是牛客网的一个模块, 他还没有必要去往下拆分, 所以微服务就不用去做了。

然后我们这个项目只有浏览器一种客户端, 也没有那么多客户端, 所以 data flow也用不上, 我们会用前两种。

那么 spring全家桶它所有的内容文档手册, 那么都在它的官网上, spring点!上能够查阅到, 咱们可以看一下。这个就是它的官网, 你往下看, 它这里有一些大概的介绍, 你的应用如果只是做基本的外部相关的开发, 你用死不让步的是可以搞定的。

Sprint是build安全性能构建一切, 然后如果你想做微服务, spring cloud可以帮你搞定, 它是call it安全性, 就是说它能够协调一切, 然后如果你想有更多的客户端数据集成可以用 data flow, 它能连接一切, 这就是它的整个产品的一个生态的结构。

总之无论是什么样的应用都能用它来解决。当然这些内容它都是基于思维框架的, 它是基础是核心。

这一页是对整个 spring产品的一个概要的介绍, 那么更详细的内容你可以在这里找到projects, 包括死人不得什么work得服了, 我可闹的等等, 然后其他的模块其实也是死不用的子模块, 因为它相对独立, 然后也比较用的比较多, 因此它也单独的拿出来, 让你能够直接方便查到我们的课程里后面会用到斯通贝特访问数据库, 用到 spring的secret做安全的控制, 权限管理, 还有 amqp用来做这个消息队列, 主要是用到这些内容。

好了, 那么我们再回到 PPT, 我们这次课重点是讲 spring, 然后 spring它有很多功能, 我们课程中也不会都用到, 我们用到的是4大部分, 第一部分是spring的核心, 它包括IOC和aopaop两部分, IOC和aop它都是用来管理对象的一种思想, 那么由死不硬所管理的对象, 我们通常叫做BIM, 它是用来管理病人的思想。

loc也是基于面向对象的一种管理对象的思想, 而aop是一种面向切面编程的思想, 它是对面向对象思想的一种补充, 总之是不愿有了这两种思想能够管理一切病。

所以说哪怕是你有一个别的框架, 第三方的框架, 那么拿过来他也能够进行管理进行整合。所以spring它牛不是牛来说它的功能多, 而是它牛在它有这两种管理病人的思想能够管理一切。

那今天我们重点学的是 IOC后面的课程里我们会学LP那么第二个是spring data access, 就是spring的访问数据库的功能, 我们会用到用它来管理事物, 我们会用它来整合my best。

第三个是外部开发, 我们会用到它的 spring mvc这一块。最后一部分是集成, 我们会用 spring去发邮件去做定时任务, 然后去做消息队列, 发送消息, 然后去做安全控制, 这是我们课程中要用到的内容。

那么spring还有更多的内容, 你可以从官方的手册里面能够查阅到, 咱们简单看一下。就 Spring进来以后这是一个概要的介绍, 详细的是在论之内。

然后标有Ga的是它的正式发布的版本, 最高的版本是5.1.7, 然后你可以点这个reference, 然后这里面它是归类, 核心有测试, 有访问数据库等等。我们课程中所体现的内容其实就是这些内容中的一部分, 那么你想看哪一块点进去看就是了。

好, 因为这个内容比较长, 所以说我们课堂上不去给大家看了, 这个是你们课后自己可以去查阅的, 作为一个课堂内容的补充。

好，我再回到 PPT，那么关于spring我们首先要学的是IOC其实spring所有其他的这些个功能都是构建着LC之上的，或者说他们都是以LC为基础的，所以必须先学好LC，其他的功能才容易学会。关于IOC首先你要理解什么是IOC那么IOC它是一组单词的缩写，叫做 it wasn't of control。

翻译过来就是控制反转的意思，这是一种面向对象编程的设计思想。

初步解释一下，比如说我们不用 Lc，我们自己管理对象通常是这样的，我new一个对象a再new一个对象b然后 a点site传入b这样的话 a和b就有了联系，我可以通过a去调用b对吧？这是我们经常会做这样的事情，这样做的话是比较直观的，也比较直接，但是它有一个缺点就是a和b产生了耦合，直接就耦合在一起了。

那么当我们的项目规模变得比较大的时候，可能就会不好去维护，可能你改了a b改b也要顺带改a这是他的问题。

那么LC思想就是说他不用这种方式去建立病人之间的关系，他用另外的办法去建立病人之间的关系，从而减少病人之间的耦合度，让这个项目能够便于维护，是这样的。他是怎么去实现这个思想的，主要靠的是依赖注入的一种方式，这是它的实现方式，而依赖注入的实现方式。它又基于IOC Ioc容器，它是实现依赖度的关键，其实它本质上就是一个工厂，那么一会我们看代码能够深有体会。

右侧这个图是我从官网手册上截取了一个图，他演示的是 LC大概的一个原理，其中 spring container就是LC容器，那么容器帮我们管理BIM的前提，于是你要给他提供两份数据，两种数据，一种是你管理哪些病，这个病的类型你要给他第二个你要给他配置文件，给他配置，通过配置描述这些病人之间的关系，然后它能够自动的实例化这个病，然后让你去调用，所以说在容指标容器管理病人的前提下，这个病与病之间不会直接产生关联，他们的关联关系是体现在配置当中的，那么病人的耦合度就降低了。

好，这个思想还是有点抽象，没关系，下面我们写一些代码，通过代码去体会一下它到底是怎么回事。

好，下面我就打开这个idea，在写代码之前我先解释一下我们上次课所写的代码当中的核心的类，community application，就解释一下这个类它运行的时候它都做了哪些事情，然后你了解了大概的过程以后，对理解 LC也是有帮助的。

类其实代码挺简单的，类方法就一句话，spring application点乱，就是spring应用运行了启动了。

当然我们知道它的底层启动的Tom cat对不对？因为我们访问过，那么其实它的底层不只是启动了tomcat，除了启动他们开的之外，他还做了另外一件事儿，他自动的帮我们创建了这个spring容器，就是说在外部项目当中是不用容器，不需要我们主动去创建，它是被自动的创建的，在这个方法底层做了这样的事情。

然后 Space容器被创建以后，它会自动的去扫描某些包下的某些并将这些并装配到容器里，哪些病会被扫描到，我们得看这个类，它是乱方法的参数。

Spring应用在启动的时候，它是需要配置的，这个类其实就是一个配置文件，因为你看这个类上不有注解所标识的类，就表示说它是一个配置文件，我们可以点CTRL进到类的底层去详细看一下，这个注解是由其他注解所组成的，你看注解，spring boot condition表示类似一个配置文件，再看注解你内部outcome，微信比方说要启动启用自动配置，所以为什么我们几乎不用做配置，也能把服务启动起来呢？

因为它启用了自动配置，给我们配了很多东西，这个是组件扫描，那意思是它会自动的扫描某些报社的某些病，装配到容器里，所以说为什么它会自动装配？并因为你启动了组件扫描它会扫描哪些？B不是全部的，是这样，它会扫描配置类所在的包以及子包下的包，你想我们的程序当中像CTRL以后还会建别的包，所有的类都在这个包下，都是它的子包对吧？所以说所有的病都有机会被扫描到，这是第一。第二个还有一点，这个病上需要有像CTRL这样的注解才能够被扫描，如果没有这样的注解就不会被扫描。

那么和这个CTRL就是等价的功能一样的，还有几个注解，还有搜一下看还有 service在这，service就是说如果你加service注解也是能够让这个病人被扫描，然后你看这个service其实它是由 control的来实现的，所以说如果在bean加上complement也可以实现被扫描，还有一个叫 repository，它也是有可能的实现的，所以说你加它也可以被扫描，其实这个control也是有control的实现的也可以被扫描，就是说我们平时一共有4个注解，可以用你把任何一个注解加到一个类上，这个类就能够被容器所扫描了。然后他们的实现机制其实都一样，他们都是有恐怖的来实现的，所以说他们的功能都一样，那区别是在于语义上的区别，就是说如果说你开发的是一个业务组件，你最好用 service注解来标明，如果你开发的是一个处理请求的组件，你用CTRL来标识，如果你开发的是一个数据库，访问的组件，你用这个注解，如果你开发的它在任何地方都能用访问数据库处理请求或者是处理业务都能用，你可以用 computer通用的好了，总之这4个注解加到并上，使得这个并都能被自动扫描到，然后用哪1个都行，但是最好是按照刚才所说的建议去做，你一开始可能记不住，没关系，那么后面的话我们的课程里会经常的用熟了就好了。

好，那么关于 community application这个类，他大致做了哪些事情以及注解，他主要做了哪些事情，我们做了一个解释，让你知道了这个容器是什么时候被创建的容器中的，并是怎么来的，了解到这一点。那么LC的话，其他的内容就容易理解了。好了，那么介绍完以后，接下来我们就来写一些代码，来演示一下 LC的使用方式，这个代码我在这个测试类里进行演示，在测试类的演示。

那么我们正式运行程序是运行这个类，那么刚才也说了，它也是个配置类，那么一运行就以它为配置类来执行程序，我们测试的代码当中肯定也希望也用配置类，也希望和正式环境用的配置类是一样的，对不对？

怎么在测试代码中启用类作为配置类呢？其实也很容易就加一个注解叫context，computer reason，然后括号里面加上classes等于类型community application，点class，我们一会运行的测试代码就是以它为配置类的。

然后我们IOC刚才说了，它的这个核心是spring容器，而容器又是被自动创建的，我们怎么去得到这个容器其实也很方便，哪个类想得到spring容器，那么你就实现这样一个接口，叫application，context aware，然后容易有一个方法需要你实现。

这个测试代码我先不要了，一会用的时候再加上，这是一个赛的方法，传入参数，叫application context这个参数其实就是space，容器它是一个接口，看一下这个接口它继承于接口，这个接口要继承于它并factory，那么并fact就是并工厂，它其实 space容器的顶层接口，然后 application context是它的子接口，它比父接口扩展出了更多的方法，它的功能更强一点。

所以通常我们会用子接口、application、contacts。

好再回到测试类当中，那么如果说我一个类实现了接口的方法，那么使用容器会检测到，然后使用容器在扫描组件的时候会检测到这样的，并然后调用它的赛的方法把自身传进来，把容器传进来。

所以我们只需要把这个容器暂存一下，记录下来引用一下，我们后面的话就能够使用它了，所以我在这个类当中加一个成员变量，用来记录什么？容器空态。

好，加个成分变量，然后这里边把它记录一下。好了，那么当这个当程序运行的时候，Application就自动的被传进来，我就自动的做了记录，然后我在其他的地方就可以去使用了。

下面我就写一个测试方法，然后在这个测试方法里面去使用一下，十分容器这个方法名我叫做test application contact测试，spring容器在这儿怎么测，我就这样，我就直接把这个对象打印出来看，看它有没有值，看它是一个什么样的对象。

好，然后我就运行一下测试方法。

好，然后我们看通用台，你看它打印出来信息，打印出来的是类名希克的，这是对象的输出的形式。然后这个类型是generic web application context，这是实现类的一个类名，我们 application contact是接口好了，那就证明了这个容器是存在的，是可见的。

下面我们演示一下这个容器我们怎么去用它来管理BIM，当然你要用它管理bean，首先你得有bean让它去管理对不对？

下面我们就写一个bean，这样我再建一个包，比如说我要访问数据库，我要去查询某些数据，我再建一个包，这个包下专门存放访问数据库的那些个病，好这个包我叫做dao就是data access ok数据库访问的对象，然后我的包下创建一个访问数据库的接口，这选成接口，那么这个接口名字我们现在不是真正去访问数据库，就是一个演示，所以我就取得稍微随意一点这个名字叫阿尔法do。

然后这个接口我需要定义一些方法，因为是演示我也简单一点，定义一个非常简单的方法，谁来比如说我要查询没有参数，返回一个字符串就可以了。当然了，这个接口还不能直接用，你还得有实现吧？我再给它加一个实现类，那么我再创建一个该接口的实现类，这个类我叫阿尔法 dao，haven't apple假如说这个时间内我是采用haven't的技术来实现的查询，所以我叫哈尔滨的叫IMP I实现的意思，Ok。

当然它需要实现刚才的接口，阿尔法道增加接口的未实现的方法，这个方法我也简单一点，就直接返回一个字符串，habit完了就这样，当然这样还有所欠缺，你这样的话，这个病是不能够被容器所管理的，或者说容器是不能够扫描并装配它的，那么容器扫描并对什么样的病包下他满足了对吧？

其次它的上面有一个注解，所以说我给他加一个注解，那么访问数据库的b应该加什么？Reporter。好，这样就可以了，只要我们运行程序，那么spring容器会自动的扫描到这个病，把它装配到容器里，我们可以看一下，再回到 test方法。

好，那么我还是在刚才的测试方法里进行一个演示，这回我要从这个容器里获取它自动装配的病，好，它的类型我可以把它向上造型为接口，阿尔法道等于application，content，容器你获取病就调它的获取病的方法，get然后你可以通过名字获取，可以通过类型获取，我们通常可以按照类型获取，那么这个并的类型就这个类型，当然你写成接口类型也可以，阿尔法到there class，我要从容器中获取类型的病，得到了以后，那么我调用一下阿尔法到点select，我调用它的查询方法，并且将结果输出出来。

好了，下面我们运行一下这个方法，再看你看最终的输出结果还有问题，成功了对吧？我们成功的获取到了这个病，然后得到了查询的结果，没有问题。那么有的同学可能会有所怀疑，我倒是看明白了，但是这样有点麻烦，还不如我们自己实例化，这有什么好处，我们得了解它背后有什么好处。下面我再写一个病再写一个病，通过这个病你去体会它的优势。

比如说有一天我们这个项目发展到一定阶段，有一个新的技术诞生了，比如说买卖类似，那么假设它比heaven更为更有优势，我们在项目中想把 happiness替换为my best，在这样的项目中我可以这样做，我可以再新建一个到的实现类，而不用把它删掉，我叫阿尔法到I bet is apple也是实现类，好，然后也是要实现接口增加待实现的方法。

这个方法我要返回的是my best，说白了实现方式和刚才同时它也得加上注解，也可以被扫描到被抛弃的人。但是你注意这样还不行，这样还不够，你这样去运行刚才的程序就会有问题了，有什么问题呢？就是说我要按照类型去获取，并这个类型是个接口，那么这个接口满足条件的并有两个，一个是一个是它，对吧？

这俩都满足接口，所以说死病容器就不知道该给你哪一个病了，有歧义了，这个时候就会报错，这种情况怎么解决也好办，我只需要呢在我希望的病上再加一个主角 prime，加上以后这个病会被优先的装配，或者会当你调用的时候，它会优先给你，它有更高的优先级，好，加上以后咱们再试。

我再回到刚才的测试方法，然后再次运行一下，好，你看这回它打印的结果就是买卖点，所以你想一下，这就是它的优势，就是我的程序当中，原先所有的到的时间内都是由happen的实现的。

然后那么我调用这个并的地方会很多的，会很多的。那么当有一天我想把这个病替换掉的时候，其实很简单，我只需要再加一个病，然后多加一个promise，prime注解就行了。

那么我们调用的地方完全不用变，因为我们调用的这个地方依赖的不是这个病本身，依赖的是它的接口，那么这也是利用了面向接口变成的思想，我依赖的是接口，那么它实现类变了，我这是不用动的，这非常的容易好。



所以说space容器就通过这种办法降低了并执行的额度，我们调用方和时限内不会发生任何的直接的关系，是这样的。

但是这样可能会带来一个就是新的问题，比如说我程序的某一块儿，我还想用这个habit，我不想用 my best，就比较特别，我就这一块儿要用 habit，但是你这样获取的话，永远都获取的是my best的实现，我怎么能得到哈尔滨的实现的病也好办，其实病每一个病它都是有这个名字的，那么being的默认的名字是类名，首字母小写，那么如果你觉得这个类名比较长，你也可以拿去自定义一下他的名字，我们怎么定义一个病的名字，还是用注解加个括号，括号里面写上字符串就是并的名字，比如说这里我写上阿尔法，还有问题，这个病的名字阿尔法还有问题。

好了，那么给这个病重新自定义完名字以后，我可以通过名字强制容器返回这个并给我，我们来演示一下看看。

那么在这个位置我再重新获取一次病，阿尔法到等于 application context点get，并这回你要通过名字获取就不是类型了，那名字是阿尔法到阿尔法还不能听，然后他得到的类型是默认是object，因为他不知道是什么类型的，因为你这样get并没有给它类型，它返回的是一个object，你需要强制转型，或者是你可以再加一个参数。阿尔法到克拉斯意思是说你得到的 Offer价格你帮我转型成它也可以，然后我就再访问一下到时代，那么写完以后我再次执行这个方法好。你看那么第一次输出的是my business，因为它优先级高，第二次我是指定了病人的名字，那么就得到了哈尔滨的病，没有问题。

好了，以上就是我们从这个容器中获取病人的基本的方式，那么这个容器管理病不只是局限在能帮我们创建病，然后它还有更多的管理并的手段，下面我再给大家演示 spring容器管理病的更多的方式方法，那么使用容器它除了创新病之外，它还能够帮我们管理病的初始化以及销毁的方法，接下来我再写一个病给你演示一下，这样我这回就不用这个道了，我怕把道的代码改给它改乱了，我们再新建一个病，假如说我要开发一些业务组件，处理一些业务，我们通常会把业务组件单独放一个包下，这个包叫做service会有很多这样的组件，然后现在我在我就新建一个例子，比如说我取名叫阿尔法service，这个名字也是比较随意的，然后我也希望这个病是由通过容器进行管理，那么所以我需要在类上增加一个注解，那么它是一个业务组件加的注解，应该就是service。

好了，那么刚才我说了，说这个病我希望由容器来管理，不只是创建，它还希望容器管理它的初始化和销毁的方法，怎么管理？我首先给并增加初始化方法in it。这里我随便的打印一句话，叫做初始化阿尔法service。

那么要想让容器帮你管理这个方法，其实就像容器在合适的时候自动的调这个方法，你只需要在这个方法之前加上一个注解叫post construct，注解的意思是这个方法会在构造器之后调用，初始化方法通常是在构造之后调用用来初始化某些数据，为了便于观察这个方法，它调用的时刻到底是不是在构造器之后，我给这个类加一个构造器，好，咱们好对比一下。

构造器里我也是打印一句话，就是实例化阿尔法service除了管理初始化方法，还能够管理销毁方法，我再给这个类加一个销毁方法，叫destroy。

好，然后这里也打印一句话说销毁阿尔法service，然后这个方法上也需要加一个注解，然后这个容器会自动的调用它，那么注解是l destroy就是说的在销毁对象之前去调动它，为什么销毁之前，因为如果是在对象效果以后你就没法调了，对吧？

一定是在销毁之前调一下，你可以在这里释放某些资源，写完以后咱们测试一下，看一看我们通过容器能不能自动的去调初始化以及销毁的方法，我再回到测试类进行演示。那么我就再写一个新的测试方法，这个方法我叫test，并 many的就是测试一下病的管理的方式。

首先我要演示的是我要通过容器去获取 Service，看一下会出现什么样的情况，阿尔法service等于 application。Contact点get我们就按照类型去获取，因为这个阿尔法service就是它是一个类，我没有写接口，这样方便一点。阿尔法service their class。好，那么获取到 b以后，我去把 b就直接打印出来，看一下这个对象是否存在。好，然后我就运行一下，看一看会是什么样的结果看控制台最终他确实打出了这个对象，说明确实能够实例化，b没有问题。

然后我们看一下这个点，这个是程序启动的过程，输出的内容，包括 space容器的一个创建的过程。

然后在这个过程当中你会看到有这样的输出，实例化，阿尔法service，初始化阿尔法service，最后销毁阿尔法service，也就是说并的初始化方法确实是在构造期之后被调用的，然后销毁方法确实是在程序结束之前，对象销毁之前调用的，然后在初始化和销毁之间，然后我们打印出来病人的实力没有问题，好了，然后因为我们看到了这样的现象，就是说在程序启动的时候，BIM被实例化，然后程序停止的时候，BIM被销毁，能说明一个什么问题呢？

就说明这个病其实它只被实例化一次，它只被销毁一次，它在容器中只有一个实例，它是单立的，为什么这么说？你想象一下我们的外部程序是不是只启动一次对吧？Main方法只启动一次，所以说并被实例化一次，它就只有一个实例，不信的话我们可以再来一次，我再去获取一下这个病，再获取一下这个病，把它打印出来。

然后我们看一下它是实例化两次还是一次就能就知道了，这个结果我运行一下这个方法，首先你看打印结果，打印出来这两个对象好奇客户的是一样的，是一个对象。

在程序启动的过程中看日志，那么也只实例化了一次，就说明那么被容器使用容器管理的病，它默认是单个实例单立的。那么如果说你不希望它单立，你希望每次get并它都新建一个实例，也不是不可以也可以怎么办呢？我们就需要在并上再加一个注解，叫scope。它的作用范围是整个容器中有一个还是整个容器中有多个？死扣不它的默认参数就是辛苦的，心都疼，然后这是单例的，这你不用写，如果是想多个实例，你要写这样一个单词，PRO to tap。

当加上这个单词以后，我们每次访问遍，他就会给我们创造一个新的实例，咱们来试一下，再运行刚才的测试方法，好，你看这个控制台输出结果，这回就不是启动时初始化了，而是在具体我们盖的病的时候，每次盖的病他都实例化一个，每次盖的病都实例化一个，是这样，所以实例化两次就是两个病奇克的也不一样。

当然了，我们通常情况下都用它默认的方式，这个并都是用单例的形式，很少说有用多实例的形式比较少，基本上我们的项目中都是单立的，所以说我把注掉，以免让大家以为我们要经常这么做，以免对你造成误解。

好了，这是spring它能帮我们管理这个病，其实是管理病的作用域，它的有效范围这么一种情况。

那么以上我们用死病容器管理的都是我们自己写的病，都是我们自己写的类，有些时候我们可能希望这样做，我希望在容器中装配一个第三方的笔类是人家写的是在扎包里，这个时候我就不能像这样，我去类加上一个注解，因为那是人家写的类对吧？

你不能轻易改，再一个他打到扎包里去了，可能还没有源码，这种情况怎么做？其实也很方便，我们需要自己写一个配置类，然后在配置类当中通过并注解进行声明来解决这个问题，下面我就给大家写一个配置类，然后装配一个第三方的BIM。

好，那么我们项目当中会把以后会有更多的配置类，我们会把所有的配置类都放到同一个包下，新建一个包叫然后我在这建一个配置类的事例，也是没有什么业务含义，我还是叫阿尔法可能这一块。

好，那么当然了我们标识配置类也可以用类似于这样的注解，什么sprint application，但是其实没有必要，通常是程序的入口用注解，我们一个普通的配置类，一般我们用叫腾飞哥瑞森表示这个类是一个配置类，不是一个普通的类，然后你要定义第三方的BIM需要加一个并注解在方法之前，这里要定义一个方法，我要装成一个什么，并假如说我要把咱们Java自带的 simple date format装配到容器当中，因为simple的，因为我们在一个项目当中可能是几乎所有的地方日期的格式都一样，我把 simple对对方卖的实例化一次，装配到b类，然后可以反复用，达到这样一个目的。

好，我要装配的是c不对的放卖的，所以这个方法的返回的类型就得是same不对的方面，然后给它取一个方法名，注意这个方法名就是bin的名字，BIM的名字是以方法命名的，然后BIM类当中这个方法当中，我就实例化cmd的方面，给他指定一个格式，好这样就可以了。

那么这段话的意思就是说这个方法的返回的对象将被装配到容器里，这个病的名字，那么写完以后我们再回到这个测试类，我们去看一下能不能取到这个病，那么我就再写一个测试方法，这个方法名我取名叫taste，并挺big病的配置，我要从病中从容器中取的是cmd的方脉，先说明这个类型，我换个行按照类型来get，simple date for MAC点plus。

好，然后我就使用一下，我就用 cmpod的方脉去格式化一个当前的日期，看看能不能达到我的预期。好，那么我就执行一下这个方法。咱们看结果是当前的年月日时分秒。没错就是这样，好了，那么使用容器基本上就是这样去用的，但这种用法是我们主动的去获取容器，去从容器中的拿一个并过来去用，但其实这种方式很麻烦，那么这个死人不死不用的，还有更简便的使用一个容器的方式。

刚才我们PPT上不是说了，它的 Lc的实现方式是依赖注入，而我们现在主动获取感觉也不是主流，对吧？

所以说这种方式还是一个比较笨拙的方式，为什么要演示之前这种主动的获取的笨拙的方式，因为我是希望你能够理解那么死笨容器它到底是一个什么东西，它的底层到底有什么方法，我们到底是怎么去直接使用，这是它的底层的一些内容，你需要了解。

然后我们从使用角度来说怎么方便怎么用，下面我再给你演示一下，怎么用会更方便一些，什么叫依赖注入？

那么比如说我当前的病要使用阿尔法道，我没有必要去通过容器去get获取，我只需要声明，我要给当前的币注入阿尔法道就可以了。那么注入的话需要用一个注解叫 o to o to where然后加在一个程序变量之前就可以我加了一个阿尔法到程序变量之前，那么这句话的意思是说我希望十分容器能够把阿尔法道注入给属性，我就可以直接使用属性就可以了。好，咱们试一下，我再写一个测试方法，叫test。

Di是什么意思呢？就是依赖注入，对攀登c infection的缩写，测试依赖注入。好，我在这个测试方法里面直接使用这个程序变量，看看能不能取到这个病，我就直接把它打印出来看一下运行。好，你看打印出来结果是这个 my best这个时间内没有问题对吧？同理，那么你想获取阿尔法service，也可以这样做，然后你想获取 simply的放卖也是可以的，好，我在这也把那两个病打印出来看，再执行这个方法看，结果这个是my business实现这个是阿尔法service，这个是safety的方面都取到了，比方说没有问题可以用，然后注意还有一种情况，比如说我二发到我希望注入的不是 my best，不是默认的优先级，我希望注入的还是happened，怎么办？

这种情况我们需要再加一个注解，叫考利菲尔，括号里边写上 Bing的名字，阿尔法 happily，那么此病容器就会把这个病你认为这个的病注入给他，我们再来测试一下，你看这个病就变成了heaven的实现。

所以你看说spring他管理并就是用这样的方式，就是依赖注入的方式，我们只要声明一个属性，然后写一个注解，这个病就有了，我就可以直接拿来用了。然后像这种情况，我当前病依赖的是接口，那么底层的实现我是不直接跟它耦合的降低了耦合度，好，而且它也很方便对吧？不用你去实地化。

好了，那么依赖注入的基本的使用方式就是这样的，非常的方便非常的简单，我们通常都是这样用的，然后其实注入注解，也可以加在类的构造器的前面，通过构造器注入也可以或者是把注解加到赛的方法之前，然后通过赛的方法注入也行。

但通常我们都是把它加在属性之前，直接注入给属性，这样更简洁更方便，我们项目中基本上都是这样去做的，所以另外的两种方法我就不给大家演示了，如果你有兴趣的话，可以翻翻手册去自己查阅一下。

好了，以上是我们演示的一些语法小例子，我们在实际的项目的开的过程中怎么去运用，下面我们在综合一下来演示一下依赖注入，演示一下我们项目中怎么去使用这种思想解决问题。

那么我们的开的过程中是这样的，由controller来处理浏览器的请求，那么它在处理浏览器的请求的过程中，会调用业务组件去处理当前的业务，然后业务组件会调用deo去访问数据库，所以说是这样的，d controller调 Service调deo他们是彼此互相依赖的，他们的依赖关系就可以用这种依赖注入的方式去实现。

好，我们来演示一下，首先我打开阿尔法service，比如说我要调道，阿尔法道我就可以把阿尔法道注入给阿尔法service，当然你需要加 how to win，我希望注入默认的my best的实力就可以了，所以我就不加科雷菲尔了，然后加上一个注解，成分变量，好这就注入了。

那么我在处理查询的业务的时候就可以调用它了，下面我就写一个方法，模拟实现一个查询的业务，这个方法我返回死菌方法名叫犯的，然后这里我调的是阿尔法到来实现的查询，我直接把查询结果返回，这就是service依赖于到的方式。

然后 controller在处理请求的过程中可以叫service，那么同样的道理，我首先在 CTRL那里将service注入给他好，然后我这里写一个处理查询请求的方法，模拟处理一个查询请求，那么这个方法比如说我还是简单点返回字符串，因为是查询叫get查询某些数据，我就直接return阿尔法service点find，把 find的结果返回给浏览器。

当然浏览器要访问这个方法的前提是它得有注解声明声明它的路径就和一样，按照 copy然后路径我给它改名叫data写完以后，我们可以运行一下正式的代码，然后访问一下试试。

好，运行一下，运行完以后我打开浏览器访问一下，阿尔法 data，你看它的结果是什么？是买卖s怎么来的？是因为我掉了 service的饭的方法，而service的饭的方法调用的是 Dao的 select方法。

那么 deo我们注入的时候，注入的是默认的优先级的，也就是买卖点，实现了这个病好了，所以说你通过这个例子可能就更容易去理解我们之前所讲的 LC的概念。那么在这个思想的管理BIM的前提下，我们项目中的BIM不是我们自己去实例化，也不是我们自己去赛的，而是通过容器统一的管理，降低了病人之间的吻合度。好了，那么这次课我们关于LC的内容就演示到这里，咱们下次课再见。

## 1.4 Spring MVC 入门

---

4-Spring MVC 入门.mp4

说话人1 00:01

这。

说话人2 00:08

节课我们来学习spring mvc框架，这个框架是用来做外部开发的，之前我们也说过，所以我们在学这个框架之前，还需要先了解一下 HTTP协议，因为这个协议它是做外部开发的一个基础，整个外部开发基于这个协议进行的，HTTP它是一组单词的缩写，是这组单词， help text transfer protocol。

那么翻译过来就是超文本传输协议，超文本其实就是网页，网页就是超文本标记语言，所以这个协议是用来传输html以及和html有关的内容的一种协议，它是位于应用层的协议，它的底层是基于 TCP的，然后这个协议它大概规定了两方面的内容，第一个它规定了浏览器和服务端之间怎么去通信。

第二个它规定了你们双方通信的时候，这个数据的格式是怎么样的，那么具体的内容规定在官方的手册里可以详细的查阅到，那么从这个网站上可以看到，不过这个网站它的内容，它的手册写的比较学术化，很学术，然后也非常的长，所以阅读起来的话比较吃力不太方便，因此一般其实从事外部开发或者从事前端开发的人，他更愿意去看这个网站某z了，点org。

那么木子叶拉是一个组织，它是从 fax也就是火狐浏览器的团队发展而来的一个组织。

那么因为他们是做浏览器的，他对这个协议理解得很深刻，然后他们官网上也有对这个协议的解读，而且这个手册质量是非常高的，另外它还支持中文这一点，特别适合我们去阅读，所以今天我们就看这个网站的相关内容，好，我打开看一下，当然这个网站打开它默认的是英文的，然后在最下方最底下你可以选这个语言，我这是已经选好了，所以已经是中文了。

然后这个网站它里边是包含了html css和GS相关的手册，也包括 HTTP相关的手册，我们看的是 HTTP。

好，这个手册一开头对这个协议做了一个概要的介绍，然后后面有的是详细的内容，这个内容是很多，我课上没有办法给大家都看一遍，那么我们挑重点的就是看一看，了解一下，然后好学习后面的内容，那么其他的内容大家课后有精力，你最好自己去阅读一下。

好，我们看的就是 http 概述，主要是了解了这个协议它到底规定的是什么东西，这个内容也挺多的，我们主要是看这两部分，我们先看这一部分，这一部分它讲的是 HTTP 协议它所规定的浏览器和服务器通信的步骤，浏览器和服务器之间的通信大致分为4个步骤，第一步他们之间需要建立1个连接，这个连接是 TCP 连接。

第二个是浏览器要向服务器发送一段消息，这段消息叫 HTTP 报文，它的格式大体上是这样的，这个不用记了解。

第三个是服务器收到了报文以后进行解读，然后知道了你的诉求，于是它会解决你的问题，并且给你返回一段信息，这段信息也叫报报文，返回的报文。是这样的格式最后浏览器要么关闭连接，要么让这个连接保持开启的状态，然后一会儿复用，这个是得看这个协议的版本。

好了，这就是大概浏览器和服务器通信的4个步骤。然后下一小节又介绍了他们通信的时候传输的数据的格式，然后浏览器向服务器发送的报文叫请求浏览器向服务器发出的是请求，请求的报文大体格式是这样的，看这一段信息。

首先第一行这个单词指的是请求的方式，请求的方法，分为 get、post 等等，具体来说我们后面代码中会有演示，到时候你对他就有一个了解了。

这个是请求的路径，它默认的话就给了一个斜线，其实可能会更复杂，我们在项目中然后这个是请求的协议的类型，这里他给的例子是 1.1 版本。

然后还有一段信息，这段信息其实不只是两行，其实还有更多的行这个叫请求的消息头 head 头，然后它包括你请求的你的域名，你的语言等等，其实还有一些其他的内容。

然后其实还有一段消息就是你向服务器传送的业务数据，因为每。

说话人1 05:44

一个。

说话人2 05:45

团队开发的业务不一样，所以他没有办法预计你是什么业务数据，所以他就没有列举出来，但是还有一段是封装的业务数据叫请求体 request body，然后下面是。

说话人3 06:02

服务器。

说话人2 06:04

向浏览器返回的报文，报文叫响应报文或者叫响应的数据，那么响应的报文其实也是由三部分构成，这个叫状态行，这个叫响应的头其实还有一个响应体，响应体也是封装的是业务数据，他也没有列举出来。

那么状态行第一部分是协议的类型版本，第二部分这个是状态码，比如说 200 就表示说响应成功了，没有问题。

然后 ok 是响应的状态的一个解释，然后下面这一大段是响应的头，它里面包含了很多服务器相关的消息，包括响应的日期，服务器的类型数据，最后修改的时间，以及它的长度格式等等。

那就是大家先对 HTTP 协议它大体上有一个了解，那么没有记住没有关系，后面我们在开发的时候，当你哪块忘了哪块有困扰的时候，随时随地再来看一下，一来二去的话就熟悉了。

好，然后除了概述以外后退一步，那么针对这个协议的每一部分，它后面还有更详细的解释，比如说请求方式，这里有更详细的解释，比如说 headers 这里有更详细的解释，还有那状态码这里有更详细的解释等等，那么其他的内容我们就不详细看了。

然后。

说话人3 07:44

那么。

说话人2 07:45

当我们需要对哪一哪个地方做出更深入的了解时，我们再看。

那么关于这个协议，我们除了去阅读手册以外，其实我们通过浏览器的插件，或者是我们通过一会写代码，其实也能够看到他这个请求，或者是响应相关的数据，我们现在不妨就看一下，我们现在不是有浏览器吗？

我就打开浏览器的插件去看一下请求和响应的相关的数据，然后怎么看是这样的，右键检查，当然了不同的浏览器插件的使用方式可能会有所区别，但区别不是很大，我这里用的是谷歌浏览器，chrome，所以说我也建议你和我用一样的浏览器，这样的话操作方式就一样了，就比较方便。

好，打开了这个插件以后，那么要想看请求和响应的数据，我们需要点 net work，网络，但是目前它底下什么都没有，没关系，我直接刷一下当前的网页，你一刷网页，这个浏览器不就是访问了某贼拉的服务器，它就不是产生了一个通信，那就有请求数据就有响应数据我们就能看到，好我刷新一下看看。

好，那么你看底下每一条其实就是一次请求，这个请求怎么这么多呢？我一会再解释，那么我们先看第一个请求，好点一下，然后右侧打开的窗口就是与请求相关的内容，然后 general里边他列举的是请求数据和响应数据当中的一些关键信息，包括请求的路径，包括请求的方式，包括响应的状态，包括 IP。

说话人3 09:48

包括。

说话人2 09:50

其他的一些内容。

好了，然后还有他列举了请响应的消息头，risk boss响应还得是服务器给浏览器返回的数据的头部的内容。

那么你看它包含这么多内容，其实和刚才我们在那个文档里看的也是大体上是一致的。

然后还有request，harders请求的消息头就是浏览器向服务器发送的数据的头部的信息有这么多内容，比手册上的都很多好了，这些内容不用记。

反正有个了解，我们开发的时候其实也不用直接操作这些数据，他们都是自动的，但是你有印象，万一将来你有什么特殊的需求，可以需要用到这些数据的时候，有个印象会比较好一点。

除此以外，我们在这个地方点 response是响应的消息，就是服务器给我返回的消息，你可以看一下，其实它返回的不就是一个网页的代码，然后还有preview，你可以预览一下，响应的代码的效果是这样的效果，当前的网页这都没有问题。

其实我们通常看的就是 Heider cz，尤其是我们将来调试程序的时候，有些时候报错了，我们可能会看一下我们发的请求数据或者是想的状态是不是有什么异常，将来的话我们讲调试的时候再说。

好了，再回过头来再解释一下，刚才我们在 net work里看怎么有这么多次请求，这个我画图给大家稍微解释一下，好，我打开我的画图工具，那么假设这个是浏览器，这是我的浏览器，假设这个是我的服务器，假设是某贼拉的服务器，两者之间产生通信，那么浏览器访问服务器，服务器给它返回的是什么呢？

给它返回的是刚才我们也欲览看到了是一个html对吧？是一个html，浏览器得到这个aychem以后会怎么样？会进行解读，进行解析，然后去渲染相关的内容。

然后在解析的过程中，他会发现网页里这段代码引用了一个css文件那个地方，引用了一个js文件，还有地方引用了图片文件，所以它在解析网页的过程中会发现网页是依赖于各种各样的文件的，那么他发现了某一个文件就会再向服务器做一个请求，然后服务器给它返回一个它所需要的依赖的文件，比如说本次请求返回的是一个css文件，然后比如再来一个请求，服务器给它返回的是一个js文件，以此类推，就

是你网页上依赖了多少个文件，所以它后续还会发送的多少个请求，以加载这些文件。

对于我们来说，我们重点关注的是第一个请求网页返回来的有没有问题？如果网页没问题，这些个css js 图片通常问题不大，我们不用过度的关注好了，所以说这样的话就大概了解了为什么有这么多次请求，是因为网页上所依赖的资源造成的。好，我们再回到PPT，了解了这个协议之后，下面我们就可以开始学习 spring mvc了。

那么关于spring mvc，我在说他之前先说一下，我们做服务端开发，其实我们的代码是有层次的，你看上次课我们开发的时候不是加了好几层包，其实就是对服务端代码进行分层。如果你不分层，你用一个类去写也能写，但是这一个类当中做的事太多了，太杂了，耦合度太高了。那么当这个代码规模变大以后，它就变成不容易维护，甚至不可控了。我们分层的目的是为了了解耦，是为了让这个代码是更有利于维护。

服务端的代码通常是分为三层的，它是三层架构，分为表现层、业务层和数据访问层。那么浏览器访问服务器首先访问的是表现层，期待表现层给它返回一些数据，而表现层它会调用业务层去处理业务，业务层在处理业务的过程中会调用数据库数据层。

当然这个调用数据层可能是一个多次的行为，表现层访问业务层可能也是一个多次的行为，最终表现成得到业务层返回的数据以后，稍微加工一下就返回给浏览器，整个请求就会结束，这是服务端的三层架构。

那么spring mvc中的 Mvc指的是什么意思？它和三层架构有什么关系，这个有必要把它搞清楚。那么mvc它是一种设计模式，这种模式的理念是将复杂的代码分为三个层次，注意也是分为三个层次，所以很多人很容易将 mvc这三层服务端的三层的结构去做一一的对应，但其实是对应不上去的。

那么 mvc的m是model，模型层其实就是数据层，数据will是视图，control是控制，这三者是怎么协作的？是这样的。

Mvc它主要其实解决的是表现层的问题，然后他们的关系是这样的，当浏览器发送请求访问这个服务器的时候，它访问的是 control的控制器，仿的是组件，那么这个组件会接收请求中的数据，然后调用业务层去处理，处理完以后它会将得到的数据封装到model里面，然后把它传给这个视图层，然后视图层利用 model数据生成一个html然后返回给浏览器，所以浏览器最终是通过视图上得到了一个html总之CTRL是用来处理浏览器的请求的，负责调度的，而view只负责渲染，只负责展现，他们是各司其职，而他们之间联系的纽带是 model，是这样的一个关系。

好，所以你一定要记住咱们服务端三层架构分为这三层，而mvc这个模式也是分三层，但是这三层是集体解决表现层的问题，是这样的。

好了，另外这个是不是mvc这三层是谁去调的？其实它是不是mvc里呢？有一个核心组件，核心组件叫做前端控制器，那么。

说话人3 20:20

实际上。

说话人2 20:21

就是一个累就dispatcher，servlet其实comptroller也好，还是veal也好，都是由这个类去调度的，它怎么去调度的，有必要了解一下，因为这涉及到是super mvc底层的一些原理，我们在写代码之前最好把这个原理弄明白，Dispatch server类的和这些组件之间的关系，我们从 spring官方的手册是可以查阅到的，我们去看一下。

好，那么我去看一下 spring点io官方手册，然后点projects，spring for work。Learn好，我看最新的版本5.1.7，然后我们要看的是spring mvc的内容在这儿。当然了我们要去阅读大段的文字，内容太多了，我主要是想给你看一个图，通过这个图能够得到一些信息，也是比较直观的看一下，好，就这个图，它这个图大概的意思是 Dispatcher servlet是整个spring mvc的核心，那么 spring mvc当中所涉及的所有组件都是在他的管理之下的，你看他画这个框就表达了这个意思。

当然他管理这些个组件的前提是基于spring容器， application context其实就是space容器好，它能管理谁？它能管理我们所写的controller，他能管理我们所写的当然CTRL了能去调 Service以及 repository，其实业务层和数据访问层是通过 CTRL的他们去调的，然后它controller除了管理controller之外，它还能管理一个组件叫will resolve，其实就是视图解析器就是图层，然后还能管理一个组件叫handle my pin，这个映射的一个组件，其实就是我们挑一个路径，然后路径与control的相匹配，由谁来匹配？

其实这就是 handwriting，我们所写的注解，瑞克斯买品注解，其实就是由他来管理的。

好了，总之 controller试图以及。

说话人3 23:14

映射。

说话人2 23:15

相关的注解，相关的组件，其实都是由 despatcher servlet去调度去管理的。

但是这个图它有点笼统，它没有。

说话人3 23:24

划清楚。

说话人2 23:26

它们之间调用的顺序是怎么样的，只是有一个大概的说明。

说话人3 23:30

那么。

说话人2 23:32

新版新的手册不知道为什么把一个图给去掉了，但是在比较老的手册里面还有一个图更直观一点，不管出于什么目的，我们去查阅一下那张稍微早一点的图，也能够得到更直观的信息，我们再看一下，我后退一下，我看一下4.3.24，这里边就有那个图，好稍等。

然后它目录挺长的，因为它没有分组都揉在一起了，我搜一下我搜 m VC不在这儿，在这儿。就是 web mvc for the work点一下看看这个细节，文字我就不看了。

然后我主要是找那个图就是这个图，但这个图画的不好看，但是它的意思这个流程还是比较清楚的，比刚才那个图我感觉要更直观一点，然后他画的大方块，什么是serve layed？ Ange。就是servitor引擎，其实就是服务器，比如tomcat，然后这些程序都是位于Tom Ford之内运行的，那么浏览器访问服务器由谁来处理？请求有一个组件叫前端控制器来处理，前端控制器是谁？

它上面也有介绍，前端控制器它就是despite a salade，好了，那么所以请求统一都由前端控制器 despatcher servlet处理，然后 despatcher servlet会根据你写的注解，映射的注解，就是你的 CTRL上面的路径，或者是方法上路径去找到 controller去调用它，然后 controller最终会把数据封装的 model里返回给 controller，当然整个封装我们这次可能会有演示，这里就是封装了页面想要的的数据，那么CTRL得到这个数据以后，它会去调用 view template就是视图模板，然后并且把 moto给这个模板是一个是一个文档，然后它里面有一些特殊的符号，可以用 model中的动态的值去替换完以后就可以生成一个动态的网页，因为这个数据是动态的，替换完以后生成的网页是动态的，生成一个aychem以后返回给 dispatcher caelate，那么他再把 html返回给浏览器或者说响应给浏览器，返回响应的消息就是这么一个过程，当然这个过程是spring mvc它底层的一个原理，你最好能够理解，你最好也能够记住，然后可能在面试的时候需要你讲一讲，然后我们开发的时候其实不用对前端控制器做什么处理，它是自动去运行的，需要我们编码的地方就是CTRL了，就是这个模板就是 model这三部分，其实就是mvc这三部分，好了，我们再回到 PPT以上就是我对 spring mvc的一个解释，总之重要的有两点，第一点就是你要理解 mvc这个模式它解决的是哪个层次的问题，它是怎么解决的。

第二个，那么你要理解 spring mvc的底层核心的组件是谁？他是怎么去调度mvc这三层代码的，你要了解。



好了，那么我们之前也写过一些死，不然mvc的代码 CTRL我们所写的CTRL其实就是控制层控制器，但是我们还没有写过 view，这个视图没有给浏览器返回过，html这次课我们会做演示，但是如果说你想给浏览器返回一个动态的网页，我们还需要另外的一个工具去支持。

那么这个工具叫做模板引擎，模板引擎其实有很多的目前最流行的一个模板引擎就是叫time leave。好，其实无论是什么是什么类型的模板引擎，它的这个作用它的价值都是一样的，它都是用来生成动态的。Html的就刚才我也说了，最终的话模板引擎利用猫的数据给你拼一个html，然后反馈给浏览器对吧？所以它的作用就是用来生成动态的html的。

那么模板引擎是怎么生成动态的html看这个图，这是模板引擎其实就是一个对象，那么它要想生成html，你需要给它两个东西，第一个是模板文件，那么这个模板文件包含了网页基本的结构，除此以外还包含了那一些表达式，这个表达式就可以被model中的数据所替换，那么你猫都给它什么动态数据，它生成的html就有所变化，所以说模板引擎它需要两个东西，一个是模板文件，一个是猫的数据，最终它会把这二者合二为一拼成一个网页，是这么一个原理。

好了，那么众多模仿引擎当中 time leave比较流行用的比较多，为什么？因为他的理念比较先进，他倡导的是叫做自然模板，什么叫自然模板？就是以html文件为模板，其实其他的模板引擎它可能以其他类型的文件为模板，比如说很多人应该听过GSP，Gsp它是以点JSP文件为模板，不同的文件这有什么区别呢？

区别就在于 html文件，基本上做外部开发的他都懂，他都能看得明白，比如说我们服务端的程序员，我们开发这个模板，遇到点什么问题需要调一下样式，需要改一下标签对吧？

需要改一个GS可能有些地方水平有限，我要求助于前端工程师，前端工程师一看这是skml我能搞得定，你反过来的话你给他看一个JSP，可能他就不敢帮你去改，因为什么？他看不懂Jsp的文件，明白吧？

所以说你用 html便于团队成员之间的交流，谁都能看得懂。然后如果用其他格式的文件，那么就会增加别人学习的负担，你还得多学一种格式不是。好了，那么关于 time leave，其实别的模板引擎也是那么无论是什么模板引擎，我们重点要学的就是三方面的内容。第一个是它的标准表达式，就是页面上哪些数据哪些地方需要被动态的数据替换，那么这个是由标准表达式来解决的。

第二个是判断与循环，那么我们在渲染数据显示数据的过程当中，可能有的时候需要判断一下这个数据是不是空值，有的时候判断一下这个数据，比如说等于0的时候我要怎么处理它，等于一的时候我又要怎么处理，经常有这样的逻辑，所以判断是很重要的。另外还有循环，因为有可能model给我的数据是一个集合或者是一个数组，我需要循环的去处理，所以判断与循环也非常的常用。第三个是模板的布局，因为。

说话人3 31:47

你。

说话人2 31:48

看过咱们我给你发的，当然我传到网站上的。

静态页面的时候，你会发现其实很多页面它的结构是非常相似的，上中下上边和下面还都一样，我们怎么去将一样的区域能够复用，这就是模板布局所做的事情。

总之无论是什么模板引擎，那么这三方面都是我们学习的重点。

这个课上我们后续都会陆续的用到，然后 time leave官网是这个网址我们可以进去先看一下，好这是time leave的官网，它就没有中文了，就只有英文了。然后我们要看的是文档，你就点 DOS，然后你可以点 read online，在线阅读，然后这就是它文档的详细内容。

这个内容是非常多的，那么我们课上就不去一点一点的看了，反正它的核心的内容重点的内容我都进行了提炼，然后会运用到项目中，那么常用的内容我们在项目中就都会学会，那么其他的内容有时间你自己看一看好了。

说话人1 33:10

那么。

说话人2 33:11

关于spring mvc我们入门需要了解的一些的概念，我都已经介绍过了。

下面案例演示的环节，我打开这个idea给大家演示一些案例，让你去从代码的层面去深入的理解 CMV它到底是怎么回事。好 ID我已经打开了，那么在写这个代码之前，我想先对 time leave做一个配置，在 application的practice里做配置配什么呢？我想把 time立法它的缓存关掉，它默认是启用缓存的，因为有 simple的他们自动的配置为什么要把 timeless的缓存关掉？

说话人3 34:01

因为如果模板。

说话人2 34:03

的缓存你不关掉，那么它就有缓存以后可能是我改了页面，然后你看的还是缓存的内容还是旧的，内容没有刷新，因为它有一个。

说话人3 34:13

延迟。

说话人2 34:15

所以说我们开发的时候模板的缓存最好是不要开启，要不然麻烦。

但是我们系统上线以后，模板的缓存就应该开启，因为它会有缓存吗？会降低服务器的压力。然后 time leave的缓存的关闭，我先写上这样写。是spring点儿，time leave点儿开始等于force。好，顺便我想讲一下配置文件为什么是这样写的？就这个字符串它代表什么含义？为啥是这样的？包括以前我们写的为啥是这么写？那么 Application叫practice里，它各种配置，其实我们从spring boot的官方手册上，它有一个附录，能够查到它针对每一个方面的配置，常用的配置都有一些示例，然后你通过示例能看到，我们去看一下，我再访问一下 spring点io然后我看的是 spring boot对。

2.1.5，好了，那么他常见的配置的举例是在附录里，是在比较靠后的位置，我找一找非常靠后了，它目录很长附录。

好点进去看这段代码它所有常用的配置的一个列举，当然并不是全部的常用的基本上都包含了，然后如果你想找time leave，你可以搜一下，搜一下很容易就搜到了 time leave。那么你从这就能够找到，你看 time leave它的缓存是怎么启用或怎么关闭的等等。当然其他的配置也有就很多了，所以说如果你想知道哪方面怎么配，就查这个手册这一部分去看就可以了。

然后再说一下我们写的配置其实是配的是什么，其实是给一个配置类注入数据是给一个配置类，注入数据，数据配置类是什么？它这有介绍，比如timeline复制，timeless output condition我们可以看一下类，数据是给一个配置类注入数据，这个配置类是什么？它这有介绍，比如timeline复制，timeless output condition我们可以看一下这个类可以搜一下 Ctrl n售假贪利凹凸consideration好放大一点。

这个类它有一个注解，说enable configuration practice配置的practice文件，指的是time live practice，其实这里是详细的参数的配置，所以你还得看 time leave practice，那么time leave它是给了一个out to consideration，通过out to condition找到什么practice，而其他的其实他都给了什么 practice比较直接了，这个有点绕。

好，其实我们要看的就是这个东西，time leave practice，然后这个类上写了这么一个注解，注解上写了说 practice等于这句话也就是说我们在配置文件里写 spring点time live，其实引用的是类 bean，然后我们写点catch，其实是给这个病的 catch、属性、赋值，所以这个类其实就是time leave的配置文件它默认的catch是two，然后通过配置文件就可以改变这个值。

当然了同理你也可以写 spring time live点其他的属性，改变其他的属性的值。

同理这个server也是类似的，这个server其实它的配置文件配置类是谁也可以从手册上找到，我估计是在比较靠前的位置，以server开头的，找到了这个server点什么server的配置对应的是类 server process我们也可以看一下，看中文搜一下server practice。

你看它的前缀就是server，所以你server点给里头配置，server点part改的是然后server点servitor搜一下，server点serve late。

其实改的是它，而它又对应的是一类，所以搜点搜类的其实指代的是这个类，然后 contact pass指代的是类的属性，所以你看其实他这个配长其实很简单的，其实我们这样写给某一个并注入值的意思，好了，我把具体它是配置哪个类在这里写个注释，课后你也可以自己去看一看，然后这个是time leave practice好了以后，那么再有这样的配置的时候，我就直接把配置写出来了，然后并且注释里写上配置它对应的配置类是什么，具体配置类是什么样，你自己去看一看。

因为通过这一个一次讲解，基本上原则你就应该能掌握，能明白了，以后的话自己就能看懂好了。

那么配置好以后，下面我就要开始写代码去演示 super mvc的一些常用的语法，那么这个代码我们需要写在哪儿？因为spring mvc刚才不是讲了吗？它解决的是视图层的问题，而这个视图层我们第一个要写的是controller对吧？当然controller会组织数据封装到model里，Model那个类本来就有，你不用写，我们可以拿来用，另外还要写这个模板引擎所需要的模板，这个模板放在哪儿放在这儿，所以视图层的两部分代码，一个是在这，而这个是业务层，这个是数据访问层，就是这样的一个结构。

好，我们这次课是小的例子演示，它的语法我们就不去调service，不去调到了，只是一个事例而已。好，我再打开阿尔法CTRL，然后在这里边给大家做一些实例的演示。那么首先给大家演示什么？演示一下我在spring mvc框架下，我怎么获得请求对象，怎么获得响应对象，请求对象显然里边就封装了请求的数据，响应对象就封装了响应的数据，他们分别能够处理请求处理响应。

当然了这个是比较底层的行为。

实际上 Spring mvc对它做了一些封装有更简便的方式，但我想先讲，因为我想你最好能够理解比较底层的一些机制，那么我们简单的方式都是对就是这种麻烦的方式做的封装，或者说对这些底层的组件对象做的封装而已。

好，那么我就写一个方法，这个方法的访问路径先声明一下，request卖品就叫HTTP好，然后这个方法我先不写risk boss body了，然后写怕被y然后就叫HTTP，好，为什么没有法规的类型了？为什么没有返回值了？因为我们通过 Response对象可以直接向浏览器输出任何数据，就不依赖反馈值了。

那么如果你想获取请求对象，响应对象只需要在这个方法上加以声明，你声明了这两个类型以后，那么 dispatcher servlet去调这个方法的时候，就会自动的把那两个对象传给你，它在底层就把那两个对象创建好了，request，对象的常用的类型，其实它是接口，好几层接口，我们常用的接口是 HTTP，servlet request，请求对象放大一点，然后响应对象和它类似HTTP servlet response。

好，request、请求，response响应。好，那么我们在这个方法里就利用这个对象处理一下请求，处理请求就是读取请求当中所包含的数据。怎么读，我给你做一个演示，先写个注释，我在这要获取请求数据来利用这个对象，然后取到数据以后我就直接输出了get master获取请求方式。好再来它有很多数据我举几个比较常见的 get。So really pass就是请求的路径。

好，然后还有请求行有很多数据，那么他做了一个封装，我们可以通过request点get hide names，得到所有的请求，行的key是key value结构，所有的key我们得到的是一个迭代器，迭代器是一个很老的迭代器，叫e number瑞士，然后它里边 key是一个字符串类型。

In, number reason，其实它的功能其实和艾特瑞特是一样的，只不过它比艾特瑞特更老。现在不推荐使用了，但是 Request对象很早就有了，所以它一直用的是这个类型没有变。那么对于迭代器对象，我们需要通过while加以遍历，elaboration点儿还small，elements是否还有更多的元素？如果有的话，我就从中获取enough reason点。

Next element就取到当前的一个值，取到的是什么呢？是一个k其实就是请求行的一个名字，name，然后Name有对应的value，这个value我们通过request可以得到也是字符串，request.getHeader，你把name传入会得到一个value，然后在这我把key和value一起输出来看一下，中间我们用冒号分割一下。

好了，这就是请这两个数据其实是请求行就第一行的数据，中间的这些这个是请求的消息头若干行的数据，除此以外其实还有请求体包含的业务数据，其实包含的参数，一会我在访问这个方法的时候随便传一个参数进来，我们假设传一个扣的进来，我通过request怎么得到这个参数，其实非常方便演示一下。

就像request的getParameter参数的意思吗？参数名叫code就写code就可以了。所以以上就是我们通过request对象获取相关数据的办法，主要是这么几种办法。然后下面我再演示一下Response怎么用。Response是用来向浏览器做出响应的对象，它是给浏览器的返回响应数据的对象，我也给你演示一下返回响应数据，那么返回响应数据首先response你要设置一下返回的数据的类型，你是返回一个这个网页是返回一个普通的字符串，返回一个图片还是什么？

这里比如说我要返回一个网页，你得这样写，text/html是网页类型的文本，然后分号再加一个charset，就是它的字符集utf-8，这样好像网页能够支持中文。

然后那么要想用response向浏览器响应，网页其实就是通过它里面所的输出流，向浏览器的输出就是了。ResponseWriter我们可以通过它获取到一个输出流，它底层的自动建好的，不用你去自己创建，然后得到的是一个PrintWriter、类型。

好，那这个需要处理一下这个异常我就给他去开始一下。好了，然后。接下来我就通过这个response向浏览器打印一个网页，怎么打印怎么输出是write。然后比如说我要输出一个一级标题，我就可以这样写，输出一个一级标题留客网吧，当然你这样写这不是一个完整的网页，应该是在它之前你还要输出黑的head body这样的标签对吧？在他之后还得有body结尾，还有更多其他的内容。

总之网页很复杂，标签很多，我们就是这样一行的去write的，所以你看我们直接利用底层的对象去做这样的事情会非常的麻烦。但是你了解了以后就知道了它封装的底层到底是干了什么事，其实他就是干了这样的事情，我就偷懒了，我就不输出完整的网页了，因为我们将来不会这么做，只是给你这个演示一下而已。

最后你流你用完以后把它关掉，放在那里把它关掉。

不过Java7它有一个新的语法，你在try的后面加一个小括号，然后把这个response在这个小括号里创建，那么编译的时候它会自动的加一个finally，然后把writer在finally里的close，这是Java7的新的语法，这样的话会让这个程序更简单，就省得写finally，当然前提是你这个writer必须得有close方法，当然这个肯定是有，好这个方法就写完了，写完以后我们启动一下这个程序，咱们看一下，我选中主要的类核心的类，然后点启动，启动以后，那么打开浏览器我来访问一下刚才的方法，community阿法，然后HTTP你看返回的就是牛客网，对吧？

没有问题。

当然你也可以右键检查，在插件里更仔细的看一下，还是选network，网络刷新一下，看这个请求，我们只返回了一个请求，因为网页上没有依赖任何资源，非常简单，然后你看这个请求看响应的类型是html看响应的具体的代码，一级标题，相当于是一个简化版的网页，再回到开发工具。

你看一下控制台它所打印的请求的数据，你看这个get咱们第一个不就是请求方式吗？Get请求，然后请求路径是阿法HTTP没问题，然后中间循环输出的是这个请求的消息头有非常多的数据，我们就不用去细看了，当然我这是两次，第二咱看第二次，然后路径中间有一段是请求消息头，最后业务数据参数是空，因为我没有传参好，我传个参怎么传很简单，在路径后面拼一个问号，然后扣的等于比如说123就可以了，这个问号就代表后面是带参数的，扣的是参数名，123的是参数值，当然你也可以带更多的参数，如果再来一个参数你就中间加and比如说扣的，比如说name等于张三，假设是这样，回车看控制台，你看控制台这回输出的扣的是123没有问题，当然内部我没处理，就是这样。

好，总之 HTTP 协议当中的请求数据，响应数据，我们通过浏览器插件能看到我们在服务端利用这样的对象也能得到，那么这是我们对底层的对象的一个直观的了解。后面我会给你演示它封装之后的更简便的处理请求的方式，处理请求的方式是什么样。

好，下面我们就立刻来演示更简便的方式怎么做，那么我们处理浏览器的请求其实分两个方面，一个是我要接收请求的数据，一个它基于 request，一个是像浏览器返回响应数据基于 response，那么因此我们就需要学习两方面的内容，怎么接收请求数据，怎么返回响应数据，那我分开来演示。

我首先先给大家演示我们请求的数据我们怎么去处理，就是用简便的方式。

好，那么首先给大家演示一下这个 get 请求怎么处理。Get 是获取的意思，get 请求一般用于获取某些数据，当我们向服务器获取某些数据的时候，通常是盖的请求，而且我们默认发送的请求就是盖的请求，就像刚才我发送的请求，get 请求。

因为你看刚才我们打印出来的这个值不就是 get，默认就是 get。

好，我举个例子，比如说我要做啥，比如说我想查询所有的学生，假设我要查询所有的学生，比如说我们查询的路径是这样的，students 然后问号查询所有的学生学生可能会很多，我希望分页显示，而分页的时候就需要带上一些条件了，你要告诉服务器我当前 carry 的当前是第几页，比如说等于一当前是第一页，然后我每一页最多显示几条数据，比如说 20 那么分页条件需要两个条件，一个是当前是第几页，第二个当前最多显示多少条数据，就这样传过来。

我这个服务器怎么处理，我就给你演示一下。

好，我还是先通过瑞克斯麦平声明一下请求的路径，可以像刚才这样声明，但这种是简化的方式。其实注解里边还有很多的参数，可以有更丰富的用法，我给你演示一下，你可以这样通过 pass 来指定路径，我说的路径不是 students 对吧？Students 访问路径是 students，然后我可以再加一个 method，这个参数声明一下请求的方式，request，method 点 get。就是我强制这个方法必须是 get 请求才能访问到。

其实像之前的方法我们没有加这样的限制，那么如果是 post 请求也是可以进来的，但我这样限制以后，他就只能处理该的请求了，这样比较合理，就是说你明确你自己到底要处理什么请求，这样比较合理，因为请求方式有很多，如果什么都能处理的话，其实但是你实际上又只用了其中的一种有可能会有漏洞，所以说最好是这样指定清楚，然后我们关注的是请求，那么响应的话我就用简便的方式，我还是用之前的 response body，然后返回一个简单的字符串就行了，后面我再单独演示响应怎么做好，那么写成 public 返回字符串词句，然后这个方法名我叫 get students，最终我就返回随便写个数据上 16 等次。

意思一下，我们重点是关注我请求当中带的这两个参数，我怎么取，其实很容易，你只要加一个参数就行了。比如说我想获取这个参数，我就 int carrot 就行，我要取这个参数我就再来 internet，只要这个参数名和你传过来的参数名保持一致，其实就这样就能得到。

那么 dispatch server 的检测到这个参数以后，会把 request 当中的与之匹配的参数直接复制给他，这样其实就可以，但是有些时候可能比如说我初次访问学生页面的时候，可能我没有加这个参数，就有的时候这个参数可能是没有的，如果没有的时候这个值怎么处理？

如果你想处理这种情况可以加上注解，再进一步就对它做一个更详尽的处理注解，可以这样写。

Request, Paro, 请求参数，括号里边我们可以写 name，current 意思是 request 当中的名为 current 的值，给这个参数，current 与它对应给它，复制给它，好，然后还可以再加一个条件叫 require 等于 force。意思是你也可以不传参数进来，没有问题。

好，然后可以再写一个 default value，如果你不传的话，默认值是几？默认值是一，我们可以通过 request、panel、注解，对这个参数的注入做更详细的声明。

同理 Limit 也是这样的，我再给他 copy 一份，这个就得叫 limit。Request 当中名为 limit 的参数，复制给这个参数，然后它也是可以传，如果不传的时候它默认是 10。好，我们重点就看一下这个参数我能不能得到，我在这里打印看一下。打印 carry 的。再打印里面的。好这样就可以了。那么写完以后我需要重新编译一下这个程序，你可以按快捷键 CTRL 加 f9，然后看一下它运行的情况有没有，它已经重启了，然

后我就打开浏览器去访问一下阿尔法，然后是16.4返回的，结果这个字符串没有问题，我们主要是看控制台，你看我刚才的路径里没有敲条件对吧？

所以它默认是1和10默认值有效了，我来把这个条件输入一下，看看问号，开问题等于3，and limit等于50，好，再看控制台三五十，所以你看在spring mvc的框架之下，我们想获取浏览器中的一些参数，在该的请求当中就非常的容易非常的方便，就是这样去做。

那么除了这种方式以外，其实还有一种方式，当然也是该的请求也是比较常见的，比如说我不是要查询所有的学生，我要查询一个学生，根据学生的ID，根据学生的编号查询一个学生，我这个路径可以这样设置 Student，但是你可以这样，比如说ID等于几，我们通常也这样做，就16档的斜线，然后加一个编号123，直接把参数编排到路径当中，成为路径的一部分。

当这个参数成为路径的一部分的时候，就不是这样获取了。怎么去获取？再给你演示一下。我再写上瑞克斯卖品注解，声明它的访问路径是student斜线，ID Id表示指代的是它是个变量，用大括号括起来，指代的是这一级数据，然后逗号我再要求他的请求方式也是那么响应我们还是用 response body返回一个字符串就完了，那么增加一个方法返回试卷，方法名叫get student反馈一个学生。

好，然后我就直接把反馈值先写上，16的一个学生。那么现在的问题是我怎么得到这个参数其实也比较容易，我们需要用另外的一个注解去搞定，这个时候我们需要用主角叫pass where路径变量，然后括号里面写上变量的名字，它不就叫ID对吧？然后我们注解会帮我们到路径当中得到这个变量，然后复制给你的参数，我们这个参数我还叫ID，这样就可以了。然后我把 ID打印看一下，看能不能得到，写完以后再重新编译 CTRL f9好。

然后我打开浏览器，访问一下刚才的查询一个学生的路径，student然后比如说101回车，结果2460的没有问题，看控制台101没问题。

总之我们在get请求当中有两种传参的方式，一种是问号拼，一种是把参数拼到路径当中，这两种方式获取参数的方式有所区别，用不同的注解，大家要了解我们后面的项目中都会用这种有些时候适合用这种具体情况，我们到时候再具体来说了。

好，那么盖的请求是向服务器获取数据，那么如果浏览器是向服务器提交数据，其实有好几种请求方式都支持，但通常我们都用post POS的请求就是浏览器向服务器提交数据的时候使用的，那么其实虽然说请求方式有好多种，但是我们只需要用到get和post就能解决一切问题，我们的项目中就只用这两种请求就够了，其他的我们就不用了。

那么至于其他的请求，他可以做什么，你可以去看看手册做一个了解。

那么接下来我就给大家演示一下 pose的请求，当浏览器向服务器提交数据的时候，这个时候我该怎么处理，我该怎么获取它里面的参数，那么浏览器要想上服务器提交数据，首先浏览器它得打开一个带有表单的网页，你通过表单填写数据以后才能提交给服务器对吧？

现在我们还没有网页，我就在项目中创建一个静态的网页没有变化的。

注意我们在time place当中存放的是模板，其实是动态的，因为它里边有动态的这个表达式，是能够被model中的数据替换的，我现在要创建静态网页，静态的资源要放到时代的一个之下，你静态网页静态的图片，静态的CS文件，GS文件都放到这里来。

那么这样我就在这个时代之下建一个目录，你有一个directory就叫html我要在这里创建一个静态网页，那么这个网页 new html我取名叫student，因为比如说我要新增一个学生，然后发送给浏览发送给服务器。

Ok。

好，网页的基本结构是有了，标题我改一下，叫做增加学生，然后我要在报道之内就加一个表单放表，单里需要有一些

文本框用来输入数据添加一个段落，然后比如说我要输入学生的姓名加一个文本框text，然后要注意文本框里的数据，你填完以后，当你提交的时候，你这个数据传给服务器，你需要给它取个名字，那么我们通过文本框的name，属性给它取个名字，这个名字我就叫name，参数名就叫name。

好，再来一个学生除了有姓名再来一个年龄，也是一个文本框类型，也是test参数名，我叫edge。好，就填两个数据，然后你还得点一个按钮才能提交数据对吧？这个按钮是一个300台按钮，提交按钮的名字，我取名叫保存。

好，那么这个表单要提交给谁，用什么样的方式提交，你还得加以声明，你要写成master的请求方式是post，我想要post的请求提交数据，然后提交的路径是action，action说明提交的路径那路径是写上community阿尔法，然后student community阿尔法student就是这样一个路径。

好网页我暂时先写好了，能不能访问咱们试一下，我们先不着急去写CTRL了，我把它CTRL f九重新编译一下。

好，那么编译之后我去访问它，你注意你访问它的时候，它是放在Style的一个目录下，style这一级路径你不用敲，但是style下级h开发路径你需要敲，怎么敲呢？

我给你演示一下就community然后下面的话我要访问的是静态页面，就不是那个CTRL了，就不是阿尔法了，它放在的是html路径下，名为student的html，你看我就打开了网页对吧？剩下就填数据了。我先不着急填，因为我还没有在服务端加路径已处理这个请求，有人说你不有Studio的路径，有吗？看一看有吗？没有，这个是斯诺登的不假，但是他下有个下级，而我们刚才所声明的路径只有斯诺登的这一级跟他层次不一样。

下面我就再写一个方法来处理pose的请求，当然我这里再强调一下，其实我们通过盖的请求也可以向服务器传，数据其实也可以把name is传过来，但为什么不用这个get呢？因为get请求传参的时候，这个参数是在明面上，即便是你保存，它也会在路径带上问号什么？一长串，这是其一。

第二个你用这个路径上面带参数，这个路径的长度是有限制的，它不是无限长的，所以说如果你这个参数有很多项的话，盖的请求传不下，所以盖的请求传站一个是站明面上，第二个它传的数据量有限，所以我们通常提交数据的时候就不用get请求，你要注意，我就再新写一个方法来处理post的请求，那么声明访问路径pass等于student然后强调一下，那么请求方式是request，master的点，post这个是一个post的请求，那么响应我还是简单处理，response body返回一个字符串算了，那么当前的方法的目的是为了保存增加学生，所以我叫save student。

好，我先把这个字符串的返回，比如说我就保存成功了，我就返回一个success。那么这个方法怎么去获取post的请求当中的参数，其实很容易直接说明参数，这个参数的名字与你表单中数据的名字一致，它就自动会传过来，就这么容易。

那么我就增加一个使用类型的参数名为name，我再增加一个int类型的参数名为h和表单一致，它就能传过来。好，下面我再打印出来，咱们看看，好，我们就试一下，重新编译。那么编译完以后，我再回到刚才的页面，我得刷新一下，然后填个名，这里孙悟空就孙悟空年龄15就15，然后点保存success。那么看控制台孙悟空15没问题，你看我们提交数据就是这样的，简单非常的方便，当然了实际上你也可以像这种方式就加Require注解，但其实我们通常都不加，只要这个名称对应它自动传过来就可以了。

好，那么以上是get请求，post请求的处理方式，那么请求的处理方式我就演示完了。

接下来要演示的是我们如何向浏览器返回响应，数据响应是我们关注的焦点，那么之前我们都是响应字符串，这种比较简单的内容，下面我就给你演示一下我怎么向浏览器响应的是一个动态的html，响应html数据，那么假设我要浏览器，我要查询一个老师服务器就帮他查询到了一个老师相关的数据，就要把老师相关的数据响应给浏览器，当然我们想的是一个网页的形式，我们看一下怎么做。

好，那么首先我还是声明一下，访问路径pass等于提成我要获取一个老师，然后请求的方式get因为是获取get，这个时候你要返回html，你就不要加注解了，不加注解默认它认为就是返回html。

好，这个时候你返回的数据类型就先不要写成string了，我们通常写 model and view类型。换个名我就get提成，返回的数据类型这个对象你一看就能看明白，它返回的是model和view，两份数据，我们之前不是讲了spring mvc的原理吗？所有的组件都是由dispatcher servlet去调度的。那么despite serverlet会掉 CTRL的某一个方法，这个方法需要给它返回猫的数据，也需要给它返回视图相关的数据，然后他把model和视图这个数据都提交给模板引擎，然后由模板引擎进行渲染，生成动态的html，对吧？

所以说对象里面封装的就是我要给dispatcher servlet返回的model和view两份数据。

好，那怎么去做非常容易，我先实例化这个对象 model and view实例化6等于6model and没有然后我需要往里传动态的值，假如说老师的名字，老师的年龄，我就model and view点I的object，然后参数名叫做name，参数值我就写死了。

张三儿。再来一个model。Under view点I的object配置，比如说30以此类推，就是你那个模板里需要多少个变量，你就传多少个数据进去就可以了，那两个就已经能说明问题了，然后你还要往这个对象里设置一个模板，它的模板的路径和名字。那是这样社长。Model and view。点site, view, name。好，那么这个模板我们需要把它放在哪放在下。Time please这个目录你不用写，它的下级目录你要写，比如说我一会儿会把模板放到template下面的一个demo的目录下，所以我这个demo要写上，然后后面要写上这个模板的名字，什么点html因为time leave引擎它默认的它的模板就是sm文件，怎么说是一个固定的这个方式，所以说后缀不用你写，你只要写上文件名就可以了。

比如说这个文件名我叫will，就这样写就可以了。

那么你要知道 will其实指的是view的sm file，好，写完以后他要求返回这个对象，我就返回model and view就行了，那么这个方法就完成了，但是还缺一步你这个模板需要写出来，那么才能够从这儿走到那个模板，才能浏览器才能得到一个html那么我下面就需要在 times的下面创建一个目录，叫demo，然后在demo的下面创建一个 html，叫will，注意这个html不是静态的，是timely模板，那么我们需要进行一个声明，你得让这个服务器知道它是模板，而不是普通的html那么首先你要在这写上这么一句话，xmlns等于th等于HTTP冒号，双曲线三w点thymeleaf点org。

这句话就声明说我当前的网页是一个模板，这个模板的语法是来源于他们利福官网是这样，所以要做这样一个声明，声明完以后，你在这个页面上可以写出你想要表达的内容。

比如说我就希望写两个段落，第一个段落显示姓名、老师、姓名，然后第二个段落我要显示的是这个老师的年龄，但是我把标题改一下，改成teacher。好姓名和年龄是动态的值，我们需要利用模板引擎的语法去解决怎么写是这样的。我在p上写th Th其实就是time leave的缩写，冒号，然后 text等于意思是什么？就 P的文本应该是等于这里边的一个变量，这个变量你要用表达式来写，到了大括号name，这个name就是从model中取的 name的值，总之你这个是 time leave的语法。

那么这句话在模板引擎渲染网页的时候，它就会获取变量，然后放到 p里。好，你这个会的 a的就跟它一样对吧？也是这么写，只不过这个是a的变量。

好，那么完成以后我再编译一下这个程序，然后我们试一下路径，叫teacher试一下，这个是committee的阿尔法，teacher你看我就得到了网页张三30，但这个网页比较简陋，看起来好像不像是个网页，你可以检查通过插件去看刷新，你看teacher这个请求再刷新一下，你看teacher请求它的类型是document，document就是网就是文档，就是网页一点进去看，他想用的类型是html对吧？

然后看响应的具体的内容也是网页的格式对吧？你看 P ta是没有了，那么变量跑到了p的内部，这个是由谁来做的？就是模板引擎来做的，就是这样一个开发过程。

好，那么实际上还有一种响应的方式，也可以返回网页，然后比略简单一点，但是差不多也给大家演示一下，比如说这个例子是什么，比如说我要查询学校，跟这个类似我要查询学校，我还是要写上声明一下这个路径， pass等于辞工，也是该的请求查询学校，然后这个方法我就返回10卷，这个方法就不返回model and view了，因为简化了，然后方法名我叫get持股。



好，然后我把瑞特写一下，rett什么，其实它Ryton就是veal的路径，我还是利用刚才view的html，所以路径就是它。你注意如果我们返回的类型写成死菌，它指的是返回的是will的路径，那么model数据怎么传？因为它返回了死菌这里不可能带model的数据了，Model怎么办？我们需要加一个参数，类型就是model，model对象不是我们自己创建的，是dispatcher servlet，在调这个方法时，他一看你有model对象，它就会自动实例化model对象传给你，然后因为这个是一个并是个对象，那么dispatcher他是持有对象的引用的，那么你在这个方法内部往这个对象里存数据，那么它也能够得到好，所以说我们就往里装数据就可以了，也就是说其实这个方法和上面这个逻辑是类似的，只不过上面这个方法更直观，是把model和vivo的数据都装到了一个对象里，而这个不是那么装的，是我们把model数据装到这个参数里，把veal的仕途直接返回，那返回的值给了despatch ursolate，而这个model的引用dispatcher servlet也持有着，所以这两份数据他照样能得到结果和这个是一样的。

那么我就再给这个学校加加上一个名字和一个年龄就model点。Model是点l的。At，它和model and view的方法不太一样，但也大同小异，参数名叫name参数值比如说学校，比如说北京大学好，再来一个年龄猫都有点l的h标的配置，抱歉，北京年龄北京大学的年龄我不知道，我就随便瞎写一个对不对？就是一个例子，80。

好，那么完成以后我们测试一下，看能访问重新编译。好服务重启了，然后我打开浏览器访问一下子贡北京大学80是吧？效果也是一样的。

对比来看很显然是第二种方式更简单一点，第一种方式略微的麻烦一点，所以我是比较喜欢用这种方式，我建议大家课上也开发的时候也尽量用这种方式，毕竟它简洁一点，那么这是我们像浏览器响应html数据的两种办法，除了能够响应html以外，那么服务器还能够向浏览器响应，其实任何的数据，那么还有一种情况比较常见，就是像浏览器想用json数据，想用json数据，那么什么时候要响应json数据？

一般是在异步请求当中。好，什么样的请求是异步请求？

其实这个例子很常见，我随便给你举个例子，比如说比如说我要注册B站哔哩哔哩，我就打开b站，然后这在哪注册在这儿，在这点我去注册，那么我注册b站的时候，我要我输入我的密码，输入昵称，那么你输入昵称以后，当文本框失去焦点的时候，或者说当你光标切换的时候，他就会判断昵称是否被占用。

咱们试一下，比如说我的昵称叫游客网，你看我输入完以后，他立刻就告诉我该昵称已被他人使用，你注意那很显然他要判断昵称没用，一定是访问了服务器查数据库，知道的对吧？

所以说在我刚才输入完优酷网以后，他查了数据库房的服务器，但是我们这个页面没有被刷新，那么这种请求就叫异步请求，就当前网页不动，当前网页不刷新，但是他悄悄的访问了服务器一次，得到了一个结果，很显然本次结果不是一个html因为整个网页不刷新，它只返回了判断的结果还是否，对还是不对而已。

那么了解了什么是异步请求以后，我们再回到这个程序里，当然我当前并不会给你演示怎么发送异步请求，后面我们项目中会有这方面的需求，到时候我们再演示，现在我们重点是关注说我们怎么下浏览器，响应这样的数据，有人会想这个json数据是什么样的数据，或者说它有什么价值，你注意我们当前所编写的代码是Java代码，面向对象的语言，所以我们这个代码时时刻刻都能得到一些招对象，然后比如说我要把这个账号对象这些数据返回给浏览器，假如我要把这个账号对象数据返回浏览器，浏览器解析这个对象用的是GS那GS也是面向对象语言，那么他肯定是希望能得到一个js对象，有可能将Java对象直接转成js对象吗？

显然不可能，因为这是两种语言对吧？

这不兼容的，那么我们通过json就可以实现两者的兼容，杰森实际上是一个具有特定格式的字符串，什么格式咱们一会就能看到，我们可以把Java对象转成这种字符串，然后给GS给浏览器传过去，那么浏览器可以将这个字符串转为GS对象，因为任何语言都有字符串类型，而这个字符串的格式是比较通用的格式，任何语言都能够进行解析，都能够很方便地将其转为对象，所以说它起到一个衔接的作用。

所以有了这种格式的数据，那么Java对象可以转成GS对象，也可以转成c sub对象，也可以转成其他任何语言的对象，这样就很方便。

所以在跨语言的环境下，Jason是我们非常常用的一种字符串形式，那么尤其是在异步请求当中，我们客户端需要你返回一个局部验证的结果，是否成功失败这样的结果，那么用这种方式就很方便。

好了，那么下面我给你演示一下我怎么去向浏览器响应这样的字符串，好，首先我再写一个例子，request，卖品声源访问路径没有写错了，pass等于什么？比如说emp假如说我要查询一个员工，Emp是员工，然后查询请求方式是request，method点get，然后如果我们需要向浏览器返回的是节省的话，那么你就得加上注解，response bardee，因为如果你不加他的话，他会认为你反馈的是html，加上他以后才可以返回这样的字符串。

好了，然后我就写一个方法，这个方法我反馈的是emp员工数据一个员工，我可以用一个map对它进行封装，当然你也可以自己定义一个类叫emp也可以，那效果其实和这个是一样的，然后死卷不对方法名我就get emp然后这个方法之内我就实例化 map，然后往里存一些值，最后再返回就可以了。

那么员工我要往里存什么呢？存一个name，张三，好再存再存一个年龄，比如说23再存一个薪水，比如说8000，好，最后我就返回延聘，那么dispenser service调这个方法的时候，一看你加了注解，并且你声明返回的是这样的类型，它自动的会把 map转成一个Jason字符串，发送给浏览器，咱们可以测试一下，重新编译。

好，编译完以后我打开浏览器访问一下 emp员工，好，你看就这样的字符串就是Jason字符串，然后它的格式非常的简洁，首先大括号代表这是一个对象，然后里边是key value结构，name k值是张三儿，塞勒瑞是k值是8千，配置是k23是值，那么每一对py6之间用逗号隔开，所以这个结构非常的通用，别管你是map还是一个我们自定义的一个类型，比如说T恤类型，时间段的类型有属性和属性值，所构成都能转化成这种字符串，非常的方便，好了这个值也没有问题，你可以看一下这个插件，刷新一下，看这个请求，那么他响应的类型是杰森，这是自动的。

那么还有一种情况也比较常见，有些时候我们返回的数据可能不是一个员工，可能是一组员工，可能有这种多个相似数据的结构的情况。

那么我们再尝试一下这种方式，比如说我要查询所有的员工，好我就就跟这个类似复制一下，查询所有的员工路径，我叫emps也要加上risk boss包点，因为我要响应的是Jason返回的方法名叫加个s返回的类型，多个员工我加上一个集合，每一个员工是一个map，所以集合里装的是一个map。

好，那么我在方法之内就创建一个集合，然后最终返回的就是集合类似的，然后我需要像集合之内装几个数据，我先把张三装进去，然后我再来两条数据太少了。再来一个这个内容比如说改成叫李四，比如说他24岁，假设他9000块钱，我再来一个王五，王五假设他是25岁，1万块钱，好就这样了。

那么返回集合以后重新编译，然后我再打开浏览器，访问一下，查询所有员工的这个方法。好，你看我们得到的是这样的数据，外面是一个方括号，表示这是一个数组，然后方括号之内有三个大括号表示这里面有三个对象，每一个对象就是ky6、ky6、ky6所构成的这样一个结构，它里面这就是集合形式的这么一个阶层字符串是这样的，所以你想一个map可以拼成这样多个map，又可以拼成这样一个数组，所以说基本上我们所有的形式的数据都可以用这种简洁的方式所表达是非常方便的。

好了，以上就是我们在spring mvc当中处理响应的最常用的几种方式，我们后面的项目中会经常反复的用到。那么这次课我们先演示到这里，咱们下次课再见。

## 1.5 Mybatis入门

5-Mybatis入门.mp4

说话人1 00:01

这次课我们来学习my best。My best它是一个访问数据库的框架，所以说我们在学它之前还得把数据库装好。这节课我们要装一个my circle server，My circle的服务端，还要装一个my circle的客户端，my circle，rock本子，那么这两个软件在它的官网都能够下载得到，这个链接是下载 server，链接是下载客

户端，那么这两个网址我已经提前打开了，咱们看一下，当前这个就是下载my circle服务端的页面，然后我们需要往下拽到这个地方这里让你选一下你的操作系统，我默认 windows就是适合我的。

然后当前我们要装的 co的版本是8.0.16，也是最新的版本，后面它有两个GIP的压缩包，那么第二个稍微大一点，然后它带有调试，还有测试的一些工具我们不需要，所以我们现在第一个就可以了。

你点 download然后进入到这个页面可以开始下载了，然后它会提示你是不是要登录什么的，不需要你直接点这个链接 no就可以了，然后就会开始下载，那么我已经下载好了，所以说我就不点了。

然后另外的一个页面是my circle，本子它的下载页面也是往下拖拽，也是这样的界面，选这个操作系统，然后这里边只有一个文件，你就点登录的，然后这个页面也是点漏，然后直接开始下载。

那么这两个包我都已经提前下载好了，给大家看一下。在下载目录下，这个是my circle的server，这个是my circle本，咱们一个来装，我先给大家装一下 my circle，server它是一个压缩包，你解压缩就可以。我把它解压缩到 d盘work目录下。

好，然后我去打开目录 d盘，work my circle。

然后 my circle直接启动是这个有问题，我们是需要对它做一些配置，做一些初始化，那么我们需要在它根目录下创建一个配置文件叫卖点ini这个文件其实不难，然后我提前准备好了，在这买点imi另外还有几个社科文件，这都是我们后面的课程里面要用到的。

那么这些资源我课后都会上传，大家可以下载得到，好，现在我把配置文件拷贝一下，然后粘贴到my circle的根目录下，这里要稍微改一下，把它打开，这里什么支付及ututf8这个端口3306，然后最大连接数20，这个不用搞得太大，因为我们是个人电脑，支持不了那么多链接，20个也足够我们用了就可以了。

然后默认的引擎in the DB这都没问题，那么唯一需要改的是路径，这就是你my circle安装的目录，因人而异，我们得改一下，我把我的路径拷贝一下，然后粘贴到这个位置，配置文件就配好了，配好以后还得需要对采色卡做一些初始化，需要执行一些命令，然后才能用。那么执行这个命令我们需要进入到 B目录下，然后去执行。

就当然了，即便是你给 b目录配置了环境变量也需要这么做，然后你配初始化好了以后，正式使用马赛克的时候就可以不用到目录下执行了。

好，这样下面我我先把你们先配好，省一会忘了，我把 b目录拷贝一下，然后我的电脑高级系统设置环境变量，然后双击pass，新建一个条目，把 my circle b目录配到这里，以后我们直接在命令行里访问my circle，直接就可以敲它的命令了。

但再强调一遍，就是我们初始化命令一定要CD的 b目录下去敲才可以直接敲，不行，哪怕是你已经配好了黄金亮也不行，一定要注意。

另外我们初始化的命令还不能说直接普通的方式打开命令行，你还必须用管理员的身份打开命令行工具去初始化才可以，这个也是一定要注意。

好我就点窗口，然后搜一下 cmd，命令行这个工具，然后右键以管理员身份的运行，然后像刚才说的，我们要 CD到 my circle的b目录下，我的目录在d盘，CD的d盘，work my circle b目录下，然后我们需要执行几个命令，第一个命令是 my circle，d就是my circle的后台服务，然后 initialize ina TIAA elyse ee 初始化，然后将初始化的过程打印到控制台，我们好能够看到一些关键的信息回声，那么它就会读取配置文件，然后帮我们初始化相关的配置。

这里面关键你要记一个信息，你看这句话我放大一点，就放大一点。看这句话就倒数第二句他说 a temporary password，一个临时的密码is generate生成了for root at Google host就local host是本地root。默认的账号，一个默认的密码，针对本地的root账号，一个默认密码生成了，那个密码是谁，就是这个字符串。

每次生成都不一样，而且很诡异这个字符串。所以最好你把它选中，然后右键拷贝一下，一会好用我把它记录到记事本里，反正一会不小心丢掉了，是这样的。好，这还没完，接下来我们还需要就初始化好以后还需要正式的安装服务，你敲这个命令很快服务就安装好了，安装好以后我们要启动服务。Net start my circle。

好，到这 My circle的服务就启动成功了，启动完以后这个界面就不需要了，我就把它关掉。My circle我就可以正式访问了，我再用普通的方式打开这个命令行工具。房mesko命令是Escaut杠Yoo，然后后面跟的是账号，就是root，然后杠p表示说我需要按照密码的方式去访问，回去以后让你输入密码。刚才我不是复制了吗？

右键粘贴到这个地方，回车就可以了，但是你用默认密码登录，他不让你做任何操作，因为他也知道这个密码你记不住，下回你又忘了，所以你必须第一次登陆，必须去修改你的密码修改密码，用一个比较简单的命令就是out user root艾特多个house，我就把本地的root用户的密码进行修改，然后后面要跟着这样一个单词叫 identified by一个字符串，这个字符串就是你的密码随便写，我写上我的名字，这样好记不容易忘。

好，然后分号结束回车，这个密码就改好了，改好以后行不行，我试一下退出一下，然后我再重新登录一下 my circle这块我就手动输入，我自己写的密码没有问题，好了，好了以后这样我们游客社区这个项目我们给他建一个库，然后我们把这个项目所需要的那些个表给它导入进来，然后一些测试数据也给它初始化好，这样后面的话我们就省事了，通过命令行导入比较方便，好，我先创建一个库 create database，那么取名就叫community。

接好了你可以看一下瘦一下。好肯定的，没有问题。我要使用这个community，我要使用这个库，然后在使用库的前提下，我要去导入建表的脚本，刚才我说的那个地方 community引力的circle，课后我会上传，然后这有三个社科文件，注意我们暂时一时半会用不上，比较靠后的位置在用。

然后 scheme点思考就是建表的脚本，这个data点circle是测试数据，我得先建表先导入，然后它的路径我得拷一下，导入的命令很简单，SARS然后后面写路径，然后写上文件名。

好，分号结束回车，他说我哪错了，可能是这个路径。左斜线才对他这块不能这样写，这样我拷过来然后再改。

那么在买sirk的模式下，你得写向左的斜线，不能写向右的斜线，然后文件名拷过来，分号回车就执行成功了，这是建了表，我们看一下这个表， show tables就这么几张表，表不多，然后表里什么数据都没有，我们还能还得导入数据，那么导入数据刚才和刚才的命令一样类似，然后只不过这个文件名要换一个换成 In it，data点server好回车，很快就导入完了，导入完以后再看一下show， tables有这么多 table，我们随便的找一个表看，比如说 user表，这个表里有很多数据，当然在控制台里看，在命令行里看不太方便它都折行了，那么所以我们才需要装一个好用的客户端。

总之经过这些操作，马斯克服务端是装好了，简单能用不方便。

好，接下来我们继续再把马斯克沃本斯好用的客户端安装一下。

那么my circle本色这是一个普通的安装程序，安装它就简单多了，就是双击，然后一步一步安装就可以好。我就下一步，然后这里我改变一下存放的位置，我把它改成这个地盘 work目录之下，然后在work下我得指定一个新的文件夹，我叫my circle杠。Work best好。My circle干活本色。Ok好。然后下一步安装就可以了，下一步然后基本上就一路默认就可以了，好。安装这需要等一会儿，咱们稍等一下。

好，完成了，完成以后它就自动的给我启动了，启动以后在这个地方我们需要稍微配置一下，因为你客户端访问的是哪个服务器，他不知道你要配一下，然后右键edit，编辑一个连接，但是它默认的话就是访问的是本地local house，端口是3306，默认端口账号是root，这个没有问题，但是密码需要改，他肯定不知道密码对吧？

这个地方点这个按钮，给他设置一个密码。

好，然后 Ok这个地方是你要填上默认你要访问的是哪个库，我们要访的是游客社区库前面那天。好，然后 test测试一下，测试的时候他又让我输入密码，我再输入一遍这回 save一下，保存一下，成功了我就 close就可以了。好，然后这就是我们编辑好的链接，你就点击它一下，它就启动了就进来了。

进来以后它默认这个打开了选中了 community库，然后我们可以做一些操作，这里面你看它默认了就访问了 user表有默认的一些内容，其实这是因为之前我装过，然后可能是卸载了没卸载干净，它有一些这个缓存，其实默认你打开应该是这样的，你们看到的是这样的界面，然后这边的大概是这样结构，大概第一次打开是这样结构。

好，然后基本上右边这个界面我不需要，下面这个界面暂时也不需要，所以我就点把它关掉，再把下面关掉。

然后平时我们左侧不看这个界面是给管理员用的，是管理数据库的，而我们是使用，所以选然后默认选中了我们要用的库就可以了，这里贴吧展开以后能够看到我们导入的表，说明没有问题，然后这儿你右键比如说谁来可以查询一下这个表里的数据，有这么多默认数据。

好，那么经过这样的操作，我们也看到了这个表里的数据就证明安装的没有问题，然后买适合本上也可以做一些这个配置也是有必要的，那么我给你演示一下它需要配什么，它的配置是点 edit，然后 preference首选项，然后通常我会配这个字体， forms and colors，它默认的字体我忘了，这是我配过又是一个缓存的数据，就这样了我就不动了。

然后的话如果你需要你的默认的字体你不喜欢的话，你可以在这儿进行修改，我都把它改成了这个字体 12号。

然后另外还有一个地方也最好配一下点 circle editor，最后它有个选项，这个选项它默认是勾上的，把它去掉，叫安全的更新。什么意思？就是说如果你勾上的话，如果你再删除数据或者是修改数据的时候，如果你没有写via条件，它是不允许你提交的，他怕你这个是忘了写外部条件，但是我们平时开发的时候经常是删掉点数据，修改点数据其实是比较随意的开发这个数据又不是正式的，为了方便还是把它勾掉，这样我们开发是省事。

好，那么到这我们 My circle的服务器，还有 my circle的客户端都已经装好了，那么课后的话你自己也安装一下。好，我们再回到 Ppt看下一页。

现在我们终于可以正式的开始学习my Betty's了，那么my Betty's首先说它的官网，它的官网的话是买白的4.22g但是它的官网主站你必须配置VPN才能访问，默认的话还访问不了，但是它的下级目录，下级路径是可以直接访问的，不用VPN，所以我把下级路经常用的给你列出来了。

My Betty是三它目前最新的版本，那么my Betty是基本的使用的手册就看然后我们用死不用整合买卖铁丝看，所以这两个链接是我们学习的手册所在的地方。

咱们先进去看一下，我先点前面，然后稍等一下，好。

打开了 my business的手册有一点好，就是它支持中文，你可以点这切换中文，然后只有麦德斯简介，当然他这个文档写的比较简略，不是那么的这个就是不是那么的详细，不是比较粗略，然后入门后面还有配置验证文件等等，那么我们课堂上所讲的内容，从这些内容当中就是提炼出来的重点，那么这些内容我就不在课堂上给大家逐个看了。

回到PPT我们再看另外的一个链接，就是死不用整合，买不得斯的手册点进去。

好是这样一个界面，然后这里也有简体中文，那么你会发现买白铁丝也好，还有死不容易整合，买白铁丝也好，他这个手册都在买白铁丝的网站，而不是在spring。这是为什么？这是因为买卖的意思，他发版的时候晚了一步，死不应发版的时候，本来是想带着my best，但是他晚了一步就没带上。

后来的话，为了能够让spring整合自己，是他自己做了一个自己做了一个整合的方案，所以说 spring整合my best的方案是my best团队做的，而不是spring团队做的，所以这个比较奇怪，所以说它整合的技术叫my best spring，而不是叫super max，所以文档是在买是在买卖的官网，你注意。

好，这个我也不详细看了，我们课堂上也会提炼出重点给你进行讲解。

好我们再回到PPT，那么关于买卖的事我们想使用它的话，需要了解它的几个核心的组件，那么它的第一个核心组件叫circle session，factory就是circle，session的工厂是一个工厂类，用来创建SQL session，什么是circle筛选？

Circle筛选是买卖特斯的核心组件，最核心的组件，它是用来向数据库执行circle的一个对象，就相当于jdbc当中的 connection式的。

好，第三个它还有一个主配置文件，是一个xml配置文件，然后在这里可以对买PETS的底层的行为做出详细的配置，比如说连接数据库的参数，比如说一些连接池相关的配置等等都在这。

那么注意我们实际开发的时候，其实这三个其实可以不用去写代码，因为我们是在spring boot的环境下开发的，那么死不认不得。

这个环境它会自动的去初始化，factory自动的去创建succession，然后它也自动的整合了配置文件，我们可以把xml中的配置配到application点practice里，所以说前三个组件尽管是核心，但是都被整合了，我们在开发时其实可以不用设计，但我说说一下是希望你能对它有所了解，因为这毕竟是买卖的核心。

我们开发的时候需要主要是开发后两个组件，一个是map接口，实际上就是dao接口，那么在买卖帖子里，那么他习惯性的称dao接口，为什么什么map？

其次就是map映射器在映射器里面编写circle，这个其实就是一个文件了，我们可以用xml实现，可以用注解实现都可以，为什么需要这样一个东西？

就是我们使用买卖点去访问数据库，我们只需要写出接口，不需要写出实现类，那么它的底层能够自动的帮我们去实现接口，前提是你要把那个每个增产改查的方法所依赖的circle告诉他，否则他也不知道你要干啥。

映射器其实就是用来编写它所依赖的circle，然后映射器里不只是有circle，那么你的circle是查询到了数据或者是向数据库提交数据的时候，那么这个circle需要和我们Java中的实体类做一个对应关系，你比如说我查到了数据库里的结果，我得用一个类去封装，我查的user表，我得用user类去封装，因为这表里有ID字段，有内部字段，我这个内容也得有ID属性，内部属性它得一匹配，对吧？

所以说映射器也要体现circle以及实体类之间的一个对应关系，就是这样的。好了，大概是这样。那么我们写几个例子来看，这个就非常的清楚，那么接下来我就使用my business，咱们对用户表进行一个增删改查的操作，Crud的操作。因为用户表可以说是我们刚才所导入的那些表中基本上是最简单，也是最容易理解的表了。

好，那么说到这，我们先去详细的看一下用户表它里面都有什么字段，把这个表搞明白以后，我们再去做一个这样的操作，这块我们可以直接用word本上去看。

刚才我们也看了这个数据，我们再仔细看一下，首先它有ID不用说的主键，然后是uz name就是用户名就是账号，password是密码，注意这个密码是经过加密的，然后后面这个sought叫盐，这个是个什么东西？

为了防止用户的密码过于简单，虽然说我们对用户的密码进行了加密，但如果你的密码太简单的话，也能够被破解，比如说你就搞一个简单的123456这样的密码，也是很容易被破解的，为了防止这种情况出现怎么办？

那么我们会在用户的密码的后面拼1个随机字符串，5位拼上以后，对拼好了，随机字符串的密码再加密，这个就不容易被破解了。

所以后面这个sort就是随机的字符串。当然我这里是手写的，其实在程序当中应该是随机生成的，就这样的。这个叫盐，我们给密码加点料，让它有趣一点，有味道一点，大概是这样一个感觉。

后面的话是邮箱不说了，tat是用户的类型，0代表的是普通用户，一是管理员，二是版主，status是用户的状态，那么零代表是没有激活，一然后 activation code是激活码，后边的hide URL是用户头像的图片的访问路径，create time是这个数据的创建时间，也就是用户注册的时间，因为注册是在创建用户，是这样的。

好，总之这个表其实很容易理解，主要是 Type和status，可能你不理解它01代表什么含义，你记不住也没关系，你在这点！它可以打开一个建表的 ddl语句，这个语句中有注释，说0代表什么意思，一代表什么意思，二代表什么意思，你可以详细的看每个字段是什么类型，有什么约束都可以在这里看到。

好了，这个表我们就了解到这，下面我们就写代码对这个表进行一个征迁改革的操作。那么在写代码之前，我们还得做另外一件事，咱们得把包导进来对吧？我们的项目中还没有买SQL的包，还没有买卖s我就打开这个没问题，抛弃特瑞斯打开这个网址，然后去搜索扎包。好，然后我要说的是my circle，因为我们需要依赖于my circle。

然后就因为你看它的使用者仍然比别的多，就 My circle connect接好点进去，然后我们选择最高的版本8.0.16点进去，然后我们选的是每本的配置，我不要注释这句话，我可以把勾去掉，你没有注释，然后把这段代码拷贝一下，粘贴到我们的项目中的pome的xml当中来。我放到后面去站到后面去在这儿，好，然后 Anybody out to commit凹凸import允许它自动下载自动去导入。稍等一下，等它颜色变成白色就是下载好了。

好，然后再回到刚才这个网址，刚才这个网址我们从没问的专库去搜包的一个网址， maven reporter.com你最好记住，其实搜索没问的包不只是这一个网站可以搜，其实别的网站也可以搜，包括阿里云，但是这个网站是比较好用的，因为它有对包的一个使用量的一个排名，你比较容易知道哪个包是比较热门的，也是便于我们去选择我要用哪个包。好除了买SQL以外，我再搜一下 my best搜一下，你看他有单独使用买威特斯的包，有spring整合买威特斯的包，有spring boot，整合买卖特斯的包，我们就用这个，因为我们用的是spring boot的框架，你用spring boot的包肯定是它就更简单了，它就有自动的配置了对吧？

点这个买卖的是不是不得是大特，然后还是我选择最高的版本，然后 Mabon把 copy一下，粘贴到pom的xml当中来，也是需要稍等，下载完了那么包就处理完了，包处理完以后，那么我们还要对数据库，还要对买卖点是做一点配置，因为即便是simple的它能自动配置，他也不知道你连的数据库的路径是什么，账号密码是什么，对吧？

你希望启动的最大连接是什么等等这些他不知道，这必须得自己配，没有办法。

好，那么在sprayboard的整合下，我们就不用创建买卖dace的 xml文件了，直接在application process里配就行，我就在这里进行配置，买circle还有买卖点他们的配置比较多，当然你从spring boot的官方手册附录里能够找到它的配置的模板，我课上就不挨个去给大家找了，因为太多了比较麻烦。

我就直接从我之前配置好的代码中把它拷过来，然后需要改的地方再改一改，给大家介绍一下每一块是什么意思，课后的话你配的时候也可以直接把我这段配置贴过去加以修改就可以。

当然如果你想了解更多更详细的配置，可以去查阅手册，可以自己去看一下。好，我就打开找到我之前的配置文件，在这。好，我把它打开这一段，然后我把它复制一下，粘贴到我的现在的项目中来。那么第一段配的是马赛克数据库以及连接池，这一起配了后面配的是my business。什么叫连接池？连接池也叫数据源，它是能够统一管理连接的一个工厂，然后它的作用是它能够统一的去初始化一批连接，让你反复使用，能够让人家复用，这样效率高。

第二个它能够管理连接的上线，比如说上线我这里配的是15个，那么如果超过15个请求，第十六个第十七个你要等待，这样的话避免了我这个数据库由于过多的人次访问他瘫痪，避免这种情况出现，所以说我们的访问数据库是一定要使用连接池的。

好，那么关于数据库的连接是4前4项，第一个是数据库的驱动， My circle的驱动是从my circle的官方手册可以查到，然后第二个是my circle的连接的路径，是这样写固定的格式，jdbc固定的，马赛克固定的，然后是IP地址端口号，根据自己实际情况写。

但其实你在本地装也就是这样。后面的是库名，我刚才建的库不叫community，就这样写。后面是一些参数 cap in coding，编码utf8支持中文，use SSL force就是不启用安全连接，还有 server time中，服务器的时区设置为中国的一个时区，香港，好，然后后面跟着的是账号密码，根据你的情况去改。

再往后是连接池的配置，这个是连接池的类型就是类，那么连接池其实有很多产品，那么 spring boot它内置了多个连接池，这个是性能最好的，所以我们选择一个性能最好的就可以了。

然后这个配置是连接池的最大连接数，我这里因为是个人的电脑不适合搞得太多，15个就够了。

然后是最小空闲连接，比如说刚刚一大波请求来了，连接都被用了，现在一大波请求走了，连接都空闲了，那么需要回收几个关闭几个，保留少量的几个就够，那么保留几个5个，然后这个都太超时时间，那么他的意思是我不能说连接空闲了，我立刻就把它关掉，我需要等多久看它还空闲着，我才把它关掉，3万毫秒。

好，以上是连接池相关的配置，然后这个是my Betty's的配置，第一项配的是 map的case，就是映射文件的存放位置，xml Cirque的文件的位置，我们通常会这样，我通常都会把配置文件放到resource下面，我会在这建一个目录叫map，然后 map的目录为什么选class pass？

因为你一编译以后 map就会被编译到target，classics下面就像style timeless applications放的地方一样，那么这个classes就是class pass类路径，所以说 class pass指的是 class文件夹，map，然后它里面的文件都以xm结尾星点 xml，这个是指定你的，你不是有表吗？

我们需要通过类用类，用实体类去封装表里的数据。那么这个要声明尸体类所在的包的包名，当然这个包我还没见我见一下。

在这创建一个实体类，用于存放实体类的包anti，好将来我就会在类之下创建在这个包下创建实体类，用来封装某一张表的数据，那么我们做了这项配置以后，那么在配置文件里引用实体类的时候就不用写它的包名了，这样就省事了。

这一项是启用自动生成组件，我们insert的时候 ID是自增长的，我要启用一下才能支持。

然后这个是什么意思解释一下。咱们那个表它是不区分字段，是不区分大小写的，然后一般我们字段是这样命名的，比如说 PAD URL好，我们建一个实体类与表对应，实体类的有属性和字段对应属性，我们通常是驼峰式命名，这样的字段和属性名是不一样的，但是有规律的。

那么这个选项设置为树的意思是让下划线的命名方式和驼峰的命名方式二者能够匹配起来。我们让表里的字段和类中的属性去匹配，就可以达到一个自动的状态，就不用我们挨个去设置，这样就方便了。那么以上是数据源，还有买卖TS相关的配置，这么多我们写好以后就可以直接在这基础上做下一步的开发了。

好，现在我们包也导入了，配置文件也配好了，接下来我们终于可以开始写正式的代码，然后去访问优则表里的数据了，对优则表进行一个增删改查的操作了。

好，我们要仿这个表，首先要写的是实体类，因为你需要有一个类能够封装表里的数据，然后我们才好对它进行操作。

好，所以我在安推的包下，新建一个类取名叫user和user表相对应，然后根据表里的字段，我这里有多个字段，我有与之对应的多少个属性，只不过我的属性名是按照驼峰式命名的方式来命名的。

好，刚才我们也看了，首先它有ID整数，其次它有 User name、用户名，然后有密码有这字符串有言也是一个字符串 sort，然后还有邮箱 EMAIL，它还有太了就是类型以及状态，然后还有激活码，activity code，还有拍的 URL就是头像的防路径，好。

最后还有一个是日期类型，叫create，看就是这个用户他注册的时间，这条数据的创建的时间。



好写完属性了以后，你要给他提供相应的盖特赛的方法，你可以按快捷键就是out加insert它会弹出一个框，这里面你选get on the set，然后勾选上，你要生成set的这些属性都选上。Ok，最后我再给它生成一个toss卷，也是out加insert的快捷键。To，spin给它生成一个to spin方法，这样我们打印对象去看这个数据的时候会比较方便一点。

好了，那么实体类很方便，没什么逻辑性，就是很简单的就写完了，写完以后我们要访问数据库，我们需要在do这一层写一个组件，那么买卖这次开发，我们管数据访问组件叫map，通常命名为什么map？

而且它只需要写接口，不用你写实现类，因此我这创建一个优点，map是接口，ok好。

另外我们需要加一个注解才能够让space容器装配，这个并按理说dao层你应该用repulsive，Terry实际上你用注解是可以的，但是买白铁丝也有一个注解来标识这个病买白铁丝的注解叫matter，我一般习惯于用因为毕竟这个单词的简短对吧？

它简洁就用这个，然后我们需要在map里声明一些增强改造的方法，然后再给它写上对应的circle的配置文件就搞定了，那么这个方法我就按照我们后面这个业务中需要的来写，我需要什么我就写什么，然后这样的话我们user map写完以后，现在就可以直接用了，我们到后面的阶段的时候就能省点事儿。

好后面首先我要用的是根据ID查询用户的一个方法，那么我就这样写，select by ID，根据ID查询的是user得到一个user，那么参数就是ID，好再来，另外我们还会用到根据用户名查user用户名是唯一的，所以也能查到一个user select by name。那么参数就是邮政内容，还会用到根据邮箱查邮政 user select by EMAIL那参数就是EMAIL。说白了这三个方法都是查询，而且逻辑都相似。

其实我们会写一个，其他两个肯定就会然后我们后面还会就增加一个用户，然后返回一个整数，返回的是你插入的数据的行数，你成功其实就一行，换个名我叫insert user，然后参数是整个user所包含的数据，应该用user对象加以封装的，对吧？然后后面我们还需要对user做修改，修改了它的状态，update status返回的是什么？是修改的条数，修改了几行数据，那么修改的话你得有修改的条件，我们通常是以ID为条件，然后改的是状态，所以你要把最新的状态传入进来。

还有后面还需要update head，更新头像，更新头像的路径，也是跟上面这个类似，以ID为条件，然后参数是high点，URL Url是路径的意思，好再来还要更新密码，那么条件是ID参数是最新的密码。好了，这些方法是我们后面开发的过程中需要用到的一些方法，我在这提前把它先正好利用这个案例就把它实现后面好用，这样比较方便。好了，这个方法其实也很容易理解。根据ID查询，根据内容上根据修改什么东西也很直观了对吧？但逻辑上并不难理解，那么麦克就这样就写完就可以了，生命好就可以。

那么要想实现它，我们需要给他提供一个配置文件，那么配置文件里需要给每一个方法提供他所需要的circle，然后这样的话买卖的底层会自动的帮我们生成一个实现类。

好，那么我就在map这个目录下创建一个xml文件，创建一个xml文件，这个文件其实你叫什么名字都可以，我叫user杠mica点xml，那么这个文件的结构，咱们从买MAC的官网是可以把它找到的，我们去找一下，因为这个东西我没有办法去手写，记不住好。

我们再打开一下my circle的官网是my business的官网，my business3.5.1，那么在入门这块就能找到，这块就能找到，这个是主配的文件，我们把它省略了，我们用的是application的practice，这个就是他map映射的配置文件，我整个copy一下，然后把它粘过来，就是这样一个结构。

里边的内容我先删掉，一会我重新写。Xml文件它的根是mapper，然后namespace这块有说法，这个地方写什么？写user map的权限命名，因为你这个配置文件是为哪个map服务的，要在这里体现，不然的话它跟哪个map对应起来不知道对吧？所以这里面写的是Map的权限的名com点，脑壳的点儿肯定的，点儿deo点儿，又在迈克尔这块一定要写清楚好。

然后在迈克尔之内我们需要写出各种各样的标签，每一个标签要对应一个接口中的方法好，很简单，如果是查询方法，你要写一个谁来给的标签里边写上circle，如果是增加数据，你要写一个音色的标签，写上音色的circle，然后update的标签，得力的标签，我们这里没有得力的，但是你明白了，insert、update、delete其实也很容易理解，好我就写了。

首先我要实现 select by ID的方法，我就需要写一个circle的标签来实现一个查询的circle。

然后他报错了，你要加上一个ID，这个ID又有说法，就是ID要写什么呢？要写它所对应的方法的名儿，方法名儿不叫写来给他办ID吗？这样的话买卖的就知道，circle是为那个方法服务的，是为了实现这个方法服务的，就有了一个对应关系，你要这样写。

好say let by ID，然后还需要声明result type，那你是一个查询，你查询以后你返回的是什么类型，你要声明，不然的话他不知道你要返回什么类型。我这里写优点，按理说你得写全限定名，com点脑壳点什么？带上包名，但因为刚才的配置文件里写了这段话，你告诉他你的市值率会放到哪里，所以前面的报名就可以省略了，直接写user就可以了。

好，然后我在这里面就写上查询的语句就很简单了。

See like ID啊然后等等，你要把他的各个字段都列举出来，那么养成习惯尽量不要写谁来给他牺牲，一个字段写邮政内容，它是word sat，EMAIL tat，当然写这个，因为他没有检查说你的单词对不对，其实挺容易写错的，所以说你写的时候谨慎一点，尽量把它写对了。

Activity扣的 hide URL以及create time。我要查询这些字段，然后 From user表where我们是以ID为条件来查询，ID等于ID等于应该等于谁？等于这个参数对吧？我怎么引用这个参数这样写？井号大括号ID，你注意我们前面所写的这字段，这是表，这是字段，而井号ID意思是我要引用这个方法的参数名为ID，这样的话就把参数嵌入到 Circle里去了，就这样这就可以了。同理不是还有两个查询语句和它类似，我可以就复制了 Copy。

第二个是select by name，反馈类型也是一样，查询语句也一样，就条件不一样，name不是name，user name等于user name。好，这是第二个查询语句。再来一个。

第三个，select by EMAIL。其他都一样，这个地方EMAIL等于EMAIL。好，查询就完成了。完成以后你会发现有的地方有点麻，有点问题，就什么呢？我们写字段这个地方挺麻烦的，尽管说我后面是copy的，但是你想一想，万一说将来有一天要加一个字段加两个字段，你这得改这得改三个地方对吧？不合适。

怎么能够把提炼出来，让它能够复用，其实提炼可以这样，就是circle你可以在前面专门定义一个circle，然后给它取一个ID，便于引用，这个ID我叫谁来 feels就查询的字段把它拷过来，这是我要查询的字段，然后我就可以这个地方就可以用 Circle来替换，这地方可以改一下，改成什么呢？改成叫做include，然后引用的ID。写上 select feels比如说我这个地方引用的是这段搜狗，好，这样的话就让麻烦的东西能够复用了，这也是这么改，这也是这样去改。

那么查询语句搞定，接下来我们再来处理那些个增加修改的语句增加，你要写一个insert，也要写ID，也要和方法名对应，insert user。

好，然后还要写上就是说你增加的时候有一个参数，这个参数你要声明好，这是个什么类型的参数，需要做一个声明。

当然了，如果说是这种简单的参数就不用如果是一种复杂的参数，是一个bin就需要声明，因为它默认它识别不了这个东西，它不是Java自带的类型，你要声明，说明参数类型是用 primary type，那类型显然就是user对吧？

好，然后还要写上一个属性在增加的时候叫key property。你要告诉他你的ID的字段对应的属性，就邮件内容与ID组件对应的属性名是什么？就是ID，key property等于ID。为什么要做这样的配置？因为增加的时候，那么 My circle它的底层会自动生成ID，生成ID以后，那么买卖的时候会从买色中得到ID，然后填入 user里，我们增加的时候 user的ID是没有的，增加以后就有了。

这个ID是买卖点是从数据库获取到，然后给你回填进来的，他得知道你属性是谁，所以要做这样一个声明。

好，然后这里边我们要写一个insert语句了，insert into user，后面你要写上字段，这里不要写ID，因为ID是自动生成的，其他的要写要写字段了，我们其实可以想办法再让字段再复用一次，可以这样的，SQL ID等于insert feels我再定义一个共享的circle，那这个insert语句的字段，然后把拷过来，这样只有只是他没有ID而已，然后那么我这儿就是要引用音色的feels，也是用这个音刻录的。

Refid等于insert是有。看看有没有写错，没有问题。好，这是插入的表里，要插入这些字段，对应的值从哪来？对应的值是从你这个参数user里来，这里面你要引用参数，还是用井号大括号的方式，然后这里边写什么？

写user里面属性名，user、name属性，然后它是word属性，然后是sort，反正就慢慢写一个写EMAIL，太太使得的是然后是ak为人透的，注意这个是属性，我们user中的属性是驼峰式命名，不是下划线，这块要注意好，继续，然后是黑的URL最后还有一个是create time。

好，就这样了，那么因为设成语句我们就写好了，我再数一下字段的对不对？Use name, password, sort, EMAIL, status, I given code. Hide URL, create time。应该没错，好，这是insert，然后还有update，三个update其实就比较简单了，因为它的参数少不对的。

Id对应方法，名update status，然后它是一个简单的类型的参数，你可以简单类型参数可以不用声明，然后你像insert也好，还是update也好，其实它返回的都是整数固定的，所以你不用声明查询，需要声明返回的类型，而往往的音色搭配的需要声明参数类型，但因为我们这个update只是改了一个字段，参数比较简单，类型也不用声明。

好在这里我就写了update，优点赛特斯特尔斯等于参数斯特尔斯 where条件 ID等于参数里边的ID。另外的两个update和相似，Update header那就是site, header, URL，注意这是字段等于参数，either URL然后条件依然是ID。好，最后还有一个复制一下update password，然后set password等于password，然后条件还是ID。好了，到这儿我们就把映射文件编写好了。当然这是因为我事先把我们后面的课程中要用到的方法都列出来了，我们一起去实现，有点有点这个时间长有点麻烦，不过好在后面就简单了。

那么实际开发的时候，比如说你要如果说我们先开发一个方法，两个方法，可能就快很多。好，那么写完以后咱们得测试一下，看这个东西不可用，对不对？那么我直接写一个测试类，在测试类中对它进行测试，那么我在这个包下我新建一个类，不用原来的类了，新建一个因为都写到一起去有点乱，map test这个类的注解和之前的测试类也是一样的，我把它拷一下粘贴过来，然后在这里面我要测的是麦克对吧？所以我要把麦克注入进来 as well。Private user my phone注入进来，然后我们把查询新增修改分别加以测试。

那么首先我先测一下，查询Select user，测试查询user，先根据ID查，我们得看一看有什么数据，看一下数据，比如说我要查这条数据，刘备ID是101就查他了，那就是user map点select by ID101，返回的是user类型的对象，然后我直接把它输出来看一下，看行不行？好，那么我就执行这个方法。好，执行完以后它是勾表示是对了结果对不对？你看ID 101 u the name刘备，password什么？没错不用细看看。两个就知道了。同理我们再测一下，比如说根据用户名查，用户名是刘备。我再来一遍user。等于user map点。Select by name刘备。然后打印user再把也撤了。User map点起来把EMAIL刘备的邮箱是什么？看一下就瞎写的，就用它来查，按照邮箱的来查，查完以后也打印悠着，我就这三个查询就一个方案一起测了，然后再执行一下这个方法。

对了，看结果查到的结果都一样，因为我们就是查刘备没有问题，下面我们再来测一下增加的方法，我再写一个测试方法，Test, insert user，那么你要增加的话，我们需要给那个方法传的是一个优质对象，你提前要把优质对象构建好，里边的数据封装进去，你就实力画一个u字，ID不用传，因为数据库会自动生成user点site, user name, 用户名要写，比如说用户名叫test测试，然后user点site，它是word，我这里就不加密了，就直接写一个直观的123456，user点site sort在这个sound是ABC随便写了user点site, EMAIL，比如说test QQ.com，悠着点site, either ul当然还有tag，还有status，那两个整数，如果你不传的话默认都是0，那就默认值，我就不修改了。

然后头像的路径，咱们优客网有默认的头像，它的路径是这样的，HTTP冒号，双斜线、三w点，now code.com，然后斜线是一个数字，然后点拼劲，比如101瓶颈，这个数字是从0~1000，超过这个范围小于0或大于1000就不对了，一共是0~1000这么多个图片。

好，再site一下，创建的时间就当天时间，好，终于构造好了 user我就要insert了它，返回的是插入几行行数，又在卖票点，音色的优质又在传入，传入以后我们打印一下 rose，另外我再打印一下悠着点get ID，看这个ID有没有生成。

好，那么写完以后运行一下，好看结果一行 ID生成的是150，数据库里有没有这个数据，咱们再查一遍，执行一下 circle。

后面150test没问题，这是一个测试的数据，好了，这是新增，最后再测一下修改，那么修改的话，一个是修改这个状态，修改路径，修改密码我就改刚才的数据 test就改它了，就user map点儿不对的 study的状态，他的ID是150，状态默认是0，你看一下，默认是0对吧？

比如说我要把它改成——表示激活态，返回的是修改的行数我会打印这个行数，好我再把直接再把那两个也改了。Rose等于user map点update还得150。

原来的路径是101，我比如说把它改成102，你看刚才的库里是不是101，咱们确认一下101没错，然后就这样没错，101。好回来这是改了头像，改完以后再打印返回结果rose，然后还可以改密码，他的密码原来是123456对吧？我改一下给他150，最后把它改成hello的。打印一下，rose。好，那么洗完以后执行一遍。好，然后看结果，结果都是一就表示说他都是改了一行，每次改的数据对不对？我们再去查询这个数据库，看一下。

Hello密码改成hello没问题对吧？然后状态改成了一没问题，路径改成药量也没有问题，那么经过测试我们就了解了我们写的程序还是ok的，然后大致的买白铁丝的用法你也就明白了，课后的话多练一练，把这个套路记住，我们后面的项目中会频繁的使用，然后会再加强练习。

好，最后再补充一下关于 My best的调试的方式，因为 myce其实你写代码的时候很容易把它写错了，尤其是 map的xml，Java类还好说，因为写错以后它会编译，会有提示，但是 map配置文件这里边的表名字段名或者是这些个key写错了，它没有任何提示，那么我们怎么去调试比较好，我们可以把 dao这一层的日志级别调为比较低的级别debug，然后控制台会打印出更多的信息，它会打印出 circle，所以不管你哪写错了，最终影响的是circle，如果你看到circle以后看这个circle对不对？

它错在哪？你就容易排错了。

好，那么当然了日志是怎么回事，日志级别是怎么回事，我们后面会详细讲，我们现在启动一下，便于你调试。我打开配置文件在这写了。写配的是日志 Logo，写个注释，然后我写老给点儿level点儿com点儿，然后扣点儿点，看没有那天。这样我就直接把我们的代码的日级别都设置为第八个，自然也包括do包。

这句话的意思就是说把这个包的级别调为第八个，也包括了deo好，写完以后我再回到这个测试类，这回我执行这个查询运行一下看有什么变化没有，我看一下，你看这回控制台不只是打印了，查询结果在每一个查询结果之前，它打印了本次查询它所执行的circle，图片预编译的circle，然后 circle的参数是多少，最终返回的数量是几，查到了几条数据是这样的，所以说你根据看这个蛇口看这个参数，看这个结果的提示，那么就便于你去调试了。

好了，以上就是买bad使用方式以及调试的一些技巧，那么我们这节课就讲到这里，咱们下次课再见。

## 1.6 开发社区首页

---

6-开发社区首页.mp4

说话人1 00:03

经过这几次课的学习，大家已经了解了spring还有spring mvc my bet is他们的基本的使用方式，那么这节课我们把它们综合到一起，开发出我们项目中的第一个功能，社区首页的功能。那么在开发这个功能之前，我想先再给大家梳理一下咱们开发的步骤或者说开发的流程。

因为我们做的是web项目，而web项目主要解决的是浏览器和服务器之间的一个交互的问题。

那么之前我们聊过 HTTP协议，大家也知道浏览器和服务器之间是由一次一次的请求进行交互的，所以说我们任何功能都可以拆解成若干次请求，那么你只要掌握了每一次请求的执行的过程，然后按照这个步骤去开发每一次请求，基本上这个项目就能够逐渐的把它完善起来，那么之前我们也说过每次请求，那么它的执行过程是这样的，请求会提交给我们服务器的，那么视图层主要有CTRL还有模板所构成，然后CTRL在处理请求的过程当中，它需要访问业务层，然后让业务组件来处理具体的业务，那么业务组件在处理业务的过程中会访问这个数据库，所以说它会调用数据访问组件，当然数据访问组件就是和数据库打交道了，所以说请求的执行过程就是这样一个顺序，那么很显然 service它依赖于到对不对？

然后 CTRL是依赖于service，我们开发的时候，如果是先开发到再开发service再开发controller，可能会比较舒服一点，按照他们依赖的顺序去开发比较舒服一点。

好，所以说我们后面的各开发各个功能当中，我都会按照这样的一个既定的顺序进行开发。

然后我们在实际开发的时候，可能对于一个新手来说，我们没有这方面的经验，我们可能一开始开发会有一些困难，你最好可以把一个功能给它做一个拆解，比如说第一步先实现一个什么样的效果，第二步我再完善一下。

然后的话其实很多这个程序它并不是说一步到位就开发好的，往往就是在不断的优化不断的完善的过程中，最终才把整个功能实现的。

那么对于社区首页，其实它就是一个帖子列表，查询帖子的功能，我们可以把这个功能拆解为两步，第一步我们可以先显示前10个帖子，因为那个表里有很多个数据，不止10个，我们先显示第一页10个，然后这一步完成以后，我们在此基础上再做完善，下一步我们把分页的组件开发好，然后可以分页显示每一页的帖子可以有一下一页或者点击某一页的这样的功能。

接下来我就给大家演示社区首页它的开发的方式。好这样，因为我们要先开发数据访问层，所以说我们先把表建好，并且把它讲解一下。

当然之前我们已经把这个表建好了，那么帖子表是这张表叫discard post讨论帖，然后表的结构我给大家看一下，你可以点 I然后点ddl就是建表的ddl语句，通过这个语句你就能够了解这个表的包含哪些字段，每个字段的它的一些含义。

然后你看第一个字的ID自增长的主键没什么好说的。第二个字到底是user ID，就是这个帖是由哪个用户发布的，那么我们记录了一下用户的ID，显然字段它可以关联 user表。

然后第三个字段，第四个字段分别是帖子的标题和帖子的内容很好理解，然后因为内容会比较长，所以说它是一个文本一个大文本，而不是一个普通的外插。

然后是帖子类型这里有解释，这个零一表示置顶的帖子，status是这个帖子的状态，零表示正常的状态，一表示它是精华帖，二表示他被拉黑了，然后create time是帖子发布的时间，comment count是这个帖子的评论的数量，实际上我们有单独的评论表，comment它和这个帖子是有关联关系的，所以说其实我们通过查询 com的表是可以查到这个帖子有多少个评论的，但是因为我们每一个帖子都要显示评论的数量，这个行为比较频繁，因此我们在这个表里冗余的存了一下评论数量，这样我们取这个值的时候就不用去关联查，效率会高一点，所以这是冗余的存储这么一个字段。

最后一个字段是score是帖子的分数用来将来给帖子做排名用的，按照它的热度进行排名，我们需要算一个分，这个是比较靠后的位置，我们再去演示这样的功能。

这次课我们先不用，好了，那么这个表我就介绍完了，目前我们这个表里已经有了一些测试数据给你看一下，有了很多数据，当然这个数据是我硬造的，所以说很多数据它是一样的，但是只要有数据，我们查的时候能把分页显示出来，效果能够表现出来就可以了。

好，这个表介绍完了，介绍完以后，我们首先要开发的是 dao 数据访问层，那么开发数据访问层首先我们要把实体类写好，用来能够封装表里的数据对不对？然后接下来我们再去开发数据访问组件，也就是迈克尔。

好，下面我们就来写实体类，我打开我的开发工具 idea，然后在安贴报价再新建一个类，那么根据这个表名我给它取个名，叫做 discus post 采用头风式的命名办法。

好，然后根据刚才我们看到的那些字段，我给它加上与之对应的属性写错了，首先是一个整数 ID，其次也是一个整合用户的 ID，user ID 好，然后是一个字符串开头，就是帖子的标题，接下来还是字符串，content 就是帖子的内容。

好，再往下是帖子的类型是一个整数，tap，那么零表示普通一点的置顶，接下来是帖子的状态也是一个整数，然后零是正常，一是精华，二是拉黑，再来是帖子创建的这个日期 create time。然后是冗余的字段，存的是评论的数量，评论我们就要 comment，然后数量 count。最后是记录一个帖子的分数，那么我这里用的是一个小数杠。

好了，我们这个帖子表的字段一共这么多个字段，那么接下来我给这些字段生成 get on the set 全选，然后为了测试打印数据方便，我再给它生成这个 to c 并。好，实体类非常容易，它没有什么具体的逻辑，反正表里有什么字段你就建什么属性就完了。

有了这个试题了以后，下面我们就可以开发 Map，那么我们通常就是一张表一个 map，所以我需要给这张表创建一个新的 map 是一个接口，那么取名可以叫 discus POS map，然后别忘了在 map 之上打一个注解，叫 map 你打上注解以后才能够被容器扫描，接口才能够自动的去实现它，才能够装配它。

然后我们是要做查询的功能，所以说我们需要在这里声明查询的方法，然后在配置文件里写出它与之有关的 circle。然后我之前说了我们做的是分页查询帖子的功能，所以说你查到的肯定是多条数据，因此返回的应该是一个集合，然后集合里装的就是帖子的对象。Discuss post。然后这个方法名你随便取，我习惯于叫 select discourse。Post。

加个 s 表示复数，然后这里面我给他来一些参数，然后我再解释一下，首先我加一个优质 ID，因为刚才你也看到了帖子，表里是有一个字段，是 uzid 的，当然了我们首页上查询的时候其实是不用传入 u 站 ID 的，就是把所有人的所有帖子都查到 U 站 ID 是给哪用，将来我们会开发用户的个人主页的功能，个人主页里有一个我发布过的帖子这么一个功能，我发布的帖子其实就可以调这个方法，然后传入用户的 ID，因此这个方法我是考虑到了将来的功能的适用的程度，因此才加了字段。

那么在首页上我们查询的时候，不会传出这个值怎么办？或者说你不会传出这个值，它就默认就是 0，因此当 uzid 等于 0 的时候，我们就不管它，或者说我们就不把这个条件拼到 Circle 里，然后当优质 ID 不是 0，它是一个正常的值，比如说它是等于一二 100 这都可以，那么我们就需要把它拼到 circle 里。

所以这个方法我们在实现的时候，circle 它是一个动态的 circle，有时要拼 uzid 有时不要拼。如何实现动态 circle，咱们一会儿写 Circle 的时候你就知道了。

好，那么还有虽然说我们第一步在页面展现上先不去管分页，但是我们在开发这个方法的时候，也要考虑到让这个方法能够支持未来分页的可能，所以说我得把分页的情形考虑进去，一步到位就把 map 写好。

分页的话咱们用的是 my circle，数据库它的分页非常简单，我们加一个 limit，然后在里面的关键字的后面跟着两个参数，一个是我这一页的起始行的行号这么一个参数，第二个是这一页最多显示多少条数据，所以需要两个条件，我就把这两个条件也加到这个方法里。

好，那么这个off site每一页起始行的行号，然后 limit指的是每一页最多显示多少条数据，那么我们拼SQL的时候，那么把这两个条件拼到分页的关键字里面的，后面就可以了，然后我们因为页面上最终要支持分页，而分页的时候你要知道我一共可能有多少页，一共可能有多少页，它是通过计算而来的，需要两个条件，第一个就是你一共有多少条数据，第二个你每页显示多少条数据，两者一除就可以了。

那么每页显示多少条数据，其实我们是可以固化下来，比如10条20条可以固化下来，那么一共有多少条数据，这得看表里的数据。

所以说为了我们将来显示页码它的方便，所以我们还需要提供一个方法，就是要查询出这个表里一共有多少条数据，我把这个方法也声明一下，那么它返回整数就可以了。然后方法名我叫select discourse。

Post rose查询帖子的行数，然后也需要加一个参数user ID，那么在首页上我们不会传入user ID，那就是0，当它等于0的时候我就不拼这个条件，当我们开发个人主页，我的帖子的时候就会拼这个条件，它也是一个动态的缺口。然后这里边我还要多说一句，我们在写这个方法的时候还可以说，其实在写其他的方法的时候也是我们在可以在参数之前加上这么一个注解派入，注解的作用是用来给参数取一个别名，比如说有的参数比较长，你嫌频道SQL里面写的时候麻烦你可以用parry给它取一个别名，就像这样写，当然我这里别名和参数名一致的，这也没关系。

然后这里我一定要写上 prm是为什么？如果说我们在circle里需要用到动态的条件，并且这个条件里需要用到这个参数，而恰巧这个方法只有一个参数，这个时候这个参数之前是一定要去辨别的，否则它会报错。

好，我再给大家再总结一下，就是说如果说我需要动态的拼一个条件，并且这个方法有且只有这一个条件，这个时候这个参数之前就必须取别名，你像刚才这个方法它有三个参数，你不取平面也无所谓，所以说细节大家注意，这里我写个注释给大家做个提醒，就是说他注解它是用于给参数取别名的，其次如果这个方法只有一个参数，并且在动态社保是用if标签来拼的，并且在if里使用，那么就必须加别名，也就是说你必须得用parry。

好了，那么我们首页的查询功能用这两个方法就够了，那么将来我们增加帖子等等对帖子做其他操作的时候，我们再去补充其他的方法。好，那么说明了这两个方法以后，下面我们就可以实现这两个方法还是比较容易的。

我需要给 Map再新建一个配置文件放到 map目录下，好新建一个配置文件，这个配置文件我取名叫做discuss poste，map点xml，那么它的结构我们之前因为有了user Micro，你可以从user map里复制，主要是这段，因为这段没法记，把它copy一下，然后粘贴过来，当然你别忘了把麦克还得给它结个尾，然后你需要改变一下这个类名。

当前我们用的是discus post map，声明完以后表示说配置文件是为接口服务的，不会有偏差，然后我们需要是做两个查询，而查询的话我们像上次课一样，我们也要把查询语句，查询的字段给它提取出来，然后公用一下，所以我先声明一个公用的circle，它的ID然后取名叫做select feels，然后把我们要查询的字段在这列举一下，一个是ID，然后是牛仔的字段，所以说中间是下划线和数据库去比较开头，然后是content。

Type。Status。Create。Time。Comment。Count。好，那么写这样的代码的时候一定要注意，如果说你单词写错了，它是不会有提示的，所以说你最好仔细一点。

好，写完以后，下面我就开始实现我们刚才的第一个方法，这个方法把它copy一下，因为我在写circle的时候需要指定它的对应关系，利用方法名，那么我是给这个方法写circle，那么这个方法它返回的是集合里装的是 discuss post。

那么你注意我们需要声明返回的类型，因为它返回的类型不是查自带的类型，那么他默认他不知道，你要告诉他，然后它是一个集合里装的是discourse post，那么这种情况你只需要声明这个类型就可以了，历史的你不用声明，因为历史的它能检测到，而是Java自带的类型，它能够解析出来这个含义，你要声明好好，那么我们返回值的类型需要通过result进行声明，然后写上这个discourse post。

好，接下来我们就来写查询的语句也非常的方便，我要查什么？查的是这些字段。那么用include把它引用到这里，然后 Frown的是discourse，post这张表，然后再加一些条件加什么条件，刚才我们看那个表也看到了，那个表里有一个 status状态的字段，然后这个状态有0122表示说拉黑，那你查询帖子的时候显然拉黑的帖子不能展现对吧？

所以你应该首先写上 status，它不等于2，这才比较合理。

好，然后还有一种情况就是说 uzid有可能是0，有可能不是0，有可能传入，有可能不传入，我需要对它作出判断。当它不是0的时候，我把它拼到circle里。好，那么刚才说了，我们是通过 If标签对它做判断，这个标签是这样写的，然后 if里面需要写一个test属性，test属性里边写上一个条件，我要写什么？就是这个参数它不等于零，参数名叫user ID，我这里就写uzid不等于0，这样就可以了，非常的简单。

此时我要再拼一个条件，拼什么 User ID字段，它等于 user ID那个参数，那井号大括号引用参数user ID，那么大家这里一定要注意，我哪个地方引用的是方法的参数，哪个地方引用的是字段，你一定要搞清楚，你看这样写的是字段，这样的是参数。

好，那么批完条件以后，最后我们对查询的结果，因为不是一条最好给它排序一下，给它一个合理的排序。

Order by首先我order by type倒序type是帖子的类型0，普通帖子一是置顶的，我道具的意思就是说置顶排到前面去了，这个是很合理的，置顶就应该排到前面去对吧？

其次我再来一个 create time desc就是在你类型一样的前提下，我其次是按照创建的时间，倒叙，就是新的帖子在前面，当然这是我们权宜之计，将来的话我们会开发按照热度进行排行的一个功能，那么那个时候就不是简单的按照这个时间来排序了，我们需要算一个分。

那么那个是将来要做的事情，我们现在先不管，因为现在我们还没有到那一步。

好排完序以后，最后我们还要实现一个分页，分页的话在 oracle里头用关键字limit两个参数，一个是起始行，我们访问奥赛的参数，一个是显示的最大的条目数每一页，访问的是里面好，这个方法我就这个circle我就写完了，我把另外的一个方法的circle也写完以后，咱们来进行测试。

好，那么这个方法还是一个查询语句，它的ID复制了是然后 resolve tag，声明一下是整数，其实不声明也可以就找出它是能识别的，然后这里边我们要查的是数量，其实查询的方式和前面一样，只不过前面查询的是每一个字段，而这个里面查询的是只是数量而已，因为你完全可以把它copy再改。

我这个查询的是数量 com can't ID from this class post。然后实在是不等于2，这也得加上，然后 uzid不等于0，也偏上uzid这都没有问题。因为我们查的只是一个数量，所以你就不用排序了，也不用分页了，就一个值而已，这些可以去掉。

好你看很快我就把蛇口写完了，整个 dao这一层就开发完了，这一层其实是我们比较容易出错的地方，主要是配置文件里写错了它不提示，因此最好我们对它做一个测试，看看有没有问题。

这样我还是在上次课的 Micro test类里对它进行测试，那么我在后面再加上一个新的测试方法，这次方法我叫test，谁来poss。好，然后我把刚才我写的那两个方法都在这里，就一起测了。当然我测的时候要调用对应的map，我得把这个map先注入进来，目前我还没有注入把它注入进来。Out where this cars post member。

然后我就在刚才的方法里调用了 discus post map点select this cast post。这样我先传一个0进去，就是我先不按照用户来查，然后比如说我要查第一页的数据，第一页它是从行号从0开始，然后比如说我要查10条数据就这样，返回的是集合discount post，好，那么我把查询结果遍历一下，然后把它们分别输出一下，好，这是查询每一行的数据我先执行一下。

好，你看看这个查询结果他查到的是10条数据吗？1234、6789 10没问题，10条数据没有问题，然后是倒序排，那么最新的会在上面。这个你可以看一下这个时间，最新的会在上面没问题。好了，接下来我在这个方法里接着写，我再把总的行数也查出来。Rose等于 discuss post member。



第二select rose。我还是传入零不加 User的筛选条件，然后我直接看一下总行数是多少，我再执行一下刚才的方法，看结果一共是149条数据，是不是这样？我们可以在库里判断一下。Select count ID from discuss post。执行151，我这里忘了加那个条件了。Where status不等于二，忘了加一个条件了，所以不一致了。加上以后149，没错。好，当然了，我再加上 Uzid看一看加上uzid以后它会不会正确。

那么我们有很多优质，我看一下哪个优质的帖子稍微多一点101 0203挺多的，这样的道具，因为我不想看这些数据，这些数据都一样，看不出哪个数据是哪个数据最好看不一样的数据。

这样我就查用户ID为149的他发的帖子好，我把刚才的优质ID就改一下就可以了，149这个地方也是149好，改完以后再执行测试方法，然后看这个结果，最终的数量是三，然后他查的打印的值也是三，没有问题。

好了，经过测试发现我们写的 map没有问题，那么如果你写的时候有问题怎么办？你就其实像我上次课所说的，你就去看一下控制台它打印出来的circle有没有打印出来，我看一下，有打印出的出口，然后通常你一般都是字段写错了，表面写错了，所以说你通过去观察 Circle很容易就能够找到问题所在。

那么数据访问层开发完以后，第二步我们就可以开发业务层，开发的时候它会调用数据访问层，好我就在service包下新建一个业务组件，然后取名叫做this class post service。需要加上一个service注解，好让它能够被容器扫描到，然后很显然我这里要查询，我肯定得调用 map，我需要把这个map注入进来，discus post买法，然后其实我们当前方法他没有什么业务，反正就是把数据查出来就完了，就那么几个参数，其实所做的事情和迈克做的事情其实是一样的，他查到数据以后没有什么额外的处理过滤等等没有，所以说这个方法看起来非常的简单，可能有的同学感觉好像一写这么简单的方法似乎就没有必要，我干脆CTRL了就不调service调map行不行了？

你不要这么做，因为即便是当前我们这个业务方法很简单，但将来没准什么时候我们需要对查到的结果在业务层面做一些处理，这个时候你就缺少这个层面就不好加那个代码了。

再一个我们的代码的层次要有严格的保障，你跨层次去调用的话，那么我们后续在处理它的安全性等各个方面也会遇到一些困扰，所以说即便是我们 service的方法，再简单你也需要 control了去调它，而不要直接去调map，一定要注意严格按照标准结构来。

好，那么我声明一个业务方法查询出某一页的数据，所以说返回的是一个集合。Different knowledge discuss post，放好名，我叫friend discuss posts。好，那么参数一个是可以传入优质ID，一个是可以传入起始行，一个是每页显示的最多的数据的数量，我直接调用map就能把它实现了，点select discourse。好，直接把条件传进去就可以了。同理，那么再添加一个查询航速的方法。Find discuss，post。Rose然后条件用这ID，然后我就是这样return。Discus。Post map点select，rose传入有的ID好业务层的方法很快就完成了。

然后在这一层面其实我们还有也需要处理的地方，就是我们查询到的 Discourse post这个结果，它里边有一个外键uzid那么我们在页面上显示的时候肯定不会显示uzid肯定是要显示用户的名称，用户名对吧？

所以说两种办法，一个是我们可以在写SQL的时候关联查询用户，然后的话去处理，就是把用户的数据也一起查到，这是一种方式。

还有一种方式是我们可以就是单独的得到这个数据以后，单独的针对每一个discourse post，单独的给他查一下 user，然后把查到的user和discourse的组合在一起，反馈给页面，这也可以，我采用后者，因为这样的方式虽然说看起来好像麻烦一点，但是将来的话我们使用rides去缓存一些数据的时候会比较方便，到那个时候性能也会非常的高，然后我们代码的看起来也会非常的直观。

好，所以说我用第二种方式，第二种方式的话，我们就需要再提供一个业务方法，能够根据user ID查到user这个方法写在 service里不合适，因为这是跟user有关的操作，所以说我们可以再新建一个service，叫user service写在这里比较合适。

好，然后加上service注解，那么我们查询的是用户显然需要把用户的 map注入进来，对吧？User map，然后这里边我要提供一个就是根据用户ID查询，用户的这么一个方法，也是非常的容易 public查询一个用户，所以说返回优点就可以了。发个名。写得直观一点，我叫find user by ID，然后传入的是ID，我就直接return又是map点谁来给他ID就完了。那么这个方法是我们之前写过的，这里就直接复用了，我们就省事了。

好，到这我们把这个业务层要用到的方法也准备好了。那么接下来就是关键的一步，视图层我们就可以基于service去把它实现了。

那么这个视图层我们需要用到一些我们之前准备好的页面以及页面所依赖的样式文件 Gs，因为我们现在是开发正式项目的功能，我们应该把真实的页面把它都引入到我们项目中来，我们还没有做，现在就去准备。

你看这儿现在 style和template的下面它只有一级包，这个时候它是这样一个结构，我需要再多建几级包，因为时代下面还得放样式文件，图片等等还得多建几包，然后timeless下面也一样，但现在是这样的显示结构，不方便我再新建一级包了，我一建都建到这个太庙之下了。

你可以这样，你可以点齿轮，然后把勾去掉，它默认是把中间的包给你隐藏了，折叠起来的，你取消勾以后它就按照这个树形能够正式的展现。

好，下面我就把我们项目中之前我上传的这些静态资源考到我们乡镇来。首先样式文件图片和GS，这是静态资源，我们把它复制一下，然后粘贴到时代的下面放这里来就行了。

其次我要把网页粘贴到 template之下，我把 site注意这个index是首页set是其他的页面 copy，然后粘贴到 temps之下，这样我顺便把没有也粘贴过来，没有是我们将来学习发邮件邮件的模板要把它拷过来，然后也是粘贴到 txt之下好了，反正你看我就已经把我们这个项目中要用到的这些个页面和静态资源都拷贝进来了，这节课我们只用一个页面就够了， index首页。

好，反正这里先准备好了，一会用的时候我们再细看，好。拷贝完以后我就可以再把勾勾回来，让它中间层次的空包不显示，这样的话看起来清爽一点。好了。

接下来开发这个视图层，我首先要开发的是 Controller，那么我就新建一个CTRL，这个CTRL我取名叫做说这么大写 home很臭了，我取名叫home很臭了，然后前面加上很臭的注解，然后 controller我就不加访问路径了，你注意controller的访问路径是可以省略的，那么访问的时候就这一级就没有就空的直接访问方法就可以了。

那么CTRL在处理请求的时候，它查询它显然需要调用 service，所以我需要把 service注入进来。Discount post service除了 service以外，把u的service也注入进来，因为你通过它查到的数据中的 u的ID只是ID而已，我们通过user service可以再多查询一下，把优质的详细的数据查到，这样数据就全了。

那么注入这两个组件以后，接下来我就增加一个处理请求的方法，我先定义一下这个方法的访问路径放大一点，把这个不用的先关了。放大一点。Pass等于 index，因为我们访问的是首页，所以路径我叫index，然后请求方式查询它，就用的是 master点干的请求，然后方法它响应的是网页，所以说你就不需要写response包点，然后加上说明这个方法，然后我们可以返回末端的view，也可以返回死菌，死菌指的是视图的名字，这样比较方便，我们用这种方式方法名我就叫get index配置，然后这里面加上model，因为我要通过model去携带数据给这个模板，好，然后最终我返回的是这个模板的路径，那么我们访问的是谁呢？

是 template之下的这个index点html，所以说你就直接写index就行了，那么下面我就来处理，请求的逻辑无非就是查询，我们第一步先查到的是前10条数据，而不是每分页查询只是前10条数据，就相当于于是第一页，我可以这样写 discuss，post service，点犯的discount，pose，然后主页，他查询的是所有用户的数据，所以说 uzid传入0，然后第一页从0开始我要显示10条，我先写死，那么一会我们在优化的时候会把这两个参数替换成动态的值，到时候再说了，然后我们得到的也是多条数据类似的。

好，那么查到的数据它是不完整的，主要体现在优质ID，就这个对象里的优质ID，我们只是ID而已，不是用户名，但我们需要那个名字好在页面上展现，这个时候我们就可以把这个集合便利一下。

然后针对每一个 discourse post，我们再根据uzid查到user，然后把这个数据组装一下，放到一个新的集合里，然后返回给页面，所以这里我再新建一个集合，集合里装的是一个能够封装 discourse、post以及user对象的那么一个对象，其实你再写一个类也可以，或者你用一个数组也行，你用一个map也可以，这里我用一个map它的key是string，value是object，你存什么都行，然后我给它取名叫discuss，实例化一下。

好，然后这个值我需要遍历它，然后得到我先判断一下，你万一这个值查到的是空的，对不对？判断一下，如果这个list它不为空，在这种情况下我才去便利它。This cost。好，那么每次遍历我都需要把最终的结果装到 map里，所以说我先把 map在这里实例化好。牛希map最后一定别忘了，你要把 map装到新的集合里对吧？我就先写上省的忘了，然后在中间我需要往外边去装东西，首先装的是 post，帖子你得装进去，post，然后值就是这个对象。

其次我需要装的是user，这个user我们需要调用user service把它查出来对吧？Find by user ID。那么uzid我们从帖子里找，找到以后，一查询得到了它所对应的user完整的数据，好了，最终我再把优质也装到map里，你看经过这几次循环以后，最终迪斯卡斯普斯它里面的每一个map里，这两份数据就都有了，我们页面展现的时候就好办了。好，然后那么我们需要把最终的要给页面展现的结果装到mod里页面才能得到，所以说你还得写一句话，model点l的h build，然后给这个数据取一个名字discuss，值是集合好，这样就可以。

好了，那么CTRL来完成以后，最后一步就是我们要处理这个模板，那么接下来我就打开这个模板，我可以点在这里或者我可以搜，你搜的话就是CTRL CF的n搜，所有的文件我搜的是index，点html找到它，那么首先我们得写成那句话，我需要指定我要用的模板引擎是谁，对吧？Xmlns冒号th等于timeline f点2g其次还要做一件事情就是说我们这个页面上它依赖了一些样式文件或者图片，你像这样的图片我们不用改，因为它是一个绝对路径，写得非常的详尽，没有什么歧义，像它依赖的是一个不是抓捕的样式文件，这也是绝对论证没什么歧义，但是这种路径是相对路径，我们开发静态页面的时候，是这样一个关系，我们引入到我们动态的项目里的话，这个关系会发生一些变化，所以说就不能这样写，这样写可能就会有问题。

这个像这种路径我们也可以通过time leave的语法去处理。

我这里可以这样，th冒号，比方说 href里边引用的这个文件，我是需要通过他们内部模板去帮我找的，然后这里我可以这样写，at括号开头，括号结尾，然后就把这样写斜线css然后global点css那么这样写的意思是让 time leave去哪找这个文件去这个时代下面去找这个资源，所以说这样写他固定会去这样死在下找资源，就不会产生歧义了。

好，注意我们在页面的结尾处还引入了一些GS那么也是类似的处理，反正页面上的你看到的这样的相对路径的图片样式，GS所有的静态资源都用这样的方式处理，我在找到最后的位置，这样GS也是这样的，下面也是一样。

好了，那么这三个不用，这三个是绝对路径没有问题，然后页面上你看起来挺长的，有400多行代码，但其实我们需要处理的没多少，尾部不用管，每个页面都一样，不用动，然后上面头部你暂时不用动，将来需要动的时候再说，我们现在要改的地方主要是集中在这个内容里，而这个内容里，比如说他一开始这个标签我们还不用管，不能都折叠起来，这个标签我们还不用管，弹出框我们也不需要处理，提示框也不我们只需要处理是这个帖子列表的部分，然后因为我们引入的是静态页面，他的数据都是死的，他硬写死了10条数据在这个列表里，10个mi我们需要把它变成动态的，显然我们留一个li，然后以此为模板进行循环输出就可以了，对吧？

其他的9个li就不需要了，我把其他的几个li给它删掉，删掉，只保留一个。接下来我就以 Li为模板，要对这个数据进行遍历输出，那么使用timely for模板引擎做循环输出也是挺方便的，那我要循环输出all抑制就可以了，表示说我要进行抑制循环，显示的 li然后循环内部我得指定我要循环，对哪些数据进行循环，所以说这里我要写上表达式，访问的是集合或数组多条数据，我们在CTRL那里不是往帽子里装了这么一个集合，它叫discuss对吧？

所以我在表达式里引用集合，比如说我要遍历的是集合，每次遍历我们会得到集合中的一个map，对吧？

我们得说明一个变量好能够指代 map，所以我们可以到了前加上一个变量名，然后冒号，假如说我要询问的是集合，每次循环我将得到一个map对象，当然这只是一个变量名，你叫别的名字也无所谓，为了和我们的逻辑相对应，这里我叫map，因为我装的就是map好了，然后在循环的内部我就可以通过map去获取我要显示的数据，我们看一下哪些地方需要显示动态的数据。

首先你看它这个地方是显示的是用户的头像，用户的头像他目前是写死的，就是显示第一条第一个图片，这肯定需要变成动态的，对不对？你应该显示的是帖子的发布人的头像。

好，那么把这里的值改成动态的，就得需要依赖 TMD模板引擎的语法，所以说这个地方我要前面加上th冒号，意思是我后面的这个里边要包含time leave的表达式动态的数据，可以写上打了括号，然后我要引用变量中的值 map点，map里不是有两个值吗？

一个叫post，一个叫user，我要的是user里的图片的路径对吧？

所以说就是map点，user hider，URL那就行了，这是这个地方有一个动态的值，我们处理完了，好，再往后看，这里是帖子的标题，显然也得动态的对吧？

说话人2 51:44

那么。

说话人1 51:46

当然这上面还有一个链接能链到帖子的详情页面，目前的话我们还没有详情页面，先不用管，先这样，或者说你为了怕误点点错了，你先把它改成井号，就列到当前页面顶部，这样不会误操作，然后我要改变这里边的内容，我们之前也学过，我们需要用 th冒号 text等于然后的话到了这回这个值我还从map里取map点，post点开头，你注意比如说我这里解释一下我们这个表达式，比如说我这里写 map点，user其实它访问的是map点get user，因为 timeline能识别 map是一个希map类型，那么它会调它的盖的方法，然后以u的为t去访问，然后我们最终得到了一个u的对象，然后我们又点high点URL，那么它会访问的是这个u的对象的 Goette、hider，URL，那么对于我们的实体内，或者说对于普通的我们自定义的JavaBean，它会调用我们的 get方法去访问属性，因为这个属性它本身在这个类当中是私有的，你直接访问是访问不了的，但是他知道通过盖的方法去访问他，所以我们在写的时候写的是点一个属性，但其实它底层叫的是get，这个也是一样的。

好，然后 text其实还有一个表示方式还可以这样写，you taxed两者的区别是 test，只是就是说按照最原始的方式显示文本，而u text有这样的好处，比如说你文本里有转移字符，它能够把转移字符显示成它该有的样子，比如说你这里头有一个转移字符是and lt这表示小于号，如果你用这个text，这里边就会显示这样的一个小你页面上看起来就会很别扭，如果是UTech，那他会把这个转义字符显示成他该表现的小鱼儿的形式。

所以像这种标题内容，这里边他有可能会包含转义字符的情况，你最好用这个Utecht，所以这里我用you taxed。好，再往后看，这儿置顶和精华也需要判断，因为不是每个帖子都置顶，每个帖子都精华，

说话人2 54:37

那么。

说话人1 54:39

必须你是置顶的类型，我才显示这两字对不对？

你是状态是精华，我才显示这两字儿，需要加判断来显示，判断的话也非常容易 th冒号if等于然后这里需要写上判断条件，我们根据什么判断？置顶。应该根据帖子类型对吧？Map点post，这是帖子，然后点太这是它的类型，1=1的时候它才是置顶的帖子制等的类型，这个时候它才会显示，如果这个条件不成立，义务不成立，Span标签是不会显示的，这是if标签的一个作用很方便。

所以你看到了括号里边写的表达式，我们不只是可以访问数据，其实在访问数据之后也可以对这个数据直接做一些运算加减乘除判断相等，包括三元运算都可以去处理好，你这个会了这个也就容易了。

Th冒号if等于这个看状态对吧？Map点post，点status，状态等于一才是精华。好，那么这个精华也置顶和精华咱们也处理完了。

然后再往后看这个地方显然显示的是帖子的作者，那么目前是写死的，你应该把它改成动态的，对不对？那就th冒号，Utecht等于这回是macke的user，从user里取值显示的是user name，后面发布于时间，那是帖子的创建时间，对吧？也是th冒号，然后这个时间你就txt，它里边肯定不会有特殊的字符等于。那就是mad dear post，dear create time。那么你注意如果你直接这样写的话，那么它显示的时间格式是西方的那种方式不符合我们的阅读习惯，所以我们还需要对这个时间做一个格式化。

那么timeline它内置了几个工具，可以直接对这个时间进行格式化，那么这个工具的引用方式是这样的，这样用我先把它剪切一下，diss它有一个diss工具，然后我们可以通过井号引用这个工具号是引用它的一个工具，然后Dx点format括号括号里边先写上你要访问的这个数据是动态的吗？

Map点post点great time，然后再来一个参数写上了，你要指定的格式，我希望这个格式是这样的，歪歪歪mmddhhmmss好，所以格式化日期格式化就是固定的一个语法，它内置的一个APA你记住就好了。

好，再往后看。后面就没有了赞回帖目前我们先不去处理，因为我们现在回帖的功能还有点赞的功能还没有开发，所以我们先不处理了。现在我们可以把服务启动一下，然后试着访问一下首页，看一看它有没有什么问题，如果没有问题的话，我们再去完善分页的功能。

好，下面我就把服务启动一下，一下，run的过程中没有报错，没问题，我打开浏览器，然后访问一下host8080，然后community index我们的首页的访问路径就是index。

好，那么你看没有问题，这个效果还是ok的。然后这个数据你看就不是静态页面自带的那种状态了，就是我们数据库里的实际的数据了。

当然目前显示的是第一页前10条没问题，但这个分页肯定是不管用的好了，那么接下来我们就要继续完善分页的功能，那么分页其实大家都操作过，它的逻辑很容易理解，我点第二页第三页，你就给我查询出对应的那一页的数据，当然你点上一页下一页其实就调整页码，和你点23也没什么区别，点首页末页也是一样，总之你就点击这里面的任何一个链接，其实最终都会转换成一个页码，比如说三，然后你是要把页码传给服务端，查询出第三页的数据，仅此而已，对吧？

再一个客户端它可以决定我这一页最多要显示多少条数据，如果有些网站还可以做成可配的，你要显示多少条，我们这个没有可配，相当于就固定了，这个如果说可配的话，也可以把每页显示最多多少条传给服务端，然后服务端得到的是你当前要显示第几页，最多显示多少条，然后我帮你查出相关的数据，然后再给你。

这个时候服务端给客户端返回的就不只是这一页的数据了，他还得给服务端返回一些额外的东西，比如说最多有多少页，然后你当前是第几页就这个信息也要给他反馈回来，因为你看第一页点中的话它是点亮的，对吧？

所以说当浏览器访问服务器，服务器响应过来的时候，还得把这个是第几页带回来，我好知道到底是第几页，我好把它点亮对不对？再一个因为我要显示一些页码，从几到几页，我得知道那一共有多少页，我会不会超出这个范围对吧？

所以说服务端还要给客户端返回一些和分页有关的信息，客户端要给服务端传一些信息，然后服务端计算完以后还给他反馈回来，所以来回传的数据是比较多的，也比较杂的。

那么我们最好写一个组件把它封装一下，封装完以后能够便于复用，然后将来我们还有很多其他的功能，那么在那些功能里，其实我们也可以把封装好的逻辑能够复用，最好能达到这么一个状态。

好，下面我就给大家把封装分页的组件给大家写一下，这样我就在安推的一个包下，再新建一个类，用来封装的分页的一些条件，我就叫页码对吧？配置。

好，那么这个写注释，封装分页相关的信息是这么一个组件，那么首先我们可以利用这个对象，从让服务器服务端接收页面传入的信息，我们先解决页面传入的信息的接收的问题。

首先页面你必须得告诉我你当前想显示第几页，所以你要传入的是当前的页码，显然页码应该是个整数对不对？这个变量名我叫做carry给他一个默认值，你也可以不写，如果你不传，我默认认为是第一页，好，这是当前页码。

第二个页面还可以传入一个就是你最多要显示多少条数据，那我称之为这个显示的上限，那显然也是一个整数 into limit，也给他一个默认值，你也可以不传，我就按默认值来走。

好，三第三个不是页面传入进来的，是我们需要自己查出来的，我们需要一个什么？这个数据的总数，就一共有多少行数据，这个是我们服务端查出来的，也要塞进来。

我们用这个数据干什么？我们用它来，我们用于计算总的页数，因为我们把响应返回给页面以后，页面上要显示页码，那么它一共要显示到多少页呢？然后能不能超出它总的页数，所以你得页面得知道我一共有多少页，我被显示错了对吧？

所以说他要算出总页数，而总的页数怎么算呢？你表里有多少条数据我除以一下上限，每页显示多少条？一除就知道了。所以这个数据总数需要给我，那么显然也是整数，肉丝。

最后再来一个是查询路径，查询路径，为什么需要查询路径呢？你看页面上我们首页末页某一页这都可以点，很显然这是个链接，我们在页面上处理这个链接的时候，链接上你得带一个路径，那这么多地方都要带路径，如果你挨个去拼也可以麻烦对吧？如果说我组件里带上了查询路径，然后能够把它返回给页面上，直接就复用一下变量不更方便对吧？

所以说查询路径的作用是用来复用分页的链接，就让我们页面上写的逻辑简洁一点。好，那么这个路径是字符串，我叫做pass。

好，那么前两个是页面要传给的东西，后两个是我要自己要设置好的东西要返回给页面页面需要用到的。

那么。

接下来我要给它生成get set的方法，因为这些属性是私有的，外界需要通过get set对它进行访问，然后注意这些数据比较特别，我们在赛德的时候还需要做一些。

判断，避免它传入一些错误的合理的数据，你比如说site，current如果万一他用户乱写，他把这个路径里边的条件写成了负数怎么办？写成了一个特别大的数怎么办？对吧？

所以凯尔我们需要做一个判断，如果说the current大于等于一，这个时候我才会去把它记录下来，我才认为它是有效的，否则我认为它无效。当然我们主要判断它是不是0或负数的情况，至于它特别大也有可能，因为也可能有很多页我们就不管了。

好同理 Set limit也是一样，你不能传入一个特别奇怪的数，那么我判断一下，如果 limit它大于等于一，我认为是合法的，并且 limit小于等于100，我认为是合法的。

就是说我对 limit每页显示的上限有限制，介于1~100之间，那么万一用户说我这一页想显示1万条10万条数据，那么我们一下给查出那么多的数据，一个是对我们服务器的压力太大，再一个这么多数据给浏览器浏览器就卡死了，用户体验也很糟糕，所以说我限制成最多100条。

好再看那么set rose这个也是一样，也得对它做个判断，那么如果rose大于等于0，我再对它inside好，这个pass就不用了。

好，除了这些条件以外，我们还需要补充几个方法，再额外的提供几个条件，这个条件一个是数据库它在查询的时候需要用到的，一个是页面它在显示的时候需要用到的，我们原始传进来的这几个条件是不够的，但是我们另外的条件是通过他们计算而来的。

好，首先我要加一个什么条件 get off site，你想想，咱们数据库查询的时候，它不是需要你传入一个 off site，他要的并不是当前的页码，它需要是当前页的起始行，所以你需要通过当前页的页码算出当前页的起始行给它。

好，我写个注释，这个方法的作用是获取当前页的起始行，是这样一个作用，然后通过当前页来算，其实很好算，当前页不就是current对不对？当前页乘以每页显示多少条数据，就是当前页最后一行，对不对？你再减去这个里面它不就是它的起始行吗？所以你就可以这样委托这个公式，或者你可以提一下，你可以这样开问题减一乘上里面就搞定了。非常的容易，这个方法是在我们访问调用查询的时候，那么需要用到这个方法获取 Offset。

好，然后我们还需要再来再加一个方法，提供这样的条件，get total。这个方法的作用显著是是用来获取总的页数，因为我们一会在页面上要显示的是要显示一段页码，页码的范围你不能超过它一共可能出现的页数，这是为了显示页码，做边界判断需要的一个条件。

总的页数其实很好算，总的行数咱们不是以当前传进来了吗？总的行数你除以每一页显示多少条数据就可以了。当然有可能不整除，有点余数，不管是余几行都再单给单多一页，对吧？再单独多出来一页，所以还有可能是加一的情况，这块就得判断一下了。

如果rose是能够长出里面的，我就委特肉丝除以里面的整数没什么好说的，否则不能整除，我就委屈肉丝除以里面，最后我再加上一就可以了，这块写错了，整除 Rose除以name，0忘写等于0了。

好，这是获取总页数，那么接下来我们还要提供另外的两个方法，我先把方法写上 get from。

Get to就是从第几页到第几页，咱们页面上是要显示页码的，比如说我有100页，我不可能从第一页写到100页，太多了，我们经常上网也知道它一般都显示一段，比如说我当前是第八页，我可以显示6789 10我可以显示当前页前两页以及后两页，其他的我就不显示了，离我最近的两页我把它显示出来，对吧？

所以你还得给他算一下，你根据当前页算一下，我起始页截止页是多少对吧？好，那么我们回来把这个方法写一下，是获取起始页码，这个是获取终止或者说结束页码。

好，那么起始页码刚才说了是当前页，离它最近的两页，当前页起始的得减二，那就是让吧等于凯瑞当前页减二，但是有可能当前页恰好是第一页，你减二以后变成负的了，就不对了，对吧？

所以你return的时候还得判断一下，如果这个数让他小于1减完以后，小于1变成0或负数了，这个时候你就认为它是1从一开始从最小值开始，否则就是它该有的值，所以把左侧的边界定好，不要小于-1的范围出现，get to和这个道理就一样了，it to，就等于当前页加上二，但有可能你当前最后一页第一百页你再加二不就超了吗？对吧？

所以你还得判断一下，抄的时候就按照最后一页来算，最后一页整个页数干得头疼，我就return to，如果它大于了头头大于总页数，我截止就显示到总页数为止，否则就显示它本身好了，这样的话我就把我需要的条件都封装在了配置里，那么有些条件是页面传入的，有些条件是服务器自动查传入的，有些条件是我经过这些已知的条件算出来的，那么它总有它用的地方。

我们接着去看一看他们这些条件都是在哪传进来，我们都怎么用的。

我在回到这个CTRL了，我需要改造刚才的查询方法，让它能够支持分页，分页的时候我们也说了，页面会传入分页有关的条件，而这个条件我就可以用配置进行封装，所以我加上配置这个条件，然后配置我们需要给它设置一些我们服务器需要给它也设置一些值，比如说它的总行数就需要我们在这进行设置。Set rose这个总行数我可以查出来对吧？Discuss post service点find rose。因为我们是首页用户ID不以用户ID来查询，传入0，page点set，pass路径，当前的访问路径不是index吗？传进来页面上我就可以复用这个路径了。

配置还有一个当前页码，还有总的每页显示多少条数据，页面可以传进来，这就不是服务器要操心的事了。然后那么还有一些值那是需要返回给页面的配置可以自动算出来，根据这些条件自动算出来，就不用我们去操心了，那个算法已经写好了。然后我们这儿查询帖子的时候，这块就不要写死0了，奥赛的从配置里取，他给我们算好了吗？

配的盖奥赛总的页码，从开始取点get limit，limit从页面传进来，如果不传就默认值好了，然后这个配置我们在这儿利用完了查询的时候，那么页面上也需要它，我们也需要把它返回给页面，你可以这样model点的actually build，然后的话把配置装到model里，但其实这一步可以省略，为什么我在这解释一下，首先我们知道在spring mvc框架之中，那么这个方法参数往往是由不是往往就都是由dispatcher asseverate帮我们初始化的，model是由它初始化的，配置显然也是由它初始化的。

那么配置的数据也是由他给我们注入进来的，this spot除了帮我们初始化这些数据以外，它还额外做了一件事儿，它会自动的把配置装到model里，所以在当前的方法里你可以直接调用配置，其实也可以通过model获取配置。

所以这里我解释一下，写这个故事解释一下就是方法调用之前，么spring mvc它会自动实例化model和配置自动的，而且它还会将配置注入给model，所以我们在timeline这个模板里就可以直接访问配置对象中的数据，你就不需要model再艾特一次，这块有一个隐含的逻辑大家注意，所以到这我control就改造完了，就这么两两个地方，接下来我们就可以打开首页，然后把分页的逻辑给它处理好，能找到分页的逻辑在这儿。

好，那么这个逻辑处理起来有点繁琐，但是大家要保持耐心，咱们一点一点来别着急。首先我们先处理首页和末页的逻辑，显然首页末页逻辑是应该是相似的，对吧？我们先把它处理一下，因为我加上标签以后代码会变得长一点，这样我把它调整一下，让它有点层次感，这样的话一会不会超出一屏看起来比较舒服一点，是这样的，然后你注意中间这个是显示的是页码，12345这个之前是写死的，这很显然我们一会要把它改造成动态的应该是这页码是循环显示，而不是写死，所以我只保留12345就三了，一会在这个模板基础上循环再来再处理一下好。

调整完结构以后，下面我就去处理一下它的显示的逻辑。首先我对整个分页的区域做一个处理内容做一个处理，因为什么？如果说你没有查到数据，假如说一条数据都没有，假设极端情况出现了，一条数据都没有，这个时候你还有必要显示分页的区域吗？就没有必要了。所以没有数据时它不显示，有数据时才显示咱们搞得严谨一点，因此我写成th if等于好。

那么我怎么判断它有没有数据，那么刚才不是说了吗？

一个你可以看Discus poss对吧？它里面有没有数据，再一个你也看可以看配置，因为配置里不有行数，如果你一行都没有，也就是没有数据，这里我就看配置，我就hit点rose，实际上它掉的是get rose方法，如果它大于0，这个时候我才显示这个区域，这是这个地方的编写。

然后再写首页的逻辑，主要是处理这个链接，这个链接里边得是一个动态的拼了一个链接的字符串，所以我们还需要用到time leave th前面加上th然后艾特括号括号里边我要拼上当前的页面的防路径配置里有对吧？

你需要访问配置里的变量，你就这样写到了括号配置点pass，然后访问到了这个路径以后，你可以后面加上还得加上这个参数，因为你首页其实就是访问第一页，你得加上开始等于一，你可以这样写问号，它让它等于一，然后到了括号语法里边还允许你这样写，你可以直接写个小括号，然后carry等于1，那么它会帮你转换成问号，carry等于1，其实都可以，然后这样写的话，如果你还有一个条件你就逗号，然后比如说limit等于比如说等于5，这样也可以，中间以逗号分隔，那么他最后给你拼出来的效果就是这样的。

我说你这样随便写一下，比如说他平常是这样，index，然后问号，carrot等于1and limit等于5，是这样一个结果，但我们只需要这样写，当然我现在我就不拼limit了，默认10正好我就只写一个凯伦的首页，你首页会了末页逻辑不跟他一样对吧？

末页也是路径复用，然后当前页等于几，它就等于总的页数，而总的页数配置里不是也有吗？变量通过表达式获取，有没有少写一个大括号，整的页数里有配置点，偷偷就可以了，所以首页末页咱们就处理完了，处理完以后再来处理上一页和下一页它的逻辑肯定是相似的，对吧？



上一页这个路径也是类似的写法，提示冒号，然后艾特，然后也是路径也是配置点pass，后面也是要带上当前页是几，当前页那得等于这个条件就等于实际的当前页减一，因为你上一页对不对？而实际的当前页是个变量，它不断的变化，你是配置点，carry减一对，是这样的，然后检查一下，千万不要少写了一个括号的结尾，没问题。

好，同理下一页也是类似，下一页就简单了，加一是吧？加一就搞定，最后再处理页码，页码不是永远显示一，它要显示从几到几那显然得循环，那么循环的话我们还是用 thh但th一是之前我们用它里边我们要传一个集合进去，现在我们要从比如第三页到第八页，这是两个数字怎么办？

其实贴 timeline它也有相关的知识，然后这样去做，它有一个还是一个工具叫numbers，numbers这工具有一个方法叫seconds，这个方法会产生一个返回给你返回一个数组，返回一个由数字连续的数字组成的数组，数字的起始和终止的那两个数你要指定好，我指定一下，这个一个是配置点 strong，另外是配置点，你注意这个地方你不用选dollar了，因为我在这已经写过了，表示这里边都是变量，这里边我们调的是一个工具里边这是变量，所以这地方你就不用再写一个dollar括号了。

好，那么这句话的意思就是说我要生成一个从这个数到这个数之间的连续的一个数组，得到数组以后，我还得有一个变量来引用每一个数字，我就前面加个I吧，发一个冒号，然后每次我们要显示的页码 I对不对？I所以我在这儿 th冒号，tx等于通过表达式引用这个变量a就可以了。

好，那么除此以外还有一些细节要处理，什么？如果你当前是第一页，上一页就不可点，如果你当前是最后一页，那下一页就不可点，首页末页无所谓，因为你是最后一个点下一页就超了，你第一页点上一页就超了，这个不行。再一个页码，当前是第几页要点亮，我们把再处理一下就搞定，好。我们先来处理上一页和下一页，你看上一页我们页面上默认它就不可点，这个不可点样式是通过disable实现的，只要你在 Li加上disable，它就不可点，你不能永远写死，而是用来动态判断。

所以 class这里边样式你要动态的添加，它是一个动态的值，所以前面我们用加上th表示说这里边我需要有动态的数据去处理，那么当然这里边不只是有动态的数据，这个是固定的，那么我们在这种情况下，用固定的数据再拼上一个变量怎么搞，可以这么处理。

就是你加上两条竖线表示说这里边有静态的值，这是静态的值，然后还有变量，有空格还有个变量，你就到了大括号拼变量就可以了。

好，那么我要不要拼 disabled，你得看一下我当前是不是第一页，我是就拼上disable就不让他点对不对？而当前页是配置，第二carry当成一2它等于1吗？如果等于一我就不让他点，就把样式加上disable，如果它不等于一就可以点，我就什么也不加空的，ok这就搞定了。

同理下一页不是一样的逻辑，th对吧？然后也写上竖线，也是在这竖线里边拼上一个变量，那也是看当前页配置，carry它等于等于就不是一了，它得等于最后一页，最后一页是配置的total，如果他等了我就加上样式不可点，如果不等我就什么也不加，可以点。

好，上一页下一页搞定，最后还有页码。好，页码点亮与否，你看一点亮了2345没点亮对吧？我当年刚好留的是一，点亮与否看的是哪个样式是样式 active。如果你在xli上加上active，它就被点亮，不加active就不点亮，所以说很显然的 Class还得是动态的去拼，也得是th对吧？

也得是竖线，然后让一个静态的值，最后拼上一个动态的值，什么时候我会把当前页码点亮，那就是当前页，刚好 I刚好是当前页的时候对吧？

因为我们页码 I I刚好是当天日的时候就点亮了，所以这里面你可以写 I它不等于配置 character，如果它等于我就加上IQ把它点亮，如果不等于我就什么也不加就可以了。

好了，你看这个地方写这个标签还是挺挺麻烦的，不过好在我们写完一次以后，那么其他的页面上所有的页面的分页它的逻辑都一样，以后就可以复用了，我们再也不用写第二次了。

所以你写的时候你也是按照我这样就是一步一步耐心点，把它一次性的把它搞定，下一次就好很多了。好，那么完成以后，下面我们把这个代码重新编译一下，我按一下CTRL f9，然后我再重新访问首页看一看刷新，刷新以后数据默认第一页没问题对吧？

还是第一页没问题，然后你看这个页码就不是12345了，它显示的是当前页码的相邻的两页，前面没有，后面就两页。没问题。我点下一页可以吗？第二页没问题，然后当前页相邻的2页到4了，下一页没问题，下一页没问题对吧？上一页也可以，末一页没问题，当末页第十五页的时候，下一页不能点了，对不对？首页没问题，首页第一页上一页就不能点了，对吧？这回我们终于就把首页功能完成了，那么把分页稍微麻烦的功能也完成了。

好了，这次课我们就演示完了，我们要演示的内容，那么你课后一定要把第一个功能好好的把它自己把它搞定，然后这样的话会比较有成就感，你第一个功能搞定，这是一个好的开始，那以后的话我们一个功能的去把它做出来，那么这个项目慢慢的完善起来了。好，这节课我们就讲到这里，咱们下节课。

## 1.7 项目调试技巧

7-项目调试技巧.mp4

说话人1 00:03

这次课我给大家讲解一下项目的调试技巧，因为咱们已经学会了第一个首页的功能，那么你在自己课后开发的时候，我想肯定会遇到各种各样的问题。那么经过这次课我们讲解调试技巧，你课后把它运用到你的开发当中去，那么一个它能够帮你解决问题，能够让你把功能调出来。

第二个，不管是首页的功能还是以后的其他的功能，你都要反复的去练习这些技巧，然后逐渐的把自己的独立解决问题的能力锻炼出来。

那么其实企业很关心的你能不能解决问题，这一点在你就业的工作的时候，它体现的尤为明显，所以说这方面是很重要的，一方面大家要多去运用多去锻炼。

那么这次课我们讲调试技巧主要是从4个方面来进行讲解。第一个方面就是说首先我们了解响应状态码的各种含义，因为我们通过浏览器向服务器发送了一个请求，那么这个请求回来他得到的是什么信息？

我们通过状态码其实大概能有一个判定，这次请求是成功了还是失败了，能看出来再一个如果是失败了，这个错误是发生在服务端还是客户端也能够判断出来，这样的话我们通过状态码能够大概的判定一下这个问题的所在。

然后判定出问题的所在以后，如果是也是服务端的问题，我们可以在服务端进行断点调试。如果是客户端的问题，我们也可以在客户端进行断点调试。

当然了，那么不光是断点调试，我们在断点调试之前，或者是在调试的过程中，也可以结合着日志，然后去跟踪这个程序去找到你问题的所在，所以说我们这次课主要是讲这4个方面的内容。

第一个咱们先来了解一下响应状态码的含义，这个我们可以通过 Http协议它的手册上面有介绍，我们可以看一下，好我打开浏览器，这个网站莫罪了，网站上次课之前我给你介绍过，然后我们要看的是这部分内容 HTTP状态的返回码，好点进去，然后他这个状态码有好多，咱们课上就不一个的去解释了。

然后我解释一下比较常见的这个状态码，基本上我们写的程序可能百分之90都是这么几种状态码，其他的如果说你在开发的过程中遇到了不常见的，你自己来查一些手册，看一看大概是什么意思，然后如果说你查手册，他可能是有的地方说的不是那么的详尽，你觉得不够详细，你可以在百度搜一下这个状态码是什么意思，那也会有人遇到类似的问题会给一些参考的意见。

那么我们平时最常见的第一个状态码是200，其实以2开头的都表示成功了，那么200 200就表示请求成功，然后成功了没什么可说的，就这个程序基本上应该是没有问题的。

好。

然后其次我们经常看到的是302以3开头的都代表着是重定向，那么302是我们最常见的重定向的一种形式，当然我们还没有去介绍什么是重定向，借这个机会给大家讲一下什么重定向，然后后面的功能里我们会经常会用到要说这个成像，我需要简单的画个图，假设这个是浏览器，我写上假设代表的是服务器，二者之间要产生通信，比如说我服务器一有两个功能，我已经开发好了两个功能，一个是我要删除什么？

一个删除的功能，比如说删除一个帖子，然后比如说我还有一个这个查询的功能显示，所有的帖子这两个功能它都部署在服务器上，我们浏览器都可以进行访问，比如说我访问的是删除的功能，那么他把数据给我删掉了，删掉以后他应该给客户端什么响应呢？

给他反馈什么数据，我们的项目中大部分的请求返回的都是网页，而你把数据删掉，你返回什么页面呢？

所以说其实这种情况其实我们应该是返回的，是查询的数据，因为把数据删掉以后，我返回的是那个列表页面，你一看数据没了，查询是删除是成功的，你很确定很直观，所以说我要返回的其实是查询功能能够返回的页面。

当然我们可以这样做，说我删除在删除功能里我去调查询，然后让查询返回查询页面，一客户端的一看那数据没了，ok这是可以的，但是这种情况是一个不好的编码方式，因为咱们的查询功能和删除功能其实是两个彼此相互独立的功能，它俩之间其实不应该产生依赖，而你这样去调他就产生了依赖。

那么一旦功能变得复杂的时候，一旦或者是业务发生变化的时候，有可能你给他就要牵扯到他，那么这有一种不太合适的方式，那么我们利用重定向就是能够很好地解决这个问题。

所谓的重叠性就是我这种情况下我删除功能可以给浏览器做一个响应，但这个响应里其实我不给他返回什么网页，而是给他返回一个建议，首先状态码给的是302，浏览器一看是302就知道了说你并不想给我返回什么网页，而是建议我去访问另外的一个功能。

那么返回的响应里还会带上一个就是一个地址，一个网址，我用location来代表，除了302之外还会带一个建议你访问的目标，那么浏览器根据你给我的路径，他就会去访问，自主的去访问他，应该访问功能，比如说查询，然后查询给他返回一个列表的页面，是这样的一个过程。

所以其实302是这样一种感觉，就是说我功能结束以后，其实我没有什么能够给予你的，但是我给你一个建议，你可以访问另外的一个功能，然后的话它能够给你很好的一个数据的展示，就是这样，你看这个最终也是调了删除的功能，但是不是通过也调了查询的功能，但是不是通过客户端去调的，而客户端是发两次请求去调，他们这两次请求是彼此独立的，所以说这两功能没有产生耦合，这就是重定向的一个价值。

然后我们项目中比较常见的地方，举个例子，比如说我要注册一个账号，我注册完以后注册完了，我去哪个页面再回到注册页面也不合适，对吧？你就可以直接跳到登录页面，这个时候就可以进行重定向，因为注册和登录彼此是互相独立的对吧？它们之间不要产生耦合，就可以用这种方式去做一个跳转。

总之就是重定向，它是能够以一个非常低的耦合的方式进行一个功能之间的跳转，那么后面实际的案例中我们再去演示，反正大概大家先对他有一个了解。

好，我再回到这个页面，网页那么所以302代表是这样一个含义，能够让浏览器自主的再发一次请求，实现一个功能的跳转这样的一个功能。

好，这302我们也很常见，有的时候我们可能会看到以4开头的状态码，这个时候以4开头的状态码就表示说有错误了，我们最常见的是404，我想可能大家平时上网的时候也会经常见到这个状态，404代表的是我下6下服务器发出请求，我要访问的功能不存在，这个不存在，它往往是因为我们写路径写错了，你在地址栏写路径写错了，或者是在超链接上写的链接写错了，或者是你这个表单上，你写的路径写错了等等，往往就是我们路径配错了，所以你要检查你页面上的路径写的是否正确，这是一种比较常见的状态码。

好，再往后看，然后还有15开头的状态码，最常见的是500，那15开头表示这个服务器接收到了你的请求，但是服务器在处理的过程中它发生了问题，发生了错误，所以遇到这种情况，我们应该去检查服务端的程序，大概是这个意思。

好了，总之我们最常见的几种状态码，一个是200表示成功了，一个是302表示要跳转要重新下，一个是404表示说我的路径很可能不对，所以找不到服务器的资源，一个是500表示服务器在处理的过程中发生了问题，那么我们需要去跟踪服务端的代码，好再回到 Ppt，所以我们通过这个状态码就能够大概的猜测到这个问题，它到底是我客户端路径写错了，还是服务端处理的程序写错了，然后我好去有针对性的去调对吧？

这是一方面。

如果说具体来说应该是服务端的问题，我怎么去调呢？如果是客户端的问题我怎么去调？其实我们的开发工具它都支持我们在程序当中打断点，然后你程序在运行的时候能卡在断点的位置，然后我们可以逐行向下执行，以慢动作的形式向下执行，这样的话你可以很好的跟踪程序，然后哪一步你发现他执行的和我预期的不一样，或者是哪一步他得到的数据不是我说我想要的，可能就是出现了问题，我就可以根据来进行排错。

好，下面我把服务端和客户端这两种调试技巧给你演示一下，怎么打断点给你演示一下。我先打开这个 Idea，然后我就以上次课我们所写的首页的功能，来给大家演示一下这个断点调试的方式。

比如说假设我发现服务端程序它就出了问题，这个结果不对或者是报错了，我想跟踪一下程序，你可以在这个方法的第一行方法内部，你不要把断点打到方法之外，因为他调的是这个方法，只是从这一行开始执行程序，你可以在这打。

当然如果你确认这两行没有问题，你在这打断点直接在这打，也可以那程序就卡在这儿，比如说我想从头开始在这打个断点，然后如果说你想让程序卡在这儿，那么你在启动的时候还需要以debug模式启动程序，而不是普通的run，怎么以debug模式启动程序，比如说你选择这个类右键debug就可以了，不点转点 debug叫虫子，或者是在这儿选中这个类，然后点图标也是ok的。好，我就点一下把这个服务启动起来，服务启动好了，为什么没有卡到这个地方？因为只是启动服务而已，我们还没有访问这个功能对不对？

只有你访问的时候它才会它在执行的过程中走到这儿才会卡住，好吧，我们再打开浏览器，我去访问一下这个功能 Local house, community index，好，你执行你会发现浏览器一直在这楼顶它不结束，因为什么？因为我们服务器的程序卡住了，就卡在了断点的这一行，然后我们要跟踪程序，有的时候可能是让程序往下走一行，看一看有没有问题，有的时候是要进入到当前行的某一个方法的内部，看有没有问题，你就需要掌握一些快捷键，当然你点其实点可以这几个按钮，也就是我们快捷键其实就是操作这几个按钮，这个按钮比较多，比较常用的其实就三个，我直我就直接说快捷键了。

首先如果说你想让程序向下执行一行，你就按这个F8，你看它就向下执行了一行，你再按F8他又执行一行。

所以F8的意思是让程序就是逐行的向下去执行，然后比如说到这行了，我想进入到这一行的方法内部去看一下，你就按这个F7 F7是进入到当前行所谓的方法内部，F7你看我们当年好像不是掉了，get outside，所以说就走到这里来了，我一看好像没有什么问题，我就 F8过去刚开始是进入到了 get offset，然后你站F7他就进到了get limit，然后站F8，又回到了这一行，然后再F7就进到了这个方法内部就是这样的。

然后当我们这个程序走到某一行时，其实我们可以看到这一行代码之前变量的值或者是参数的值，你比如说在这个范围之内，我可以很容易地看到这个参数 uzid它后面写出来了等于0，offset等于0，limit等于10，你能够看到这个参数的值对不对？或者是你鼠标悬停到参数的上方或者是变量的上方，它也会有一个提示很方便。好，我这F8又回到这儿，然后在F8就走下去，然后——行走。

好了，然后跟踪到这儿的时候，可能我们就再往下就跟到了循环的内部，这个循环如果说它只循环10次，你一次跟可能还好一点，如果是循环100次1000次，你一次跟就太麻烦了。

如果说你想我这个for循环好像没有什么问题，我不想跟了，我想直接走到这儿来，你可以在这一行打个断点打断点了，没点一下就打断点了，打个断点。

然后我希望我这儿不是逐行去跟了，而是直接跳到这儿，你就按快捷键f9，f9的意思是程序直接向下执行，直到执行到下一个断点为止，如果下面没有断点就直接到执行到底了，就执行到结束的，你看就卡在这了。

如果我再按f9整个程序就执行完了，我们打开浏览器就看到了这个结果。

当然了我们通过打断点逐行跟踪的方式，你可以——行跟踪你的程序，看一看它执行的顺序和我们编码时所预期的那个样子是不是一样的，是不是走偏了，这一个看程序执行的路径对不对？这是一方面。

第二个在每一步，比如说在这一步，我可以看到这一步之前的变量，比如说配置对不对？好，我们再演示一下，我刷新一下，回来我要看配置对不对？但这配置它显示的是一个地址，希克的看不到内部的情况，你想看的话其实也很方便，在这儿 where boss变量这里有配置，展开看这里有相关的值，还有model，你看这个model你可以展开看，你看model里边是不是就已经有了配置，所以上次课我就提到过，我说这个是不是mvc在调这个方法的时候，它会自动的实例化model和配置，然后将配置注入给model，所以在这儿就能够从model中得到配置的数据了，在后面的页面当中自然也可以得到你从这就可以印证一下，因为你一看model里确实有配置，已经装进去了。

好，所以你利用断点也可以很容易的看到它每一步之前执行过的程序当中变量的值对不对？也能够帮助你排除一些问题。好了，那么就这么几个快捷键，大家把它练熟了，多一跟就好了。再一个如果说我们程序中就到处打了断点，比如这打了断点，这打的断点，然后这打的断点，可能是你掉到service里你也打了断点，也可能是你掉的配置里你也感到动力。

我们在跟踪的过程中，可能我到处打了好多断点，然后那么我跟踪完了，排除完这个问题了，我想把断点去掉，你一个点肯定是很麻烦的，其实 idea它可以统一的帮我们去管理所有的断点，那么管理端点的功能是点通过点这个按钮弹出个界面，那么在界面上你能看到这个你曾经打过的所有断点，然后你点击一下就能切换到断点代码所在的位置，在这能看到切换可以切换，然后如果说我暂时比如说配置里的断点，我想不想把它删掉，但是我想让它不可用，你可以把这个勾去掉，它就变成空心的了，这个时候就表示不可用，但你想恢复的话再勾上就可用了。

好，如果说你想把断点删除，你可以选中若干个断点，然后你是右键删除或者是delete就没了。所以这个界面可以统一的管理，我们所打的所有的断点也非常的方便。

当然了其实断点调试还有其他的一些快捷键，还有一些其他的管理的功能，那么我就不挨个给大家介绍了，那么你首先把现在我所讲的这些个快捷键把它用熟了，然后其他的功能你慢慢去摸索，或者是你百度一下搜一搜，其实有很多这样的文章，你可以参考一下。

好，这是服务端的断点调试，那么客户端的断点调试其实就是调试GS，那么和差不多，只不过打断点的位置不是在这里，然后快捷键不一样而已，那么客户端的代码是运行在浏览器上，浏览器执行的是客户端的程序，那么我们想给客户端打断点在哪打，当然首先我们得有GS我们先看一看我们的源代码，看看有没有js文件可以让我们去试一下，其实是有的，我之前已经把所有的静态的文件放到了stirr之下，包括GS，然后咱们首页就有一个对应的js文件，这里有一个方法就public就首页上你可以点发布一点发布调的是这个方法，当然这个方法目前只是一个就是视觉效果上的呈现，然后它的实际的服务端逻辑没有，我们假设就要看一看，跟踪一下这段代码，给你演示一下断点怎么去打怎么去用。

好我需要打开浏览器在浏览器上去调，那么对于谷歌浏览器而言，你可以右键检查，然后打开了它的调试工具，Alex这里来说主要是调试页面html以及那样式的，主要是前端的工程师去用功能，控制台可以看到一些js输出的结果，然后 SASS可以看就是我们客户端的所执行的代码，那么当前我们执行的是index，首页首页它依赖了index的GS比如说我就在这儿我要调 public发布这个方法，我在这可以打个断点，也是点一下，这就是断点。

好，我给你演示一下，比如说我点我要发布，然后比如说我输入了一些内容，当然目前都是假的，然后点发布就调到这来了，你看就卡到这来了。然后它这个插件也是点这么几个按钮，然后实现就是一个调试的功能。

然后我直接说快捷键，因为这个按钮不好记，首先如果说你想线下执行一行，那就f10f10线下执行一行，然后比如说你想进入到某一个方法内部，那么就是f11，它是进入是f11，当然我GS太简短了，它就不太方便f11，而且你一进了到几块的底层的代码里，它这个代码还是经过压缩的，然后只有一行就很长，所以基本上就没法调了，所以基本上我们调GS进入到开发就底层的第三方的工具，内部去调的情况比较少，所以f11用的比较少。

然后如果你想说我调到这，我就想执行到下一个端点之处，或者是让程序执行到底，那你就按这个f8，f8是执行到底。

总之客户端调试和服务端调试逻辑是一样的，它的快捷键你记住向下一步f10进入f11，然后执行到底是f8，然后前端的调试它的断点的管理，就在这个界面上。这break points这里头它就列出来了，有两个断点你可以把勾去掉，让它暂时不可用，或者你可以点点中它把它delete，这个还不能点右键remove右键remove把它删了。

当然了，如果说你想看变量的值，我这里还没有变量，比较遗憾，再来一遍点发布。

那么如果你想看这个变量值的话，应该是我这里没有变量，应该是在这个位置，或者是你勾选上，比如说我想看这个值倒了括号，就是几块语法，选中了某一个节点，我想看到它的值，你可以选中它，然后右键 Add selected text to watch。

那么它就把你选中的数据在这个位置展现它的值在这里给你展现，你这里可以点开看它里面是什么值，当然这是js对象。几块对象这里我就不细讲了，这前端的内容。

好，我执行到底 F8。

好了，总之就是说你学会断点调试以后，当这个服务端或者是客户端遇到了问题，你不知道问题的所在，你就跟踪一下看到哪一块他出的错，或者到哪一块他走的地方不对了，不是你所预期的，你去重点看一看那个地方到底有什么问题再去仔细的看。

那么其实我们在用断点调试之前，其实还最好还要做一件事，你程序出错了，你最好看一下日志，看一看有没有报错信息，往往报错信息它很直观，能够反映出就是说你报错的位置在哪里，然后大概报的是个什么错，其实可以分析出来。如果你一开始学编程，你说我不会看这个错误信息，你可以把日志截出来，百度搜一下，可能别人也遇到这个问题，那么他会有一个反馈，有一个借鉴给你，那也是比较快的。

所以说其实我们看日志其实要优先于断点调试，但是它比断点调试要麻烦，所以说我把它放到后面来讲，那么咱们用的框架是spring boot，然后 spring boot它内置了它支持各种各样的记录日志的工具，然后它内置的内置默认启用的记录日志的工具叫look back。这个是它的 look back的官网，好，我们这节课就介绍一下 look back日志工具我们怎么去用，然后我们利用这个工具尝试打一些日志在控制台或者是在文件里看一看，然后以后的话你程序一旦出错，就多看一看日志，基本上能够发现很多问题。

好，我就看一下这个网址，我已经打提前打开了，这个是这样一个界面，然后那么我们去看看他的文档，你就点 documentation，好，然后点手册点进去以后，他这里有张的选择默认是第一张，然后你就第二张第三章等等，那么我们先看一下第二章有关的内容需要我们去了解一下，在这那么记录日志其实是一个很简单的工具，它就有一个日记录日志的组件，然后的话能够记录各种级别的日志。

么log back里记日志的组件是log这个接口，然后它有几个方法，打印日志的方法，trace打印一条消息，debug打印一条消息。In, for one era。然后这些方法其实它的作用是类似的，就是打印一些日志信息，你要把一个字符串传给他，他帮你打印出来，然后不同的方法名代表不同的日志级别，跟踪级别、调试级别，这是普通级别，警告级别，严重级别或者错误级别。

也就是说这几个方法它的严重性是越来越高了，或者说它的级别是越来越高了，这有什么用呢？

那么我们可以在配置文件里启用某个级别，比如说我启用infer级，可能你程序当中每一种级别的日志都有，但我配置文件里启用了infer级以后，那么只有in for one error会打印出来debug和确实会被忽视，明白这个意思吧？

就是说我启用infer级别的日志以后，那么infer级以及infer更高的级别的日志会被显示，低级别的日志就不要了，这就有一个好处，就是我们在开发的时候可以打印各种级别的日志，然后它都能出来，我们都能看到我们看到信息比较多，这样便于我们调试程序，然后我们上线以后，因为你把日志输出到文件里比较耗性能，那么上线以后一些无关紧要的日志就不需要了，像锤子第八位甚至盲法可能就不需要了，可能我们只需要一些记录一些警告，甚至是错误的消息就可以了。

所以说按照级别记日志，然后我们去动态的指定启用的级别，这是我们技术的一个关键。那么说了这么多，咱们还是去写点代码，写几个日志，打印几个日志看一下，一看就明白了。好我就回到我的程序里，GS我先不用了，不用的类我先关掉，这个也关了，然后我现在要打日志，我就写一个测试类，我在测试程序中去记日志，然后这个服务也先不用了，把它关掉让窗口大一点。

好那么我新建一个测试类，这个类我取名就叫做 log test，然后类和其他的类也是一样，把那几个注解引入进来，我copy一下好。

那么要记日志，首先我们需要实例化记录日志的组件，logo接口，好把它放大一下，我们就自己实例化就可以了，然后每一个类我们记录日志都单独为类实例化一个 logo，不同的用不同的logo，然后为了便于所有的方法去调用，通常我们把它设置为静态的，这样的话哪个地方都能用，然后 final不可改变，然后它的接口名叫老葛，你注意这个包它有很多包下都有这个接口，那么我们选的是 org点slf号接，然后变量名我就叫log。

那么室内化logo有一个工厂是 logo自带的叫logo，factor包下的，我们先做了包下来，然后点get log，然后我们传入一个类，这个类其实就是 logo的名字，你传入一个类，这个类名就是 logo的名字，那么通常我们都会以当前的类传进去作为它的名字，这样的话以不同的logo在不同的类项它就有个区别。

那么打印到日志上，我们也能知道 logo是属于哪个地方的logo。

好，那么有了 logo工具以后，我就测一下怎么利用它来记日志，我需要写一个测试方法，方法名我叫 test log，然后这样我先在控制台直接输出 log的名字。我们看一下logo点get name就是它的名字，这个只是看名字，这不是记日志，然后我就要用 Logo记日志了。

Log点debug当然比debug级别低的，还有一个确实跟踪机，基本上我们不会用它级别太低了，没必要，我们一般用的最低的级别就是debug，我们开发的过程中 debug有效，上线以后debug就取消，就是这样的。

然后那里边传个字符串随便写 debug log，好再来log点，Info，普通级别的日志 Info，log点word，警告级别，log最后log点错误级别 那么打印完资质以后，我们还需要在配图片里声明一下，我要启用什么级别的日志，这样的话 log back才知道我应该给你打印什么样的信息。

Look back其实它有自己的配置文件，但是经过死人不得的整合以后，我们也可以在 application practice里去配，这样我就在application practice里去配一下，这样比较方便。

打开配置文件，其实我之前已经配过了，你看咱们之前不是为了显示 circle，所以说我把整个我们的包下的代码的级别都指定为了debug，也就是说debug级别以上的日志都会打印出来。

好，我就不用改了，就这样。Log in点level其实就指的是日志的级别，后面跟着的包表示说这个包下的级别是怎么样的。好，那么我就回到测试方法当中执行一下是吧？好。我们看一下打印的结果，首先你看 log name，其实就是一个类名对吧？因为你传的就是类名，它是以类的名字，logo的名字，然后你看日志的信息，那么debug也输出了，因数也有了，望也有了，艾瑞也有了，就4个级别都有了，因为它们都大于等于debug级别，它的格式是这样的。

好，我改一下，比如说我把 debug改成我然后我再去执行这个测试方法，你看这回就只打印了两个日志，一个是one，一个是I对吧？就是用起来非常简单，那么以后我们在写代码的时候怎么去做？

其实我们写程序的时候比较常用的这三个级别，一个是air，比如说我们端到端开始捕获到了一个异常以后，你肯定是要记录一个错误级别的日志，无可置疑的对吧？

然后比如说以后我们可能会比如说启用一些线程池，启用一些定时任务，处理一些特殊的任务，这个时候可能我担心任务执行的过程中会有问题，我可能会打一个普通级别的日志，就是正常的记录一下，这个也有用。

然后平时我们在开发的过程中，比如这个只是我暂时调试程序要用以后就不需要了，你就debug所以基本上平时我们用的就是debug Info，l one用的也相对来说少一点。

当前我是把日志打印到控制台，默认它就是打印到控制台，那么如果我们系统上线以后打印到控制台就没有意义了，因为你的软件最终是部署在服务器上，服务器的话，它就不会安装开发工具idea，他直接就安装一个应用服务器，你就没有控制台，所以说而且打印的控制台它最终不会留存下来，只要你应用程序一关闭就没了，所以说为了能够长久地保存这些日的消息，便于我们日后的一个追踪，甚至是对日志的一个分析，我们经常也需要把日志存到一个指定的文件里，下面给大家演示一下我们怎么把日志打到这个文件里，那么简单的方式就是我们还是在 practice里加以声明，这样我把它改成第八个，然后多一点，这样我们打印到文件里，我好看得直观一点。

好，我们可以做这样一个声明，不是 log in点 five，等于那意思是说我日志的文件要存到哪里，后面就跟一个路径跟一个文件名就可以了。比如说我要把它存到这个地盘，然后 work的下面，然后我希望在work之下新建一个名为data的目录，在data下新建一个名为now coder的目录，然后在这里面创建一个community点log，把日志打印到这个文件里，通常的日志文件都以log结尾。

好，你看我现在 d盘应该是没有 work，没有好我就再执行一下这个测试方法，执行完以后控制台也会打印出这个消息，空还是一直都打的。

我们再看一下 d盘work就多了一个目录，data刚刚创建的，你看这个时间刚刚创建的，然后对着下面然后扣点，然后到下游肯定对 log，然后你把它打开。这就是刚刚记录的一些日志的消息，包括第八个in for more air都有，然后程序执行完以后，他就把程序杀掉了，就是杀听到对吧？是整个一个程序执行的过程都能看得出来。

好，那么这种机制的方式感应到文件中的方式比较简单，其实我们在实际的应用当中还会比这个做得更复杂一点，日志很重要，你这样简单的打到一个log文件里，这个文件可能会非常的大，而且文件里会混杂着各种级别的日志，其实不方便我们将来去做分析的。

我们在实际的开发的时候，往往是把日志按照不同的级别存到不同的文件里，infer就存到infer的文件里，one就存在one的文件里，l就存在l的文件里，那这样比如说我只想分析错误，我就看l的log，然后我只想那些调度相关的信息，我就看Info的log，以此类推，这样会更合理一点，而且最好是这个文件不一直往一个文件里存，当这个文件比如说达到5兆以后，我们让他再拆分出一个另外一个文件，就5兆以后就再分一个文件，5兆以后就再分一个文件，总之我们实际开发的时候会更复杂一点，我给大家演示一下怎么办？

那么通过 application practice就不方便做这样的配置，因为这样你想达到这样的效果，配置会非常的复杂，这样我把注掉这两种方式演示一下就完了，我们就暂时先不用了，那么如果你想达到刚才我所说的那种效果怎么办？

我们需要写一个log back所对应的 xml文件，这个文件在look back官网上每一张它有一个讲解配置的一张，第三张，讲解配置的一张你可以看，那么课上我就不给大家去看了，因为比较长，而且这个文件我们即便是看去手写也非常的麻烦，我提前已经写好一份就是我们平时用的东西，然后的话我把这个东西拿过来，引用到我们项目中来，我给大家讲解一遍，你能明白其中的意思，然后会改就可以了，这些东西不用把它背下来。

好，那么我先找到我提前准备好的文件，好像在这儿了，那么这个文件一般 springboard的建议叫lucke，bak杠是bring的xml，只要你取这个名字，并且把这个文件是放到了resource根路径下，就把它放到resource之下，那么sprint就自动的能够启用这个文件里的配置，你就不用说声明说我要使用这个文件，它自动就会发现，但是如果你取别的名字它就发现不了，他就认为不存在。



所以你要注意好，我们打开 loc的xml，咱们看一下，好，你看这个文件一开始是声明了contact name其实就是项目的应用名，这里叫community不用改，我们这个项目就叫community对吧？然后有两个属性，其实有两个变量，然后一个是log pass，就日志文件存放的目录，我这里原来写的是c盘，我把它改成d盘，然后你拿到这个文件以后，根据自己的情况去改，我就放到d盘work data之下，然后这个是appdirapp的目录，我叫community，就是说我希望把日志文件放到目录下，然后再建一个这级目录放到这个目录下，因为什么？

我们可能一个服务器上会部署，有可能会部署多个项目，那么这个目录下可能会放community，它可能会放其他的项目的日志，那么通过项目里面加以区分比较合理。

然后后面的话，air file就是说配置我什么样的日志要存到air file里，什么样的日志要存到one file里，什么样的日志要存到iPhone file里，其实如果你了解一块，埃尔菲其他的恋爱环境都了解了，因为他们的配置方式都一样，就是级别不同而已。

好，首先你看这个地方配的是具体的日志文件哪，应该是在目录下，apptir下就这个目录加上它，然后文件名叫log l点，log只是这个文件名。

然后后面还有一个叫rolling就是滚动的策略，就是说如果你这个文件存到了一定程度，比如说5兆存不下了，或者说我希望你在再建一个新的文件来存，建新的文件的策略是什么？

然后这里写了你看file pass pattern，创建新文件出在哪里也是log pass appdir然后新建一个l文件夹，文件夹下的文件叫log error百分号d然后yy杠杠DD，yy杠mm杠DD其实它是一个日期的格式，百分号d是一个日期，这是一个日期的格式年月日，然后在后面百分2l是一个变量0123，所以最终的话拆分出来的日志的文件，它是这样的一个这样一个名字log error，然后年月日，然后一个序列号，然后点log，然后这里写了最大的文件大小是5兆，也就是说那一个文件存到了5兆存不下了怎么办？

再建一个新的序号再累加一下。

然后max history就最长存多长时间？30天，当超过30天以后，那么把超过的那些个历史的数据就清了，不会一直存着，因为服务器的存储空间有限，没有必要永久的存那么多的文件，判的等于处，意思我是以追加的形式存，日志不是覆盖是追加，这是合理的，然后这个是日志的一个格式，其实就是我们控台看到的日志输出的格式，然后百分号d是日期，level就是日级别，spread是哪个线程在执行Logo日志的一个所处的，它是在哪个类中去执行的？

去打了日志，然后后面是哪个具体来说哪个文件多少行，还有详细的提示消息，详细的内容，后面的换行大概是这个意思。

然后字数集有点支持中文这个是一个过滤器，就是说什么样的日志会被我组件去处理？他说era级别的日志，那么on，match如果匹配到了era级别的日志，我就允许去记录，如果miss match就不匹配，我就带你拒绝就不记录。就总而言之组件只会筛选出l级别的日加以记录，然后记录到默认是这个文件里，如果存不下就拆分到这样的文件里，是这么一种感觉。

同理Oneself就存到了乱word这样的文件里，Info就存到了log in for这样的文件里，以此类推，然后还有一个与众不同的是看守，这个比较简短，就是说它是把日志打印到控制台，便于我们平时调试，方便打印到控制台格式是这样的，其实跟上面一样，编码是这样的，然后控制台允许打印第八个级别的日志，往上比较简单，后面还有我单独声明说，我们肯定得包下的日级别为第八个这卡夫卡我们还没有讲到就先去掉。

然后后面还有一个声明root，表示我们整个项目所有的包的根目录，默认的级别是音符，然后因为什么？我们根目录包含其实包含所有的炸包里面的类，那么你这个地方不能debug，否则日刷太多了，所以说最低级别是英法，然后这里边又启用了引用了刚才的那几个组件，Fire arrow就是组件叫fire arrow，赢的是他，比方说我要启用这个fire组件，记录air级别日志，我要启用万组件记录万级别日志，我要启用一份组件记录一份级别日志，我要请组件在控制台打印日志，就这个意思。

好了，总之这个文件就是配好了，你就拿来用就可以了。然后基本上需要改动的地方，头上的路径，如果你开发项目的时候，这个包名跟我不一样，你包名也要改一改，其他的地方基本上就不用动，这个格式也是比较合理的。然后还有这个文件的大小5兆，如果你觉得不合适你也可以改，但其实也是没有必要的，这样就可以了。好，那么现在这个文件已经有了，下面我再执行这个测试类，我们看一看那个文件会不会有效。

好，我执行一下这个测试方法，首先看控制台，你看这回打印的日志，原来没有这个文件之前有点区别，它变得更简短了，因为我在配置文件里对日的格式进行了重新的配置，它和默认的方式不一样，变得更简短一些，更直观一些。

然后我们看一下 d 盘 work data，community 你看就多了三个文件，分别是 infer l 然后没有问题，有人可能会想你怎么没有 log in for，不是第八个级别的日志。因为我们配置文件里不是只起只是声明了，我要在文件里存 l1 以及 Info 级别的日志。第八个级别的日志我不在文件里存，我是打印到控制台了，这个表示控制台，所以控制台能看到第八个级别的日志，对吧？

所以这个逻辑是没有问题的，调试的时候再过来看第八个，然后上线以后这三个级别的日志就都有了，好，这样的话我就把项目相关的调试技巧给大家介绍到位了，至于说怎么运用比较舒服比较方便，另外的话运用的时候那种感觉怎么能找到，这需要你在开发的过程中不断的练，才能慢慢的去锻炼出这种感觉和这种能力好。

我再大概的回顾一下我们这次课的内容，就是当你在开发这个程序的时候，如果你发现问题，首先应该怎么办？首先你要看一下状态码，大概的定位一下说是我客户端程序有问题，还是我服务端程序有问题，我看这个状态码能看得出来。当然有人说我状态码上哪看呢？如果报错的话，这个页面上就会显示出来 404 500 就在这就能看到。

其实不用到哪个地方特殊去看，就在页面上就有，然后的话判定了状态码大概给我提示的是客户端还是服务端有问题以后，我再去或者是有通过断点去跟踪或者是看一看日志，当然能看日志，找到问题就看日志，如果看日志还搞不定，跟踪一下，跟踪到位了，应该是可以搞得定的好。了，那么这次课我们就先讲到这，咱们下次课再见。

## 1.8 版本控制

---

8-版本控制.mp4

说话人1 00:03

这次课我们来介绍版本控制，那么版本控制的作用先说一下，第一个我们通过版本控制可以对代码进行一个备份，万一说你的电脑的硬盘坏了对吧？有备份的话还可以恢复，不至于把代码丢掉。第二个那版本控制它是会将你的代码每一次变更都记录下来，有一个历史记录，那么万一说你这一版的代码改坏了，还不如上一版，你可以通过历史记录恢复到上一个版本。

然后再一个我们通过版本控制工具，还可以让团队成员之间互相协作，比如说我把代码上传到版本控制服务器，那么你可以下载到我的代码去看去改，同理你上传我也可以下载，所以团队成员之间是通过版本控制工具去共享各自的代码的。

好，所以说我们基本上上班的第一天就是你把jdk安装好，集成开发工具安装好，然后一定也要把妹妹或者是版本控制工具都安装好，基本上这是你第一天上班就要做的事情，所以说我们在第一章就对它进行讲解。

而且我们现在是开发完了一个功能，这个代码你是时候去上传去备份了，那么以后每开发一点代码你都上传一下备份一下，这样的话对这个代码的安全性有一个保障。

那么版本控制工具是有很多的，早先有什么cvs然后有Svn等等。目前的话比较流行的是用git，我们课堂上给大家讲git怎么去安装，怎么去配置，然后怎么去使用。基本上我们讲的内容分为三个方面，第一个先给你介绍一下这个git，然后我们把git安装好。

第二个就是说给大家讲一下 key的常用命令有哪些，我们通过命令行，然后去将代码提交到本地仓库，然后再将代码上传到远程仓库，把地的它最核心的命令给你演示到位。

最后再演示一下在idea当中我怎么去集成date，我怎么利用idea去操作date。

当然我们最终在以后做项目时往往是用第三种方式，可能第二种方式用的少，但是你了解这些命令，也就能明白那idea他帮你做了什么事，或者说 idea当中我们每一步操作其实操作的是是什么，它背后执行了哪些命令，然后 git它的官网是这个网址，我们可以去看一下，我已经提前打开了，咱们看一下，就是这样一个界面，然后那么如果你想下载的话点 download，一会我们再下载，然后如果你想去看一下这个手册，就点 documentation，这里面 reference是它官方的手册，然后 book是它官方的推荐那本书，我倒是建议大家去看这本书，因为那个手册它那个东西比较罗列的比较详细，看起来比较繁琐，而这个书看起来比较有层次，关键这个书写的确实挺好的，而且它还有中文版，你在这儿你可以选简体中文，所以说我们阅读起来会非常的方便，你看看这个书就好了。

那么这个书其实你要看的话，首先看第一章，看看这个版本控制的一个介绍，然后怎么去安装，这是我们需要初学时要看的。再一个重点是看第二章，第二章主要是讲 Get的基本的命令，那么其他的你有精力看一看更好，如果没有精力的话，看这些反正也能用起来。

好。首先我们先看一下关于版本控制工具，其实关于版本控制早先的话是采用中央服务器的方式去管理代码，就是说我们所有人把代码都存到一个版本控制的中央的服务器，它这里有一个123各种版本代码，不断的升级不断的维护。

那么 ab不同的成员，不同的开发者都需要从服务器上去下载代码，然后在本地修改再上传回来，他再下载，以此类推。

好，这种情况就有一个问题，如果说这个服务器他的硬盘坏了，可能就比较糟糕了，那就会对这个代码造成比较大的影响，整个历史版本的信息就都丢掉了，这是传统的以前的方式的问题。

那么 git它采用一个比较新的方式叫分布式版本控制系统，那么它也有一个中央服务器，这里面也有各个版本的代码的存储，然后我们ab每一个成员也是从这去下载代码，然后把改好的代码传回去，也是这样的。

但是我们 ab每一个成员每一台个人电脑上也会存一个一每个人也会存一套代码，或者说我们每个人的主机上也有一个仓库，服务器，我们叫远程仓库，我们本机上存的代码我们叫本地仓库，我们我们的代码是先存到本地仓库，然后觉得没有问题了，我们再统一的推送到远程仓库上去，这叫分布式部署。

那么这样的话就是说你服务器上和每一个成员的电脑上都有一套版本控制的代码，万一说哪天这个服务器，那么每个人的电脑上还有这么一套备份，你去询问一下你的成员谁的代码是比较新的，我们利用这套代码去恢复就可以了，这样比较安全。

然后还有一个好处是我们在没有网络的环境下也可以去往库里提交代码，因为我先提交的是到我本地的库，我先提交到我本地的仓库，然后有网络的时候再推送到远程仓库，所以这样的话它的效率也比较高，速度也比较快。

好，这是 git它的一个特点，它是一个分布式的版本控制系统，设计里面就比较先进，然后如果要安装的话，我们可以看1.5，这一节他这里介绍了在Linux上怎么安装，MAC上怎么安装，然后以及在windows上怎么安装，在windows上你可以点这个链接去下载，或者说在刚才我们官网首页上也有一个download的地方去下载都是一样的，比如说我点字儿，然后它就会建立连接，然后开始下载64位的，其实我已经下载好了，这里我就取消了。

我们来看一下我下载好的程序，接下来我就把它安装一下，安装其实没有什么特殊需要说明的，我们就下一步你放到这其实也行，我还是把它放到d盘 d盘下面然后下一步默认就可以，下一步不用动，下一步默认就可以，下一步一路好，总之改一下路径，其他的地方一路默认就可以安装好好。

我的git已经安装好了，我来测一下，安装好以后你可以在 windows里去找到它， git，然后它有几个图标，一个是git bus，其实这个就是它模拟了一个 Linux环境，你点开以后它是一个命令行工具，这里面写的是 Linux系统下的命令，不是 windows，所以如果你熟悉 Linux命令的话，你可以在这里去用。

然后除此以外，还有 get come on windows的命令行工具，还有g的gui是一个可视化的界面，但他做的也不好看，然后的话我也没用过，因为我一般都会用这个命令，一般我们用这种工具都是用命令行，不会用这种很少用这种gui的工具。

因为我的电脑是windows的，所以我就用这个我就不用 Linux命令行工具了，好，那么我打开了以后，我就随便哪输入一个这个命令，然后看一下不起作用，测试一下get，然后看一下它的版本。Git version，它打印出来的版本是2.21.0，没问题，这就可以了。

好，那么在此基础上我再给大家演示一下，Get它的常用的命令有哪些，怎么去用。首先我先演示一下，我在本地仓库里怎么去使用 Git去管理我的项目中的代码，然后再演示远程的，然后再演示idea怎么去用好，我先演示本地的，那么如果你想用这个git不是装好以后马上就能生效起作用的，因为你是要把代码存到仓库里去，它得有一个标识得能识别你是谁，所以你首先需要对 git进行一个配置，你要告诉 git我的用户名是谁，我的邮箱是谁，这两个都需要进行配。

好，我就配一下，首先我们敲这个命令it凭杠杠list，就是我看一下，当前它默认已有的配置有这么多，然后这里面现在没有用户名，也没有邮箱的配置，我们需要配的是用户名和邮箱，我就配了。配用户名can fake，然后杠杠个logo全局的用户名，然后悠着点name，然后后面跟着一个字符串，你随便写我这里写成就是我的名字的拼音李洪贺，回车这就可以了。

然后再配一下我的邮箱 get挺杠杠，格罗伯也是全局的，悠着点。

Email，然后再写上我的邮箱，那么我在牛客网的邮箱是李红鹤艾特到com。好我就用这个了，然后回车也就ok了，没报错就行了，然后我们检查一下配置里会不会多这两项，还是用刚才的beat挺杠杠。List好，你看它下面多了两项，一个是u的name李鸿鹤，一个是EMAIL，李鸿鹤脑壳的到com，没有问题。好，配置完以后，我现在就可以利用 git去管理我的代码了。比如说我现在有一套代码，我需要把它传到这个仓库里去，存到本地仓库里先怎么去做好，我先用它管理哪套代码，我看一下。

我先随便找一个测试，咱们之前不是用面板工具创建过面板demo一吗？我就先管理一下，那么你要管理哪个代码，你需要先 CD，这个项目 CD到目录下才好，对它进行管理好。我就CD一样，我先去d盘，然后CD d盘 d盘work。Work space。没问demo一。先这儿，到这儿以后我们需要先做一个初始化，比方说我这个项目需要交给git这个工具去管理，那么get需要初始化一些数据，你这样写get you need很快就完成了。其实这件命令会导致get工具会在文件夹下创建一个隐含的目录，叫get这里面有一些答案，需要用到的数据隐藏，我们也不需要去看你，了解这一步他干了什么就可以了。

这里也做了一个解释，然后我们可以看一下当前我 Git管理的这套代码它的状态，hit status不用细看一看，他说泡沫的xmlsrc他盖的这几个文件和目录都是红的，就表示说我们还没有把这些个文件和目录添加到库里，还不受管理，还不受库的跟踪。首先我们需要把他们添加到我们的本地仓库中去，怎么添加呢？也非常容易get ade。比如你要添加所有的xml文件，就新的xml，所有的Java文件就新招，如果是你一股脑什么都添加，直接就星。好，然后爱的星以后，我们再去看这个状态，get status。他会列举了你看这么多绿的，绿的就表示什么，这些文件已经被添加到了本地仓库中去，只是加进去了，只是临时存在，但是还没有正式的提交，所以你还要提交以后才能够永久的在仓库里进行一个存储。

怎么提交呢？是这样的。Get to meet杠杠杠m杠杠杠m的意思是我给这次提交写一个备注。好，知道我这次提交了什么，这里我就测试随便写了，就test一回车，然后他就开始提交了。很快速度很快本机操作，然后我们再看状态，他说nothing to me的就没有了，这些刚才的那些文件都已经被提交过了，没有新的文件需要提交了这样一种感觉。

好，比如说我去改一下那个代码，看看会怎么样，我随便比如说改 Test，我就随便改一下记事本，比如说我就在这写一个一注释一，好，改完以后我们再看这个状态，你看它提示这个文件是modify的，是被修改了，但本次修改的内容是红色，比方说本次修改的内容还没有提交到仓库中去，还没有加到仓库中去，我需要把修改的代码添加到仓库中去， get还是爱的。行。

添加去以后我再看状态变绿了，比方说添加进去了，但没提交，我再提交一下，写个备注也就是test二。好，然后再看状态，没有东西可以提交了，比如说我都提交上去了。好，那么总之我们在本地仓库就是管理这个代码其实非常方便，就那么几个命令，盈利的一下，然后实力的看状态，爱的就添加，肯定的就提交，就这么简单。

好，这是在本地仓库，我们去管理代码，我们最终的目的一定是想把本地的代码，本地已经提交过的代码，不是临时的传到远程仓库上去，那么远程仓库当然你到工作单位可能公司他自己会搭一台服务器，搭一个远程仓库，那么也有的公司可能直接用 Get up第三方的，现在很多人都听过get up，你也可以把代码传到get up上，然后你像我们也可以把这个代码传到牛客的 git仓库中去，咱们牛客网也是有公共的git仓库的，只要你有游客账号默认就能用也很方便。

所以我在课堂上给你演示的是，我怎么把代码传到牛客的远程账户上去，要想把这个代码传到远程仓库上去，首先你要注意 git为了传代码安全，避免这个代码被偷。那么它在传输的过程中需要采用 Ssl安全连接安全的方式去传输，所以说它首先需要我们去创建一个密钥，需要配置一下密钥，这样的话才可以启用这种安全的传输方式。

那么怎么去创建密钥也是很简单，就一个命令我们来执行一下，在这直接敲了 ssh杠提给然后空格杠t当然这个命令你在手册里能找得到，我就直接写了。

然后是rsa杠c后面的话要写上你的你的邮箱，就是你的 git账号的邮箱，not code.com好回车，然后在生成的过程中，它会有一些询问说你要把 key生成的密钥存到哪里去，怎么样，其实我们都可以一路默认，你就什么也不用填，直接回车，然后直接回车，直接回车就完了，看到这样的界面就表示生成完毕，好。

我们去看一下它生成到哪了，其实它这有提示它生成到了这个地方是这个文件，这个文件好，我们打开看一下，是c盘，然后 users用户automatic data，点ssh这个文件我需要把它打开，用技术把它打开，然后把整个一长串密钥复制一下。

好，复制完以后，我们需要去远程仓库的外部界面上去进行一个配置，把我的密钥添加到远程仓库上去，这样远程仓库才能接受你传的代码。

好，我说了我要选择是牛客网的远程仓库，但是如果你用其他的远程仓库也是一样的，使用方式牛客网我已经打开了，那么我们在这儿点个人主页，然后左侧有一个项目的目录在这儿配，然后这儿有get账号的设置，那么这个地方你要注意你的牛客的账号需要绑定邮箱，而且需要绑定和你 git配置的一样的邮箱。

这个要注意，你不能说我给他配的是一个邮箱，这是另外的一个邮箱，这俩不一样不行他俩得一样，你注意，所以如果说你的邮箱没绑定你，然后先绑定一下邮箱，然后就可以用了，然后在下面去添加你的密钥我已经添加过了，因为我是之前用我的笔记本往上往上传过代码，然后的话笔记本它的密钥是我现在讲课用的是一个台式机，我可以再添加一个台式机的密钥，可以用台式机也往上传，你要添加密钥就点添加ssht就可以了。

然后这里把刚才复制的这一大段拷过来就行了，上面你给密钥取个名字，这就是我的PC机就PC确定这就行了。

那么配完以后我要往这上传代码，首先我们得自己建一个仓库，或者说我们建一个项目，然后我的代码要传到这个项目里，我就在这儿新建一个项目，当然这个是我之前传的一会也有用，我再新建一个。

那么这个我就叫没问demo一，因为刚才我不是在本地提交了，没问demo一，我就希望把没问demo一传到远程仓库里来，所以我创建的名字和它保持一致，然后这里可以是选择是公开的还是私有的，我就公开，然后创建好远程的项目就建好了，当然里面什么都没有，你可以点进去看，什么都没有。

然后这是它的分支，它可以有多个分支，默认它就只有一个master，主分支我们就是要往分支里传。

代码目前什么都没有，后面的话它有一些提示说让你去做配置要怎么办？

什么命令，然后你的邮箱要怎么配，你创建仓库的命令要怎么去敲等等，其实也有一些提示你也可以参考好，我就不给你细看了，我就直接演示，你要注意我们创建好这个项目以后，我们要访问这个项目的他的访问的路径，或者说他的 git仓库的访问地址是什么，其实就是当前的路径，后面加上点git，其实后面也有，你看他传的时候就是当前的路径点git就这样。

好，我把拷一下，把它拷一下，这是当前我建的项目的仓库的远程的路径，我要传的话得依赖这个路径好。

我再回到命令行工具，那么首先我需要在本地先声明一下远程仓库，或者说我先关联一下远程仓库，主要是给远程仓库，给刚才的路径取个别名，以后我再去访问远程仓库的话，用别名，而不是每次都敲这个网址，这个不好记。

好写命令 `get remote`远程操作，添加一个远程仓库取个名字，一般它默认名字叫orange，我就按照这个来取名，然后后面把刚才的路径靠过来，别忘了后面带上 git，回车这就建好了远程仓库，在本地声明的远程仓库，它的名字叫rng然后现在我要把明文demo一中的代码传到这里来，那怎么传非常容易 `get`，`post`就推送杠u然后 orange比如说我要远程去传到他的仓库里哪个分支，master目前就一个master，Twitter回车以后，它让你输入账号密码，其实就是你牛客网的账号密码，但这个账号要输的是邮箱，密码是你牛客网的，就是你输入你注册的牛客网的邮箱和密码就对了，好我输入一下，输入完以后确定以后他就开始传了，因为这个代码比较小，很快就传完了，传完以后咱们可以去看一下，我再回到优客网，刷新一下你看没弹幕一下，就有了 src target他们的xml比如说我去点 src test，我要进去看，而且你看它这个版本也有，test一test二，它的历史版本都是有的，然后我进去看一下这个代码，对不对？

没错，这个注释一不是我刚才加的，对吧？好了，那么如何把本地的代码传到远程登录上去，就是这样就很方便。

然后我们还有一种需求，比如说我远程仓库上本来就有个项目，他我想把它下载到我本地，我好去看，我好去学习，或者我好去在这基础上去改怎么办？你可以点这个项目，先去查看一下这个项目的路径。好，其实你点点开以后，它这有一个克隆或下载，其实克隆就是我要把这个项目克隆复制到本地仓库中去，你点一下它就会给你一个提示一个路径。

这路径有两种形式，一种是https的格式，一种是ssh的格式，用哪个都行，我就用这个复制。

好路径以后，我就要从这个路径 copy，就是克隆代码到我本地仓库，那么我们再回到密码工具，你想把代码放到还得CD到哪个位置，我看一下我放到哪合适 Work，我就放到work space之下，因为这里边没有叫spring boot demo的项目，我就放到这里来，所以我就CD点点回到work space这集目录下，我要在这里，然后去克隆远程项目，那就是这样写 `get`可弄，然后把刚才的路径复制到这个地方，然后回车，他就开始克隆了，很快就下载完了，完了以后你看一下本地就有了这个项目 spring boot demo就有了，而且不只是有项目，其实它也包括，它也包括其实给对它的历史的管理的一些信息都在 `get`目录下都隐藏着。

所以说这个项目不只是我们看到这个项目代码本身它的历史的版本都有好了，那么命令行常用的命令我就演示到这里。

当然git其实还有很多的命令，我不挨个演示了，然后的话其他的你可以自己来查阅一下手册。另外的话如果你喜欢看视频的话，牛客网还课程里面其实有一个专门讲地的课，而且是免费的，你也可以去看。那么最后一步我想利用 Idea去管理git，我们在idea当中把git配置好，利用idea去操作`get`去管理我们相关的代码，这样就不用敲命令了，那么我们写完代码以后点一个按钮，那么很容易就传上去了，这样比较方便。

好，这样我就打开我的这个idea，然后比如说我就要管理我肯定的项目，我要把它传到远程仓库上去看怎么办？首先我们需要配置一下`get`，因为这个idea他不知道你给他装在哪，你要告诉他你可以点 file，settings设置，然后在这working control版本控制工具这里边选，然后这个地方主要是你要选择一下你`get`的安装程序在哪里，点一下这个图标，我是装在了d盘work get下，那么你要选 b目录下的git点exe好。

Ok, 然后这样就行了。

怎么好像没选中再来一遍, 怎么刚才好像没选中似的。

Get be git点exe好, 然后选中以后 apply ok好, 这是beat的配好了, 配好以后我要管理这个项目, 首先是要它把这个项目初始化一下, 然后要把这个项目添加到本地仓库中去, 然后再推送到远程仓库中去, 和我刚才的命令行操作是如出一辙的, 是一样的。

只不过这回不敲命令是通过点按钮的方式去做好, 我要操作的是整个项目, 传的是整个项目, 那么选中这个项目的根目录, 然后 vcs, 版本控制工具版本控制系统, 然后 import into working control。我要把它导入到哪个控制工具中去, 这里我选择创建get仓库, 其实就是初始化在本地。

好点中, 点中以后他让你选中这个项目项目目录, 那就是一个community没有问题。Ok, 然后很快你会看见这里的文件变红了, 那就意思是说他已经把项目给我初始化好了, 而且他还做了一个添加的动作, 爱人帮我做了。你展开看, 你看这里都变红了, 说明了这个文件已经被添加到本地仓库中去了。我们在本地的路径下看, community get隐含的目录也有了, 说明它确实做了初始化, 做了添加的行为。

好, 那么我要把这个代码提交到本地仓库怎么办? 还是点选中项8点cvs, 可以直接点to meet或者是你点get。

下面的to meet也行。我就点这个to meet, 然后它弹出一个对话框, 这里让我可以选择我要提交的代码, 你可以有选择的, 提交一些代码, 一般像 maven自带的一些文件我们可以不去提交, 因为那个文件提交上去也没有用, 我们主要是要提交 Java, 还有一些配置文件, 像腾飞的目录下, 其实这个目录下就都要, 然后 resource之下的文件也都要, test之下也要Tom点xml也要, 其他的, 其实这种文件就不需要了, 好。

然后但是这儿我这窗口有点小, 然后这儿还要写一个提交的备注, 说首次提交第一次提交代码, 然后点to meet。好, 我要确认就是to meet再确认。它有一些提示你不用管you can meet就可以了。然后你发现提我们选中的提交过的代码变白了, 就表示他已经入库了, 其他的这些文件不需要不用管了。

好, 就这样。那么要想把它提交到远程仓库, 首先我还得在远程仓库上建一个项目, 我就回到牛客网, 然后回到这个界面, 然后我新建一个项目叫community, 也是公开创建, 建好以后我还得知道他的路径, 它路径其实就是一个网址, 加上点get, 或者是你从下面去找, 就这个好。

我再回到这个idea。那么我要把这个代码提交到远程仓库, 还是点 Vcs然后get然后是pose, pose是向仓库推送数据, 如果仓库中的远程仓库中已经有了别人更新的数据, 你可以破把它拉取到本地。

我这里就演示pose, 因为没有人去帮我改代码, 我只能pose好点pose, 点POS以后这里需要先定义一下远程仓库, 就像刚才我们敲命令, 执行 remote命令是类似的, 定义一个远程仓库, 名字叫orange, 默认的URL拷过来, 然后 ok那么ok完以后, 右侧它列出了, 那么我们当前本地仓库已经提交的代码, 我们就是要把这些代码提交到远程仓库对吧?

我就点 pose在提交的过程中, 它要求你输入你的 get的账号密码, 远程仓库的账号密码, 牛客网的账号密码, 账号是邮箱密码, 就输入你的密码, 然后记录然后落地, 这个时候再连上以后就开始存了, 最后的提示说成功了。

好, 我们再打开远程仓库刷新一下, 你看这个代码就有了。Apart的xml sa si SC下may may下有这个 Java calm Lowder community等等这样的一些代码。好了, 我们就已经把我们的代码上传到了远程仓库做了一个备份, 当然你也可以通过这种方式和别人去分享你的代码。好了, 这次课我们就是把 Git安装好也用起来了, 希望课后大家把它装好, 然后把你现在的开发好的代码都提交到游客的 get仓库中去, 把代码做一个备份, 好吧? 这次课我们就讲到这里, 咱们下次课再见。

# 第2章 Spring Boot实践，开发社区登录模块

## 2.1 开发注册功能(续)

10-开发注册功能(续).mp4

那么这个name也就写完了，写完以后还有问题，就是说如果说我这么一提交数据提交给服务器，服务器注册成功了，它跳转到了首页，那就和这个页面无关了，这都皆大欢喜。但是如果说我这个数据有问题，提交给服务器注册没有成功，它不是还会回到这个页面，而回到这个页面的时候，不是给我们携带了一些错误消息对吧？所以我在错误页面上还得把错误消息做一个处理。

好我们一点一点处理，有点繁琐，别着急。首先首先如果你回到错误页面的时候，那么肯定是希望 User、name、password这些值还在对吧？而不是说回来以后这个值就空了，它还在，所以我们回来的时候就要给它填一个默认值，框的默认值是通过value来进行设定的，每一个框它现在比较长，我做一个折行，我先把这折一行，然后默认值我在这写 thy0默认值肯定是个变量，等于。

好，那么默认值我们从哪取？你看这个方法，当我们跳回页面的时候，这个出页面是可以直接访问 user的，为什么？因为 User它已经存在于model里，在spring mvc在调这个方法的时候，他就做了这样的处理，既然又在model，我们可以在页面上可以直接访问到它，对吧？所以我这可以刚才那个地方默认值我可以写成什么？就写成 User点user name。

好，但是这样写是不严谨的，因为如果说我是不是通过错误发送返回这个页面，我是直接点注册功能，跳到这个页面的时候，user就没有，对吧？我直接仿这个页面图，user没有，你这样写不就控制异常了吗？所以你还得判断如果user为空的时候，你就什么也别显示，如果不为空的时候我再把它显示出来。所以这块也就是说 register是在两种场情景下复用的一个模板，所以你要考虑到兼容性。

好，因此这个地方我得这样讲，user对象不等于now，在这个时候我就在这里外流表显示悠着点儿，悠着内对吧？如果它等于now我就什么也不显示是这样的，同理密码也是这么处理就可以了。

我这个代码拷过来逻辑是一样的，这就是 user点pass word，User不等于now，那么显示的是user点pass word。

好，那么确认密码其实也是一样的，因为确认密码和密码一定是一样的值对吧？当你能提交的时候说明密码和确认密码一致了，那回如果说错了回到这个页面，那他俩也是一致的，所以确认密码也是这么处理，我把刚才这个代码贴过来，那就显示密码就可以了，邮箱也是类似的逻辑，只不过这显示的是悠着点，EMAIL。

好，那么回到错误页面默认值的显示处理完了，那么最后还有一个逻辑就是说你这个错误信息要显示在这里，但这里我们目前是写死的一个消息，我们应该把它改成一个变量，所以你应该在 div上写上th，冒号，text等于某个变量，等于谁？就是我们control当中传的这些个错误的消息对吧？这个账号的问题你应该写user name，message。

当然你首次访问这个页面的时候，如果 Message是空的，它就显示不出来，因为它没有下级的调用，所以它不会出现控制异常的问题。那么同理，如果密码有问题，就是整密码上显示那个消息，然后确认密码不需要 pass是不是写错了pass word，password，message确认密码不需要，然后邮箱需要，邮箱是 EMAIL message。

好，然后还要注意一个小的细节，就是错误的消息默认的它是不会显示的。

那么什么时候它会显示就需要，因为我们这个页面是通过 but stop做的。那么它是有一个连带的关系，我们看一下什么连带关系就这个错误消息是否展现，它基于 input框上面的样式，class，那么如果 Class里带上了样式 is invalid。

如果说框里的值表示框里的值是非法的，那么 div就会显示出来，如果没有样式它就不显示出来，有没有样式得动态的判断，对吧？这是一个动态的情况，不能写死。



所以 Class应该是一个动态的，我们要传一个动态的值给他，所以也需要用 tap来处理。其中 form、control、样式一直有的，然后后面还要拼上一个动可能配上一个动态的值，对吧？所以你要用两个竖线表示说这是一个固定的值，后面可能带着一个动态的带一个动态的值，我就可以这样写了，打了括号对吧？好。

是否拼一直歪着的，你看一下就是 user message它是不是空值，如果user message它不等于那就表示说有错对吧？如果有错我就应该加上样式is invalid，否则我就不讲样式，这里要写上三元运算。那么同理你这个账号是这样的，那密码也是这样的，也是竖线。好拼上一个变量，但这个叫password，message逻辑是相似的，copy一下，整体都一样，确认密码不需要做这样的判断，然后邮箱是需要的，这个地方是EMAIL，message。

好了，到这儿我们终于把注册页面的逻辑处理好了，总之处理页面的逻辑的时候比较啰嗦，那么你不用追求的时候我一步到位把它处理好，你可以先按照逻辑，我要提交数据时我要做什么处理对吧？

我回到这个页面时我要显示原来的值我要怎么处理，我要显示错误消息，我又怎么处理，那一层一层去追加就可以了。

甚至来说你可以就是说我先把内幕添加好以后，设置好以后我直接先测一下，如果没有问题我再返回来再去设置 value，再去设置 class也都可以。

好了，写到这里我们就可以做一个测试了，看看有没有问题。我们写了这么多代码，出现问题是很正常的，那么出现问题的时候，我们主要是多看一看控制台，多看一看它的一个提示，按照我之前跟你讲的调试技巧去解决问题就好了。

好，我就把代码重新部署一下，我按一下 CTRL f9算了，这样我还是点这个按钮就完全把服务重启一下，因为它CTRL f9编译的时候，有的时候可能会把 html这些东西可能会有一些遗漏，有的时候它是不及时的。

那点这个可能稍微靠谱一点，好我再打开浏览器，我先访问首页，再访问注册页。

好，我要注册了，我先我先看一下数据库，看一下表里的数据，我先写一个错误的故意写个错的试试。这样 order办ID倒叙一下，你看我之前已经有了一个用户叫牛客，假如说我还想注册一个用户名叫牛客，看看行不行，然后牛客的密码，你要问他密码。

我记得好像是123，咱们试一下。牛客密码就还输123，确认密码123，邮箱，我看一下表邮箱，我就填写我现在的邮箱就可以了。别重复。

好我就点注册了，点立即注册，成功了，因为牛客可以存在对吧？他返回了这个页面给我带来这个消息，而其他的没有受影响对吧？表示ok我这个逻辑是对的，我就账号不让我注册这个账号，我就改一个，改一个改成什么？改成换一个不叫扭亏了，随便写个嘻嘻密码123确认密码123邮箱是这个。

我这点注册对我换个邮箱试试，换个邮箱是邮箱也可以重复吗？我看其他邮箱里面我用 Test QQ.com，其实这个邮箱根本就这不是我的邮箱，不存在瞎写，看能不能验证出来注册，他说该邮箱已被注册，说明我们服务端的逻辑对于账号邮箱没有问题，那么如果说我什么都不写，什么都不写，他其实前台已经有了判断，他不允许你过，所以其实后端走不到，我想应该也没有问题，因为判空逻辑比较简单。

好我来填一下这个数据，账号是CC密码123，邮箱是我的邮箱，然后我点立即注册，这个有点慢，然后成功了，跳到了中间的中转页面，说注册成功，我们已经向你的邮箱发送一封邮件请尽快激活，然后倒数8秒跳到了首页，没有问题，没有问题。

那么接下来就要看了，我这个邮箱里有没有收到邮件还挺快的，收到了。收件箱，激活账号。某某你好，你正在注册牛客网，这是一封激活邮件点链接，而这个链接你看鼠标全程的时候，你看我们网页的左下角路径对吧？路径，然后IPTV151151是我的优质ID，后面 Ff500什么什么，那就是自动生成的一个字符串是激活码，但是现在还不能点，因为我们激活的行为逻辑还没处理，但不管怎么说，我们把注册的这件事已经处理好了，对吧？

就是说我们已经把第二步处理好了。

好，那么到这儿我们终于可以开发最后的一个环节了，这个请求开发完以后，咱们整个注册功能就都搞定了，这个相对来说就容易多了，其实这三个请求当中就是中间的这一步比较繁琐，那么激活账号无非就是做验证比较容易一点，好，我们接着写激活注册账号的代码，我打开我的开发工具，这样我先把刚才我说编写的代码关掉，因为这个内容放不下有点乱，好，我们重新整理一下这个思路，我要激活这个账号，我首先要做什么，首先我需要在业务层，数据访问层不用，因为激活账号其实最终是要改变一个用户的状态。

我们数据库访问层面，我们其实有的map已经写了，所以不用写了，我们直接在service层加一个业务就可以了，再加一个业务方法。

那么我们在处理激活账号的时候，其实有几种情况，一种情况就是说你激活成功了没有问题。还有一种情况是什么？我点了邮箱里的链接，我点了不止一次点了多次，就第一次你激活成功了以后，其实重复激活就没有意义了，所以还有一种情况是你重复激活，我们要给个提示，还有一种情况是你激活码可能是伪造的，你是瞎编的，就不对就失败了。

所以激活最终有三种结果，成功了，或者是重复的激活或者是激活失败了，首先我想把这三种结果做一个敞亮的声明，写一个把它放到接口里，我们好在不同的地方能复用好，能够得到这个状态。好，那么这个常量接口我把它放到YouTube下，这是一个接口就可以了，用起来方便，那么我给接口取名叫做community constant，就是常量的一个接口。

然后这里边我就定义几个常量，主要是激活的一种一些状态，首先是激活成功这个状态，那么常量我们通常以大写的单词来命名，ATV的小火线，success等于0，这种情况表示成功。

好，再来一个还有可能是重复激活，这个复制一下 active region，然后是重复激活repeat，大写的好办，你可以有快捷键，好像是CTRL CF的u能把它做单位转换，然后再来还有激活失败的情况，然后也copy一下，然后失败那就叫failure。

Failure当然这个值不能都是0，都得有区分01二我就把常量定义好了，那么我在user service里头去处理激活的逻辑的时候就需要用到常量，所以我让当前的类实现接口，a place community constant。

好，那么刚才我们写的是register方法，现在我们写的是激活的方法，他们这个方法就返回激活的状态，你是成功了还是失败了，就返回int就可以了。然后报上名儿我就叫activity。那么你要激活账号的话，你看我们刚才的激活链接，你会传给我什么？你点链接的时候你会传给我用户的ID是多少，要给谁激活对吧？以及激活码是多少，所以我们能得到这两个参数值，所以我就要求你把优质ID告诉我，要求你把激活码给我，然后我就可以帮你激活好了，那么我们处理这个机构的逻辑也非常简单，就是说首先你把用户先查到，然后查到以后你判断一下他激活码对不对？

大概是这么一个逻辑，对了，我们怎么处理错了怎么处理，对吧？

首先是作为一个查询 User，等于利用 map对谁来办ID方法，你把 uzid传进去，我们会得到这个值，那么有这么几种情况，如果说悠着点get斯特尔斯，它等于1，你注意我们注册的时候斯特尔不等于0，而你激活以后它会变成一，如果我们发现斯特都已经是一了，可是它已经激活过了，你这次机会就是重复的对吧？

所以我觉得return什么？Active region，repeat这是一个重复的激活。

好，再来如果你不是重复的激活，还有可能出现什么情况，还有可能是这样的。User的get activation code点e cos错了，不是按照位置，是e cos点e cos code。如果你激活码和传输的激活码是一样的，那就表示说你激活码没有问题对了，你就可以成功对吧？激活我最终要返回的是 Active vision success，但是你在激活成功之前还要做一件事儿，你得把用户的激活状态改了，改成一对吧？

这才叫成功，所以你要调you the map的update的方法，我们之前不是写过吗？Update是特色，你把用户的ID传进来，对吧，把状态改为一就比较成功了，还有否则这是扣的相等，否则就是不等，不等就失败了，return，activity，非典。

好这样就行了，激活的业务层面的方法就写完了，写完以后我们需要在 control里再把这个请求处理一下。请求的页面是邮件的邮件就是一个html，所以我们已经把这个页面做好了。

好我需要打开controller，还是刚才 Log in controller，我就是无非是再加一个方法处理这个请求。那么先声明请求的路径，这个路径你不能乱写了，因为之前我们在注册的时候规定了这个路径应该是这样的，对吧？我把它拷过来再做一个强化，应该是这样，前面的都是默认的，就是我们访问的时候它自动的后面的话你这个就是路径名对吧？

以及它的参数，你要按照这种方式来写，所以 Pass应该等于它，但这里不能写死，应该写成变量对吧？

这个是user ID这个地方是扣的，然后这个请求是盖的请求就行，因为怎么说我激活的时候，因为它不是我们邮件上不是给提供一个表单，对吧？

只是利用路径简单地携带了两个条件，他最终期望你返回的是激活的一个结果，其实这相当于是一个获取的行为，我到底成没成功识别失败对吧？

它其实相当于是一个查询的行为。

好，然后我这个master就等于request点，你看那是点get，盖的请求就可以，然后他不理他是这样，方法名我叫 activity，那么这个方法上面我还需要加上 model好，用来像我们将要访问的模板传参，然后我需要用注解叫pass无二宝。

从路径当中取这个值，咱们之前不是讲过，首先我要取的是uzid对吧？我们得到的是一个整数。

好，其次我要取的是把这个复制一下，Code我们得到的是一个字符串。好，就这样，这个方法就说明到这。那么我们得到了这两个参数以后，我们去调service的方法去处理这个结果就可以了，非常容易 use。Service点aptv点把优质ID传给他，把客户的传给他，他会给我们返回一个结果，这个结果的含义，我们从接口中可以去识别，你不要去用手写012对吧？

我们最好是依赖于接口，为了访问接口方便我这个类也实现这个接口，community cost好，然后我要对结果做一个判断，如果说 Result你等于的是active vision，success你成功了的我怎么办？对吧？

还有一种情况是without等于activity。Repeat。那么就是一如果你重复了我怎么办？还有否则我不用比较就三种情况，否则我要怎么办？这三种情况其实第一种情况是成功的，后两种都是失败的，无论成功还是失败，我都希望给用户一个提示。

怎么提示？还是刚才的 Opera的中间页面，我们跳转到中间页面给他一个明确提示，你成功了还是失败了，然后提示完以后，那么跳到最终的目标不一样，成功以后我就跳到登录里面，因为你激活成功了，可以马上就可以登录了，对吧？如果失败了我就调到首页，你还不能登录，调到首页你先看看别的帖子就这个意思。

好，那么总之无论如何我们最终这儿要访问的是 operate，就写上return，operate这里有就他好，那么登录成功的时候，我们给一个这里可以复制，因为这里不是有了这样的逻辑，对不对？可以复制 opera的页面需要两个参，一个是message一个target对吧？登录成功的时候我给他一个message，注册成功我也给他一个message，不准激活成功，老说错激活成功。然后说您的账号已经可以正常使用了，然后我们最终跳转的目标是登录页面log。

In，假设页面的路径是log in，其实我们登录页面的登录页面还没有声明的对吧？一会还得声明一下，但我先定义好它的路径是login。

好，然后如果是这种情况就是说它是重复激活的，我们要给予提示 Message说这个是无效的操作，那么该账号已经激活过了，这个时候就不要去登录页面了，就去首页好，再否则这是错了你就激活失败，然后您提供的激活码不正确，这个时候我们还是也是要去首页，我就不去登录页面，好，完了这个逻辑就处理好了，比较容易。

好，那么处理完以后，当然这个operate我们刚才已经处理好了，这个不用管，我们还需要处理的就是登录页面，我们现在还没有登录页面，我就再写一个访问登录页面的请求直接在这里加一个就可以了。

我在这加访问注册页面再来个访问登录页面，其实和它很像复制粘贴，稍微改一下就可以了，这个我就要烙印也是盖只是得到一个登录页面的登录页面，只是得到一个页面而已，放个名我要get哪个in配置，然后他的页面的名字叫login的html，在sett目录下，所以这地方就写log in就完了，这就写完了，当然写完以后这还不行，你还得把 log in点is太慢，把它添加上模板相关的声明，所以我需要找到烙印，点html，加一下xml ns，HTTP冒号双曲线是三w califf点LG，然后这个页面的逻辑我们先不处理，因为我们当前开发的事，注册功能不是登录功能，我们只要把这个页面能正常的展现给用户就可以了。

在这一步，所以你需要把样式文件的路径什么的处理好就行了，所以这里我写th然后艾特好，把样式文件处理好，这个也是要做这样的处理。

好。当然底部还有js，这三个不用，就这一个可以了，然后别忘了，黑的每个页面都要复用复用的都是index的页面对吧？所以我把黑的也处理一下。Th replace。Index header好了，登录页面我就配置好了，然后还差一点点事，因为我们现在已经增加了访问登录页面的请求了。

那么我们在这个页面上是不是应该这个可以点，现在还不能点，因为你看这个路径还不对对吧？

所以我们在首页上再把它配好，在首页上index把登录的页面的路径配好，在这和一样类似 th，好，然后艾特老毕。

好了，终于把这些零碎都配好了，配好以后咱们可以尝试一下，那样这样重新重启一下点，它就自动重新编译，然后自动重启服务了。启动好以后，那么我先打开页面，看一下我登录页面对不对？能不能访问？可以，没有问题。好，接下来我就要去激活我的账号了，刚才我邮箱里不是收到了激活码对吧？我就点一下点了一点以后向服务器发出请求，服务器做出了响应，说正对了几乎成功了，然后倒数几秒以后跳到登录页面，你就可以去登录了。登录页面我还有的地方没处理好，就这个验证码是一个图片，现在是一个固定的图片，将来我们需要动态生成，这样看不好看，把它也改一下。

好，打开 Log in，点开mail找到图片，这儿原来写的是这样的路径不对，我先把它改一下 th，就是说它不是不对，是在怎么说是在就是静态页面的环境下是对的。当我们把它拿放到模板引擎下，放到模板里，它的路径发生了变化，那就找不到了。好，处理完以后咱们再可能f9一下，然后打开浏览器刷一下这就可以了，当然你现在点的话没反应，因为我们还没有做刷新的处理，但是我们看起来顺眼了就可以了。

好，那么到这儿我们终于就把注册功能开发完了，虽然说这个功能比较繁琐，但是你仔细想一想，我们把它拆解成三次请求，一次请求逐个的去攻破，攻破完以后你发现这个很简单，没什么特殊的内容略微麻烦，但是你把它拆解成几步以后逐步去实现，还好，这一步相对来说还算比较容易，所以拆解以后拆解成三步以后，再根据具体的逻辑再去做一个小的拆解，就是总之我们一个复杂的功能就是做层层拆解，直到我们可解为止，这就是我们开发的一个套路。

好。那么课后大家好好的把这个东西写一下，你去体会一下这个套路，然后遇到问题好好利用我之前所讲的调试技巧去自己去调试程序。我们这节课就先讲到这里，咱们下次课再见。

## 2.2 开发注册功能

11-开发注册功能.mp4

这次课咱们来开发注册功能，那么注册功能还挺复杂的，像这种相对复杂一点的功能，我们开发的时候要额外的注意，复杂的功能，我们想办法把它拆解，然后拆解成若干部分，使得每一部分都相对简单一点，那么就好开发了。

那么对于一个外部项目来说，我们每一个功能其实可以把它按照请求去拆解，因为每一个功能它都是由若干个浏览器和服务器之间的请求交互所构成的，这个是很容易分析的，你像注册功能，你怎么去访问怎么去使用，你很容易能想到，那么首先我在首页上点注册的链接，就能打开这个页面，这个很显然访问服务器就有一次请求，只不过只是打开页面而已，这次请求会比较简单一点。

第二个我需要填写数据在表单里，然后点注册这个数据就会提交给了服务器，服务器去把它存下来对吧？但是显然是第二次请求。

然后服务器保存了这个账号以后，它会给你发一个邮件，然后让你去激活一下，这个时候你收到邮件以后可以点邮件中的链接，其实一点链接也是访问了我们的服务器，然后进行了激活的服务，这是第三个请求，所以很容易分析出来我们这个功能由三次请求所构成，那么我们一次请求一次请求去把它搞定，就相对来说就容易了。

而每一次请求我们还是按照之前的思路，我们先开发数据访问层，然后开发业务层，然后开发视图层三层架构对吧？但是你注意有些功能它可能没有数据访问层，有些功能可能没有业务层可能会有一定的缺失，你比如说首先第一个我们访问注册页面只是打开页面而已，这个就没有什么业务，也没有什么访问数据库，好，我们先把第一个环节搞定，好，我打开开发工具，那么对于这个视图层，首先我们得有 controller 对吧？

我们请求提交给 controller controller，然后当然这里这个业务比较简单，没什么处理，他直接去调用模板，然后有模板给浏览器做出响应就可以了。

那么这个模块登录模块我单独的给它创建一个新的 controller，在这里，这个 CTRL 就叫 log in CTRL，那么创建好以后，我们需要在这个类上声明啃出了注解，当然通常我们会加上瑞克斯麦克这个类我就不加了，不加以后我就直接访问的是方法。

好，接下来我就在这里增加一个方法处理访问注册页面的请求，他被返回死病。

然后这个方法名我叫 get register 配置就是获取注册的页面，那么返回的是模板的路径，我们之前已经把所有的页面都引入进来了，注册页面是很好找的，它在 site 目录下，所以我 return 的是 site，然后 right，当然写完以后前面别忘了给它说明访问路径，pass 等于 radius，然后 master 请求方式就是 get 好，这样就可以。

那么当然了这个是模板，我们还需要对模板进行一些设置，加上他们那些个标签相关的内容。

好，我就打开 register 把它放大一点，然后首先加上 timeline 的声明，你可以拷贝，我懒得切换，界面我就直接写了。好，然后我们页面之上所要依赖的样式文件，js 图片，那么它的路径我们需要通过 time leave 的方式去管理起来。那么这种绝对路径你不用管，主要是这种相对路径我们需要进行一个处理。

我就写上 th:hief 然后艾特括号这两点就不要了，然后 css global 点 css 这个也是好，然后我们页面的结尾之处还有 js，我们也需要做类似的处理，总之开发页面是一个比较繁琐的事，因为页面的代码看起来会比较长比较乱，我们需要挑着去改，很容易会遗漏一部分，所以说你在处理页面的时候需要比较富有耐心了。

好了，那么这个页面的这些个路径静态资源的路径我就处理好了，处理好以后，那么我们访问注册页面是通过点击顶部的注册的超链接点过来的。首先我们整个项目首先我们访问的是首页对吧？所以通常都是通过首页去点过来，接下来我把首页的顶部的注册的链接给它写好，练过来。

好，那么我打开首页的模板，找到 hide 的区域，在这儿这里面我不只是要把注册改了，把首页也得改，对吧？因为首页当然练的就是他自己这个也要改一下，不然的话你一点就会有问题，这个是不对的，这里我写一下 th:冒号 hief 然后后面就不是 html 了，我们首页的路径是 index 对吧？Index 接下来再把注册的路径来处理一下。

那么注册的访问路径就是 register，然后其他的功能我们还没做，那么其他的路径我们就先不去写了，先这样了。另外你注意咱们这个项目中其实每一个页面头部黑的区域都是一样的，对吧？如果你每个页面都把这些个链接都这么处理一下，会非常的麻烦，所以最好的方式是能够让这段代码能够在多个页面复用，那么他们内部也是支持的，而且很容易。

那么如果想复用的话，首先我需要在还得这个标签上写上一个属性叫 th:frige 问题。

等于说白了其实就是给这段代码取的名字，你好通过名字去复用它，我给它取个名字就叫就叫黑的。好，那么这儿我取了名字给这段片段，我 register 想复用那段程序怎么写呢？

咱们来写一下，我首先找到 hide 的区域，那么这里我也需要写上 Timeless 的属性 th 冒号，我们要写 replace，那意思是我要用这项内容替换当前的标签的内容，我要把它替换掉，用谁来替换 index？Index 就是 Index，模板他并不知道他的位置，然后冒号两个冒号后面写上还得刚才我们给代码评论取的名字，你这样写就可以了。

好，那么写完以后咱们可以进行一个测试，我就启动一下这个项目。好，以后我们启动项目都是用 debug 模式，那么出了问题咱们好打断点，随时去调试。好启动完了以后我打开浏览器，local host 8080/index，访问以后主要是看一下这个链接对不对？你可以把首页悬停到首页之上看左下角这个位置，你看的链接是对的，放到注册上看左下角的位置也是对的，咱们点一下。

没问题点注册，没问题，这点首页就可以了好了。

那么开发注册功能第一步访问注册页面我们就处理好了，这里是主要的要演示的内容就是 Timeless，我们怎么利用它把一段标签一段什么标签的内容复用，这里就演示到了。

好，接下来咱们就可以开发第二个请求了，要提交注册的数据，在开发请求之初，我想先再导一个包，那么我们在这个功能里以及我们后面的功能里会经常的用到这个包，主要是给我们提供一些这个判断字符串，还有一些比如说集合等等这些常用的数据，它的一些空值的情况我们经常会用到。

好，那么这个包我们去搜一下，那么我打开 memory post 库，然后包的名字叫 comments long，comments long 回车好。

然后第一个使用的最多的就是康姆斯朗三，我就点进去，它是阿帕奇推出的，然后我们选一个最高的版本，3.9 就行了，这里我把美本配置拷过来，然后我再回到我的程序里，这样我把这个代码先关掉，然后我打开 pome 的 xml 把它拷进来，然后稍等一下，好这个包已经下载好了，导进来了，这是第一步。

然后第二步我还要做一点配置，我要在 application 在 practice 里把我们这个网站的域名配置好，当然我们现在没有域名，就是配的一个本机的 IP 地址，因为我们在注册的过程中要发邮件里得带激活链接，这个链接链到哪去得？链到我们的网站，所以你得有一个链接对吧？因为这个链接是在开发阶段，在测试阶段，在上线的时候它是不一样的，所以说你需要把它做成可配的。

好，下面我打开配置文件，application practice，那么我就追加了这个内容，其实是我们自定义的内容，要给它这个配置我给它取个名字叫 community，然后 t 也是我们自己取的，k 我叫 k 没有那天点 pass 路径什么路径都没域名，当然我们现在也没有申请正式的域名，就是写我们本机的地址，local host 端口是 8080 这样就行了。

第二步我们准备工作也做好了，然后第三步还需要做一点额外的准备，我们需要再写一个工具类，然后在工具类里头我们去提供两个方法，然后我们后面注册的时候好用，不然的话我们献血有点麻烦。

好，那么我就在 YouTube 的包下再新建一个工具类，这个工具类我叫挺牛的挺 you tell，然后这里头我就提供静态的方法，都是很简单的方法，然后我们直接调就完了。这个工具我就不由容器托管了。

第一个我要加的方法是它生成一个随机字符串，我们一会生成激活码，每次是一个随机字符串，再一个将来我们可能还就一定会做上传头像的功能，或者是提供一些上传文件的功能。那么每次上传的时候，我们需要给这个文件生成一个随机的名字，那也是一个随机字符串。所以我们在这个以后的开发里经常需要生成随机字符串，所以这里把它提前封装好比较方便。

我写一个注释，这是生成随机的字符串，它是一个公有的静态的方法，返回的是死病，那方法名我叫 generate uuid 因为我们生成随机字符串，它其实用到了一个工具，就是 uuid 工具，其实这个工具可以直接生成，但是我们需要对它稍微封装一下就可以，我就直接 return uuid Java 里自带的工具，然后 render uuid 它会生成一个 uuid 对象，然后点 two string 我们就得到了一个随机字符串，这个字符串它中间有字母以及横线所构成，通常我们不想要横线，我们只想要字母和数字就够了。所以我就这样写，就是点 replace are 替换掉所有的横线，把它替换为空的股票，这个逻辑就完了。

除此以外，我们再封装一个方法，是叫做MD5加密，我们采用 MD5算法的方式对这个密码进行加密，因为咱们注册的时候你提交的密码其实是明文，那么我们存的时候要把它加密，不然的话密码是容易被泄露出去的，或者容易被盗的。

那么 MD5算法加密它有两个特点，第一个特点就是它只能加密，它不能解密。

比如说假设举个例子，我有一个字符串密码，hello，假设它加密成的结果是这样的，ABC123、def456，那么这个算法只能把这个字符串加密成这样的值，而且你每次加密它都是这个值，但是不能解密，那么看起来好像说我们得到这样一个加密的结果，好像是很安全的，其实还不够，为什么？

因为如果说你的密码真的像这个单词一样hello或者是一个生日一样，非常的简单，那么因为它每次加密的结果都是这个值，所以说想盗取你密码的人，那些黑客其实这个值他也知道，因为他自己有一个简单密码的库，hello生日都包含在内，他也会利用这个算法把这些简单的密码加密成这样的值，所以说其实你密码简单加密，结果其实人家手里也拥有着也是容易被破解的，因此我们一般是这样处理的，你看user表里不是有一个字段叫salt，盐我们不管你是密码是简单也好还是复杂也好，我们都给你加上一个随机的字符串，没有规律的，比如说加上一个3 e4 a然后这样的hello加上随机字符串以后，那么加密的结果假设是这样的，那么因为加上一个随机字符串以后，这个单词其实就没有规律了，那么黑客的他库里其实没有这样的单词，所以说他就相当于他库里就没有这样的密码，他就盗不了你的密码，这是其一。

其二如果说他想硬性的去破解密码，因为后面字符串是随机的，它破解的难度是非常大的，而且随着这个字符串的越来越长，你10位20位越来越长的话，那么它破解的难度是呈几何级增长的。甚至如果说你在这个字符串里还加上了中文的话，那就更难破解了。

所以说我们这样去处理，就在密码的后面加上一个随机字符串再去加密，是为了让安全性更提高一些，是这样的。

好了，那么这个方法我这样去封装还是公有的，然后静态的它返回加密的结果是一个死菌，然后这个方法名我直观一点就叫MD5，然后要求你传入一个key，Key其实就是这样，就是一个你原始的密码，我们给它拼上一个言拼成一个随机字符串传进来，然后我对这个字符串进行一个加密。

那么其实加密 spring它有一个自带一个工具就一句话就搞定了，就实现了这个算法。

当然我们首先最好对这个参数做一个判断，因为如果你这个参数为空，我就不去加密了，也没有什么意义，我就直接返回空了。好，那么判空的话，我们就可以用这样一个工具叫stream YouTube。

我们选这个你看阿帕奇commerce浪三其实这就是刚才我们导的包里所带的工具非常的好用，然后你可以判断is block key，那么他会判定你这个k如果是now他认为是空的，如果是空串也认为是空的，如果只是一个空格，他也认为是空的，所以非常方便。

好，如果说你传的是空值，我就返回空，我就不去处理，否则我就去调用spring的工具，它叫 dig digest dig it。Youtube，YouTube这是spring自带的一个工具，然后点MD五，然后是得给as x就是说他把你的传输的结果加密成一个16进制的字符串返回，但是它要求参数传入的是带，但我们这个参数是string。没关系，我们转换一下，就key一点get back就可以了。

好了，总之工具的封装非常的容易，那么主要是你要了解，那么这个MD5它加密背后它的逻辑是怎么样的，为什么我们去在设计优则表的时候，那么有 sat字段为什么要这样处理？主要是为了提高安全性就可以了。

好，那么有了工具以后，接下来我们就可以这些包配置工具都准备好以后，我们这回终于可以去开发我们注册的业务了。我就打开 User service，我们之前写过这个工具，因为注册是针对用户表的操作，所以我们可以把逻辑写到类当中去，打开 user service。

然后那么我们在注册的过程中，因为要发邮件，所以说你得把这个邮件的客户端注入进来，对不对？你还得把这个模板引擎注入进来对吧？我们先把注入进来，首先我注入这个邮件客户端没有class，然后再注入进来这个模板引擎他们的安检。

另外发邮件的时候要生成一个激活码中要包含域名，还得包含我们的项目名，所以我们要把配置文件中的域名和项目名都注入进来，那么域名刚才我们配的key对吧，community pass都没有，而这个项目名这很早以前我们就配了 Sova点spotlight点，context parce，把它们注入进来，当然这个是注入一个固定的值，是不是？

一个b所以你需要用到 value注解要换个注解。

然后这里面写到了括号，表示说我用表达式的方式取 t的值，这里我写community，这字符串不要写错了，因为字符串它是不会检查你是否正确的。这里我说明一个变量来接收这个值。杜妹好。再来一个我可以copy，第二个是项目名，咱们通常习惯于叫context pass就应用的路径，然后这个是刚才我们看了是 Server点 context，god pass，ok，就是这样。

好了，那么终于我把注入进来的，内容都注入好了，下面我们就开始编写注册的业务，我就写一个共有的方法，便于别人复用，public这个方法需要返回一定的结果，这个结果就是要其实你返回什么都行，比如说你返回一个整数代表不同的状态可以你返回一个集合，包含了很多信息可以，你返回一个自定义的类都可以。

实际上我们返回的内容中包含什么？主要是包含比如说你这个账号是空的，不能为空，密码不能为空，邮箱不能为空，账号已存在等等这样的错误信息，它包含的信息不是一种情况，是多种情况，所以最好我们声明反馈是能够封装多多个内容，这样我就写一个MAC，我把返回的结果通知到map，词句ok，然后注册这个方法名我叫 resist，注册的时候你需要把用户的信息传给我，包括账号、密码、邮箱这样的内容，所以说我要求你转入一个，因为这对象给我，我来帮你解决这个问题，最终我暂时先写维特纳，然后一会我们再具体的去实现。

好了，这个注册方法这个结构就定下来了，定下来以后接下来我们就开始编写它内部的逻辑，这样我首先先把 Map给它实例化好，对六希map，然后最终的话我觉得返回 map先把这个结构写好，省得一会忘了。然后我们这个逻辑的一开始应该是对这个参数进行判断，因为万一你传的参数不对，你账号密码都没有输入，对吧？

所以先对这个值进行一个判断，那么这些注释首先我们是对空值做一些判断的处理，我就写空值处理。那么如果优点是闹我就直接抛异常了，这是绝对不允许的，你要是闹我就没法处理你。阿根问题 exception，非法的参数我就写参数，不能为空给一个提示就完了。

好，那么如果说user不等于我们继续再判断user里边的内容，可能对象不为那，但是里面的属性有问题，所以再判断，那么如果说user我用的工具是病，are you tells点is block？

先判断user点get u在内，这是一个很关键的信息账号，那么如果这个账号是空的，我要这回要给这个不报错，因为账号是空的，是一个他业务上的一个怎么说一个漏洞，但它不是说程序的错误，因此我们要把这个信息封存到map里，返回给客户端，告诉他说你这样做是不对的，而不是抛个异常。

好，所以我就往map里装一个提示信息，这是提示你账号不对，我这个提示信息的名字叫using name。Message提示的内容就是账号不能为空，好，完了然后我就不往下走了，直接就返回就行了。

好，那么同理你账号可能为空，密码也可能为空，所以你再把按照这样的方式再把密码也给他判断一下。这个是user点get它是word，然后如果密码为空，那么我们要返回一个密码相关的提示，我就叫 password，message，密码不能为空，然后如果是这种情况就回退了，还有那么我们注册的时候还要输入邮箱，邮箱也可能为空，所以也是做出类似的判断。

点get EMAIL，那么这种情况我就提示你的是EMAIL message，然后这里面写的是邮箱不能为空，逻辑都一样，所以说你拷贝复制粘贴很容易，那么如果上述的关键的数据都不为空，我表示说我可以开始去做注册的处理了，要处理的时候不是上来就去注册，而是你要验证一下你传的账号是不是已存在，对吧？

你的邮箱是不是已经被注册，所以你要做这样一个业务逻辑的判断。好，我先判断这个账号，我验证账号，验证账号就是说你把它传入的又在内，去库里查一下，看他有没有对吧？我就调用user map，然后点select by name。我条件就是传入的user点get user name。



好，那么我会得到一个user，user的名字就叫简单就叫u和参数做一个区分，这是页面传进来的，这是我查到的数据库里真实存在的数据，当然如果你账号不对的话，就账号这个怎么说是不存在的话，查到的结果就是闹，对吧？如果存在这个结果就不是闹，所以很容易就能得到一个判断的结论，如果u它不等于闹，那就意味着我查到了意味着这账号存在了，那就不对了，我就给他一个提示放到map里，这个还是账号的问题，所以我的返回的k依然是user，那么这个提示的消息就是该账号已存在，此时我就return map不往下走了。

好，这是账号的验证，同理邮箱的验证也是类似的，就是验证邮箱，因为这个逻辑比较长，所以我把每一步的注释写的清楚一点，验证邮箱的话就是通过邮箱去查一下用户看存不存在对吧？我还是复用U变量名user map点select EMAIL，然后user点get EMAIL，好，然后判断他是不是等于如果说它不等于就比方说这账号邮箱是存在的，我还是做类似的处理好。

这个地方一没有message表示该邮箱已被注册你得换一个，或者是你要找回密码。

好，那么经过这两步以后，账号密码邮箱都不为空，然后账号邮箱它也不存在，就说明这个信息没有问题，这个时候我就可以注册用户了，那么所谓的注册用户就是把用户的信息存到库里，我就写个注册用户，然后我在保存之前还得对密码加密码，对吧？而密码加密是他输入的密码，你要加上一个salt，然后再加密对吧？所以你还得先生成随机的salt随机字符串。

好，所以我们需要对user里边的数有些数据做一个补充，悠着点set给他一个随机的字符串儿，community YouTube咱们刚写的工具类点，MD5。抱歉写错了，是生成随机字符串，get点generate uuid然后这个字符串非常的长，我们不用那么长，我们用5位8位就比较短就可以了，这里我就来5位，我就在萨德斯郡一下，然后的话从0~5一共有5位。

好，那么有了so了以后，我们再对密码进行一个加密，然后再设置到u的对象里，对他原来的没有加密的密码进行一个覆盖。Site、password，然后这里边需要加密码，那就community、YouTube点MD5点m第五，然后传入的参数应该是它原来的密码，没有加密之前的密码点get password，然后加上sort user的get sort。好了，那么经过加密以后得到一个值再覆盖这个password就行了。

然后我们再把其他的字段也进行一个设置，因为它传进来的只是邮箱，账号密码，其他的字段也需要我们重新设置一下悠着点site，tap类型注册的用户默认都是普通用户user点set status状态默认都是0表示没有激活，然后你需要激活，我需要给你发送激活码，所以悠着点set activation的激活码，这个激活码就是一个随机字符串，还是用这个工具点generate就不错了。

Janet uuid就行了，那么激活码可以长一点，没关系，我就不截取了。

好，那么用户注册的时候，我们在给他一个随机的头像，因为他注册的时候不要去上传头像，我们给他一个随机的系统自带的，然后的话他可将来注册，成功以后登录以后可以去自己再改一改。

那么牛客网有好像是1001个随机的头像是路径是给你看一下是image点not q点.com，然后是head，然后下面是一个比如说一t一t点瓶颈，这是一个随机头像，然后只要你改序号一就行了，改成23一直到1000，好像0也可以。

从零开始到1000，一共1001个随机图片，所以我们随的就是这个数字，然后把它拷进来，所以我需要给它设置一个随机的头像set，hide URL，这个因为那个数字是随机的，这个路径当中包含一个数字，我用死菌的放卖的方法，然后把这个字符串写进来，这个地方就不能写死为0了。

应该写一个占位符百分号d比方说这个地方是一个数字，然后这个方面的方法我可以给它再加一个参数，那么它会用这个参数来替代占位符好，这应该是一个随机的整数，我就这样写new render账号自带的一个工具，然后点next int随机整数，那么这个整数的范围我希望它在1000之内。

好，就这样，那么头像的路径也随出来了，最后再给user设置一个创建的时间，或者说他注册的时间就是当前的时间，ok。

到这我终于把优质对象里边的数据补全了以后，那么我们调之前我们写过的insert优质方法，去把它添加到库里就可以了。

好这一步就完成了，完成以后别忘了我们还做一件事儿，你要给这个用户发一个激活邮件，那么发送激活邮件，我们当然发的是html邮件，你们好带这个链接，首先得考虑那模板是谁，模板需要什么参数对不对？

其实这个模板我们之前在拷贝静态页面的时候已经拷进来了，它就在 mail的目录下看名字就知道了，Activision的xml。

好，那么我把它打开，咱们先设置一下，因为目前的话还是只是一个静态的html你需要把它改造成模板，这里首先我要写上xmlns thhttp三w timeless表。Rg好，先指定先声明它是一个time leave的模板，然后这个链接绝对路径不用改，这就不用动了。那么看里面它的格式，他说叉叉叉.com，这是邮箱，你好，显然这个是我写的一个例子对吧？

你需要把它替换掉，因为刚才都是把它替换掉，所以这里我们需要写th text等于一个变量，假设我要求这个变量叫EMAIL来替换这个值，一会你要把 Email传给我这个模板。

其次这里边有一个正文说您正在注册牛客网，这是一封机会邮件请点击，然后这个链接，然后注册激活您的账号，那么很显然这个链接你得动态的传进来，不能写成井号，要不然他不能点，对吧？

所以这个地方还得改成动态的值，th冒号hef等于那么URL就路径传进来一个路径，请点击，这是一个事例，就是这样一个路径。

其实你也可以这样就把这个路径去掉，写成此链接，那么我们这页面上会看到这样一句话是吧？

您正在注册牛客网，这是一封寄货邮件，请点击此链接，然后激活您的牛客账号，我们看到的是这三个字，但是这三个字是有链接的，是可点的，你一点就链过去了，所以没有必要把那个路径写的那么啰嗦，在这里就三个字，更直观一点。

好了，这个模板就处理好了，处理好以后我们再回到service，接下来我就可以利用这个模板发邮件了。那么上节课我们讲了邮件怎么发，所以说很简单，把上次跟我们说写的 Test里的代码挪过来就可以了。

那么首先激活邮件，首先我们需要创建这样一个对象叫context，aaah是timely foo包销的对象，识别化的，我们好通过这个对象去携带变量，好那么 contact、点、sat wherever你得先带上 EMAIL，对吧？而这个EMAIL发给谁就发给用户的邮件点，user点，get EMAIL对吧？就这个值。

其次你要给他晒了一个 URL，这里边我们就得规定一下 URL它是一个什么样的路径，这是由我们自己定的。就是你希望服务器用什么样的路径去处理这个请求，那你就可以自己定，假设我要求这个路径是这样写，完整 HTTP，冒号双曲线，logo house8080这是我们的域名，对吧？

然后接下来是我们的项目的访路径community，接下来是我们这个项目之下某个功能的访问路径和激活的功能，我叫activity，然后这是哪个用户在激活，对吧？

因为很多用户都可以激活，这是哪个用户，我们得给一个标志，比如说这里我就要求你传入一个用户的ID，101是用户ID，直接把它拼到这个路径里面来，然后还得携带激活码，我就后面再带一级扣的就激活码，比如说我要求激活的路径是这样的，所以那么我们得把这个路径动态的拼出来，不能写死都能拼出来。

首先前面这一段是域名，那就是我们注入的度命，接下来是项目名，那就是我们注入的康txt，再往后是activity是功能的访问名，我们这就得写死了，这就可以了，因为以后不会变，我就写错了，copy一下。

然后再加上用户的ID那好办，user点get ID你注意注册的时候，用户传进来的优质对象里是没有ID的，但是当我们调用了insert语句之后，那么 user里就有ID了，这是买卖得自动的获取了生成的ID对它进行了一个回填，为什么会这样？

因为我们这儿对买卖投资做的配置就是使用自动生成ID的这么一个机制，所以它底层在执行insert的时候会自动生成ID，然后我们在 map配置文件里还指定了 ID的属性对应着谁，所以它会把生成的ID复制给属性，所以 user点get ID就有值了。

好了，然后 ID 之后还有最后一集是激活码对吧？我们得加一个斜线，然后再加一个激活码，激活码就这里不也有了么？Get xtv 正扣的。好，这个路径我们终于把它拼好了以后，你需要把它也 site 到康 txt 之内，P 叫 URL 值也是 u 二。

好，那么拼好了这两个条件以后，接下来我就可以利用模板引擎去生成邮件的内容，此卷空间的内容等于 template 的安检点，process，那么那个模板我是把它放到了没有的路径下对吧？

然后 Activision 点 html 后缀不用写好，然后把 context 传进来，那么生成 html 格式的内容以后，最后一步就是调用 milke lend，然后散的没有方法去发送邮件就可以了，邮件发送给用户邮箱，所以优质的 get EMAIL，然后邮件的标题，这是一个固定的业务，所以说固定的标题就是激活账号，然后内容就是 content。好这样就行了。

好了，那么到这儿我们 service、业务层注册用户的逻辑就完成了，那么最终如果说你传的参数有什么问题，我们会返回 map 里边，包含对这个问题的描述，我们可以把这个描述显示到这个页面上，给你做一个提示，如果说没有问题，我们就进行一个注册激活，最终的返也返回 map，这个 map 就是空的。

那么如果 map 里是空的，就比方说没有问题，好了，那么 service 完成了，完成以后接下来我们再去开发控制器 CTRL，那么在 CTRL 当中我们去处理前后端交互的逻辑，好 CTRL 我们还是利用 log in control 了，还是在这个模块中去写，那么我们首先需要注入一些内容，因为我们要处理这个请求是需要调业务层的，对吧？你首先得把 user service 注入进来，我们好去访问它。

User service。好，那么其他的暂时不需要了，因为类提供的软件的方法一下就能解决所有的问题，只要你把参数给他就能解决所有的问题。

好，然后我们需要定义一个方法来处理注册的请求，这个请求是浏览器向我们提交数据，所以请求得是 pose 的请求我先声明一下路径 pass 等于 register，然后 master 等于为快 master 点 post，然后方法是 public 返回 string，就返回视图的名字，然后方法名就叫 register。

首先我要增加 model 参数，这样的话我好往 model 里存数据，携带给模板，这是第一个。第二个页面它会传入什么？它会传入注册时传入账号，密码邮箱，你可以声明三个参数去接收这三个值，或者是直接声明一个 user 对象也是可以的。

好，那么只要这个页面传入传入值的时候，它的值与 user 的属性相匹配，那么 Spring mvc 就会自动的把直注给注册对象里面的属性，这样非常方便。

好了，那么在这个方法的实现的时候就简单了，因为我们有 service 对吧？你只要调 service 的 resist，把 user 传给他，让他去处理就好了，然后他会给我们返回一个 MAC，我们根据 map 的返回值去进一步看我们要怎么给浏览器做响应。

好我得到一个 map，我判断一下，如果说 map 它是空的，或者说 map 它是它是或者说它是空的，总之就是卖的是空着的时候表示说我注册就成功了，那么注册成功的时候我们怎么办呢？

注册成功的时候，我们应该在浏览器上给一个提示，说你成功了，提示完以后我们应该跳转到这个登录页面，马上你就可以登录了，这样就比较方便，对对不起，我说错了，是我们注册成功以后就给它跳到首页去，为什么不是到登录？

因为它有一个激活的这么一步，你激活以后再去跳到这个登录页面，这是我们逻辑，刚才我记错了，我们跳到首页去，然后我们怎么给他一个提示，我们可以把请就这个数据返回给登录页面，然后登录页面上弹出一个提示，但那样的话不好看，我是这样做的就是说当他注册成功以后，我弄了一个第三方的页面，在 site 下这个页面叫 operate result，就是你操作的结果的页面，operate result。

这个页面非常简单，我把孩子折叠起来，这个孩子没什么可看的，福特也没什么用，它的内容就非常简单就是说你这做什么事成功了，但是这个地方的值你可以改，然后系统然后成功以后，系统会在几秒后自动的跳转的目标链接由你来定，所以它过几秒就自动跳转了，自动跳转的逻辑是底下 GS 实现了，很容易很简单的。

好了，就是说我们注册成功以后，我们跳到这个页面给一个提示说你成功了，然后过几秒再跳到最终我们想去的那个地方。而这个页面需要两个条件传进来，一个是提示信息，一个是你最终要去哪个页面，是这样一个逻辑。

好，那么我们回到 CTRL，我就这样处理。Model点儿at actually message这是给中间页面一个提示信息，提示信息我就自己随便写成功了，我就写注册成功，然后说我们已经向您的邮箱一发送了一封激活邮件，然后请尽快激活，就这么一句提示。

另外提示完以后，倒数8秒最终跳到哪去，还得给他一个目标的设置。Model点儿led actually目标也叫target，最终我要跳到首页，我写上首页的房租叫index，然后我就return我要去的模板，那么它在site之下，而且site operate result这样就行了，当然这样还没有写完对不对？

因为你这个模板还得根据这个条件去设置，我们还没设，所以接下来还得设置这个模板。所以说其实每一个环节还挺啰嗦的，但是也还好，你按照请求执行的流程的倒叙一步一步的反推去做就好了。首先我需要把它定义成他们那个模板，这里我要写上，xm Insth他们if挂机，然后这个路径是绝对的，没有问题，这个路径我们需要进行一个定义，采用 Time before的风格去定义它。

好，看底下有没有GS需要去处理，这个GS都是绝对路径，不用管。好，那么最后我们去看中间的环节，中间的内容咱们怎么去配？首先我们要就这个提示信息可能不同的业务它有不同的消息，我们要根据实际情况去动态的改这个值，所以这里应该是个变量 test，等于当然我们传的是一个message对吧？Message。

另外跳转自动跳转是跳到哪去，是跳到超链接的路径，这个路径不能写死，要动态的改变。

Thhref这是一个路径，他们的意思路径的写法是以at开头，然后但是这个路径不是一个固定的值，是一个变量，所以这里还得用表达式把它取到变量名叫target。

好就这样就可以了，这个模板我也定义好了，我们再回到 control了，继续把它永章的逻辑给它处理完。

刚才处理的是就是说他注册成功的情况，l否则 map中如果有值，比方说它失败了，失败的时候我们需要向页面传那些失败的消息，好让它显示而最终我们要return到哪去？你失败了应该return回原来的页面是谁呢？Resist对吧？

所以你看return就是它了，return回去，但这里要携带一点信息，因为 Model点l的还是不对，那么当然它的内容可能三有三种情况，一种是账号有问题，一种是密码有问题，一种是邮箱有问题。

这里我不去识别到底哪个的问题，我就直接把账号、密码、user、EMAIL那三个信息一股脑的就发送给他，但是有可能有的值是空的，一般上空的我们就不处理，不是空的就处理就可以了。

好，那就是username message，然后这个值我们从map里取map点，get usually message同理再来俩，第二个是password。Message。最后是EMAIL message。

好，总之错误的时候，我把这三个消息发送给发送回注册页面，接下来我们要对这个模板进行一个处理，模板我都打开，他这个模板我们之前已经把路径 hider说到嗨的我刚才忘了一件事儿，那么这个opera的不也有head吗？他也得进行这样的复用对吧？刚才把这个事给忘了，但这个是小事儿，最好还是把它写上，我写哪去了，怎么写到结尾去了？因为刚才我把它合并了，在合并的状态下一填就填到结尾上去了，写错了。

好，那么回退一下把它写到这个位置。行了，好，我接着去处理 Resist。它的header我们已经复用了，不用管了，内容是需要管的，负责不用管了，底下 Gs路径也不用管了，主要是要对内容的部分进行一个处理。

那么内容的部分其实主要就是对表单的处理，表单的话，首先你要声明这个表单提交到哪去，他什么样的提交方式对吧？

所以我们需要在 form上加上一个master的声明，我是以POS的方式提交数据，另外你要指定提交的路径提交给谁，那也是用 Tommy的风格去写这个路径，那么这个路径是rise的就行了。

另外你表单想提交的话，它最终一定得有一个提交按钮，button submit这就是提交按钮没有问题，只要你点这个按钮就提交过去，而提交的时候每一个框需要有name声明你这个数据的名字对吧？目前我这个name应该是没有的把它补上。

首先你看是账号，有ID，但没有内容就等于 user name这个name要和CTRL当中属性对象的属性名相对应，这样是不是mvc基于同名原则把参数传给user点user内容，这个一定要注意写错字母就不可以了。然后密码同理密码，这个name等于pass word。好，再来确认密码，注意密码和确认密码它俩是一个相同的值，所以我们把密码传给后台，确认密码不用不需要，它只是就是说怕你就是手误写错了而已，所以确认密码只是前台的逻辑，他不用传。然后是邮箱，name等于EMAIL好了。

## 2.3 会话管理

12-会话管理.mp4

这次课我们来学习会话管理，因为我们下一个功能要开发的是登录功能，那么在开发这个功能的时候，我们就必须要用到会话管理的方面的内容，那么首先我们先了解一下什么叫会话管理，这个得从HTTP说起，咱们之前看过 HTTP的手册，然后它有一个概述，概述中有一个小节讲的是HTTP的基本性质，我不知道你课后有没有自己看一下。

然后它里面就强调了说 HTTP是简单的可扩展的，并且它是无状态的，有绘画的，那么这段话其实讲的就是会话管理方面的内容。当然了为了看得更详细一点，咱们可以打开这个手册再去看一下，我已经提前把这个手册打开了。

然后我们就看 HTTP概述的部分，看 HTTP的基本性质，那么它是简单的可扩展的很容易懂，我就不详细的介绍了，大家自己看，我们主要看的是这一段，它是无状态的，有绘画的，首先你要理解什么叫无状态，他解释了说在同一个连接当中，其实指的就是同一个浏览器，访问同一个服务器，他们产生了连接，他们连接上以后可以发送多次请求，那么多次请求之间没有任何的关系，就是从服务器的角度来说，即便是你同一个浏览器向我发多个请求，那么我也把你当做一个陌生人陌生人来对待，我也不认为这些请求之间有什么联系，说白了我通过这个请求我不认识你，即便是你是来源于同一个地方，所以说这就带来了一个问题，因为我们平时上网很多网站它需要有持续性的交互，你比如说他这里给他举了一个例子，说我们访问电商网站，那么我可能是打开一个网页看鼠标把它加入到购物车，我又打开一个网页加了个键盘，加入购物车，我们知道我们最终去结算的时候应该是看到键盘和鼠标对吧？

但是默认情况下还实现不了默认的请求，服务器认为这两次请求没有什么关系，其实它不会把键盘和鼠标组合在一起成为一个订单。

同理浏览器因为服务器记不住，浏览器也做不到这一点，好，这就是问题。所以说这种状况就叫HTTP是无状态的，就是说同一个浏览器访问，同一个服务器多次请求之间彼此是独立的，没有关联的，或者说服务器它无法记住浏览器的状态，它不能识别你是谁，这种情况说它是无状态的，但是我们要想解决业务连续性的问题怎么办？有办法用的就是用cookies可以解决这个问题。

好了，所以最终有一个结论就是HTTP它本质上是无状态的，多次请求之间是彼此割裂的，没有什么联系的。如果你想让多个请求之间有一个联系，让这个业务成为一个整体，能够整合到一起，能够得以连续怎么办？我们用的是cookies可以解决问题，我们用库克斯解决这种业务连续性的问题，这就叫会话，就是说我们两两者之间浏览器服务器之间进行对话，那么是连贯的，不是割裂的这个意思。

好了，那么关于 Cookies，其实这个手册里也有相关的介绍，咱们后退一步，再后退一步，你看这就 HTTP cookie它很快就介绍了这个内容，我们也可以顺便看一下什么是HTTP cooking？

其实刚才说的本质的问题是服务器它不能识别浏览器它记不住，你想办法让它记住，其实这个业务就可以连续了。

所以库克就是这样一项技术，就是想办法让服务器能够识别浏览器能够记住浏览器，就大概是这么个意思。

所以详细解释是 cookie它是服务器发送到浏览器，并且保存在浏览器本地的一小块数据，就酷k数据是由服务器创建发起的，它会发给浏览器，然后浏览器会存到本地。

然后还有一个特点就是浏览器下次再访问服务器时，它会自动的携带这个数据给服务器，然后给服务器看一下，就好像这是一个令牌一样，就是我浏览器第一次访问服务器，服务器给你发了一个令牌，发了一个卡片，你拿走了，然后你下次再来出示卡片，ok我就能识别你是我的会员，你是我的这个客户，没有问题可以继续，大概就是这么个意思。

好了，然后关于cookie，我们读这段文字可能还是有点这个不够直观，所以我给大家画一个简单的图，然后通过这个图你可能会更直观的能够体会到 cookie到底是怎么回事。

好画个图，首先我画个方块，假设这个方块就代表了我们所使用的浏览器，我再画个方块，假设它就是服务器，那么如果能让浏览器想让服务器记住浏览器，那么是这样去做的，就是浏览器在访问服务器的时候，那么服务器它可以创建一个对象，这个对象就是cookie对象，然后并且将 cookie对象在响应时发送给浏览器。

当然库配对象中可以携带一些数据，一会我们通过例子你可以看出来，然后酷配对象是比较特别的，它在传输的过程中表面上我们看不到，因为它是默认是在请求的头里有一个参数，浏览器得到以后，他知道这是一个特殊的数据，我需要存下来，所以说浏览器就会保存这个数据在它本地，并且当浏览器再次访问这个服务器的时候，他会很自觉的在刚才说错了，这个是响应的头，当他再次访问服务器的时候，他会来请求透底把 cookie再带回给服务器，然后服务器又收到 cookie了，就得到了用户的信息，通过这种方式他就记住一个用户了，这就是库克的一个工作原理。

所以的关键是库克是一个特殊的数据，浏览器在得到它以后会自动保存，下次请求是自动传送这么一个流程。

好了，那么大家了解完以后，接下来咱们就写这个小例子来看一下，来体会一下它的语法，然后我们通过浏览器的插件也能够看到这个过程，看完以后可能你就会这个体会会更深刻一点点。

好。下面我就打开了我的开发工具，然后咱们这是一个简单的小例子，不去处理什么任何的业务，所以我这个代码还是写到阿尔法CTRL里，这里专门演示一些小demo，然后我在后面追加一些内容，流行注释。

下面我要演示的是qk相关的事例，首先我想做模拟这个请求，就是浏览器访问服务器，第一次请求的时候，那么服务器要创建一个库，可以把它发送给浏览器，把这一步实现了，然后再写一个方法把搞定了，浏览器在访问服务器它怎么去带？

Quick，咱们先来把请求的逻辑搞定，我们写的是服务端代码，所以我需要在 Ctrl加一个方法能够接收浏览器的请求，我就先声明请求的路径，pass等于 cookie，然后 set，因为我这个方法是要产生一个 cookie，好，然后请求方式get请求就可以，这样比较简单，然后这个方法最终的响应咱们就简单处理，我就返回一个那个Jason字符串就完了，我就不去返回页面了。

所以我写上瑞斯邦斯包点。

当然你返回的是字符串，不是返回页面，这个和cookie没有关系，因为你不管是给浏览器返回什么，你最终都是要向浏览器作出响应，都是要通过response去做响应，而酷k在响应的时候，它是存到了response的头部，对吧？

所以说跟你返回的具体的内容其实没有什么关系。好，我就写上他的x卷，然后这个方法名叫site cookie，好，然后我们这里可以创建cookie，创建完cookie以后，你要把 Cookie存到risk boss里，那么在响应的时候才能够自动的携带给浏览器，所以这里面我们需要用到 response对象，这里我就声明一下，HTTP super late response。

首先我要一个创建图片，这个很容易，尤其声明一个酷配对象给它取个名字，然后去遛，这个对象在使用的时候它必须传入参数，它没有无参构造器，而且这个参数必须都是字符串，并且每一个qk对象只能存一个字符串，——组字符串一个q一对q白流是这样的，我这里随便存了，比如说我要给浏览器发送一个编码扣的，假设这个编号我是自动生成的，随机的，我可以调 community YouTube对吧？

点January的uid好了，cookie对象就创建好了，创建好以后，我们通常还要对它做一些设置，我们可能经常会要设置它的生效的范围，什么叫生效的范围？

看那个图，那么浏览器保存库克以后再访问服务器的时候，它不是会把酷片发自动的发给服务器，cookie你需要指定我访问哪些路径我才会发，如果你不指定的话，他可能所有的路径都发，可能就会有问题，这个范围就太大了，对吧？

因为你每次都发 cookie，那么如果说服务器有些请求他不需要，其实一些浪费了一些网络的资源，所以我们要明确声明 cookie在哪些路径下有效，那不需要它的路径就让它无效，好回来怎么设置有效的范围也很容易 cooking，然后点set pass，声明它在哪些路径下有效，我可以这样写。

肯定的，那就意思说是在我整个项目下都有效，当然你也可以写这些下划线，比如阿尔法项有效，也可以这样我写个二，所以库克亚只有在这个路径和它的子路径下才有效，那么你仿别的路径就无效，这样明确它的范围。

还有那么浏览器得到酷k以后它是会存的，它默认是存到了它的内存里，那么当我把浏览器关掉以后，cookie就没了，当我再打开浏览器它就没有了，所以他在访问服务器就不会带 cookie。

那么如果说你想让它存的时间久一点，哪怕我浏览器关了，我第二天打开再访问服务器，这个顾客还在这还能够传给这个服务器也可以，这个时候你就要设置cookie的生效时间生存时间。

好，我再总结一下就是cookie默认发给浏览器，它存到内存里关掉就消失，但是你一旦给酷k设置了生存时间以后，它会存在硬盘里，然后长期有效，直到超过这个时间以后才会无效。好，这写故事就设置cookie的生存时间，也就是site cook，点site，max edge最大的有效的时间单位是秒，这里比如说我写的是60×10，对吧？1分钟60秒，那显然就是10分钟。

好了，那么其实我们平时对扣费的设置主要就是两项，基本上这两项是必须要设置的，然后就要发送cookie了，我就把 cookie放到risk boss的头里就行了。怎么放 Site？Response点不是site，是艾特cooking。好，把cookie对象添加到response里，那么最终像浏览器响应的时候就会带过去。好了，我就处理完以后我就 Return一个结果让它响应给浏览器，这里随便写我就写sat，cooking这个结果不是我们的重点。

好这就写完了，写完以后咱们可以进行测试，这就可以测了，我就启动一下这个服务器，好，启动完以后我告诉你怎么测，需要依赖于浏览器的插件，咱们得看那个插件，然后能看这个cook在没在，那我就打开浏览器，新打开一个页签，我去访问刚才的功能 community，Alpha cookie said好，我先不用着急回车确认一下，对这个路径，然后提前点检查，右键检查把 net work打开，然后我一发请求的时候，下面不能够看到请求的过程，我们在请求的过程里就能够看到 cookie它有没有？

好我就回去了，回去以后看 site请求点进去。

好，刚才我们话都说了，说这个请求是我们服务器创建的库被发给了浏览器，响应时它在响应的头里对吧？所以我们得看response，而不是request，你看 response，response headers它有一个属性叫sat cookie，这里边有一个code等于一个随机的字符串，我们产生的那个东西。然后 max600秒，然后最终他通过这个时间计算出了超时的准确的日期，时分秒就是有效路径。是对吧？就是我们设的这些东西好，说明我们 cook就ok了可以了。

可以以后我们接下来还得做一个测试，就是说浏览器在访问服务器的时候，它会自动的把这个数据发回来，我们把测一下，我再写一个方法，我写之前成这样，我直接访问另外的一个方法，不是阿尔法之下的，再写个页签，比如就访问 index，然后我们看一下 net work好回车，看第一个index这个请求，然后我们看一看 request，heads里有没有cookie，为什么没有？

因为我们刚才响应的 QQ 的有效路径是阿尔法，我们访问首页跟这个路径不匹配，所以对首页的路径就无效。

下面我写一个能有效的，这个也很简单，我就在这个地方再加一个方法，让浏览器访问，那么这个方法我还是这样复制一下我上面这个方法的访问路径是 get 吧，它位于阿尔法之下，夏季路径就有效，然后也是盖的请求，那么这个方法他那个时间我管他叫 get cooking 好，然后这个方法我就直接 return，随便说听个啥听 get。

Ok，完了好，那么我重新编译 CTRL f9 一下，好它自动重启了服务器，然后我打开这个浏览器，我去访问 get 这个结果没有问题，我们主要是看请求中携带的数据有没有看 request，你看酷片不就有了吗？对吧？当然有了库克我怎么用呢？这个库克怎么用？可能我们会在服务器里用，我这次请求你不是把库克传给我了吗？

我收到以后我可能在服务端区里面只是用一下，或者是服务器不用我我在那个模板上要用，我可以，反正在服务器得到以后把它给模板也可以，我们在这怎么去获取 Cookie 非常简单，因为 cookie 是返回来的时候是在 request 里，所以这里如果你声明 request 对象，然后点 get cookie 就能得到所有的 cookie，然后但是那样比较麻烦，因为你得到的是一个 qk 宿主，还得遍历从中去找，有点麻烦，如果说我就想要众多酷 k 中的某一个 k 所对应的酷 k 怎么办？

我们用一个注解可以实现在这注解了叫 cookie y6，然后你把 k 写上去，咱们刚才存的时候 k 不是扣的，对吧？K 写上去，然后写上一个参数，意思是说就这个参数，你要从酷 k 中取 t 为 code 的值付给他就行了。

然后这里面我就把参数的打印一下，看看能不能得到这个值，所以你得到了以后在这里就可以用，如果说你不想在这用，你想给模板用，那你就把扣的加到 model 里带给模板也是可以的，对吧？

好，那么我们在 CTRL f9 编译一下，这样我就让他不要再提示这个提示了，自动的去处理就好，处理完以后我就在打开浏览器再刷新一下，回到服务器我们看一下，你看控制台我们扣的值就有了，就是这样。

好了，那么 cook 就是这么用，非常简单，当然我们这次课主要是演示它的使用的语法，那么我们后面接下来要开发的功能登录，那么处理具体的业务逻辑的时候，到底在业务中怎么用，那就后面的功能在演示，那么了解了这个事例以后，我们通过插件也看到了它玩法以后，你回过头来再去体会它的意思，说 cookie 它是服务器发送到浏览器，并保存在浏览器端的一小块数据，因为一个 QQ 就能存一个 q56，它就存很少的数据，那么浏览器下次访问服务器时会自动携带这块数据，将它发送给服务器。

好，这就是 cookie 的作用，最终的结论就是服务器通过这样的手段，它能够识别浏览器，或者说它能够得到浏览器相关的一些数据特征，数据好了，那么 cookie 有它的好处是显而易见的，就是它能弥补 HTTP 无状态的这种情况，能够让业务得以延续，而且在使用的时候它很简单，对吧？很方便能解决问题。

但是其实它的缺点也比较明显，什么缺点呢？就是 cookie 这个数据是存到浏览器上的，存到客户端的，那么存到客户端的数据可不是很安全的，因为客户端是谁你不知道客户端它的系统的安全程度也远远不如服务器，它也容易被盗，对吧？

因此鉴于它的不安全性，你这个库可以不能存很敏感的隐私的东西，比如说密码，你就不应该存到这里来，否则就很容易被盗，这是第一个它不太安全。

第二个 cookie 会在很多请求当中，他都会主动的把数据发给服务器，他每次都发这个数据，每次就会访问服务器，就会增加数据量，所以说也会对流量对性能产生一定的影响，这是酷 k 的一个缺点。

那么其实还有一种办法也能解决问题，思路是类似的。



那么另外的一个解决问题的办法就是session，这个session是服务端对象，但它不是HTTP的协议的标准，所以你在 HTTP里面找不到具体的声明，它是Java ee的标准，那么你去奥瑞克的官网去查 Java ee的手册，它里面是有关于相关的 API的介绍，我们课上就不去细看了，然后它是用于在服务端记录客户端的信息，那和酷k刚好相反，酷k是在客户端记录，然后因为你数据是存到了服务端，所以说它就更安全，但是它也有缺点，就是你把数据都存到服务端，都存到内存里，服务端的内存压力就很大，因为你想一个网站的用户访问量是很大的，对吧？

你谁来访问都产生一个cookie对吧？

所以说内存压力比较大。

总体来说就是说如果说你这个数据特别隐私，你不能用cookie来存，你就可以考虑用session对吧？如果说不是那么隐私，其实一般的数据你用QQ可以服务器压力不就小了，所以说得看情况来使用它们要注意。

那么接下来我给大家演示一下，筛选它是怎么来玩的，对，在演示之前我还是再画个图，把这个筛选它的工作流程先说一下，说明白以后我们再演示。再一个。

其实30年使用的时候，它跟cookie还有一些联系，它其实本质上它是依赖于cookie的，所以这一点你要整明白，我们一会在演示案例的时候你才能看，看得出来这怎么有产生一个新的东西和QQ有关。先讲明白，然后再演示好。

我再画一个图，我首先再画一个方块，还是代表浏览器，然后再画一个方块代表服务器，还是解决的是它们之间发请求通信的时候，服务器能够记住他的问题，那么比如说第一次请求浏览器访问服务器，那服务器就会创建的不是 cook对象，而是一个session，对象服务器可以创建一个session对象。

好，然后筛选对象就存在服务器端，它不会把整个筛选对象发给浏览器，但是你注意，那么浏览器和服务器的关系是多对一的关系，很多浏览器都能访问同一个服务器，每个浏览器访问服务器都会产生一个筛选对吧？

当你下载浏览器，浏览器访问完以后产生了一个筛选，但他下次再访问的时候，那服务器怎么知道你浏览器和筛选对应，它和另外一个赛程对应，就浏览器和赛程之间对应关系怎么找，那这个就得依赖于cookie了。

总之服务器创建的session，用session存了当前浏览器或者说当前用户的相关的数据，包括什么ID密码姓名都存了，怎么让这个赛事和这个问题有个对应，他是这么做的，在响应的时候，那么服务器自动的像浏览器发送了一份数据，这份数据是通过cookie携带的，或者说服务器的底层自动建立一个cookie，然后在cookie中出了一个 session的ID，就是对象的ID，Session的ID是对象的唯一标识，那么每个浏览器来了创建一个新的session，这个session有唯一的标识，然后他把 v的标识从cookie发送给浏览器，那么cookie的特征刚才我们讲过了，浏览器它就会存这个对象在它本地那本地里就有了30ID，那么当它下次访问服务器的时候，库位的特点就是它会自动发送给服务器自动发回来，所以下次请求的时候发回来的就是他之前存的 cookie里边就是这个赛事ID，那服务器会得到这个赛事ID得到以后他利用这个赛事ID在他的内存里一找不就找到了，与浏览器对应的筛选，就能够找到与浏览器对应的数据了，就是这么一个原理，也是很简单很好理解的。

总之 session它就依赖于cookie，但是它的数据存到了服务器端，对服务器会产生一定的压力，但是更安全。

好了，那么了解了三审的就是工作的状况，或者说流程以后，下面我们再写一个小例子来演示一下session的使用，这个session其实它的使用是比较简单的，我在这个cookie示例的后面接着写筛选的示例，还是我说明一个方法来模拟这次请求，那么我们创建筛选往里存点数据，好，这个请求我copy一下pass，我叫session，site这次请求我们是要往里存数据，然后响应也是一个简单的字符串，加上了response包点，然后我写上他的死病 sat筛选，好，那么在这个服务器端创建session，其实他不用我们手动创建他和酷k不一样，服务器或者说是不是mvc可以自动的帮我们创建筛选并且注入进来。所以筛选

的使用和request, response和model这样的对象的用法是一样, 你只要声明, 那么spleen mvc就会给你注入进来, 所以我声明就好了, cookie对象的类型是HTTP session。

然后筛选因为是一直存在于服务端的, 所以它里面存什么数据都行, 它一直在服务端保存着。Cookie为什么就只能存字符串, 而且只能存少量的数据呢? 因为库克的特点是它来回传, 你要存大量数据的话不太影响性能了。

再一个你要把这个数据传给客户端, 那客户端你其他的Java类型它也不能识别字符串它是可以识别的, 所以鉴于它这种特点, 它只能存字符串只能存少量数据, 而session可以存任何类型的数据, 然后的话也可以存很多的数据, 好了, 我就随便往里存点啥, session点site attribute也是key value, 比如说我要出现一个ID, 假设是一筛选点, sat比如我再存一个name, 假设叫test, 好我就往筛选里存了两条数据, 最终我向量器响应一个字符串, 3筛选。

好, 写完以后咱们就测一下我CTRL f9, 那么打开浏览器我来访问一下, 这个请求也是要看这个插件, session set响应的没问题对吧? 很简单, 我们看这个请求, 然后我之前说了赛程的特点是在响应的时候, 它会通过cookie去传一个session的ID给浏览器存, 好, 你看有没有响应的时候, side cooking, 然后 jssid jssid就是sisid然后它的pass默认就是整个项目下都有效, 就这样一个状态。

然后没有 max age的声明, 没有最大生存时间的声明表示出什么呢? 这个数据存到了内存里, 但我浏览器关了它就没有了, 这样的。

好了, 那么我们再来看我浏览器下次再访问的时候, 会自动的传回筛选ID, 从而能够自动找到筛选得到这里面的数据, 那么把这个请求也进行一个演示, 我就在刚才的事例的后面再写一个事例, 我就copy一下, 那么这个路径我就get然后叫get section。

好, 这个方法里我主要的目的是要从筛选中取值, 这个再生也是通过思博mvc注入, 你也不用自己去声明, 你就从中取值, 我直接把渠道的值输出了, 筛选点儿, 存值是site取值就是 get attitude, 然后我们存的是ID和name, 所以我先把 ID打印出来, 然后再把 name叔叔看一下, 最后再随便的返回一个字符串。

Get session。

好, 写完以后在CTRL f9重新编译, 然后我再打开浏览器访问一下, get看一下这个部件这回看什么, 看的是请求当中有没有 sisid请求当中, 你看有一个cookie对吧? 接sisid有的我们看控制台, 我们从赛程中尝试打印的数据有没有? 应该也有e test。

好, 所以塞申也能够解决同样的问题, 只不过它是用另外的一个角度来解决的。然后的话优点就是数据存在服务端安全, 缺点就是服务器都会有一些内存上面的压力。好了, 那么再总结一下, 说我们要想让这个业务连续要想记录浏览器的状态, 我们到底是有使用cookie还是筛选, 这得看具体情况而定。那么总体来说咱们能用cookie就尽量用cookie减小服务端的压力。再一个如果是没有必要存就是这样去存的, 你就不要用有必要的再去存。

然后 Session的话其实如果是一个单体的应用, 就只有一个服务器, 我们将来只部署一台服务器的应用, 你随使用, 筛选没有问题, 但是现在因为每个网站都比较大, 用户数都比较多, 其实筛选逐渐越来越少了, 那么这个是和部署有关的, 和分布式部署有关的。

我给大家再解释一下, 为什么在分布式部署下赛程用的少了, 出现的问题是什么? 我们在实际的应用中我们怎么去解决这个的话, 你一定要多了解, 可能是面试的时候很容易会容易问你, 因为你把你的开发的功能一解释, 说你是用这样的方式去处理这样状态数据的, 面试官肯定会问你这方面的话题你为什么不用三审对吧? 所以这块你一定要注意。

这里我也画个图给你解释一下, 就是说首先我们分布式部署的时候, 为什么使用三神会有问题, 会有什么问题? 好, 那么首先还是画一个框, 它代表的是浏览器, 然后我再画一个小的框, 代表的是服务器, 然后咱们这个服务器往往是分布式部署, 所谓分布式部署就是我部署多台服务器同时跑这个应用, 同时向浏览器提供支持, 因为浏览器的请求实在是太多了, 用户太多了, 如果你一台服务器撑不了这些流量就多台服务器, 分布式部署分别是123我们一共有三台服务器, 三台服务器浏览器访问谁?

其实对于浏览器来说这个服务器分布式集群它是透明的，因为我们会在这个服务器之前再加一个代理，或者加一个负载均衡的服务器，通常我们会用NDS后面的课我们也会介绍，然后你浏览器访问的不是这些服务器，而是访问这个统一的代理Ajax，当然这个Ajax它也未必就是一台，它也可以是做组成备份，我们就不说它的群的情况了。

好，那么当浏览器发出请求访问这个服务器访问 Nds的时候，那么它会做负载均衡，它会去分发请求分发给哪个服务器，这得看策略，比如说我们是按照比较均衡的策略，就哪个比较闲我就分发给哪个。

假设这次请求是服务器一比较闲，那么恩杰斯就把请求发给服务器一去处理，这个时候服务器一要处理这个请求，它就可以，比如请求需要后面需要业务连续，它可能会创建一个筛选，服务器一就会创建一个筛选，然后当然会返回筛选ID给浏览器，它会存了。

那么当浏览器第二次再去访问这个服务器的时候，它又发一个请求给ndx携带了一个section ID但这回 n 这个是因为它是负载均衡，可能这个时候服务器一处理的业务比较多，而服务器三比较空闲，他可能就会发给这个服务器三去处理这个请求，你就问题就出现了，那服务器三当然也可以创建session，当然也可以识别三神，他一看你给我传了一个三神ID，但是我本地没有，因为这是这个服务器，三里没有存筛选，对吧？

他没有他只能发现一个新的筛选。因此你看我第二次请求，我想得到三分钟数据就得不到了，分布式部署就是这样的问题。

好这个问题怎么解决？其实有多种解决方案，一种方案就是我们去设置负载均衡它的分配的策略，一种方式叫粘性三审，就是说比如浏览器的IP是假设是101，我把101这个IP分给服务器一去处理，当你下次这个IP来了，我还得分给服务器一处理，就是说你只要是一个固定的IP，我就永远分给同一个服务器处理，这样的话 Session没他就能得就能够得到上次的 session，因为每次都是同一个夫妻处理请求，这种方案叫粘性筛选，但这种方式显然有问题，就是你同一个IP分给同一个固定的服务器，那就很难保证这些服务器之间是负载是均衡的，对吧？

因为你不是平均分的。

好，所以说这种方式有它的问题，就负载并不均衡，性能并没有那么好，所以说这是第一个粘性筛选不完美。

第二个解决方案是同步筛选，当某一个服务器它创建筛选并且存在数据以后，那么它会把筛选同步给其他的服务器，其他的服务器，那么这样的话所有的服务器这个session就统一了，浏览器无论访问是哪个服务器，这个服务器串联出来的session以后都会同步给别人，每个服务器里都存了你浏览器的一份三星数据，这样也可以，但这样的话也有问题，首先你得做同步，你一台服务器处理完以后得同步给很多台，这个会对服务器的性能产生影响。

第二个这样做的话，这个服务器和服务器之间就会产生的关联，它们之间就会产生一些耦合，而不是那么的独立了，对部署的部署的话比较有影响，所以这种方式也不太理想。这是第二种方式，叫同步筛选也不是很好。

然后第三种方案叫共享筛选，什么叫共享？开始就是说我单独搞一台服务器，这台服务器不是处理你普通的业务的，它是专门用来存筛选的，就是说如果说浏览器访问服务器，那么需要创建赛程，需要获得赛程的时候，我们就把这个筛选统一交给放到这个服务器里处理，然后别的服务器都像这台服务器去获取筛选，他们都像这台服务器去申请筛选，是这么一个关系。

他专门处理筛选好，这种方式似乎能很好地解决问题，但是它也有一些隐患，是什么？这台服务器因为它是单体的一台服务器，万一说这台服务器挂了就糟了，这些服务器都依赖于它挂了，这些服务器都没法工作了。

所以我们当初之所以要分布式部署，是为了解决性能的瓶颈，你现在的的话又搞出一个单体的服务器解决一项问题，万一他挂了他也是瓶颈，对吧？也会有很大的影响。

所以说这种方案也不太好，但如果说你要是搞个集群的话，那和我们直接这三台服务器各存一个筛选也没什么区别，它俩之间又得共享，还是有问题，好，所以这种方案也不好，共享财产。

那么其实目前比较主流的办法是这样的，我把塞我把客户端的身份的数据我就不存到筛选里去了，我能存到库克里就存库克里，有些敏感数据不方便存到库克里怎么办？我就存到数据库里。

好，那么数据库你可以做一个集群组成备份都没关系，他们之间的这个集群它是性能是ok的，他们之间同步数据没有问题，数据集群这个方案是很成熟的，那么所有的服务器都可以访问数据库的集群来得到就是关于客户端的会话的数据，这样就可以。

当然这种方式也有它的缺点就是说因为传统的关系型数据库它是把数据存到硬盘里，我们像数据库的硬盘去访问数据，显然是从内存中和从内存中读数据比它是性能更慢的，对吧？这个可能是在并发量大的时候也可能会出现一些瓶颈。

所以这是它的问题，尽管说它能够很好的共享数据，很好的同步数据，但是它性能可能没有存到内存里好，但是你要知道现在数据库可不只是关系型数据库，现在的话 no circle数据库的技术也是非常的发达了，非常的成熟了，也很好用，那么我们可以把绘画的数据不存到观音数据库里，我们可以存到no circle数据库里，比如说readiness，所以说最终我们把这样的数据存到reds里，这是一个比较推荐的解决方案，也是一个比较方便比较好用的解决方案。

所以我们的项目当中最终会是用这样的方式，但是由于我们一开始现在还没有去学 radius，所以现在我们还不能把会话数据存到这里来，怎么办？对于有些数据它适合存到my circle里，就存在my circle里边，有些数据不太适合存到my circle里，我就先存到section里，因为我们现在是单体应用存到section里，反正也能玩转。

等我们学完了 rids以后，我会对代码做一个重构，把这个存到my circle里的这样的数据，或者是存到session中的数据再迁移到rides里就ok了，这是我们后面的话关于 section方面我们使用的一个小的安排。好了，那么这次课我们把绘画的管理介绍到这儿，咱们下次论坛再见。

## 2.4 生成验证码

---

这次课咱们来学习，如何生成验证码，也是为了我们后面做登录的功能做一个准备，因为咱们登录页面上不是有一个验证码需要随机生成对吧？所以这次课我们先把这件事搞定。那么生成验证码，咱们可以用Java swing在后台的内存里头去画，但这样会比较麻烦。那么也有很独罩成的工具可以直接拿来用，其中用的比较多的就是叫卡片。

然后我们这次课就给大家讲一讲我们如何利用这个工具，然后在服务端内存里画出一个验证码，其实就是一个图片，然后怎么把它发送给浏览器，怎么显示在登录页面的对应的位置。

好，当然我们使用这个卡片主要是有这么几步，第一步你得把包搜索到，你得把包导入进来。

第二个我们需要针对他们写一个你要配置好，这个图片你要生成多宽多长，然后上面有几个字，这个字的范围是什么，什么颜色有没有干扰的点或线等等，主要是做这么一些配置。

然后第三个就是使用卡普查的API去生成我们想要生成的东西。

这个验证码其实生成它包含两部分内容，一个是它的有一个随机的字符，我们先生成这个字符，然后我们而且服务端我们需要把它保留下来，然后再根据这个字符生成图片是分这么两小步。

那么关于卡我们在学习的时候可以参考它的官方的手册扣了点Google的com，然后这么一个路径，我们可以去看一下，我这儿已经提前打开了，但是我说一下这个网站，你需要通过VPN才能访问，或者说你需要科学上网才能访问，默认是访问不了的。

然后我这里已经打开了，那么因为它是一个小工具，所以说它的手册写的比较粗略，不过好在关键的内容还是有的。

打开这个网站以后，这有一些介绍，我们不细看了，非常简略，然后具体的内容可以看 Weeks，然后他这里有说 how to use 怎么用，有一些介绍，有些例子好不细看了，然后这儿有一些详细的配置的说明，后面也有一些例子，但它这个格式就挺难看的，我们课上就不细看了，或者说我一会讲的时候就直接给你一个事例，你就参考我的事例。

如果想看的话，反正费点劲也可以做一个参考。最后还有一个因为我们是 spring 框架下去使用这个卡片，spring 可以对它做整合，对吧？在 spring 之下我们使用它怎么办？看网页也有一些介绍，也有些例子，但它格式都很糟糕，我们就不想去看了，我就直接给大家做演示好。

那么回到 PPT，刚才我说了，我们想使用这个工具，第一步你倒包，对吧？好我就去搜一下包，我访问没问 repository.com，好，然后我就搜工具的名字 Capture kapt cha 然后收到以后是好我们点进去，它目前就只剩一个版本，2.3.2 我就选这个版本，然后我要把这个代码来拷过来，粘贴到我们项目之下的 pom 的 xml 之内。

然后稍微等一下，因为它需要下载咱们稍微等一下。好，当它这个颜色由红变白以后，就证明咱们已经下载好，已经在项目中依赖了包就可以使用了。

然后在使用的时候，你看这个包它就是一个普通的包，他这个包名不叫死人，不他如何，所以说 spring 没有针对它做整合，没有给它做自动配置，因为这是个小工具，然后我们需要自己对它做一些配置，我们可以写一个配置类，然后配置好以后把配置类，然后加载 space 容器可以对它做一个初始化。

好配置类我们项目中都是写到 can 目录下，那么我新建一个类叫做差肯定配置类我们之前也讲过，我们首先需要在类上写一个注解，叫 configuration，比如说这不是一个普通类，这是一个配置类，然后我们可以通过并注解声明一个 BIM，然后这个病将会被什么容器所管理所装配。

我们要装配的肯定是工具组件的核心的代码核心的对象，那么卡普查核心的对象它是一个接口，这个接口我们可以搜一下，叫 producer，producer 这就是它的核心的接口，接口中就两个方法，一个是 create text 就是创建验证码文字，create image 创建一个根据文字创建图片，得到的是一个 8 分的 image，就是一个图片，然后这个接口它有一个默认的实现类，我们可以直接拿来用，你可以点这个箭头，它会跳到默认时间内叫 default capture。

所以我们配什么？配实例化接口，说白了好我就写上 public 要 16 号的接口，所以我返回的是 producer，接口类型，然后方法名就是 bin 的名字，我叫卡片，突然就是走，然后在这里边我们实例化的得是它的实现类，刚才我们也看到了是 default。

Default capture 记错了，new default capture。

然后我们需要给它传入一些配置传入一些参数，这个参数我们可以把它封装到一个 config 对象里，can be 对象。

好，那么 can 这个对象要求我们传入一个 practice 对象，product，对象里存的都是 k 外的，其实就是个 map 里面存的就是参数，然后我们再把这个卡片赛得很废，把可能 site 给咖啡主见，然后委托这对象就可以了。

好，所以所有的配置项其实都是通过肯菲克去配的，而肯菲克它需要依赖于 practice 对象，所以我们在此之前还得自己在室内化一个 practice。

当然其实 practice 中的数据，咱们可以通过从 practice 文件里读，对象其实为了封装 x 文件中的数据的，然后当然我们也可以自动不要不在配置文件里写，就在这实例化也可以，我就直接实例化直接往对象里塞直了，因为卡普沙特 k 比较长，他文章写得也不好，所以说我就不到处去写他的 key 了，否则很难写很难记，就在这一个地方写就算了。

好，practice 等于 new practice。然后我们需要往 Practice 里晒它一些 k86，但这个 k 我一写你就能猜到什么意思，我就写一下差不多不要写错了。点 image、图片，点 wise 宽度，图片的宽度，我把它设置为 100，它的单位默认是像素，kaptcha 没有问题，好同理再来一个参数，key 不好写，官网它给的代码又格式不好，所以说我就直接在那里写，而不在 ps 文件里去配，出于这么一个目的你也看到了就很麻烦。

好，第二个给一个高度，我希望高度图片的高度是40。好再来，这个得改一改。Capture点text。Producer found size。我检查一下text。Producer。Phone的size没有问题，那么k是我们从官网它的例子中去看的好，这是字号送的字体SARS字号大小对吧？多大？我希望这个字是三十二号。

好，再来capture点text。Producer点儿phone。

点儿卡了，这儿的颜色我写的是000，其实你这000表示的是黑色，你写个单词black其实也是可以的，或者写16进制的形式其实也是可以的，其实颜色计算机显示器的颜色，它的配色方案是基于2GB三原色红绿蓝，我指定这三个数其实是红、绿、蓝三个颜色的值0代表的是最低的值，就是说白了像素点的小灯灭了，三盏灯同时灭，什么颜色都没有，黑的就是这样，反正这个次要的，你就了解这么写，黑色可以了。

好，然后再来卡片儿点儿，test不是留着点儿。不是我们这个差叉点儿steam，咱们要生成验证码，他的有几得有随机字符，那随机字符的话就是随几个字符拼在一起，我得给一个字符的范围，在哪个范围内去随这里给一个字符串，他会把字符串拆成字符，然后去随你随便写，包括汉字其实都可以，但这里我写的就是通常是这样就可以了，0123456789，再加上所有的英文字母就可以了，好，当然我这个英文字母都用的是大写的，你用小写的也没关系，好再来一个差烂死，你要生成几个随机字符对吧？

这么多字符你要生成几个给一个长度的限定，4个通常就4个，你太多了，烦太少了，又不安全对吧？

好，最后再写一个Captain点nice input，后面就没了，就是说这个意思是说你要采用哪个干扰类，你要采用哪个就是噪声类，所谓的噪声是什么？

它生成的图片上面给你加一些干扰，加几条线，加几个点，做3d做拉伸做阴影等等，就为了防止机器人暴力破解，通常其实我们不用它图片生成的很好，默认防破解的它就有一点变形，所以其实你不加那些个干扰也是没关系，因此我们用这个内容com点我们可以搜一下，可以搜一下，要不报名不好写，他内叫做no nice。

用的就是就没有噪声没有干扰，我主要是把报名拷过来，然后报名点。

No，nice好就可以了，配置写好以后你把它传给config就行了，这个病就在服务启动的时候就能够被自动的装配到容器里，那么我们通过容器就能得到一个producer实例，我们通过它的实例，刚才不是看到它有两个方法就可以创建验证码就能创建图片对吧？

好，那么完成以后，下面我们就用一下试一试在哪里，用登录页面对不对？登录页面我就打开log in controller，我在这里用一下，然后你注意登录页面我在哪个方法里写，有人可能会想我是不是在log in是打开登录页面，是不是在这个范围写的？不是。这个方法目的是给浏览器返回一个html，而这个html里会包含一个图片的路径，浏览器会依据路径再次访问服务器，获得这个图片，所以这件事我是一开始讲，HTTP的时候给你介绍过对吧？

所以我们需要再单独写一个请求，向浏览器返回图片，当然请求会在这个模板里去引用它的路径，它是这么一个关系。好，我在这写一个生成验证码的方法，那么我先声明了它的访问路径，就叫卡片，然后master因为这是一个获取图片，是一个获取数据的方法，所以用get就可以了，我写错了。然后这个方法注意我返回的是why，为什么？因为我们现在浏览器输出的是特殊的東西，是一个图片，不是一个字符串，也不是一个网页，所以这个方法比较特别，我们需要自己用response对象手动的相当的去输出，不过也很简单，所以我就返回word，然后方法名我叫get capture这里得用到risk boss，我先把它声明一下。

好，另外就是我们生成完验证码以后，服务端需要把它记住，那么当你登录的时候，你再次访问服务器的时候，我好验证说你这验证码对不对？这个验证码不能够存到浏览器端，否则就很容易被就被盗取了，这个是个敏感信息。

要存到服务器端，在多个请求之间要用这次请求我生成存进去，然后登录的时候我再去使用，所以说要跨请求的，我们就可以利用了我们上课所讲的库克或筛选，因为它是敏感数据，我们存到筛选的比较适合，所以这里我就获取筛选好使用。

好了，那么这个方法之内就很简单了，首先我就是生成验证码，当然我们生成验证码需要使用配置类所产生的币，这个币我们需要通过容器获取，并且注入到当前的币之中，我在前面写一个注入奥斯维尔，它的类型是 producer 对吧？名字我叫 capture。Producer。好，我要使用它去生成验证码。非常简单，Create text。这句话就是生成一个字符串，它根据我刚才的配置会生成一个4位的随机字符串，范围是从这里随。

好，那么所以我们这句话执行完以后，我会得到一个字符串，我就叫 txt，得到以后我们再利用这个字符串去生成一个图片去画一个图片，点 create image，你要把 text 传入，它会生成一个与之对应的图片，这个类型是 8 分的一面。

好，那么我们生成的这两个内容做不同的处理，首先我要把验证码文字存入筛选好，在后面的请求中使用，写注释我要将验证码存入筛选，你给它取个名字我就叫看不上 txt 的文本，那么对于图片我要把它直接输出给浏览器，将图片输出给浏览器，那么人工的输出其实我们之前在阿尔法 CTRL 里也讲过，首先你要 response 点 set，空天的 tap，你要声明浏览器返回的是什么格式的数据，我返回的是图片，而且我返回的是编辑格式的图片，就这样写。

然后瑞斯 boss 它向浏览器做响应，我们需要获取它的输出流输出内容，我们从 response 里获取输出流点，get 你可以 get right，这是字符流，你也可以 get output、stream、字节流、图片，我们显然用这个比较好，out to stream 不玩是吧？

那么流我们需要对它捕获异常做这样的处理，我就搞错了，我应该是把它针对它生成拆开去，就可以点客户的手上的位置，这个快捷键但是我没记住，我就手点了，将它用 track 开始包围，然后我们想向浏览器输出图片，还有一个工具可以用 image io emit io 然后点 right。

好，你要输出什么？要输出的是那个图片，就是它对吧？然后输出的格式得说明是编辑格式，用哪个流输出？是 os 输出流去输出就可以了，那么这个流你可以不用关闭，因为整个 ris boss 是由 Spring mvc 去维护的，最终它会自动会关，我们拿来用就行了，不用我们关。

然后当然如果说你万一捕获到异常怎么办？我们只能记一个日志就没办法解决，我们默认认为它没有问题，有了问题一定是服务器就出现了什么问题我们也解决不了。所以打个日志日志我还没有声明，我在前面再补充一个创建一个日志对象 log，好，然后以当前的类名来命名。

行了喂哺过到异常以后记一个日志 Logo 点，Ehle 说响应验证码失败，然后把失败的具体的信息做一个记录，好到这个方法就完成了，完成以后咱们可以测一下，你直接浏览器访问它，他就给浏览器访问一个图片就能看到，咱们到这去测一下，好，我把这个服务启动一下，启动以后打开浏览器 local house，然后 community 这个路径是 capture，你看起来就是这么一个图片，tukv 然后它是一个做了一个类似于那种玻璃的那种纹理的那种幕后的处理，所以说本身也是防破解的，你每次刷新会得到一个新的效果还不错，就这样。

当然里边的文字的范围，还有这个字的颜色大小你可以调，但我觉得这样就可以了。

好，这只是我们测了一下说这个方法没有问题，没有问题之后，我们需要把它应用到登录页面上去。

好，我就打开登录页面，是 log in 点 html 我们之前已经把它简单的配置过，并且能够访问到它。目前访问一下试试，目前访问它图片是假的，我们需要把它改成动态的对吧？就改成我们刚才的路径。

好，找到它找到了就在这儿，之前我们写的路径是这样的，是一个固定的 PNG 图片，现在我们需要把这个路径改成我们刚才的动态生成图片的路径，那么它的路径刚才那个路径就叫 capture。

好这样就行了，我们 CTRL 还不久，重新部署一下，然后打开浏览器刷新一下，你看它就变了，我再刷一下，每次刷新就变。当然你不能这么刷，我们肯定是希望点超链接的时候它就变，但现在超链接是死的，还没有处理，我需要处理一下，每次点它都变，这个需要用 js 来实现。

你回答回过头来，你看刷新验证码的超链接，目前其实他什么都没做，他空的，它的声明，这不是一个链接，这里面是一段 GS，阿扎的冒号分号，但是里面什么都没有，所以没有做任何的处理。

我要做的就是说我吧 GS给它实现了，我需要写一个js方法，我可以在这声明一个，比如说我要刷新验证码，我可以叫reference卡片。这句话的意思是当我点这个链接的时候，我不是去访问一个链接，而是要调一段js，我要调这样一个GS方法，而这个js方法要实现我将来要实现的逻辑，就这么个意思。

好，接下来我就写这个方法，把这个代码复制一下，我在这个页面的结尾处写就在这写。

好，那么当然我这个方法是直接不是用原生的GS来写，我是用解快瑞来写，而当前页面已经引入了解快瑞，我们整个页面是用boost来做的，它就基于这块儿，我就把这个方法做一个声明，然后方法之内最终要做的事就是改变刚才图片的访问路径对吧？

用解快实现是非常简单的。刚才的图片我看看我得给它加一个ID，我好引用它目前还没有ID对吧？加个ID，这个ID我给它取名叫 captain。好，那有了ID以后，我要改变图片的路径，首先这个路径是什么？先声明一下， Pass等于 community capture。

大概是这个意思，但是这样写其实有点问题，第一个问题是这个项目的访问名，你这样写不好，因为万一将来我们要改变这个项目的房名的话，你这样写就写死了，你程序中到处都这么写得，到处都得改不太好，最好把它做成一个可配置项，我们去引用它 GS的话，我们就把它配到 global里，因为它是全局的GS对吧？

我们把它配到这里来，我搜一下 logo点GS在这个路径下你可以直接点，这里边有些代码不用管，我在一开始加上说明一个全局的变量叫因为其实这是一个常量，在程序运行是不会变，我用大写的 content pass应用的路径等于挺妙的，然后格罗波点GS在任何页面都引用了，我们直接在任何页面上都可以引用变量，回到刚才的页面，这个地方就加一下，我就把它去掉了。

好，context pass加上capture。

然后其实还有一个问题，那么你这个路径你这样写其实和这图片上是一样的，但是图片上我们没有写 community，你注意艾特大括号的作用就是补充 community，所以它将来也是有的，这个路径是 community culture，然后你改的时候还叫community culture，其实相当于没改，那么有些浏览器它就比较智能，他一看你访重新访问服务器要获取一个图片，他认为这个图片是静态资源，而你获取的路径你又不变，所以他认为你就没什么变化，他就不去给你访问，他就偷懒了，这种情况我们需要做一个善意的欺骗，我们欺骗一下浏览器，让他误以为这路径变了，怎么欺骗其实很容易，我们加个参数，这个参数每次点都变就行了。

这个参数只是欺骗浏览器，让他误以为路径变了，但服务端其实对他不用的，参数没有用的，对服务器来说我就随便加个参问号，p等于等于什么？我每次我或者加一个时间或者加一个随机数，其实都可以再加个max点 render，每次一个随机数都变就行了。

好，上面的路径以后我要改变图片的访问路径，其实改它的src属性，我要通过ID得到这个图片对象，改它的src属性，用解快很容易实现，用它的选择器并号capture ID，就是ID选择器选中了那个节点，然后改它的属性attr是h built单词的缩写，改属性改什么属性？Src属性，把它的属性改成什么值？Pass就完了，好，这就写完了，写完以后测试一下也得重新部署，可能f9，然后打开浏览器，刷新一下页面，鼠标放上去，你看左下角它 JavaScript的代码变了，掉了 reference方法。

好我点一下。行了，每次点的时候，他把路径一改，那么浏览器一看路径变了，就访问服务器就得到一个新的图片，这样就可以了。好了，那么通过卡片这个工具生成验证码，我们这次课就演示到这里，下次课我们会在登录功能中的就是做更深度的去使用它。

好，这次课我们先讲到这里，咱们下次课再见。

## 2.5 开发登录、退出功能

14-开发登录、退出功能.mp4



这节课我们来开发登录和退出的功能。那么登录功能它有两个请求所构成，第一个请求就是我们点头部的登录的链接，然后能够打开登录页面，这个请求我们已经完成了，上次课我们在做验证码的时候已经把它实现了。

然后第二个就是我们在登录页面填写数据，然后再点立即登录，那么这个逻辑我们也很容易能想到，后台应该是对填的数据进行一个验证，如果验证的是ok没问题，我们需要生成一个凭证，然后发送给客户端，这样的话好记录登录的状态，让状态能够在多个请求之间能够连续，让业务得以连续，当然这里就需要用到 Cookie了。

另外如果说你没有成功，某些数据是错的，我们还要回到登录页，然后在这里给一个提示，其实和注册是非常像的，处理的方式，只不过它对数据的处理是有一些区别。

那么退出的话就一个请求我们点这个地方展开有个退出的链接，然后你点退出以后，我们需要把登录凭证改成一个失效的状态。之前是登录成功了，现在要改的失效，然后的话还要跳到这个网站的首页，这是我们的一个要求。

好，那么咱们就一个请求一个请求来实现，我们先来处理这个就是登录验证数据的环节。

这里我要说一下，我们生成的登录凭证最终是要发送给一个key给客户端，然后让他记住下次再提交给服务端，我好能够就是识别你，但是登录凭证中包含了一些敏感的数据，包括用户的 ID、用户名甚至密码，如果说这些数据不能发送给客户端，要存到服务端，我们可以用三星来存，或者是也可以存到数据库里，这里我就把它存到数据库里，my circle里将来我们还会对它进行一个重构，把它改到存到 Rise 里，这是后面我们要做的事情。

好了。

我们开发之初，我们先来去看一下登录凭证的表，看它的结构，然后我们怎么存，我打开了这个表，它的表名叫log in ticket，然后这个表现在它里面是空的，什么都没有，然后有这么几个字段，一个是ID组建这没啥好说的。第二个是优质ID，用户的ID，因为你存的是用户的凭证，所以说你得识别是哪个用户，所以这个也很好理解。

第三个是txt就一张票，一个凭证其实就是一个随机的字符串，不重复的一个唯一的标识，我们要存的其实就是一个关键就是这个字符串，然后 Status状态，你这个评论是有效的还是无效的，其中零表示正常，一表示过期，我们看一下备注对零表示有效，一表示无效。

然后还有一个这是日期XP的，然后过期时间什么时候过期凭证，因为你不可能说把凭证永久的保存，它总有一个时间，哪怕这个时间长也需要有一个周期。

好，就这么几个字段。那么我们开发的时候还是按照以前的方式，我们先写数据访问层，然后再写业务层，再写表现层，数据访问层的话肯定我们得先把登录凭证相关的操作给它实现了。

好，我们就先写这一块，我打开这个idea，然后首先我得写一个实体类，好封装的这张表里的数据，然后的话再写它的征迁改查相关的逻辑。首先我在安铁的包下新建一个类和表面相对应就叫劳力 Txt。

然后刚才我们也看到了，它里边有几个字段，我就增加几个属性与之对应，分别是ID，然后是user ID，好，然后是一个ticket，就是一个字符串，还有一个状态，最后是一个日期，就什么时候就时间到了。

好，这是到期的日期，然后我要给这些属性生成盖德赛的方法，再给它生成一个兔子菌方法。好了，实体类我就写完了，没什么好说的，反正也没有什么业务。有了它以后，接下来我们就开始实现数据访问的逻辑，我就在 Do包下新建一个接口，接口的名字我叫love in。Take it。

好，那么我需要声明一个注解，map注解，比方说这是一个数据访问的对象，需要有荣幸的来管理。

然后。

我们的业务当中需要这么几个方法，第一个是你增加这条数据，插入一个凭证对吧？你登录成功以后要插入一个凭证，所以这个方法先声明一下。

那么增删改的方法通常就返回的是整数影响的行数，方法名我叫做insert log in。可以，那么参数很显然应该是实体类对吧？Log in KK好这是插入一条数据。

然后后续你。

插入完以后，我们肯定得利用这个状态，肯定得在某些时候把它查出来，因此我们再加一个查询方法。

那么查询的关键是什么呢？是依据什么来查？是依据 Ticket。因为整张表 t 都是围绕 txt 来设计的，txt 是凭证，它是核心数据，我们最终就要把 take 的字符串发送给浏览器让它保存，然后其他的数据是我们服务端自己要存一份，或者说我们服务端在数据库里存一份，是这样的，得要发给客户端。

因此客户端他用 cookie 了，除了这个数据以后，他再次访问我服务器的时候会把这个 txt 给我，我可以利用 ticket 查到整条数据，那就知道这是哪个用户在登录，这是哪个用户在访问，是大概这么一个逻辑。

所以我们这个查询是以 ticket 为条件的一个查询，那么 ticket 它是一个唯一的标识不能重复的，因此我们查到的是一条数据 log in ticket，这是反馈值，那么方法名我叫做谁？

Let bye take it，然后参数就 take it 好了，最后还要加一个方法就是修改凭证的状态。因为当我们还要开发，当你退出的时候，这个凭证就应该失效，对吧？

那失效其实一个是你把它删掉，第二个是更改状态，我们采用第二种方式，实际上我们在互联网行业，我们真正删除数据的时候很少，通常都是改个状态，这个状态就表示它失效了，作废了，但是我们留一个底，那么将来我们想统计的时候可能会有一些用处。

好，那么修改状态返回的类型是 int，然后方法名我叫做 update stayed。

那么修改的话也是条件是 txt，你把这个 txt 给我传进来，然后告诉我你要把它改成什么状态，我就给你改好了，这三个方法我们就声明好了，我们的整个的业务当中其实就需要这么三个方法，别的就不用了，然后就要实现这个方法。

我们之前去实现 map，都是在目录下新建一个配置文件，然后写 circle，这种方式我们经过之前的练习，大家应该也比较熟悉了，那么除了这个之外，我们还有一种方式，在 map 类当中写结果当中写注解，通过注解去声明这个方法对应的什么？

Circle 也可以搞定，那么我在 map 里就用注解的方式给大家实现一下，那么两种方式你都会这样的话，无论是企业用的一种方式你都比较方便。

好，那么这注解也很简单，比如说我这个是一个音色的方法，要执行的是音色的语句，那么你就用音色的注解，然后小括号里面写一个大括号里，你可以写多个字符串，随便多少个都可以，然后他就是把这多个字符串拼成一个 circle，就是 circle 你可以拆成多部分去写，它会帮你拼到一起。

同理这是查询，你就用 select 注解也是类似的，然后这是 update，你就用 update 的注解也是类似的，就里边可以写多个字符串，最终它会帮我们拼成一个 circle。

好，当然了我们这里头没有删除的方法，如果有的话你也应该知道它肯定是 delete，那就是 delete，好了我们一个来实现来把 circle 写出来。那么首先写我习惯于把换行，这样看起来方便一点，然后插入 Insert，into log in。Take it 我们写这个字符串的时候要注意不要写错了，因为写错了它是不会有提示，不会有检查的。好，然后我要插入的是哪些列呢？这个组件不用，因为它会自动生成，第一个字段的是 uzid 然后是 txt 的凭证，然后是状态，然后是过期的时间，你也可以后面接着把最后写完整，但你这样的话，把一个比较长的 circle 写到一句话当中，不好看对吧？

所以我们可以再换一个字符串，另几样再换一个，所以他为了你书写方便，他就允许你写多个字符串，到时候他帮你拼，然后我们写的时候需要注意的是什么呢？你在一每一行字符串后面最好加一个空格，这样他拼 circle 的时候，这两句话之间有一个空格断开对吧？

不要挨在一起，那样容易出问题，养成这个习惯就好了，反正每句话后面加个空格好了，insert into 这些列，然后 Values y64 里写什么？就是写表达式，这表达式引用的是对象中的参数属性，然后作为 Circle 的条件。

好，那么这里第一个是uzid那么对应的属性就是uzid属性。所以你会发现 circle其实我们在这写和我们在xml里写基本上没区别，就是格式有所不同，但是道理都一样。所以这里我不多的解释了，你写一遍就明白了。那么第二个是txt是对象中的txt属性，以此类推，下一个是所得的。

好，再来最后一个，x pair，好，这就完成了。然后还要注意一点，这是一个音色的语句，我们希望它的Id是自动生成的，对吧？那么这个里面需要声明一下，不然默认的话它不会自动生成。我们之前写xml的时候不也是做一些声明，这里也需要声明，声明 circle相关的一些机制，我们需要用到这样一个注解叫options。

然后我希望组件自动生成，我需要在注解里声明就是use它有提示你看 use generate case是否自动生成组件等于two，还有自动生成完组件以后，它会把生成的值就是边注入给对象，注入给哪个属性，我们需要通过key property来指定，那就是ID对吧？就是这样的。好了，你看这个insert语句就写完了，这样确实比较方便，因为我们少写了一个xml文件，但是它的缺点就是说色块如果复杂的时候，就看起来没有在x三秒中那么舒服，因为毕竟它是分了几个片段片了，分了几个字符串显示到这儿了。

好，所以总之各有利弊书写起来，简单阅读起来，麻烦你选择哪一个都是有道理的。好，然后我们再写这个查询，查询我也是这样给它断开，然后就是谁来ID。User。Id。然后 Ticket。然后是status，最后是 aspect，养成习惯空个格。好，然后另起一行再接着写。From老b txt给个条件，我们这个查询是以txt为条件，我就where txt等于参数提给他。

好了，查询的语句也很容易就写完了再来。这个是不对的语句。好，这个和音色其实大同小异，我就直接写了update。Love it。Take it。Set status等于这个参数status，然后你看跟一个where条件对吧？Where ticket等于贴给他，但是这样就可以了，然后我想说的是什么？

我们在注解里写 circle的时候，其实它也支持动态的circle，我们之前在配置文件里有的地方用到了动态出口，比如说用到了if，那么注解里也可以写if，这样的动态的出口怎么写呢？我就在 update的语句上给他立案是吧？尽管说 Update的语句它并不需要动态，我就假设他需要我就给你演示一下怎么去做，我需要写一个动态的circle，是这样的。

首先我在这前面再补充一个字符串，这个字符串我这样写 secret，然后在这个字符串之后再补充一个字符串，square的标签闭合这块加个空格，SQL的表示脚本，那意思说这里边是脚本，然后我在SQL的标签之内就可以加 if标签像if或者你可以理解为将if的脚本，我可以这样写，if开始，然后你还得闭合，if结尾，然后if上你可以写test，等于等于的是一个条件，但我这里写的是双引号，它不行，你需要进行一个转译，然后双引号之内你要写上一个判断的条件，假如说我判断 take的是否为空，比如说take it不等于no，我就再拼一个条件，你写到你会发现 if这么写其实和在xml中写是一样的，对吧？

所以其实它就是和xm中的语法是一样的，但是如果你想写if这样的标签的话，你前面必须套上 script，里边才能写这样的标签，这个要注意好。那么如果txt不等于now，我要怎么样？我就随便再拼一个，拼一段我就拼了一个按的1=1。好，那么你注意，其实我拼上这个条件和不拼没什么区别，因为这个是一个恒成立的等式，对吧？拼不拼都没有区别。我只是给你演示一下 If怎么去用，是这个意思。

好了，那么总之我们在注解里是这样写SQL的，那么好处就是方便直接就写少建一个文件，缺点就是像这样的不太方便阅读，而且你在这里写标签的话，也不是特别方便，各有利弊。

那么写完以后就这个内容特别容易出错，因为你的表名字段名都手写的，它也没有提示，万一写错的话就比较麻烦，所以我们最好对它先做一个测试，如果没有问题的话，我们再继续。好，我打开找到一个测试类测一下。我就在这里写 Map，tests因为我们这个类不是专门测试map对吧？我就在这里写，我就把刚才写的log in take the map注入进来，进行测试，我就在后面的加方法，test我们三个方法，我就先测增加 test，insert，老兵，谢谢。

好，那么在这儿我就自己先随便造一个老兵KK的，好，随便的给他传点数据，这数据可能不是那么合理，但是没关系，site uzid随便写一个，比如说101log in take it点set，txt就是一个随机字符串，这里我就不随机了，我就ABC login take it。点site，x status0，然后是log in ticket点site。Expired。

过期的时间我就new date，然后给它传入 system点current当前的时间，然后再往后推若干毫秒，比如说10001000毫秒乘以60，那就是60秒，一分钟对吧？一分钟，然后再乘以10就10分钟，假如10 10分钟的以后就到期了，就这样。好了，那么构造完这个对象以后我就要存了，我就调 log in, take the map 点insert，把它传入进去。好，下面我就执行一下这个方法，然后看一看能不能把它插入成功。

好，没报错我们看一下，没报错以后控制台会打印 Circle，但是没报错说明 cell有没有问题，然后他最后给你一个提示说这是 circle，这是你传入的参数，然后这个是更新了多少行一行都是对的，我们看一下表里有了一条数据没有问题，然后数据的ID是一，接下来我就要测一下怎么去查询，我就要查这个数据。

当然我们查不是根据ID，是根据 tk的来查，然后查完以后我就给他直接改了，所以我把这个查询和修改放到一起来测，我就test谁来 log in。Take it，好，那么就直接写log in, take the map点，select, buy ticket。传入的是ABC凭证，我会得到这个对象。好了，得到这个对象以后，我把它打印出来看一下有没有问题，然后的话我再对它进行一个修改，我就老给你txt，不点买票，直接点 Update status，我根据 take it把这个状态改为一，比如说它失效了，它退出了，然后改完以后我再查一遍，然后再看看数据对不对？

好了，执行一下看结果也没有报错，那么结果当中它打印的 log in ticket，这是我们第一次查询，没有改状态之前，数据没问题，状态是0，对吧？

然后第二次查询你看状态就变成了一了，说明我们 update也没有问题，好，经过测试 map我们写的没有问题可以直接使用，那么大家在开发的时候最好是这一步都加上测试，因为这一步是最容易犯错的地方。

好了，那么这个数据访问的开发以后，下面我们开发业务层，业务层你当然就要支持登录的业务，你登录的时候就需要接受页面传入的条件，包括用户名密码，也包括你希望希不希望这个账号被长久的记住，或者说这个账号它什么时候过期这样的一些条件，然后我们对这个账号密码进行一个验证，如果没有问题，我们就认为你成功了，我们就给你生成一个凭证，就这么一个逻辑。

我需要去写业务层service在这里这个我们可以新建一个service，或者是我们就在 user service里写也可以，因为你登录其实是用户的一种行为，对吧？

所以说我们在 user service里写也就ok了。那么我们在实现登录的过程中，肯定是要用到刚才的 log in to get the map，我把它先注入进来。好，有了它以后，下面我就开始实现登录的功能，我在后面追加一个方法，那么登录的时候一个是你可能成功，第二个你可能失败。

失败的原因还不止一个，有可能是账号你没输入，有可能是这个账号不存在，有可能是这个账号没有激活，对吧？有多种条件，说白了有点和注册是类似的，它的情况比较多，所以我们返回的数据，我们可以返回一个map，可以封装，就是多种情况的反馈结果。

好，所以我返回值生成为设置为map，方法名我叫log in，然后我要求你调用方要把 U的内容账号传进来登录对吧？

然后把密码传进来，你注意用户在页面上输入的密码是明文，就是没有加密的密码，我们得到的也是这样的一个密码，但是用户注册的时候，我们给他库里存的是一个加密后的密码，所以你这个pass word还不能够和库里的密码直接去比，怎么比你也把密码进行一个加密，按照他注册是同样的逻辑加密，然后去比。

因为MD5只要你给的字符串是一定的，它加密后的结果是固定的，因此你加密之后去比，如果说一致那就密码就对不一致就不对。

好。

另外你外界调用的时候还要传入一个，你多少时间以后要过期凭证要过期，我要求它传入一个过期的秒数 expiredxpl seconds就是我希望你给我传入一个，你希望多少秒之后凭证会过期，好我暂时先水吞闹，好了这个方法就声明完了，接下来我们就来实现。首先我先声明 Map，最终我是返回 map。

好，那么我们对整个登录的处理，首先是要处理这个参数，它是不是空的这种空值的情况，所以首先的逻辑是一做一些空值的处理，我就判断使用可以调工具类使用YouTube，点is black，我先判断user name是不是空的，那么如果user name是空的那就不行，所以我需要返回一些结果，我就给页面返回的是user name message这么一个消息，然后消息上写说账号不能为空，好然后直接就return了。

好这是一种情况。另外密码也可能为空也不行，这个逻辑和他是一样的。Pass word，如果pass word为空，我就返回一个pass word message，然后提示他说密码，不能为空，然后返回。

好，那么如果这个账号密码都是有值的，都不为空，接下来我需要对它的合法性进行一个验证，首先我先验证账号，先对账号验证怎么验证，我们就用他输入的又在内部去查一下，看库里有没有，先看他有，如果没有就账号错了，对吧？如果账号如果有，我们再去比较一下库里的密码和密码不一致，看密码对不对是吧？大概是这么一个逻辑。

好我要做一个查询，那就是查到的结果是user，那就得调 user，map点 select白，这里是banning，那我就传入user name得到user，然后紧接着我要判断一下 user它是不是no，如果是no有问题，这个时候我需要返回 map中的 k还是using message，但是这个提示消息要改一改，这不是为空了，是该账号不存在，这是一个错误的账号。好，那么如果说这个也没有return，说明账号存在没问题，你还得判断一下这个账号状态对不对？因为咱们注册以后不能直接用，它不得激活吗？对吧？

所以这一步我们先把先判断一下它有没有激活，如果没有激活也是不能登陆的。

好，可以进一步判断，如果说user点get standards，它等于0，等于0是表示说他注册了，但没有激活是一对，所以这个也不对，还是要return map，又针对message这块提示说该账号未激活，好，如果这儿也过了，那说明账号存在也激活了，接下来就可以判断密码了。我就验证密码，刚才这个地方是不是，这个是验证状态。

好，那验证密码的话也容易，首先我要对传入的明文密码，按照同样的规则加密，然后再去比较这个 password应该等于什么呢？加密后的一个值，我就调肯定 Ut有点MD5这个算法，然后把明文密码加上 sat，我们又这里有对吧？

Get sat好，密码加上 SARS盐以后再做一个加密，得到加密后的密码，加密后的密码和库里存的是不是一致，是我们所关心的问题判断一下。

悠着点，get password点，echoes password相比，如果说两者不相等，注意是不相等，不相等就表示不对是吧？不对的话你要提示密码如何，我就把这个map返回，这个地方你得写密码不正确对吧？

好了，到这儿如果程序能执行到这儿，就说明以上都没问题，账号也对密码也对，那就说明你登录是可以成功的，登录成功的时候我需要做什么，我就需要生成登录凭证。好，证明你是登录的，我这边要记，并且我需要给你发放凭证你也要记。

好了，生成登录凭证无非就是我要创建一个 log in take的实体对吧？然后往往库里存 Log in，take it设置方法以后，我们需要往里设置一些值，首先你要site ID不用自动生成对吧？Set优质ID那就悠着点get ID对吧？然后 log in ticket点set，take it凭证。凭证是一个随机生成字符串，那么我们之前在 community utu里加了一个方法叫generate uuid方法生成的就是一个随机字符串，不重复的。

好再来，然后 log in ticket点，set status。零就是有效的状态，然后最后再site一个过期时间，new date。但是你这样解释当前时间不是这个意思，应该是当前时间往后推移这些秒是这个意思对吧？New data我这里就这样写，system点carry的time，就当前时间毫秒数再加上那个秒，当然那个秒得换算成毫秒对吧？所以你就是x的second，换算成毫秒就乘1000就行了，这样的话过期时间是当前时间往后推移这么多秒，好，这个对象的构造完以后我就要存了，我需要调用 log in take the map，然后 insert，把这个参数传进去搞定。

好，那么如果说登录凭证已经生成了，登录已经成功了，那我最终返回的结果里需要把凭证放进去，因为我们最终要把凭证发给客户端，你可以放整个log in TV的对象，或者是其实只放 ticket那个字符串也是可以的，因为页面只需要记这个内容，只是说白它浏览器只需要记一个key，比如说我这是一个登录的凭证，他下次再访问服务器的时候把 key给我，我去库里找找到了与之对应的一条登录的数据，我一看状

态对，时间也对，ok我就认为你是登录成功的，从而而且我通过推给他能找到uzid我就知道你到底是谁，就这么一个逻辑。

我们这个表它所起到的作用就有点类似于筛选了，这里我们没有用筛选直接用表存的。好，我在这儿就做这样一件事儿，我就卖的点儿兔子，我要把txt存的放到map里返回，那么它的结果值应该是log in take的点干下去。可以的，好了，判定之后返回这个逻辑就完成了。

好，那么我们有了这个业务方法以后，接下来就可以开始去编写，那么表现层的逻辑，首先我们的一些controller的方法处理页面的请求，那么因为是登录页面，你登录页面的话，你有一个表单会传入三个值，这个很直观，账号密码验证码，对吧？

我需要收到三个值，然后把它提交给user service做处理，然后如果说处理完了以后，我要么去重新上到首页，要么错了，我就回到登录页面，相关的页面也需要处理。

好，下面我就写controller，我们还是在log in controller里写，那么因为这service我已经注入进来了，不需要重新注入了，我只需要在后面增加一个能够处理这次请求的方法就可以了。

首先我声明这个方法的访问路径pass等于老兵，可能有人会奇怪，好像老兵是不是被占用了，你看之前我们这个方法它的访问路径也叫log in，那两个一样能行吗？可以的，但前提是什么呢？这两个方法它的请求方式得有区别，如果我们下一个方法也是盖的，这个也是盖的路径又一样就冲突了。

我这里是应该用的是post，因为什么？这是一个相当于是一个增加数据的行为，就是你表单提交数据给我，我最终要把数据增加到库里对吧？因此我要处理表单提交过来的数据应该用Post的请求，那么根据请求方式不同，所以说也能够将这两个路径区分开，所以这样是可以的。

好，然后我上面这个方法再返回实际方法叫log in，这个方法我需要页面表单传入进来一些条件，首先需要的是user name，其次我需要的是password，第三个我需要你传入验证码，我叫code。

第四个还有一个这个页面上可以勾上，记住我，就是说不勾上的话，我就记得时间短一点，你勾上我的话就记得时间长一点，记住这个勾你也得把勾标记传给我，这个勾是勾上还是不勾上，它显然是一个布尔对吧？

所以这里说明一个参数是布理。Remember me记住这个单词似乎写的不太准确，remembermeok没问题。

然后。

我们返回响应的时候，我还需要用到model，把这个model也声明一下。好，另外我们页面传进来验证码，我需要取到之前我生成的验证码去比，用户打开登录页面生成的验证码放哪去了呢？放到了session里，这个方法对吧？放到了session里，所以这个请求当中我需要从session中把密码取出来，因此我得用到这个筛选，所以我把这个筛选也声明好。

另外如果说登录成功了，我们最终要把tk的发放给客户端好让他保存，需要用cookie来保存，所以要想创建cookie，我还得需要用HTTP server late为sports对象。

好了，总之这个方法比较麻烦，它需要的参数比较多一点。好，那么把这些逻辑理清楚以后，接下来我就开始写了，我们首先要验证什么？首先我们应该判断验证码对不对？如果验证码不对，账号密码不用看了，所以先判断这个验证码，那么验证码是在表现层就直接判断出来，不管业务层只管业务逻辑，你账号密码对不对？

好，所以我在这里直接判断这个验证码，我需要先把验证码取出来，开不成等于从三生中去筛选点get h标的。

我们当时往里存的时候，这个名字就叫卡卡对吧？然后它这个方法返回的是Object类型，但是我们需要的是死卷着，你需要做一个强制的转换，好我就得到了验证码，然后将验证码和用户传入的客户的相比，我判断一下string YouTube点is blank，capture。

如果我取到的验证码是空的，说明有问题或者 is block，如果说你船上的扣的是空的也不对是吧？或者 capture equals。

注意我用的是这个方法，because you cannot kiss就这个方法其实和一口是相等的，逻辑是一样的，但是它是忽略大小写的，这个方法是比较 session中的验证码和你传的验证码是不是一样的，如果不一样就不对。

总之这两个值有任何一个是空的不对，它俩不相等也不对，这个意思，那么验证码相比肯定是不区分大小写的，我们上各个网站都是这样对吧？

所以说用这样的方法好了，那么这种情况就是验证码错了，错了以后我们需要给页面返回一个提示，我把这个提示放到 model里，然后返回给页面提示我叫code，message然后说验证码不正确，我就委托我要去回到登录页面的，是放在 template之下，site下的名字叫log in点test mail，这样就可以了。

好，所以第一步检查验证码，是这样逻辑，如果验证码没有问题，下面我们就需要检查账号密码对吧？这样的内容，这个可以交给业务层去处理，我们在调 service的时候不只要传账号密码，还要传一个过期的时间对吧？你过期的时间你是多久过期，我们是这样去定逻辑的。

如果说你没有勾上，记住我记相对较短的时间，我也是把它存到库里，我依然是把它存到库里，但是只是存的时间短一点。

然后的话如果你供应商记住，我也是把它存到库里，但是时间要很长，所以我需要定义两个常量，时间比较方便，这个常量我还是把它写到 community constant这个接口当中来，好。定义两个常量，第一个常量是默认的登录凭证的超时间，默认状态的登录登录凭证的超时时间。好的，设立一个整数，就是多少秒，然后 defeat。Expand seconds。默认超时的秒的数字，我就写3600，3600乘以12乘以12。

好，那一小时是3600秒，乘以12就12小时，就是说我这个凭证默认给你存12个小时，就是你不勾记住我的情况下，好，那么如果你工程基础我要存很久，因为我们现在大家也有这么一个体会，我们去各个网站别管是通过APP还是通过网页，我们只要登录一次以后，其实很长时间之内都不用登录了，对吧？尤其是APP，所以只要记住我们可以给一个比较长的时间，这个我写个注释，记住状态下的登录凭证超时间这个叫remember。

Expand不懂再感觉这个时候我存的记多长时间是3600秒×24小时，那就是一天了对吧？我再乘以一个100记100天记了三个多月，这时间也挺长了，这样就可以了。

好，那么回到 CTRL了，我需要让它实现这个接口，我好利用它里面的一个常量已经实现了之前，所以我就可以直接用了。我得那我到底是用哪个值，我得看一下。用户勾没勾选上记住我，对吧？他没有勾选我就用默认的，他勾选上了我就用长的，所以超时的时间我还得判断一下，我说明一个变量 X PAD，seconds就是我们调service方法它需要的参数，超时的描述，我是根据Rimbey这个条件去判断它到底应该等于几。

如果remember me记住我应该用的是常量。Remember你remember expire second对吧？否则那就不记住我，那就短一点default，这样就好了。

好了，那么有了这个条件以后我就可以调 service了，user service点log in，我把user name传进去，我把它是word传进去，我把超时的时间传进去，那么service给我判断以后会返回一个map，我需要让他收一下。好，我得到了一个结果，这个结果我们能够判断出来它是成功的还是失败的，很好判断。如果 MAC里头包含了ticket，就说明成功了，因为成功的时候我们才往里放 ticket对吧？否则就是失败。

首先说一下如果成功了，我们最终不是要重新上的首页对吧？我就委托人我这里写上重新向前，Redirect重定向的首页的仿路径是index就行了，那么如果是不对，如果是没有退给他，说明你有错误是什么错误我不知道先不管，那我就得回到登录页面，所以我return登录页面。好，先把 Return写好，写好以后咱们就做处理。

那么在登录成功，在这个时候我需要跳到首页，但前提是我得先把这个txt取出来，然后发送给客户端让客户端存，说白了我用我就是给客户端发一个cookie，带上 ticket，好，我就这样写了。

先是另外一个 QQ等于new cookie cookie的cake，我就叫cookie的。那么它的value就是我从map中取到的一个结果，map点get tkt好，当然我们get到的是它是一个对象，我们需要把它吐死并转成字符串，因为 Cookie它k86都必须是字符串才行。

另外 cookie我们通常都要给它设置一个它的生效的有效的路径 sat pass。那么其实用户登录以后，它的凭证有效的路径应该是包含在整个项目之内，他访问项目中的任何一面，其实这个登录的状态都应该是有效的，对吧？

所以这个路径就是整个项目的整个项目就应该是community，但这里我们最好不要写死，而是写成写成一个参数，这个参数我看我有没有注入进来，没有。这个参数咱们不是在 Application process里去做了配置，你看application里边有这个参数就它对吧？

我们可以把它注入进来。

当然如果你不注入，其实我们通过request，对象也可以获得，这里我注入一下它，注入到 CTRL之内，那么我在这写上value，这是一个固定的值，它可不是一个b好，然后把 k写到这个地方，然后底下声明接受这个值的一个属性一个成员，我叫contact pass。通常都这么叫，你看他也是这么叫的名字。

Context pass好了，有了这个值以后，下面我接着写，set，pass就是整个项目都有效。除此以外，接下来你就要设置 cookie它的有效的时间了，那么设置cookie有效时间就是site max h这个时间就是刚才我们得到的一个second多少秒，那么设置完以后，我需要把 cooking发送给页面用到resource点，add cooking，把酷可以添加进来，在响应时就会发送给浏览器，正确的时候就是这样处理。

那么错误的时候我们要回到登录页面，要把错误的消息带给登录页面往model里存，model点at actual，有可能是user name出了问题，user name message，这个message应该是map要get，又在message，当然有可能不是他的问题，但如果不是它的问题，那么map点get得到的结果，因此它是闹到页面展现的时候也不会有什么影响。

好除了u的name错误以外，还有可能是密码的错误，密码的错误的话，我就 set password message，那也是同样从map中得到这个值放在mod里。好了，到这儿我就把这个CTRL了就完成了，那么完成以后，最后我们就把这个页面处理一下就好了。

当然页面处理什么呢？你要处理的就是设置这个表单，首先你表单每个框上得有内容，我们才能够把表单提交给服务器。第二个如果说我们检查发现的账号密码或者是有什么问题，我们还要回到这个页面对不对？等你回来的时候怎么去显示错误的消息，怎么显示样式？

说白了配置和这个页面的设置和注册页面是很相似的，好了，我们接下来就来设置这个页面，我就打开登录页面，login点html，好，然后我找到这个表单的位置就在66行这个位置，首先我要声明这个表单提交给谁，Master等于post，我们采用POS的方式提交，然后提交的路径action等于老b好，这是提交的路径，然后每一个框上我们需要给它加上name，我换一行，首先这个账号它的name等于u的name和我们 CTRL的方法的参数有一要一致，其次是密码，这里我也太长了，我也换一个行，然后 name等于 password，然后还有验证码在这，ID不用管 name等于我那CTRL的参数叫code，所以这里我写的是 code，然后再看还有这个记住我其实它是一个勾选的，它也算是一个框，你也得给它加上name也要传给CTRL了，name等于control，我是怎么写的，参数你慢慢密好要跟它一致。

好了，这个表单就是提交的方式路径以及每一个框的内容都写好了，这是我们发送请求的时候要做这样的设置。

如果说检查到有什么问题，回到这个页面，我们需要对错误的信息做一个展现，我们在处理错误的信息的展现。

首先是账号，如果你回到这个页面的时候，我肯定是希望把原来的旧的值显示到框上，它就有个默认的外流，所以我在这给它加上 th冒号value等于。



好，这个value从哪里取呢？Y6从哪里取的？注意是从这里取，这块有一个规则你要了解，如果说我这个参数不是普通的参数，而是一个实体，比如说user，那么spring mvc就会把yous装到model里，所以我们在页面上就可以直接得到u的对象的数据，那是从mod里获取的。

但如果说你是这样比较普通的参数字符串基本类型这样的参数，那么spring它不会把这个参数放到model里，此时model里没有有两种办法我们去得到它，第一个你人为的把它加到model里，这是第一个。

第二个还有一种办法怎么说？这几个参数它是存在于request对象里的。我们如果说在这里头写代码写request点get private也能得到这些参数，因为它是请求里携带过来的，那么当程序执行到的Html上面的时候，request还没有销毁，因为请求还没有结束，所以我们在这个页面上也可以从request中取值，那么我们在这儿我就懒得在control里把参数再装到model里，麻烦我就直接从request中取值，怎么取他们内部对这方面的知识还是很方便的，你可以这样写。

Carol user name这句话的逻辑或者意思相当于什么？相当于request点get private user内容。好，这个条件是从Request对象中取参数得到的，你注意。好了，同理密码也可以做这样的处理对吧？Thyl等于他有点pass word，好，然后验证码就不要给他默认值了，如果说他账号或密码或什么错了，每次验证码是要刷新的，所以不要给默认值。

最后是这个记住我，如果说你刚才勾上了记住我，然后一点登录，结果发现有错，回到这个页面的时候，这个地方勾应该还在对吧？

这个记住我它是一个多选框，它的状态是通过check进行设置的，不是通过value，所以我们要想让它有默认值，是通过属性来进行一个处理。

目前的话它是写死了，check它等于check，这样的话就表示说它就默认就是选中的，但我们现在需要动态的判断，或者把它改成一个动态的处理的方式，你就可以用还是用timeline。

Th checked表示属性里边的值，我是用动态的方式去处理它等于什么？它等于这个is Paro还是从request中取参数，取的参数是remember me。好曲的，remember me其实它是一个false的值，Check的属性当中，我们给false这个值也可以处理表示勾选，force表示不勾，选是它也能识别。

好，这是我们回到这个页面的默认值的情况，除了这个以外还有一些错误的提示也得去处理，我们这样回到上面再进行最后的一波处理。那么首先如果说你账号不对，你这里肯定要显示的是账号不存账号相关的提示对吧？这个提示应该是动态的。Th模型text那就等于变量user name message。好，然后这个账号就这个提示它是否显示，取决于什么？取决于它前面的Input的样式。

如果有这个样式它就显示，没有样式就不显示，我们应该默认没有样式我们首次访这个页面是没有回到这个页面，是如果说确实它有问题才加上，因此class应该是动态拼上去的，不应该写死，而且它是一个固定的样式，加上一个动态的有可能没有的样式，我这里需要这样做这样的处理。我需要判断user name, message它是否等于null，如果它不等于null，我就带上样式，否则就不带。这样好，同理，这个密码也是做一样的处理。首先我设置一下密码，它的值取值，它的取值是它所有的message，然后他也是对这个样子也是做同样的处理，也得加竖线，这是一个变量。

然后写错了 Password, message它不等于not，我就带上样式，否则就不带。

好这个是密码，那么验证码你也需要做这样的提示对吧？验证码的提示信息也得写上去，验证码的提示信息是扣的，message，然后也是一样的逻辑，一开始不带s已经办理，也需要动态的判断，我判断的是客户的变量，客户的message这个条件它不等于那我就带上一起来卖了，否则就不带好了，最后我再检查一下这个按钮，是提交按钮，没错。

好了，那么我能想到的地方我都想到了，然后把它写上去了，当然写的时候有可能哪个单词写错了，也很正常，我们就测一下，如果有问题我们再去调。好我们进行一个测试，我需要把这个项目启动一下，启动的时候看一下，控制台，如果说哪个地方后端代码有明显错误，它可能启动时就报错，我这里没报错。

然后我访问一下，打开浏览器访问local host先访问首页，然后我点登录，首次点登录，我还没输入任何条件，那么它所有的错误的样式都没有，对吧？

然后此时验证码是可以刷新的，如果你看不清就刷一下好，接下来我就要输入一个账号去登录了，我找一个账号看一下优质表。

咱们之前讲注册的时候，我不是注册了一个账号叫CC，我就用它，但我一开始不用它，我一开始故意用一个错，嘻嘻用的错的，然后密码我也随便瞎写，验证码我也瞎写，好记住我勾上，然后我点登录以后错了，显然不对，哪个都不对，他就回到了登录页面，它首先提示的是验证码的错误，因为我们在后台是先判断的验证码，如果验证码不对，我们就不访问数据库了，这样效率高一点，所以首先我需要把验证码写对了，我就看不清了，刷一下，然后不区分大小写，好我写成一个对的验证码，然后此时回来以后你看勾勾上了，表示说我们对它的处理也是没有问题的，好再登录验证码还不对，3v8接，好刚才可能还是看不清，我是写错了，这回对了以后，但是账号不对，对吧？

回到这个页面它就提示账号不存在，我给他搞一个存在CC，但密码不对，我那个密码好像就是123明显更长了，验证码还得重写，7ac0登录，回来以后提示你说密码不对，好，我就把密码改对，123，然后Rwut好，我就立即登录，这回成功了，他跳到了登录页面。

当然因为目前我们还没有做后续的处理，就是登录成功以后，比如说注册登录这些需要该隐去的隐去，然后这里显示的可能是和我登录有关的内容。

这些地方的正确的显示其实就和我们给用户颁发的凭证是有关系的，我们需要利用凭证看你到底是哪个用户，看你到底有没有登陆，然后对这些地方做一些设置，这是我们后面要处理的地方。

目前没处理之前，我们从页面上还看不出来我到底登录成没成功，看不到痕迹，我们只能去数据库看一下，怎么看你看看凭证表你看生成了一个新的凭证，uzid151151是CC吗？是的，然后状态是0没问题。超时时间是6月13号12:29，12小时以后，现在的话还是12号。Ok没问题。就是这样的，这12号刚才12号对吧？这12小时以后没问题，对的好了，那么我们就把这个登录逻辑就搞定了，最后还剩下退出的逻辑那就比较简单了，下面我们再把这个退出给它处理完。

退出的时候，其实我一开始也说了，我们首先需要把凭证改为一个失效的状态，然后跳转到首页就可以了。把凭证改为失效的状态，我们在数据访问层已经实现了log into, the map里不是有一个update吗？已经写好了对吧？

所以数据访问层不用写了，那么业务层我们需要写这么一个业务，好我就打开刚才的user service，我在这里再加一项业务处理，就是退出public，然后就没有返回值了，然后log只要你说你没有报错，我就认为是成功的。当然你退出的时候你需要你是谁退出，你要把你的凭证传过来给我，我好根据凭证找到你是谁，我好给你做一个退出的这么一个行为，对吧？

所以说你要把凭证给我，你要传的是txt。好，当你把txt传入以后，我就调log in。Take the map。点不对的去改变凭证所对应的状态，可以给它把它改为一。一表示无效那这就可以了。好，那么这块处理完以后，接下来我们需要在CTRL那里再加一个请求，能够处理页面的这个请求。

好，那么这个请求的访问路径先声明一下，路径是pass log out，master是就get就可以，因为退出它不需要传通过表单提交什么数据过来，就是一个普通的请求。

好，然后方法返回的是死病，方法名叫log out，那么这个方法当中我需要得到你浏览器之前存的cookie，对了，刚才我们页面上表面上看不到它登录成功与否，但其实我们通过这个插件能看cookie，如果说他确实存了cookie，就表示他拥有登录的凭证，用了登录的状态，那也能证明他是登录成功的。

我们这样看一下，你可以点在他word刷一下这个请求，然后看请求里有没有，或者是也可以点application，application里头有一个选项叫cookies，这里面会列举就是你这个路径下对应的cookie，你看他有一个cookie叫ticket，然后ticket这个值结尾是1a26对不对？

你可以做一个检查，1a26没错，然后它的生效的路径是肯定的，没错对吧？

它的超超时的时间太小了，超时时间也没有错，所以从这也能看到它没有问题，现在浏览器已经存了 qk 它退出的时候需要把qk传给我，先把那个值传给我，我怎么去得，它会自动传我怎么去得，咱们上次不是讲了，我们可以利用酷k五6注解对吧？

然后要求spring mvc要把它注入进来，key是ticket，然后加一个参数去接收 ticket。

好，那么我在这个方法里很简单，就调 service，然后 log out就完了，然后把k给他传给他，他把设置为失效的状态就可以了。然后你退出登录以后我们去哪，我把它重定向到登录页面，你可以重新登录，ready？登录页面就是老毕。

好，那么注意log in，我们有两个方法路径都是log in，一个是get请求，一个是post请求。那么我们重定向的时候，那么它默认就是get请求，当然这还没完，因为我们在页面上需要点这个地方才触发退出登录的对吧？这个链接你得把它配好是吧？

所以我们需要配置部位的链接，而那个链接我们所有的头部的链接都是在index页面去配的，然后大家复用，所以我还要打开首页，index，然后我就找到退出的地方，它了目前它的路径不对，我需要进行一个修改。

好，那么路径要改成退出的路径，那么就是log out，好这样就行了。

那么做完处理以后咱们再次做一个测试，我CTRL f9，好，然后重新启动了，启动以后我打开浏览器然后刷新一下，刷新完以后我去点退出登录报告的它跳转的没有问题，但是底层的逻辑对不对？不知道，我们可以去库里看一下，看这个表的状态变成一说明我们更新的成功了，没有问题。

好了，那么这次课我们就把退登录退出这样的功能开发完了，总体来说并不难，但是它能够把我们之前所学的验证码，我们之前所学的状态管理cookie都利用上，我们可以体会得到这些之前所学的这些零碎在这个过程当中是怎么去运用起来的。好，课后的话大家也认真的把这个功能做好，因为这是我们几乎每一个网站都有的功能是非常具有代表性的。好，这次课我们就讲到这里，咱们下次课再见。

## 2.6 显示登录信息

15-显示登录信息.mp4

这次会我们来实现显示登录信息的这么一个功能。什么叫显示登录信息？就是说我们在每个页面的头部在这个地方，都要把登录的用户的头像在这里显示出来，另外你点三角展开的信息当中，在这你要显示用户的名字，除此以外还有一些处理的地方就是说，如果用户没有登录，我们给它显示的是首页注册和登录这样的链接，如果他已经登录了，我们要给他显示首页消息以及头像相关的内容。

所以说根据它登录与否，我们需要调整的头部显示的这些内容，我们可以统称为显示登录信息，那么很显然大家也看到了我们的静态页面每一个页面里头部都有这些内容，或者说它都需要显示登录信息，就是说你开发这个功能，你需要每一个请求都要做同样的处理。

当然我们用一个比较笨拙的方法就是说你可以CTRL当中每一个方法都加上这样的逻辑，但是这样会很麻烦，即便是你把这个逻辑进行了一个封装去调，你也去调需要调很多次对吧？我们一个网站可能会有很多功能，可能会有特别多的请求，几百个请求上千个请求，你调了那么多次会很麻烦，万一说将来的话哪个地方要变是吧？你要调整的地方也很多，侵入性太强了，或者说耦合度太高了。

因此像这种情况，我们有没有办法说能够集中的统一的用一种很低的耦合的方式把它解决了，那答案是有的，我们可以利用spring的拦截器去统一解决问题，那么拦截器跟它的名字表现的是类似的，它的功能体现在名字上，它能够拦截浏览器访问过来的请求，当然可以拦截多个请求，甚至是全部请求可以配置，然后拦截到这些请求以后，它可以在请求的开始或者是结束的部分插入一些代码，从而就能批量的解决多个请求共有的业务，这就是拦截器的它的一个好处，总之它能够以一个非常低的耦合度解决通用的问题。

那么我们这次课是这样，我先给大家演示一下拦截器的示例，你了解一下它语法它怎么用，了解完以后我们再去解决我们最终要解决的问题，这样就顺理成章了，对于拦截器我们在使用的时候也是挺简单的，第一步就是你写一个类，然后实现接口，借口有几个方法，我们根据情况然后去实现。

第二个就是我们需要把拦截器配置一下，你要指定它拦截哪些请求，不拦截哪些请求，这两步开发完就完了。那么我们开发的拦截器和处理请求的 Ctrl 没有任何直接的关系，但是也能解决问题，所以说这就是它的好处，耦合度是很低的。

好了，接下来我给大家写一个案例，我们来体会一下拦截器它的方便之处。

那么我打开我的开发工具，然后因为拦截器它处理的是请求，那么其实它也属于表现层的逻辑，因此我把拦截器代码也放到 CTRL 这里，但是我新建一个接口，新建一个包放到新建的包下，这个包我就叫 intercept 拦截器，然后我就写一个拦截器的示例，这个类我取名叫阿尔法 intercept。

好，拦截器首先我要把他声明一下，让他把他交给 space 容器去管理，所以加上一个注解，另外要给它实现一个接口，刚才我们说了实现的接口叫做 handler intercept。

好，那么这个接口里主要有三个方法，这三个方法一个是 play handle，一个是 post handle，一个是 after come complexion。

那么这三个方法它的前面都有一个 default 标志表示这个方法是一个默认实现的方法，当然它是空实现，所以说你实现这个接口以后，你可以不实现这个方法，也可以是实现这个方法，就是根据你的实际需要挑挑你需要的去实现就可以了，这也是它方便之处，它不强制你就是实现所有的方法。

好，这里我把这三个方法都实现一下，然后我们看一下它有什么用。首先第一个它是有一个叫普瑞憨豆的方法，放大一点 Free handle 的方法，那么这个方法就是说它在拦截请求的时候，它是在 CTRL 处理请求的代码的前面执行的，它是在请求的前面执行，这里我写清楚，就站很抽了，之前执行。

好，那么这个方法是返回布尔值，如果说你方法内部你 return false，那么就是说我要取消这个请求了，那么它程序就不会往下执行，controller 就不会被执行，所以我们通常是反馈错误的，除非你有特殊情况说这种情况不应该往下执行，我才 return force，这里我就改成 return 处。

好未来一会我们看到程序执行到这，所以说我想给他在这里打一个日志，我先把日志组件实例化一下，好。

那么以当前的类作为 log 的名字，我在这里就简单的就打个日志，日志是起到一个调试的作用，所以说给它设置为第八个级别就可以了，那就是 log 点。

第八个随便写句话，我就写这个方法名，比方说这个方法被调用了，不是憨豆，然后这样我再加一些信息，因为你看这个方法它是有一些参数的，首先有一个 request，参数，因为你是拦截请求，所以说你想在拦截的过程中，你想自己怎么说加个塞儿，在处理请求过程中加个塞儿，加一些自己的逻辑，所以说你也想处理请求，因此他把 request 对象给你，response 对象给你，你可以去处理请求和响应，另外还给了给了你一个奥派克，这个 update 是什么？

我们可以打印看一下，这样比较直观一点，我就加上 handle 点兔子菌，我们看看这里这个是个什么东西。

好，就这样了，我们再看另外的两个方法，然后还有一个方法，刚才我们看了叫 post candle，我也把它在这里实现一下，这个方法我就直接告诉你他在什么时候执行的，他是在调用完 CTRL 了以后去执行的，所以说它是在 controller 之后执行，当然我们一会通过程序能够更直观的能够看到这一点。

然后它也有几个参数，第一个参数 request，第二个 response 不用说了，这个 handle 跟这个一样，除此以外它还有 model and will，因为什么？这个方法是在 control 了之后执行的，那 control 了之后表示说我主要的请求的逻辑已经处理完了，下一步就是要去模板引擎了对吧？

要去给页面返回要渲染的内容了，因此你再去模板引擎的过程中可能需要用到 Model and view，你可能需要获取这里的数据，或者你需要往这里装一些数据，因此他把这个参数给了。

你好这里我也打个日志 log点底半，然后写上pose handle，同样的我也把不知道是什么东西的对象，打印出来我们看一看它会是什么。

好，除此以外，最后还有一个方法，那个方法叫after Complete。

好，那么这个方法它是在程序的最后执行是在这个模板引擎执行完以后执行，你注意这两个是有区别的，pose title是在control了之后，模板之前，而它是在模板之后，最后执行他们的时机你要搞清楚，这里我也注释也写清楚了，它是在 template安整就模板引擎之后执行。

好同样的这里我也打一个日志 log点debug，然后还是写上方法名，然后写个冒号也是加上 handler。

第二兔子病它的参数还是这样的request，response，然后 handle最后还有一个异常对象，就是说如果说你在调用 control这个过程当中或者是调用这个模板的过程当中出现了异常，它会把这个异常信息也给你，你也可以对异常做一些处理。

好了，那么这个代码就写完了，我们只是通过它来看一下程序执行的过程，这个时机对不对？打日志，就写完以后要想让它生效，你还得对它配置，不然的话死光也不知道我应就组件应该拦截哪些请求。好，所以我要写一个配置类，我在肯目录下新建一个类，那么这个类要取名叫外的 Mvc can figure ym VC倍。为什么这么叫？一会你就知道了，首先我要在这个类上写上一个注解，表示它是一个配置类，Can figure reason我放大一点。

然后我们之前所写的配置类主要是想在配置类里声明一个第三方的病对吧？

通常是做这样的情况，但拦截器其实这个逻辑相对来说和别的不太一样，我们想配拦截器配置类，它要求实现一个接口，而不是简单地装配一个GB实现哪个接口，是vibe的 mvc can figure，所以因为当前这个类要实现这个接口，而这个接口的名字是这样的，所以说我在命名类的时候是参考了接口的名字这个意思。

好，当然了我们在配置类当中主要是要配拦截器，所以我们可以把拦截器注入进来，然后再对它进行配置，先不用着急实现这个接口好，那么它叫做阿尔法 intercept对吧？把它注入进来以后，我们就实现接口在某一个方法里去注册拦截器，那么我们需要实现这样一个接口，名字也很直观了，叫爱的 intercepts。

其实这个接口里有很多的方法都是做了默认的实现，那么你根据自己的需要去实现就可以了，我们注册接口就是实现这个方法，爱的interceptors，那么这个方法就是spring在调用的时候，它会把对象传进来，我们是利用它传进来的对象去注册 intercept，怎么用很容易 registry，然后点at intercept就添加一个拦截器，你就把这个been添加给他就可以了。

如果你只是这样写的话，就比方说拦截器会拦截一切请求，那么如果说有些请求你不希望它拦截，你可以这样就可以先就是include pass patterns。意思是你可以排除哪些路径，哪些路径不用它进行拦截，哪些路径不用拦截呢？我们的项目中除了浏览器可以访问CTRL，其实我们项目中还有一些静态资源，cssjs以及图片对吧？

这些静态资源其实浏览器访问的时候，你就任他访问，你不需要进行拦截，因为你访问一个静态资源，没有什么业务逻辑，而且你随便访问的这些东西，因此你要把这些静态资源的访问给它排除掉，所以我在这里把就写一下这个路径排除的路径，我这样写杠星星，然后星点css，为什么这么写呢？你看一下我们静态资源存放的目录放到了style之下，而且我都是把它放到了某一个目录下，对吧？访问静态资源怎么访问不就是 Logo house的80 80，然后community然后站斜线，直接就css什么其实相当于在项目的路径的下直接就访问到这儿了，所以说我们要配的话，直接项目以及域名可以忽略掉，从这一级目录开始配。

所以我杠星星表示什么意思？就比方说时代的一个目录下，所有的文件夹当然你这里写CS也可以，刚星星我觉得省点事，就哪怕说还有目录有css也是ok的，它能匹配所有的目录，就这个意思。

好，那就是说所有目录下的css文件我都要排除掉，同理 GS也是这样类似的逻辑对吧？杠星星杠星点GS我们就这样采用通配符的方式一个去排除，然后再排除图片，然后是星点偏近。

好好再来，当然图片不只是PNG格式，然后还有 Gapg还有接peg好，就这样了，我排除掉了这些静态资源，其他的请求我就都被他拦截了。

比如说我不希望其他的请求都被拦截，我希望一部分请求若干个被拦截怎么办？这里我再进一步演示，你可以再点爱的pass patterns，你添加一下你明确你要拦截的哪些路径，我这里比如说我就希望它拦截的是注册的功能，你就可以写register，这是注册的功能的仿路径，比如说我还希望它拦截这个登录就烙印，那这样就可以了，你就可以在这里加很多个路径。

当然其实如果有些路径有相同的规则，你也可以用通配符，假设我们有一个 CTRL，它的仿路径是 User 下有好多个方法，你可以这样对吧？你可以这样，当然目前我先没有先不管，好了，我拦截器就这样配好了，基本上就这么用就可以了。

那配完以后，你看我这两个类和我们要拦截的 CTRL其实没有任何的瓜葛，从表面上来看对不对？那么它能起到作用吗？我们测试一下就知道了，好，我就把这个项目重新启动一下。

好，启动完以后我就打开浏览器，然后我访问一下 Host，我先访问首页，index访问到了，然后我们看一下控制台，你看一下有没有拦截器，记录的那些日志，什么pre hand post这些东西有没有？经过检查没有，为什么没有？因为我们配的是拦截器不拦截index，它只拦截就是 Register老兵对吧？我再访比如说 Rest或者老兵看看他能不能拦截到，比如说我就访问这个登录，我点一下这个登录，打开登录页面。

好登它的路径就是log in，我们看一下有没有拦截到看工作台，在这我这个屏幕就是说分辨率调的比较低，所以说一屏显示的内容比较少。在最后你看这个地方 Free handle阿尔法intercept20行Free handle, post handle after completion。

所以说你看到了它确实起到了拦截的作用，代码执行了，而且这个顺序也是我所说的顺序，这个先然后是然后这个方法另外它也打印出来了拦截器的参数，object是什么东西，你看一下。就是CTRL点这个方法就说明什么？Object是什么是拦截的目标，我们刚才访问的是log in方法，object就是log in方法，那个方法它是拦截的目标是这样的。

好了，当然这么看的话，只能说看出来三个方法执行的顺序，但是你并不能看到说他确实是在controller之前，他确实是在controller之后，他确实是在模板之后看不出来，你想看看也很简单，我在程序中打几个断点，我们看一下断点，执行的顺序就知道了，对不对？

下面我就在 log in control这里先打个断点，我找到烙印就打开登录页面的逻辑，我在这里打个断点，然后再打开拦截器，在这三个方法当中也是依次打上断点。

好，我打开浏览器刷新一下，或者再点一下登录点，然后你看这个断点卡在哪了，首先是卡在了普瑞憨豆这个地方，这个地方是不是啃出了还没执行对吧？

因为执行的话就会卡到这，但他先在这说明他是在肯德勒之前被调用的，然后我就按一下这个f9让程序接着向下执行，直到执行到下一个端点为止。

你看第二步就卡到了登录的这个地方，然后再f9就卡到了POS探头这个地方，就证明了这个顺序，就如我所说的，他在肯出了之前，他在肯出了之后，然后再执行 f9到这这个地方就是模板，然后我让程序接着走下去，然后把这个断点都去掉。省的一会我们在用的时候它有点碍事。

好了，我们再回到 Ppt就是说通过这个小例子你就了解了拦截器它是怎么用的，就非常简单，反正写一个类，然后简单的配一下，它能够起到这么一个很强大的这么一个功能。好了，接下来我就用它来解决我们项目中的实际问题，想办法要在页面上显示用户的登录的状态。

那么怎么去处理？这里我给列举了几个步骤，首先就是我们在处理浏览器的请求的时候，我们应该在请求开始的那个时候，先把登录用户先查到，因为你要显示登录信息，你要把登录的用户查查到以后，这登录用户后面模板上要用，所以你要想办法把它存一下，存到内存里。

这里我说的是在本次请求当中持有用户数据，其实就是存到内存里，后面好用，然后在后面我们在模板上要显示用户在模板的地方要用，最终用完以后，在请求结束时你把这个用户数据清理掉就可以了，所以说是分这么4步，但是我这样说的话，可能是大家并不能够把这个逻辑怎么实现和我们拦截器就是match起来，或者说也不能够把这个逻辑和我们之前做的登录匹配起来不够直观。

所以下面我给大家再画个图，把这个逻辑再串一下，然后你把这个思路理清楚以后，我们再去写这个代码，画个图先画个框，这个框比如说代表了浏览器，我再画一个框，用它来代表这个服务器，那么假设目前用户已经登录了，他登录以后，我们服务器不是给浏览器发了一个凭证，浏览器是通过可以发的，所以浏览器就存了凭证，上一次课我们看到了对吧？

因此库克里存的数据存的是 ticket在浏览器里存了。

好，那么除了以后浏览器只要是在访问服务器，不管是访问哪个功能，那么他都会把这个数据发送给服务器，因为我们对 Cookie设置的有效路径是整个项目对吧？所以他会把库可以发送给服务器。好，那服务器就可以从库里得到凭证对不对？得到一个凭证，得到这个凭证有什么用凭证关联的是用户，对吧？因为每个用户我们生成的凭证是不一样的，凭证关联的是用户，所以我们很容易能够通过凭证得到当前用户是谁，当然你想得到当前用户是谁，你需要查询一张表，这张表叫做log in ticket对吧？

我们上次给他创建的，所以这个服务器可以根据凭证去访问一张表，然后通过这张表得到用户的信息，其实得到的就是用户的ID，得到用户的ID就好办了，你再查user表不就最终能得到这个user了吗？对吧？

所以我这里就省了查查优则表这一步了，反正我就通过它能得到优则的明确的信息了，得到优质的信息以后，最终我们是要把它运用在模板上，因为模板上你要显示头像显示用户的名字对不对？所以说你要把它用在模板上，那好吧，你就把它放到model里，然后模板就可以使用它了，对吧？

所以很容易就能把它提交给模板，就是把它放到model里。

好，最终的话这个模板像浏览器响应生成了一个html，这个html里就可以携带上当前的登录用户的信息，所以说整个流程是这么一个过程，那么这个过程当中关键点是登录的时候，我们除了 take it，所以再次访问时我们可以获取 take it。

另外我们库里除了 take it，我们在服务器可以通过take it，找到这个数据，从而找到 user，是这么一个过程。

然后这里你也应该能理解说我们当初为什么没有浏览器上直接存user数据，因为user数据比较敏感，你直接存到服务器端，浏览器端的话不安全而存一个take的只是一个key，相对来说不那么敏感，这样比较合适。

好了，整个的过程大概是这样，下面我就给大家写代码，把这个过程实现一下，好。

我们再回到我们的开发工具里，当然了还再说一下，我们这个实现这个逻辑就是获取得到 txt查询，从而得到user，然后在模板上渲染这个过程是每次请求都要这么干的，因为每次请求最终的模板上都要显示用户信息对吧？

所以这个东西是要反复去实现的。因此这套逻辑我们应该是用拦截器来处理，而不是写多次对吧？所以我一上来就应该写一个拦截器，我就在 intercept包下再新建一个拦截器。好，新建一个。那么拦截器我取名叫log in。Take it。然后 intel sector看来这个就不用了，暂时也不用把它先关掉，放大一点。那么他的书写的方式和刚才我们写阿尔法inside是类似的，对吧？我就先给它加上 complaint，注解，然后另外让他也实现接口 Handler intercept。

好，然后按照刚才我画的图的逻辑，我们应该是在请求的一开始就去获取 txt，从而利用 txt去查找有没有对应的user，然后如果有的话就暂存一下最早做这个事儿，为什么要一开始就做？

因为我们在整个请求的过程当中，可能随时随地都要用到当前用户，随时随地都要有用的，所以你最好一开始就找到他，所以我就重写一下 pray，憨豆这个方法，然后先先写成return处不要return，否则后面就不会执行了。

那么在这个方法里，我第一步就是要通过cookie中的数据，通过cook得到 ticket，那么你注意这个方法是由接口定义的，它的参数可不能说随便去加的，你说我加一个参数不行，接口定义好了，所以我们不能加参数，因此我们就不能在这里用酷ky6注解了。不过也没关系，好在他给了我们传过来的 request，那么cookie是通过request传回来的，对不对？我们从这里能得到 cookie，我从request中取 cookie，我希望把它稍微封装一下，因为我们后面其实还有别的工具，也会有类似的情况，用瑞克斯获取酷品，而它从对象中获取库还有点麻烦，所以最好封装一下以后好复用。

那么这样我在 YouTube的报价在这新建一个类，然后类我取名叫cookie YouTube。

然后在这里边我就写一个静态方法，咱们直接调就可以了，我就不用这个容器去管理它了，就是一个简单的小工具，静态的方法，那么它最终的返回的就是库克中我们得到的一个值是死菌，然后这个方法名我就get y6，从库克中取一个值，那么我们从库克中取值，首先你得给我传入这个request、对象，另外你还得告诉我你要取那个值它的 name是什么， k是什么，把这两个条件给我，我就帮你取，但是我首先得判断一下，你这个request对象传的是不是空的，你万一给我传个空值怎么办？

对吧？

第二个 name它是不是空的，如果你传个空的又怎么办？如果你传的参数是空的，我解决不了，我就抛一个异常，就伊利狗给我们exception，给个提示说参数为空，这是不对的，处理不了。好，那么如果参数不为空，我就可以去获取了。从request当中取cookie也很容易就点get cookies。

我们一下能得到所有的cookie，所有的cook对象得到的是一个数组，好，是个数组，那么我们想找到某一个值，你很显然得遍历这个数组，然后就判断当前的对象的k是不是等于这个name，如果是我就返回它的值，如果不是就继续对吧？这个逻辑很容易想到，我就当然得判断一下，cookies如果cookies不等于not，万一一个cookies都没有对不对？

判断一下我就便利它，便利 Cookies。对于每一个cookie，我去判断 Cookie的name它是否等于你传入的参数，如果是的话我就返回，就是我们要找的数据，我就返回cookie的 baidu。当然如果不是你就继续，如果说整个程序都执行完了，你还没有返回，那就说明其实就没有你要的数据，我就得听到就可以了。

好，这个工具我就写完了，还是比较简单的，咱们再回到拦截器，在一开始的地方就可以解决我们第一步要解决的问题了，获取 Cooking。从酷配中。对吃完了从酷配中获取凭证，当然你得登录我才能取得到，如果没登录是取不到的。好，我就使劲 Take it。等于调那个工具， cook you too点get value。把这个request给他，然后把 key给他， key就叫take it。

好，我们尝试取这个值，它有可能是空的，有可能不是空的，你得判断，如果teacake它不等于不等于，那就表示你登录了，那么你登录的时候，你登录的时候我要怎么办？我要要查询一下 ticket，从而找到用户的数据对不对？你要查询的话，我们还得调一个service去做查询，我们最终要查的是user以及推给的相关内容，我们就需要看么user service里目前我没有加查询凭证的代码，我给他加上去，在后面追加一个方法，他不得然后返回的是 log in。

Take it。我要查的是log in ticket，然后方法名我就叫find log in ticket。

那么传入的条件就是凭证这个字符串，这个方法实现非常的简单，就是你就调 log in tk的map，对查询方法一步就实现了，没什么可说的。那么有了这个方法以后，我拦截器又可以继续写了，如果take的不等于我就把整个take的对象查到，当然我要查还得调 Service，我还得把 service住进来。

User service，好，我这里调用user service点find log in ticket，把凭证传入我就会得到 log in ticket对象这一步写注释，查询凭证，我们查到这个凭证就能直接用吗？其实也不是，你得判断一下凭证当前还有没有效，因为凭证不是有状态，它有可能是已经失效了对吧？

第二个凭证还有一个过期时间，如果说他已经超时了，那也就无效了，所以你这里要判断一下凭证有没有有效了，我才能把它暂存后面才能用，如果无效我就当做你没登录一样。



好，所以这一步我要检查凭证是否有效，那也很好检查。比如说如果说 ticket它不等于闹，并且 ticket不是txt，只是老跟特别大，log in it不等于no，然后 log in take it点，get status。它等于0，并且log in，take it。

Dear get expired there after new date。我这写完以后再解释，你看只有说我们查到的凭证这个结果它不为空，并且状态是0，有效的，并且它的超时间晚于 after当前时间，超时时间在当前时间之后就说明它还有效，否则达到了当前时间就无效。

所以这些人都满足才有效对吧？那么如果凭证有效的话，我就可以认为你是登录成功的状态，我就可以利用凭证去查询user。

那么查询user我们调 User service的点，find user by ID，ID从哪来？从对象中取，这里有这里写个注释就是根据凭证查询用户，那么查到以后，刚才我PPT上也写了，我说了我们查到以后不是在这用，是在后面在模板上用，但或者是在我们CTRL处理这个业务时也可能用，总之我们在后面整个处理的过程中，随时随地都有可能用。

为了后面的使用，我需要把 U的暂存一下，或者说我们换个角度来说就是说在本次请求当中持有用户，让本次请求持有用户随时随地都能拿到好，我怎么去存 user呢？当然有人可能会想我就创建一个公共的类把它放进去，或者是我把它放到容器里会怎么样，其实就是这样处理简单的处理都不对，因为你要考虑什么？

咱们浏览器访问服务器是多对一的方式，就是一个服务器能处理多个请求是并发的对吧？那么每个浏览器访问服务器，其实服务器会创建一个独立的线程来解决你的请求，所以说服务器在处理请求的时候，其实是一个多线程的环境，你要考虑到这一点，所以你在存用户的时候要考虑到独者程的情况。

如果你只是简单的存到了一个工具当中，存到了一个容器当中，那么是一个变量的话，那么有可能会产生冲突的。并发的情况下，我们想把一个数据存到一个地方，让多线程并发访问都没有问题。怎么办？你得考虑什么呢？考虑线程的隔离，就说我每个线程单独存一份，他们互相之间不互相干扰，大家学Java的时候，你学过独者程的话，应该了解到有一个工具就是解决这方面的问题，工具叫此外的logo。

好，所以下面我就想办法要把优点存到那个时代的路口里，在多个线程之间隔离存这个对象，但是这个逻辑我最好把它分成一个小的工具，因为其他地方也可能用到。

好了，当然我就先写这个工具，然后写的过程中我们再去看一下那个时代的logo它是它是怎么隔离的，其实通过源码能看到，我们一边写一边看，我就在 Youtube包下再新建一个类，这个类我们习惯于叫host，当然你叫别的名字其实也无伤大雅，没有关系，按照习惯我们喜欢叫host。

好的，然后这个工具我把它放到容器里是加上一个component，然后这个工具其实主要是它就起到一个容器的作用，这里我写个注释，它起到一个容器的作用，那么它的作用是持有用户的信息，其实说的直白一点，它是用于代替筛选对象的。

因为我们程序中不太想用筛选对象，然后如果有三成对象它可以直接持有数据，而且它是线程隔离的，host其实就代替它的功能。

我们刚才也说了，我在这里要用到的是smart的logo，所以说我把 smart的logo就是初始化一下，所有的logo范型指定它里面存的是什么对象，存的是user，那么这个对象的名字名字我叫users，因为它里边存的是每个线程对应的user，那就是new时代的路口。

好，那么死的路口它主要提供了一个赛的方法，就往里存纸，一个钙的方法取值，那么在存和取的过程中，它用一个很简单的办法就实现了线程隔离，咱们去看一下它的源码，我进入到这个类当中来了，我就看一下，你看他有赛的方法往里存，纸存的是你指定的类型的对象，你看它怎么存的，它首先是spread点carry on the spread，就当前的线程，他先获取当前的线程，然后根据当前线程去获取一个map对象，然后把值存到 map里，所以你看它存数据是根据线程来存的，每个线程得到 map对象不一样，它是通过这种方式进行了一个隔离。

然后你再看盖的方法，他取的时候也是先取到当前的线程，然后取到当前线程所对应的 map，然后的话从 map 中取值，所以说它以线程说白它是以线程为 key 存取值的，是这样一个逻辑。

好了，回到刚才的工具类当中，我需要再给它加上两个方法好，让外界能够调用一个就是 site，user 方法好，让外界能够把 user 传进来，然后我这个工具把它存到 style local 里，那就很容易就 users 点 site，user 就行了。

好再来一个 get，get 就得返回 user，get user 你要从时代的 look 里取 users 点 get 好就可以了。然后我们再提供一个方法就是清理，请求结束的时候，我们把死在路口中的 u 的再给它清理掉，不然的话你每次只往里存，不往外清越来越多，它占有太多的内存不好，所以说用完就清掉。

Public、world 颗粒清理，我们直接调 users。第二为目，你看这个人物方法，他也是获取当前现成的 map，然后把里边的值清掉，所以说也是和线程相关的。好了，这个工具我就做了一个简单的封装，那就可以用了。接下来我们再回到刚才的拦截器当中，我这回就可以用刚才的工具去持有用户了，有人说为什么他能持有？

因为我们在这儿把数据存入到了当前线程对应的 map 里，这个线程只要是这个请求没有处理完，这个线程就一直还在，当请求处理完服务器向浏览器做出响应之后，这个线程被销毁，所以在整个处理请求的过程中，在后续现场一直是活着的，然后死在路口里面的数据是一直都在的，所以这里我就写上好还不行，我还得把 House toda 给它注入进来，好，然后在这儿我就可以 house toda，然后点 site，user 好了，普瑞汉德我们要做的事就完成了，在请求开始之初，我就通过凭证找到了用户，并且把这个用户就是存到了暂存到了 house 头的里，就暂存到了线程当对应的对象里。

好，我们再往下，什么时候我们需要去用 user，那就是在在模板引擎之前就要用，因为模板引擎上要用，所以我们在模板引擎被调用之前，应该把 user 给它存到 model 里它就可以用了，所以我们可以重写这个方法，post handle。还记得 Post handle 是在模板之前掉的，对不对？而且它恰好有 model and view 对象，所以在这可以怎么办？可以先从 host 里得到 User，得到当前线程持有的 user，然后判断一下 user 他如果不等于 null，然后并且 model and view 不怎么弄，因为我最终要把数据传到 model and view 里，我得判断一下万一它为 null 对吧？

好，如果这两者都不为 null，就达到了我要处理的条件，我就 mode and will 点 I 的 object，然后往里存一个值，存的就是登录用户，key 我叫 log in user 值就是 user 对象，我就存进去了，那么紧这个方法执行完以后，紧接着模板就被执行了，那模板在执行的时候，model 里已经有了 user，它就可以拿来用就很方便。

好，什么时候把 house 的厚点里的优点给它清掉，咱们刚才不是要清吗？在整个请求结束的时候，我们就得重写 after 来选这个方法这样模板都执行完以后，最后我要调它的颗粒方法，把数据清理掉就完了。

好，你看虽然说我本次课要解决这个问题，主要用的是拦截器批量的处理请求，但是我们还需要有一些技巧，需要结合着之前所学的内容才能解决掉这个问题，所以说你要把之前的学习到的东西都串联起来，综合的去理解。

当然如果说你独者程那块的知识就是遗忘了，你需要课后做一个回顾，做一个复习，再多看看，那么下一步干什么？咱们拦截器写完了，下一步不是得把它进行一个配置，对不对？所以下一步我就再回到外部 mvc config，我需要把刚刚写的拦截器进行一个配置，和上一个拦截器一样，我要把它先注入进来。Log in take it，interceptor。

好，注入进来，然后注册的方式和一上一次注册也是类似的，我就直接复制粘贴了，我们注册的是拦截器，然后这些静态的资源访问，我们也需要把它排除掉，不处理好，然后我们需要具体来说除了静态资源以外，要拦截哪些请求是全部的，因为所有页面我们都要做这样的处理，所以我就不要去写 ID 了。

好，拦截器我就写完了，写完以后我们在模板上需要进一步处理，因为最终咱们不是要在模板上显示用户的登录信息，就在头部对吧？

我们所有的页面都是复用 index的header，所以我们需要去改写 index的 head的部分，我找到这个页面找到它还得这一部分，好，咱们看一下，首先首页你不用去处理，因为无论是你登陆没登陆，首页我都能看见，谁都能访问，这个不用管。

然后第二个我们看这个消息，这个就不是了，如果说你没有登录，你应该看不到任何消息，别人都没法给你发消息，所以这个地方你得做一个判断，如果没登录不显示登录了，才显示你这里可以用 if th冒号，if等于。

好，我如何判断它有没有登录很容易，你看一下model里有没有有没有 loving user，对吧？如果你登陆了，它一定会有 log in user，因为我所有请求都做了这样的处理对吧？好，所以这里我就写 log in user，它不等于not，那么这个时候才显示它，否则我是不显示的，你要注意好，这是我们对这个消息做的一个处理。

同理其他的也是类似的，你像注册和消息刚好相反，他得是你没登录才显示注册，你登录了就不用再注册了，对吧？所以这里把这个复制一下，这个得是等于now。

Love you又等于now，比如说你没有登录，此时我就给你显示注册，你就可以注册，登录也是一样的，log in u等于no的时候我就才显示登录，否则你已经登录了，还写着登录干什么呢？有没有必要重复登录对不对？然后再来最后的这个地方就是这个位置，很显然这个位置应该是登录以后才显示的，你没登录也没有什么头像对吧？也没有这个用户名对吧？

所以这个地方它和消息是类似的，是 li复制一下，如果用户登录用户不为空，我就显示它，然后在显示的时候下面的路径我们需要处理一下，看一看，下面有些数据需要处理一下，首先这个地方是用户的头像，目前我是写死的，这个得是动态的，你要显示的是当前登录用户的头像，所以你可以这样写 th冒号s二c把这个静态的路径换掉，换成login user的tide URL属性，换成动态的值就好了。

除此以外，咱们下面不是还有一个用户名，它也得显示一个动态的值，对吧？就当前登录用户的用户名，这里用的是Utecht，因为我怕万一有些用户注册它里面带了一些特殊的符号，我就用一个u text，然后这里显示log in user点。User name。

好了，首页我就处理完了，接下来我们测试一下看看能不能行，我重新重启一下点重启一下，重启以后打开浏览器，我先访问一下首页，当前他没有登录，所以你看显示的是首页注册登录没有错对吧？我就登录了，我就用这个 CC账号密码123，这个验证码是gn7c登录密码不正确，我少填了一个3小键盘有点不好用。

Nqec登录不不对，U38g密码还不正确，难道我忘了密码ln10，刚才可能是我第一个字母就敲错了，我也不知道敲什么去了。

反正这回就好了，然后你看我登录成功以后，请求一刷新就跳转到了首页，然后的话这次请求就被拦截拦截到了，就被我们 log in take it in the Sept拦截到了，那么在请求之初它根据登录的凭证得到了用户然后放到了host hold里，并且在 control了执行完以后，他把又取到了，然后放到了model里，我们在模板上就index这个模板上写的这些代码就生效了。

他能够正确的得到 log in order这个数据。

所以你看登录成功以后，我只看到了消息和这部分这头像，然后你展开看，其实也是没有问题的是吧？然后你也可以点退出登录，然后再退出一下，一退出他又跳到了登录页面这个时候，因为退出的时候我们把凭证给他设置为失效了，对吧？

所以在这次警示当中的拦截器虽然找到了凭证，但是一看失效了，他就没有去存放，没有去找 user，没有存放 user，所以说最终页面上我们的逻辑判断是没有登录，因此显示的是这样的。

所以说你看头部根据登陆与否，它变成了一个合理的展现，这就是我们这次课要这个给大家呈现的内容。好了，这次课的目标我们就达到了，课后的话你也把功能实现了，好好体会一下拦截器它的强大之处。我们这次课就讲到这里，咱们下次课再见。

## 2.7 账号设置

16-账号设置.mp4

这次课我们来开发账号设置的功能，那么当你打开账号设置的页面，你会发现这个页面上其实有两个功能，一个是上传头像，然后一个是修改密码，这次课我给大家演示一下上传头像，那么修改密码就作为一个课后的作业，然后你自己把它完成了。

因为修改密码比较简单，它里面没有包含什么新的内容，你利用我们当前所学的这些知识是可以把它搞定的，那么关于上传头像其实就是上传文件，然后我们在上传文件的时候就是要注意三点，第一点就是请求它必须得是post请求，这是规定。

第二个我们在设置表单的时候需要给表单加一个 enc type属性，这个属性必须等于multi part form data，这两点是固定的要求。然后我们在服务器端接收文件的时候，那么spring给我们提供了这样的一个类来解决这个问题，叫multipartfile。

那么我们利用的对象接收文件，然后对它进行处理会非常的便捷。然后我们具体在开发这个功能的时候，你看它挺简单的，但是我们实际在做的时候还是要细分一分，实际上是要分为三个小的步骤。

第一步就是你得能够访问到这个页面对吧？所以说我们先把这个页面给它配置好，让它能够访问。

第二个我们就是要对这个表单进行处理，然后数据能够提交服务器能够接收到文件，然后把它存起来，那么这个文件存哪去有两种办法，一个是存到我们这个服务器上，服务器硬盘里，第二个是存到第三方的云服务器上，那么这次课我给大家演示的是我怎么存到我们的服务器上，然后将来后面我们会有一个专这个专题演示，怎么传到云服务器上。

好，那么当你把这个图像传到了服务器上以后，最终我们在其他的页面上是要用到的，是要读取到的，所以说你还得在处理一个请求，让页面能够通过请求获取到头像，所以说总体上来说开发步骤包含这三小步，好，下面我就给大家来一步一步的去实现这个功能。咱们打开开发工具，首先我们要访问账号设置的页面对吧？

所以说我需要开发的是 Controller，然后简单的声明一下那个模板它存放的路径就可以了。

这里我再新建一个controller，我叫user controller，因为我们当前的功能它是属于用户的一些相关的业务，所以说我新建一个Controller处理与用户直接有关的一些逻辑，然后类上面我先加上注解@Controller，另外我给他加一个RequestMapping，给类声明一个访问路径就叫user。

好了，然后在这里我就加一个方法，然后使得浏览器通过这个方法能够访问到设置的页面。

我先声明一下这个方法的访问路径设置我叫setting，然后请求的方式访问一个普通的页面，不是提交数据，然后这个方法返回死菌方法名，我叫getSetting配置，然后不需要有任何的参数，我就直接return这个模板所处的位置就可以了。

Site那个页面的名字叫的html，所以这里描写的是html。

好了，这个Controller咱们就处理完了，但是处理完以后你还得去去配这个模板，我把这个模板找到，它在这，然后把它打开，然后首先在这配置一下，他没付，然后再配置一下这些静态资源的路径，还有少了一个括号，然后header我们还是复用首页的header，所以这里我需要写replaceIndex，冒号，还得然后后面当然尾部我们不需要做任何处理，后面的GS你也需要处理它的路径，好了到这儿。

塞汀的xml，关于这个模板的一些基本的配置我就配置好了，配置好以后这个页面你得通过我们网页的头部点一个链接链过来，所以说我还得去改链接。那么那个链接它是在Index点html当中复用的，所以说我打开这个页面，然后找到账号设置的位置就是这了，把它改一下。

那么当前它的路径是user对吧？然后setting好了，关于页面的配置就配好了，配好以后现在我们可以测一下，看看能不能点它，然后访问到这个页面我们试一下，我就把应用的重新启动一下。

好，启动完了我打开浏览器，我首先访问一下首页，当然如果你没登录的时候还看不到账号设置，因为我们在这做了处理，所以我还得登录一下，好，登录成功以后就看到头像了，点三角就能展开这些链接，我现在要点的是这个账号设置，那么鼠标悬鼠标的悬停到这个链接之上，左下角出现的路径应该是对的，然后点一下 ok 这页面能访问到没有问题。

好了，这一步完成以后，下一步我们就来实现，其实是我们这个功能其中的一个关键的部分，就是我们怎么去上传头像，怎么去上传一个文件，那么要想上传这个文件，首先我还是需要在配置文件里做一个简单的配置，就是说配一下这个文件最终要存放到哪个硬盘上，这个需要做成可配的，因为我们开发的时候我们是用的是 windows 系统，你存到 c 盘 d 盘这都可以，我们上线以后可能就换成了 Linux 对吧？

这个路径肯定是要变的，所以你要做成可配的，因此我先打开配置文件，好在下面先打开配置文件，在这我配置一下我们上传的资源存放的位置，那么 key 我还是仿照以前的写，can 没有那些点 pass 路径点，不漏的就上传的路径，等于我把它放到我的地盘 work 之下，然后新建一个 data 目录，然后 out no 的我就放这吧。

好这就可以了，做好了这项配置。

这项配置做好以后，这次请求我们还是按照这个服务器三层架构，我们从底下往上去开发，首先考虑我们这个数据访问层要做什么处理，其实没有什么要处理的，因为我们上传文件是把这个文件存到硬盘上的某一个位置，而不是存到数据库里，对吧？

所以说数据仿真代码就没有。然后再考虑业务层，业务层我们确实需要写一些代码，其实主要写什么？主要写的是你上传完文件以后，你最终是要更新用户的。还有一段 URL，因为你是传的是头像对吧？传完以后用户的头像的路径就要变，所以我们 Service 服务端就需要提供这样的一个改变用户头像的这么一个功能，然后上传文件这个事儿我们在 CTRL 那里实现就可以了，这表现层实现就行了，我就不再 service 层实现了。

为什么？因为我们最终处理文件要用的是这个对象，而这个对象它是属于 spring mvc 的一个对象，它是属于表现层的一个对象，如果说你把它传给了 service，那么使得业务层和表现层就有了耦合不太合适，所以说我们就直接在 CTRL 那里把文件存了就可以了，那么业务层只处理了更新路径的这么一个需求就可以了。

好吧，我就打开这个 service 就在 You 的 service 里写，那么我在这个类当中再追加一个方法，这方法比较简单，因为是要更新修改头像的路径，所以说更新完以后我们可以返回更新的行数，那么方法名我叫 update 拍的，那么你要更新用户的头像，首先你要告诉我是哪个用户对吧？所以你应该把 user 的 ID 告诉我，其次你要把路径更新，为什么？你把路径就 hider URL 告诉我就可以了。

那么我们更新的时候就调 map，迈克尔好像之前我们写过更新头像的方法，我试一下调一下试试 User map 点 update head，你看我们很早以前写过俩参数，第一个是 ID 传进去，第二个就是 head，URL，当然我需要回测一下，好这个 service 我也写完了，写完以后接下来我就再去写 CTRL 了，我们再追加一个请求来处理本次上传文件的这么一个请求。

然后我们在上传文件的过程中，我就需要用到刚才我配的条件，除了那个以外，其实还需要用到这个域名，咱们这个项目的域名，还需要用到我们这个项目名，为什么需要一会我们写代码时你就知道了，我先把这几个条件把它注入进来，但注入之前我这样，我先把 logo 室内化一下，因为我们在处理具体的逻辑过程当中，基本上都得用到 logo，当发生问题的时候都要记日志。

User controller 点 class，好，然后注入几个值，通过 value 固定的值我写错了，第一个我先把上传路径注入进来，t 我 copy 一下，以免写错。

好，那么说明一个变量去接受一下这个值叫 up low 的 pass，另外还需要用到域名，把域名也传进来，这个域名的注入方式和这个一样，我 copy 一下，那么域名是 key，复制粘贴过来，然后把变量的名字改一下，改成叫读命，然后再来一个再复制一下，还需要用到这个项目的访问路径，那么配置文件里也是有的好粘过来，然后变量名改成叫康 txt pass，我把这几个参数注入进来，做好一些准备。

当然了我们在处理上传的逻辑的时候肯定得用到 user service 对不对？我们刚才写的方法，所以说把 user service 也注入进来，然后我们更新的是哪个用户的头像，就是当前用户的头像，对吧？不是别人的，当前用户是谁，那就得从 house 的后头里取，是吧？

所以说还得把 house 偷的也得注入进来。所以说别看这一个功能其实并不麻烦，但是我们写起来的话其实考虑的还挺多的，很多地方都比较灵活，所以说需要引入不少的内容。

好了，接下来就要我这些工准备工作当中处理完了以后，下面我就要开始处理上传的这次逻辑请求，我就再新写一个方法，先声明一下这个方法的反路径，Pass 等于 up load 上传对吧？然后 master 等于 request，master 点 post，这是一个 post，请求刚才我 PPT 上不是说了吗？我们上传的时候表单的提交方式必须为 post，这是固定的要求，然后我声明这个方法来处理请求，这个方法名我就叫 upload either。

那么这个方法我们接收的就是页面上的参数，一个数据就是一个文件而已，我用什么样的类型的参数来接收刚才我 PPT 上所写的，所以只不过 mvc 给我们提供一个专有的类型，叫 market part multi fire。如果说你页面上传进来的是一个文件，你就声明一个这样的类型，如果是多个你可以写一个数组，我们这只有一个。好，那么传入的是一个图片，所以说我命名的话就叫 either image。另外我们响应时可能需要向页面返回一些数据，所以说我需要声明 model 用它来给模板携带数据。

好了，接着往下写后面的逻辑，首先我们应该判断一下这个参数你传入的有没有问题，是不是空的，如果是空的，我们得给一个提示，对不对？

所以在这里我做一个判断，head emit，如果说它等于 not，我就不用往下处理了，我直接给提示，那就猫都有点爱的 action 就写一个！一个错误的提示来说，您还没有选择图片，然后我就 return 那么如果你页面点上传，访问了我刚才的方法，当我检查到你没有参数，你这个有问题，我肯定回到这个页面给一个提示对吧？

所以说我要 return 回到这个页面，这个页面的路径不就是它好，这是一个判断。

如果判断没有问题，这个文件是存在的，我们接下来就要上传这个文件，那么上传文件的时候你注意，你不要按照原始的文件名来存，比如说它传的是一个一点偏计，你不能就直接把一点偏计存到硬盘上，因为可能很多人传的文件都叫一点 PNG 对吧？会产生冲突会覆盖，为了避免覆盖，我们一般会给这个文件生成一个随机的名字不重复的，这样的话肯定是每个人都不同，就会避免这样的问题。

然后你名字是随机生成的，但是这个文件的后缀你不能变，比如说它传的是偏近，你存的时候就是一个随机的名字点偏近，你不要把后缀给它替换掉。

好，所以这样我就先读取一下这个文件的后缀，我把这个文件的后缀先暂存一下，先读取一下这个文件的后缀，我需要得到这个文件名从名中去取后缀，是这样的。我就调用 hide image 点 get，它有一个方法叫 origin of file，name 就是它原始的文件名用户上传的时候他文件叫什么名？

就是原始文件名，然后就得到这个文件名的话叫 file，name 这个范围内它很有可能它会有后缀，当然也有可能比较特殊的用户，他传了一个图片，他没有后缀，这样我们就不处理，我们只要带后缀的，因为你带了后缀，我才知道这是一个什么格式的图片，对吧？

否则我不知道你是不是合法的东西，我现在要从这个文件中截取出后缀也比较容易，你就可以 file、name、点萨姆斯菌对吧？

那文件是什么点什么，我们把那个点最后一个点之后的内容截取到最后一个点的位置，你得算出来，这也很好算，Cm 点 last index of 最后一个点的索引。

这句话的意思是我从最后一个点的索引往后截取，就把整个后续都取到了，对吧？这个后缀我给它命名叫 surface 后缀。这个后缀有可能是空的，有可能他传的文件真的就没有后缀，你最好判断一下。我利用死病 you tubes is black surface。如果说它是空的，我认为不对，我要给一个提示的方式，其实还是这样 Copy 一下，还是！然后提示说这个文件的格式不正确，然后回到这个页面就行了。

好，这是第二步判，就是判断获取后缀，然后判断一下后缀合不合理。好，如果说没有问题，那么接下来我们就要生成一个随机文件，名是一个随机字符串，后面再带上后缀。

好，接下来写一下生成随机的文件名。我们就利用那个工具肯定没有那些YouTube点，这样的ID就是一个随机字符串，再把 Surface加上就可以，非常容易，然后我得这个范围内不就变成了新的值了，发生变化，我们存的时候就是按 name来存，你存的时候去还得确定一下 file它存放到哪个路径下，我们需要给一个权限定名才能去存对吧？

所以说我们还要确定一下文件的存放的路径，然后才能存。但这个路径也很好办，我们不是做了配置，不是有一个upload pass，我们就把它放到upload pass里，这样我就根据它存放的位置初始化一个file，当然是一个空的file，然后准备往里面传东西，这是我们要存放的目标，我叫desk等于new file。

然后 New file里面你要指定这个路径路径很容易就是upload pass存放的目录，然后加上一个斜线对吧？再加上我们三项内容这就可以了。

好，我现在就指定了说这个文件要存到这个位置叫这个名字。File空的目前还没有内容，我们需要把head image里面的数据写入到这个文件里，那就很容易。

赛尔的操作很简单，它的APA你就直接这样，还得emit点儿，有一个transfer to的方法，直接指定dast目标文件，他就把当前文件中的内容写入到目标文件里去了，非常容易，但是这句话我们需要处理它的异常。

这个得揣开始我把拆开怎么办？就是把这行代码选中，然后你点 Code，然后这里边有一个surround位置，利用什么环绕在这段代码之外套上什么？格式？我想套上 try catch，这样，那快件我没记住，你可以自己记一下。

好，如果遇到了异常，我们肯定得先记个日志对吧？捞个点air，记住这个错误，说上传文件失败，然后把异常的信息也一起记录下来，然后这个时候我再抛个异常抛出去，抛出去以后就打断整个程序，这个异常由谁来处理？将来我们会讲怎么统一处理这个异常，就是controller他所抛的一切异常我们可以统一处理，现在我们只管抛后面的处理。好，这里我 Niu run time。

Xp。搞错了，ragtime except然后给个提示说上传文件失败，服务器发生异常。好，然后再把 e也带上，这样的话他带的信息是比较完整，便于调试。

好，这就是存储文件，这里写注释这句话就是存储文件，那么存完以后最后还有一步，如果你存成功了，这段代码过来了就存成功了，怎么办？我们得把当前用户的头像的路径给它更新了，那就是更新当前用户的头像的路径。

你注意我们所更新的路径是什么样的路径，可不是我们这个路径，论证是啥？是d盘work。Data什么？你想访问者他能通过d盘 work这样的路程访问到你的资源，访问到这个网站的资源肯定不能对吧？这是你本地的路径，你要给人家提供的是啥？是外部访问路径，是外部路径，外部访问路径是啥？就是这样我写一下，就这样HTTP冒号双曲线QQ host，冒号8080，当然你系统上线以后，Ip端口号就会变成域名被替换，然后是项目名，肯定那点，然后后面是啥？

当然这个路径是由我们自己来定义的，比如说我这样定义，我定义的路径叫user，然后是header，然后后面是文件名，什么点PNG什么就是随机的字符串，点PNG就是后缀当然未必是评级，也可能是别的，我规定的路径是这样，当然前面是一定的，后面的话是我自己定义的。

一会儿我们再给用户提供读取图片的方法的时候，就要按照这个路径来处理，这样的请求是这样。好了，既然如此我就继续处理了，这里说更新当前用户的头像，你得先知道当前用户是谁对吧？这个很容易我们可以通过house获取的大型用户对吧？然后我就要更新它的 head，我先把 head路径给它，按照这个格式给它拼出来，这不能写死。我就写上string，hide URL等于拼一下。

首先前面这段不就是域名，对吧？我们已经注入进来了，do may，然后中间这一段不就是项目名对不对？那就是contact pass，再往后这段就是当前control的访问路径对吧？固定了我们不会变优点优，在下面这个high点这一级也是固定的，不会变，然后后面的文件名这是动态的，因为每次传文件名不一样，是随机生成的，你得加上范围内比较亮，好了，就是我们上次允许外界访问的外部路径是这样的，我就把当前用户的头像更新的路径调 Service就行了，点update header，然后把当前用户的ID传入，对吧？

然后把更新以后的hider传入就可以了，那么到这儿整个的流程就完成了，完成以后怎么办？完成以后比如更新成功了，我们就直接重定向到首页跳到首页，然后你有人说跳到首页我能看出来对不对吗？能因为你更新的是头像，他的首页请求一刷新这头像变了，你看就对了是吧？我们就不回到这个页面了，就去首页好，所以这里我就return read write重新下到首页的仿路径index。

好这就完成了，完成以后这样我把直接把第三个请求也写了在这，然后最后再统一处理这个页面。还有一个请求就获取头像，所以CTRL需要提供这样一个外界获取头像的这么一个服务。好，我就直接再加一个方法，瑞克斯卖品说明了它的防路径是什么，就不能随便写了，因为我们这规定的是这样的，你得按照这个来处理对吧？

当然当前的类访问路径是user，我这个方法的路径显然就是header，head然后head之后跟的是文件名，这个文件名是直接拼到路径里去了，不是通过问号的方式去拼的，所以我应该是这样，斜线后面加上一个变量叫CNN然后 Master等于request点，这是概念，只取一个头像的路径。

刚才的是post有区别，好，然后这个方法我返回值声明为y为什么呢？因为这个方法比较特别，它向浏览器响应的不是一个网页，也不是一个字符串，而是一个图片是二进制的数。这个比较特别，我们需要通过流手动去向浏览器输出，所以说我就返回值写成y的，然后这里边我就直接手动的调 Response，然后往外写就可以了。

好，那么方法我叫get hide， get hide。然后首先我们需要从路径中解析到 Final name这个参数，对吧？你得用到 Pass variable password31name，把它付给非常内部参数，另外最终我们向浏览器响应的是一个特殊的東西，我们用response去拼，所以我还得声明 response。好继续，首先你给我传入的是这个文件名，我怎么去找这个文件，我上哪找呢？我们是不是得上这个路径上去找对吧？

因为我们服务器存到了本地，存到目录下，所以你还得知道找到我们服务器上存放的文件的的路径和名字，这个很好找，就是服务器它存放的路径就是upload，pass加上一个斜线加上范围内，好，那么我重新覆盖变量，这个范围内经过这样处理以后，它就变成了一个带上本地路径的概要内容是一个权限内容好了，然后我要向这个浏览器输出的是这个图片，你输出的时候咱们之前输出过这个html对吧？

你输出的时候你得先声明我输出的文件的格式是什么，这个文件的格式是什么？其实就是文件的后缀不就是它的格式吗？Png JPG对吧？所以这里我们还得去解析它的后缀，解析也很好，解析之前我们都写过一次了，就再写一次。文件的后缀， Surface等于file name点儿sup死运，然后 for name点last index of点。

你看还是刚才的逻辑，就是说我先获取文件名最后一个点，然后从这个点的位置往后截取就得到后缀，好，那么得到这个后缀以后，我就要开始响应图片了，写完这个文件了，我得写response，点set content type这个文件的格式是这样的，固定的 image斜线后面带上就是文件的忽略，我就拼上 surface就可以了。

好，那么然后我们想用这个图片它是二进制的数，所以说我们得用到这个字节流，我就 response点 get output, screen, 获取字节流， output stream。好，当然这句话我们需要对它处理异常，涉及到io操作，这里还是得再看去。Surround with try catch。好，这样我先不着急，先把异常的情况出现异常的情况打个日志。Log点era，这里说读取图像失败，加上明确的做个消息。然后在这儿我就得到了输出流，我要输出文件，那么首先我得创建一个文件的输入流，好去读这个文件一边读一边才能输出对吧？

所以还得创建一个文件的输入流， File, input, string, SIS等于new file input词卷，那么你要传入的是 file name，它会读取一个文件得到一个输入流，然后有了输入流以后，我们就要开始输出了，输出的时候你不要一个字节一个字节输出，你要建立一个缓冲区对吧？

比如一次最多输出1.24个字节，——批输出效率高一点，我先说明一个缓冲缓冲区，8所 new bat指定为1024个字节，好，然后需要有一个游标，然后利用while循环去输出 while。B等于file, input, strain, 点read，那他每次read最多8分这么多数据，每次遇到的数据就把它复制给b然后最后我判断一下 b是不是等于不等于-1？因为如果说你不等于-1，就是读到数据了，等于-1的时候就表示没读到，没读到的时候我就结束这样的。



好，所以不等于-1的时候就继续等于-1就结束，每次循环的时候我就输出最多1024字节的数据，我就output stream点Write8分这么多数据，然后但是从0开始写到b这样好这就完成了。

当然了其实输出流最终sim VC会自动的将它关闭，因为瑞斯邦斯是由他管理的，但这个输入流是我们自己创建的，他不会帮我们管理输入流，所以这个输入流需要我们手动的去关闭，应该是在find类当中关闭比较合适，那么你可以这样写，在try后面加一个小括号，然后把这个代码挪到这儿来，这是Java7的语法，就是说你这样写这里声明的变量最终它在编译的时候会自动加上final，然后在final里关闭它，前提是它有close方法，它显然是有的。

所以这块稍微注意一下。

好了，或者干脆我们把outputs dream也加上也可以，需要关闭的我们就得在这儿都声明了，让它都关闭好了，像浏览器响应图片的方法我们也写完了，最后就是要处理这个页面上这个表单怎么配，就配好以后不就可以传过来了，对吧？

下面我们就打开赛艇页面，然后去处理表单。好在这个页面是比较简单的，因为它只有一个数据对吧？只有一个数据，然后我们先配一下Phone，首先MR等于第一个，第二个enc tap等于固定的值，马蒂帕特from，data固定是这样写，然后你还要写上提交的路径是多少，提交的路径是user，然后up no的对吧？就是他另外文件框也得加上内容，它能够提交加上内容。

Name等于你得和controller匹配起来，controller我们传的时候叫header，image你要和它对应是吧？Ok最后还要考虑一下，如果报错，我们回到这个页面，我们在哪显示这个错误的消息，把这个也处理好。

那么仔细一看，这个页面它没有在这个地方预留一个提示错误的一个空间，我们可以自己加一个，我们可以参考之前我们做过的注册登录的页面去加一个，好，我就打开这个登录页面，看一看登录页面是怎么处理的，随便找个框，比如说这个账号首先它它是在账号的框的，后面有个div这div class是叫invalid feedback，然后这里边有一些数据，我们在这里去显示错误的消息，然后注意你加上样式以后它默认不显示，直到它前面的框有了样式它才会显示它俩是联动的，所以说我们还得动态的去处理样式，这样才可以。

好，我就回到setting，我把刚才的代码加到这个框的后面，你可以加到label的后面在最后加上就可以了。

然后这个是txth冒号，tx等于我们返回的叫错误消息，这里边其实显示什么都无所谓，因为它会被变量给覆盖，然后默认它是不显示的，我们需要这框里拼上is invalid，样式它才会显示，但是这个得是动态的拼。

这样我换一下，因为有点长了，更大的拼。

Th冒号，这里是一个静态的数据，再拼上一个动态的数据，所以我加上一个双竖线，然后这里加个空格，拼成一个动态的变量，我得看一下这个错误消息是不是为空，如果它不为空，我就要带上错误的样式，这里就要拼上，意思是横线。意思。明白了，是不是这样写？我确认一下Log in点etml意思。明白，干脆复制一下，以免这写错了。好，然后如果说l说是not我就不显示就这样好了，现在我表单也配，下面我们就可以做一个最终的测试，看一看行不行？

我把这个服务重新启动一下。

好，启动完以后打开浏览器先访问首页，刷一下，然后我要访问这个账号设置的页面访问没问题，然后我要上传一个文件，我选一个图片，我本机图片的文件夹下有几个图片我就随便选一个选一，选完以后我就点立即上传，他报错了，报错以后我们来仔细看一下错误的提示，然后我们看一下是什么问题。

首先我们先看一下这个页面上的提示，因为它报道错了，首先页面上也能看到，你说的话它提示了你说status是500，500之前我们也说了这个状态，比方说服务器有问题，后面它也有了一些提示，我们看一看这个提示有没有对我们有没有帮助，你看议程我们主其实主要看的是什么，看的就是这种cosplay这样的地方。

Cosplay表示说这个问题出现的原因，然后看异常的类型是file, not found exception，它的意思是这个文件不存在，这个文件为什么不存在？我们再来看一下，但是他没有明确的提示为什么不存在，我们可以看看再看看这个Idea的控制台，看看他有没有什么更详尽的提示，其实有，你看他这里提示的稍微更准确一点。

这有中文，刚才那地方其实也有，但是它乱码的返回到这个页面上，你看这个地方他说系统找不到，指定的路径找不到，那是因为什么呢？

我这个路径还没有建，我只是声明了还没有建，我把这个东西建一下d盘，然后data不漏的，我把目录创建好之前没有，好了，再回到浏览器，我们再测一下回去退回去，然后再点一下立即上传，这回没报错，然后重定向到了首页，我们也看到了图像有了变化就说明ok了是吧？

我们可以去不露的目录下看一下有了，是不是？而且这个文件名是随机的，我们再试一个账号设置我再选一个我选的猫然后上传也是ok的。到这我们这个账号设置当中的上传图像就完成了，那么课后你把完成以后把修改密码自己把它也实现了，这次课我们先说到这里，咱们下次课再见。

## 2.8 检查登录状态

17-检查登录状态.mp4

这次课我们来实现检查登录状态的这么一个功能。首先说一下什么是检查登录状态，为什么需要检查登录状态，这个我需要通过我们的项目给你演示一下。目前我已经把这个服务启动了，并且我已经打开了首页，当前我是没有登录的状态，在我没有登录的时候，我是看不到个人设置相关的内容的，因为我们在这个页面上对它做了处理判定，对吧？

它虽然说我们看看不到，我没法去点，但是如果说用户他知道你的功能的路径的话，它是不是可以直接敲路径去访问？

显然是的，比如说我知道我敲一下试试，那么用户设置的路径是这样的，user setting这里都有记录，我一回车你看也访问到这个功能，你想这是不是一个系统的漏洞，对吧？

本来是你没登录的时候应该不能访问这个功能，但现在却能访问，但这只是看，那么说如果说用户没有这个登录的前提下，他访问了一些比如说管理员才能访问的功能，访问一些特别敏感的功能，他甚至去尝试去提交了一些数据改变了，数据这个就糟了。

所以说这种行为对我们的系统会造成很大的安全的隐患，因此不是说不让他看见就ok，也一定是他无论如何都访问不到才可以。

所以说这个得怎么处理，那也很显然就是说当用户没有登录的时候，在访问这些不可以访问的功能的时候，你应该在服务端进行一个判断，你登录了可以访问没登录拒绝对吧？

应该是有一个判断的。

那么也很容易想到，其实不是说一个功能两个功能有这样的需求，有这样的判定，可能将来我们功能越来越多的时候，很多功能都需要做这样的判断，那么众多功能众多请求都有一样的逻辑，我们用什么处理呢？

你也容易想到应该使用拦截器对吧？

我们之前不是用拦截器就处理了批量的请求，那么拦截器怎么使用，我们之前也用过，但这次我想我们再变通一下，用一个也是我们工作中比较常见的一种应用方式，但是它不是一个没有什么新的语法，只是说我们的一个使用习惯而已，你最好了解这个习惯。

我们工作时有的项目主要是习惯于这样，就是说我拦截器拦截哪些路径，我不想去配置文件里去挨个指定，你想拦截哪个方法，我希望在那个方法上加一个注解，加个注解你就拦截它，不加注解就不拦截它，大家很容易能理解这个意思吧？

所以我们就需要这样去处理问题，首先你注解，我们得自己定义，第二个我们怎么去识别你当前拦截的方法上有没有注解，所以这两个是我们这次课的一个关键。

那么自定义注解其实你学账号语法的时候，如果你学的比较全面的话，这个应该是可以学得到的，但如果你没有学到，你把它忽视了也没有关系，这里我稍微讲一下，也很容易理解。

那么如果我们想自己定义一个注解，我们需要用原注解定义我们自己的注解，那么常用的原注解有4个，第一个叫target，这个注解用来声明我自定义的注解可以写在哪个位置，可以作用在哪个类型上，比如说作用在类上，方法上属性上由他来声明。

第二个叫retention，注解用来声明，你自定义注解它保留的时间或者说有效的时间，你是编译时有效还是运行时有效，用它来声明。第三个是document注解用来声明，你自定义注解在我们生成文档的时候，要不要把这个注解也带上去？

第四个是inherited，那么它是用于继承的，比如说一个子类继承于父类上有自定义注解，我子类要不要把注解也记录下来，用它来指定，总之我们自定义一个注解，基本上前两个原注解是一定要用的，你一定要声明，注解应该可以写在哪儿？注解的生效的周期是什么？至于说它能不能生成文档，能不能继承，你不写就默认就不能，所以这两个你可以省略。

好，那么我们利用原注解定义出自己的注解以后，那么我们在程序运行的时候，我们怎么去读取到注解了，那就得通过这个反射了，我们通过反射的 Mass 的类型对象调用这样的方法 get declared under tensions意思是获取这个方法之上所有的注解，你可以从中找有没有你要的，或者你可以这样。

Master的点get education，然后指定一个类型，按照就只或尝试获取类型的注解有没有？

好了，那么接下来我们就用自定义注解的方式去使用拦截器，然后解决检查登录状态的问题。

好，我打开这个idea，我先把缩小一下，我第一步就先把注解先定义好，这个比较简单，那么注解的定义，我新建一个目录，新建一个包，叫按照推选，因为I know teacher就是注解的英文单词，然后我在I know teacher包下，我要新建一个注解new class，然后你选一下选成按照推选好，那么注解是用来标识，说这个方法它需不需要在登录的状态下才能访问，对吧？

你是登录了还访问，还是说不登录也可以访问，需不需要？你是处于登录的状态，因此基于这样一个需求，我给注解取个名字叫 log in required，就是是否需要登录的意思。好，ok，这个注解我们需要用原注解对它进行描述，加上它get声明一下，这个注解我需要把它写在哪儿上才行，那应该是你的方法上。因为我们拦截器拦截的不就是方法对不对？

好，所以说这里你要这样写，Element type，点，element、type，点，method就是注解可以写在方法之上，用来描述方法，接下来撰写 retention，声明注解时机有效的时长，那这些retention policy点run time，我们程序运行的时候它才有效，是这样的，这就完了，注解里边其实什么都不用写，这就可以了。

它只起到一个标识的作用，我打上这个标记你就必须得登录才能访问，不打上标记，随便，所以只要有这个标记就可以，这就完了。

好，完了以后接下来你用它标记一下，我先用它标记一下。我们程序中其实现在有很多请求了，那么哪些请求需要登录以后才能访问，这里其实有两个，一个是 Setting，因为你只有登录以后你才能去进行账号设置，才能改密码，才能上传头像对吧？你都没登录就无从谈起，上应该写上，其实应该是log in CAD写上自定义的注解。

另外阿布路的孩子这个也是你必须得登录，你才有可能可以上传头像对吧？这可以这个也加上去，然后其实获取头像不用，为什么？因为如果我没有登录的话，其实我也可以看别人的头像对吧？我也可以访问别人头像，因此这个方法不登录，也是可以访问的。

然后至于其他的方法，比如说首页你登录都能访问对吧？登录你没登录也可以访问，因为没登录我才要去注册和登录是吧？所以说其实目前来说我们只有这两个方法需要加注解，那我就加上了，加上以后下面后面我就要用拦截器尝试拦截带有注解的方法，然后在拦截到这个方法以后，我就判断你登没登陆，登录了可以不登录我就拒绝，所以下面我要写一个拦截器来处理后续的逻辑。

好，那么拦截器我还是觉得 Intercept目录下，那么我就新建一个类，这个类我叫 log in required intercept。那么首先我们在它之上加上小加错了，加上component，另外它需要实现一个接口，叫 handler intercept对吧？那么我们应该在什么时候去判断，你登不登陆显然应该是在请求的最初，对吧？如果他都访问完了，你再去判断就晚了，既然如此我们就应该重写推憨豆方法在这里去判断，当然我默认为天数，然后只有你不满足条件的时候，我才return force。

这样这里我要判断最终判断的是你有没有登录对吧？你有没有登录怎么判断，你就尝试去获取当前的用户，如果能取到就表示登录了，否则就没登录。我们就得用到 House toda对吧？所以我把 house toda 把它注入进来。

好在这里我要写了，首先怎么写？首先你得判断一下 Object是什么，是我们拦截的目标。我判断一下我拦截的目标是不是一个方法，是方法我再接着处理，不是方法就不要处理，因为它拦截可能还拦截的是静态资源，其他的内容所有的形式它都拦截，但是我们只判定方法，所以我要判断一下handler他是不是属于这个类型叫 handler method，这是super mvc提供的一个类型，如果说它拦截到的是一个方法的话，那么这个对象将是类型，这是它隐含的规则。

如果是这个类型目标就是方法，既然它是这个类型，我就可以把它转成这个类型，我们就不要基于 update类型，因为我们这样的话不方便获取它里面的内容，我就可以对它进行一个转型。Handled myself等于handle把它转型一下，这个类型我们可以调用它的一个方法get master直接获取它拦截到的master的对象。那么有了master的对象以后，我们可以尝试从这个方法对象上去取它的注解，我就这样master的点儿get education。

按照指定的类型去取，我就不取所有的，我需要的类型就是 log in required。第二class对吧？我尝试去取注解好log in required当然有可能它是空的，所以我要判定一下，那么如果说它不等于空意味着什么？意味着这个方法是需要登录才能访问的，对吧？我这会就在判断你有没有登陆，我就从这里取user就知道了对吧？我就house的后者叫get user，那么如果说get user等于，那这个就是错误的情况，就当前方法需要登录，但是又为闹你没登录，这个时候就有问题，对吧？

这个时候我就应该是有特放c拒绝后续的请求，拒绝以后我得给他一个响应，得给他一个反馈对吧？你去哪我们这儿可以这样处理，我们就强制他重定向的登录页面，你没登录，但你又要访问这个功能，你就去登录去。那我就这样写resource，因为我们当前的方法不是有这个类似方式对象对吧？我们利用它进行重定向 response，因为这个方法它是接口声明的，你就不能像CTRL的那样随便return的一个模板，我们需要用response去重定向，那么response重定向是散的ready？

其实CTRL那里我们return read write其实它的底层也是这样写的，好重定向到哪？Log in路径，而log in前面的话可能还有域名项目名那个项目名，我们可以通过配置文件注入一个参数进来，也可以这样request点that contact us，我们从请求当中可以直接取到应用的路径也可以，好，然后加上毛病这样这逻辑就完成了就可以了。

那么拦截器开发完以后，你需要对它进行配置对吧？指定它的生效的路径，我们配一下。

当然有人说你这不是拦截了所有请求，然后你从中找哪个是带注解的，你怎么还要配？其实我配的目的是要把静态资源的请求排除掉，因为我们根本就不用去拦截那些请求，拦截那些请求是浪费效率把它排除掉，而其他的形式我们都拦截，然后我从中找出带注解的去处理，是这个意思，好我打开配置，然后把刚刚写的拦截器注入进来。

Love in recruit in the sector，然后注册的方式跟他们一样，跟这个一样，把它注册上要求他不要去处理这些静态的资源，而其他动态资源他都处理，他都处理，但处理的时候我们人为的筛选了带有注解的那一部分，其他的不管，这样就这样的好处是我希望他处理谁就在谁上加注解，而不是在这里一个去加路径，因为如果路径多的话也挺烦的，是不是？

这样也挺方便，这是我们在工作中也经常用到的一种方式，这就完了。

处理完以后咱们试一下，看看行不行，我把这个项目给它重新启动一下点，它提示我停止，然后重新启动，是的。

好，等他重启完我再打开浏览器再刷新一下页面，这是没登录的状态。

这回我再去敲路径，强制硬要访问那个功能，看看能不能访问到 user setting，你看我一访问没访问到，他给我强制呈现了登录页面，那就意味着我们刚刚写的拦截器起了作用，这样的话我们项目才算是比较安全，你想硬访问是不可以的。

好了，这次课我们检查登录状态就搞定了，那么到这儿我们第二章的目标就达成了，我们整个登录模块主要的功能就完成了，当然其他的还有一些小的功能没有完成的，是需要你自己去尝试去把它自己实现了。好，这次课我们就上到这里，我们下次课再见。

## 第3章 Spring Boot进阶，开发社区核心功能

### 3.1 过滤敏感词

18-过滤敏感词.mp4

从这次课开始，我们要学习第三章的内容，这一章的目的其实还是要加强对死人不得的实践。然后经过这一章的学习，我们会把游客社区的主要的核心的功能都开发完，并且我们在开发的过程当中会穿插着去学习过滤敏感词，事务管理，是不是LP等等这样的一些内容。

好，那么这次课我们第一节先来介绍一下如何过滤敏感词。那么过滤敏感词我想大家应该都知道，就是说我们在访问某一个网站的时候，我们在这个网站上发布一些内容的时候，那么这个网站它一定会对我们所发布的内容进行过滤。

那么如果他发现我的内容里包含一些色情暴力等等这些非法的词汇的时候，那么它会把这些词汇隐去，或者说它不显示这些词汇，或者是把词汇打码显示为星号，这种行为就叫过滤敏感词。

那么从技术角度来说，其实过滤敏感词很简单就是说对字符串进行过滤，那么你输入的内容不就是一个大字符串对吧？然后我要调用某些API去判断这个字符串里它有没有敏感词？比如说我们可以使用JDK自带的方法，string里就有一个replace方法，可以直接把字符串中的一个词替换掉，非常简单。

但是网站在实际的运营过程当中敏感词会比较多，有几十个甚至几百个都有可能，然后你输入的字符串也可能会长，因为你发布的可能是一个文章，那么在这种情况下，你用刚才所说的那种replace方法去一遍一遍的去替换，性能就太差了。

所以说在实际的工作当中，我们其实不会用这样的比较原生的API，我们通常会采用前追溯这种数据结构，然后来自自己实现一个敏感词过滤的算法。所以说我们这次课主要就是讲如何使用这个数据结构来自自己实现一个过滤敏感词的算法，讲这样一件事。

然后下节课我们讲发布帖子的时候就能把这个内容用上，那么所谓的前缀数它有几个别名，那么它的英文名叫tree这样写的，这没写错就是这样的，然后它还可以叫做字典数或者叫查找数，那么它的特点就是查找效率是非常的高的，但是它是用空间换时间，它比较消耗内存，但不管怎么说，它通过一定的内存的消耗，能够大大的提高查找的效率还是很有价值很有用的。

然后它的应用的场景主要是这么几个方面，第一个是字符串检索，就像我们过滤敏感词找这个字符串里有没有敏感词。第二个是词频统计，比如说我想统计一篇文章当中哪个词出现的次数最多，它是热点词这样。第三个是这个字符串的排序，比如说你要对一个班级的所有的学生按照名字来排序，其实也可以用这种方式来排，或者你想对某些句子或者是某些文章等等排序也都可以。

好，这就是我们对前追溯的一个简单的介绍。那么这次课我们最终目的是要开发出一个能够适用的工具，这个工具我给它取名字叫敏感词，过滤器其实就是一个工具，然后那么主要是分为三大步，第一步我们得先定义前追溯数据结构，第二步我们得根据敏感词，然后把这个数据结构初始化好，然后才能去用。

第三个我们编写出它过滤敏感词的这么一个方法，就是一套算法，大概是三步，那么在实际写代码之前，我想给大家再画个图，详细解释一下前追溯这个数据结构大概是怎么样的，它有什么特点。

然后再一个我们怎么就利用它能够过滤敏感词，它怎么效率就高了，我们去体会一下它的算法执行的过程。

你了解这个过程以后，你直观的了解以后，那么一会我在写代码的时候，你也知道我为什么这个地方出现这个对象，那个地方我是这样去处理，其实和我一会要讲的逻辑是匹配的。

好，下面我给大家画一个图来直观的讲解一下前追溯它固定敏感词的算法的逻辑。

好，那么你要固定敏感词，我们得做一个假设，我这个项目当中敏感词有哪些，我得先定义好敏感词对吧？你得先有敏感词。

这里假设就随便写几个，咱们是推演，所以说不能写的太多，假设我的敏感词有ABC这是一个词，比如说bf这是一个词 Be好了，假设我的项目中一共有三个敏感词，当然我们推演就弄得简单一点，实际的工作当中你可以把它配成任意的词，中文的也可以，英文的也行，没有限制。

然后其次你有了敏感词假，我再假设一下，假设我们某一个用户他输入的内容是什么？我们根据这个内容去推演过滤的一个所以说我在写上就是说假设用户他输入的内容换一种形式，假设它输入的是这样的，一个字符串，X w你忽略里边的空格，因为我加空格是为了让它稀疏一点，这样我们看起来比较舒服一点，我们可以把空格忽略掉。

好，假设用户输入的是这样的一组，这个字符串是这样的，我摆到这儿，好，那么要想就是利用这个算法去看一看这个字符串里有没有这个敏感词，我们首先需要构造出就利用敏感词构造出一棵树，这棵树就是前缀树。你稍等，我把这个字给它放大一点，有点小。

好，首先我要构造出一棵树，然后这个树我先说一下前缀树它有几个特点，满足这个特点的这个树其实就叫前缀树，它的第一个特点就是这个前缀数的根节点，它不包含任何字符，根节点是空的，然后除了根节点以外的每个节点只包含一个字符，这是它的第一个特点，每一个节点只包含一个字符，但根节点是空的，然后从根节点到某一个节点经过的路径上，每个路径不有一个字符吗？

每个节点都有一个字符吗？

你把这个字符连接起来，那么就是这个节点，当前这个节点的对应的字符串说白了就是把一个单词拆解到这个数里头去了，然后每个节点的所有子节点包含的字符串不同，如果相同的情况你要合并，但我这么说这个不直观，我画一下可能你就明白了。

那么我就根据当前的这三个敏感词来构造出树形结构，首先这个树它得有一个根，我先画一个方块儿代表根叫root，root里是不存任何和敏感词有关的数据的，就是空的。

好了，然后我们看一下这个敏感词，我们树的下一级第一层其实要摆放的是单词的第一个字母，第一个字母是ABB b重复了，我们只画一个b所以root下面就有什么？

有ab两个字符就完了，非常简单，好，我就画一下这是a能不能小一点，有个a然后还有一个b是根据这个词分析出来的，然后他们都是root的孩子，所以我画条这个线就表示这么个意思，把它弄的小一点，像一个线一样，好就这样了。

然后你再看下一层应该这个单词的第二个字母，第二个字母是bfe对吧？Bfe然后你要看 b b属于 a的一个子节点，对吧？

所以 b应该在a之下，就a下面有个b然后 b下面有个f这个b下面有个e就是b下面有个f和e好，我再画两个节点把它挂到b之下，一个是f一个是e总之你按照这个规律去构造任何的给你一种敏感词，你都可以去画出这样的树形结构，当然我这还没有画完，因为你第一个单词还有第三层，有个字符是c但是后两个单词就没有了，所以 b之下还得有个c好，整个根据敏感词我们画构造的前缀数大概是这样子的一个结构，这个是c就是这样子的一个结构，然后还没完，刚才我们再回顾一下刚才我所说的前缀树的一个规定，根节点，没有任何字符的，除了根节点以外，每个节点只有一个字符，确实是这样，这个是e这是f我按顺序来，这是f这个是e好，然后第二点，从根节点开始，然后到某一个节点，比如说到c那么它会途经a和b对吧？

那么它途经的这些字符连起来，其实就是我们对应的这个单词ABC，就这样的。就是第二个特点。

第三个特点是每个节点的所有子节点包含的字符串不相同。你比如说 root项，它本来按理说应该是包含ABB，但是 b重合那就把它合二为一，保留一个就可以了，就这个意思。现在树构造完了，然后还有一个地方需要注意一下，需要处理一下。就是说比如说我从入的开始往下遍历的话，我遍历到这儿，我得到的单词是ab Ab是敏感词吗？其实不是，而ABC才是对吧？我往这边便利，我便利到BB这个字母是敏感词吗？不是，bf才是对吧？

他得到最末级的节点才能算作是一个敏感词，到变成到中部的时候不是，所以我们需要因为ABC我怎么去给它标记一下，哪个不是还要做个标记，不然的话可能你编辑到b你也不知道它是不是敏感词，怎么办？

还得做个标记，我再加一个符号加个叉，这是我画图打个叉，一会我们写代码的时候，我们可以用代码的方式，比如说加个布尔值，标记它是不是末级，或者说它是不是敏感词。

好，就总之我在他有可他构成敏感词的节点上打个叉，表示到这就是敏感词中间没有叉的就不是这样我们程序的在运行的过程中便于一些条件的检测好了，到这这个数就画完了，画完以后，那么我们现在就要利用这个数据结构去检测这个字符串里它到底有没有敏感词。稍等，我把这个空格再稍微大一点。

这样一会好画，你检测的话，我们检测的话其实需要需要的是三个指针，搞错了，需要三个指针，然后第一个指针指向的是这个数，然后它默认指向的是这个根节点，这里我标记一下第一个指针，当然我们在程序中其实就是一个变量，然后第二个指针指向的是这个字符串，然后默认它指向的是这个字符串中的第一个字符。

然后第三个指针依然是指向这个字符串，默认依然是指向第一个字符，字符串第一个字符有人想你这个字符串为什么要用两个指针来指向？我举个例子，你比如说我要检我们人为能知道说你其实你的这个字符串里有bf对吧？有bf这一部分bf就是个敏感词，你去筛选或者过滤这个单词的时候，需要两个指针才能把它标记出来，因为它标记开头，它标记结尾f对不对？

你这样的话我把从指针到指针之间的字符串认为是一个敏感词，我就把它替换成比如行行不就完了，如果你用一个指针的话，只能指向一个敏感词的开头或结尾对吧？你不能同时指向开头和结尾就不行。

因此我们需要两个指针，这两个指针他们是怎么去协调的，这是我们一会要演示的这么一个过程。所以我们整个算法需要三个指针，另外我们在筛选的过程中可能是有些字符它不是敏感词，我们就会把它保留下来对吧？

如果是敏感词我们把它替换成星号，把它打码，那么我们最终一定要有一个地方存一下这个结果，所以这样我底下再画一个方块，这里比如说我有一个变量专门用来存放这个结果，我们程序中可能会用一个使用标准，这里写上使用标准，然后 String builder它等于什么？你检测到这个检测到的结果就存放到死病标的里，它应该存放我们检测的结果。

好了，到这儿我就把要准备的事情都准备好了。

接下来我们就开始让算法的运转起来。那么整个运转的过程当中，其实主要是这两个指针从前往后走，那么其中第二个指针大致他是不回头的，他是一直从前往后走，它标记的是什么？就当前你检测到了第几个字符，第一个字符x是不是敏感词？不是，第二个是不是又以w开头的是不是敏感词，以a开头是不是敏感词，它是不断的往后走。

那么它在走的过程中，它标记的是敏感词的一个起始的位置，但比如它标记到 b 的时候，指针三要配合他判断你 B 我指向 b 从 b 开始到 b 结尾，就一个字符是不是敏感词？不是，我再往下走 bf 是不是？是从这到这是敏感词。

所以说其实它标记的是开头，它标记的是结尾，然后一旦标记成功以后，可能我们下一个就会就把打微信以后就会往下再往下执行，再往下走，他就往下走，他也往下走。

然后如果说你发现不是他还回来再重新进入下一个环节再去匹配，大概是这么一个过程，就总之第二个指针是永远是不回头从前往后走。

但第三个指针它是要去结尾的，如果发现这个是不是我再回来再从另外一个下一个字母开始再回来，它有一个不断怎么说不断抖动的这么一个过程，但是小范围的不是大范围的，所以这是这两个指针的一个特点，我们大概先了解一下到底是不是敏感词，我们主要是看这个数。

我们从这个书里找，就是将当前指针指向的字符和这个数据比较整个过程是二走一遍，三在微小范围内可能稍微有点往复，那么总体来说基本上就是一大遍搞定，所以说效率还是比较高的，下面我给大家完整的演示一下这个过程。

首先 2 和 3 都指向第一个字符对吧？然后我们的算法要去看树形结构，问他说你的第一级有没有 x 字母，树形结构他去检查，你看我第一级没有 x 没有 x 它可以不动，没有就表示什么？以 x 开头的，这个单词不可能是敏感词，对吧？x 是安全的，所以说你可以把 x 就记录下来。既然 x 是安全的，你就可以往下接着走了，对吧？看下一个字母，对吧？这两个指针就往下移就可以了。

同理，那么再看说这个字符是 w 那么你这个树形结构里有没有下一级有没有 w 没有。说明 w 也是安全的，于是 W 记不起来，没有问题。

好，然后再来，w 没问题再往下走，走到了 a 它同时走到 a 从 a 开始检测，这个时候再去询问这个数说你这里有没有 a 很显然有，这个时候这个数的指针就动了，它就指向 a 找到了有个 a 那有个 a 但 a 是不是打个叉有没有标记？没有，对吧？说明 a 它不是到这儿为止不是敏感词，所以我们还需要接着往下去看，二就定在这不动了，他定在疑似敏感词的开头不动了，然后三往下接着走。

看 b 询问树形结构你下面有没有 b A 项刚好有 b 对吧？但 b 没有差，没有这个判断的条件，所以 ab 到这这不是敏感词，但是他疑似因此怎么办？再往下再看再走一步，f 再去询问这个数，你下面有没有 f 没有，那就说明什么？说明从 a 到这儿它不是敏感词。

所以说或者说以从这儿以 a 开头往下，这部分不是敏感词，这个时候我们就归位了，因为这一轮以 a 开头的字符检测完了没有问题，那就归位它回到原位。

然后的话 a 没问题，我们往下走一步，看下一个字符，所以说他往后挪了一位，这个时候三指针也要指向 b 指向下一个字符，从 b 开始检测，那就重复刚才的过程。

那么先询问这个树有没有 b 有 b 指向 b 但 b 不是敏感词，疑似，我们再三再往下走看，B 的下级是 f b 下有 f 吗？有。

而且 eph 有标记，说明 bf 是敏感词，我们找到了一个敏感词，刚才我忘了一步就是 a 如果不是敏感词，我们得把它记录下来对吧？忘记录了。补充一下，然后这回我找到了从 b 到 f 这是一个敏感词怎么办？要替换成星，于是我会在这里记录成星星，比如我们打三个星把它替换了，替换完以后，首先我把这个数回到原位，然后我们接下来要检测的就是说跳过这个敏感词对吧？

因为它已经是敏感词了，跳过它跳过它以后那去哪呢？去指针三的下一个位置，同理二也去指针三的下一个位置。好，你感受到了吗？二是一直往前走，但三可能小范围会回调一下。是这样，好。

同样的道理，我们再去检测这回又是以 a 为开头的单词，有没有可能是敏感词，也会询问这个数下级有 a 疑似，但 a 不是，那三的时候我再往下走一步，再走一步是 b 下面我们的 b 有疑似，但不是这样往下走，看是不是下面是 c 那树说我下面有个 c 这个 c 而且是有标记的，就证明什么？



从这儿到这儿这个单词是敏感词要替换对吧？所以下面还得加星星。好了，既然这个单词是敏感词，我们就得跳过它，不记住它而记住了星对吧？所以指针再往下移一位二指针就直接移到这个位置就ok了。同理这个数还得回味。同理我们再去判断以字母开头的是不是敏感词，一看这里没有f不是对吧？不是就往下移，移到这一看还是f这里没有f不是那就移到头了就完了，当然你这两个f不是敏感词，你得把它记录下来。

最终的算法经过这么一轮这么一轮的执行以后，那么我们记录的结果是这样一个结果，就是exo wa然后其中的bf被替换了，ABC被替换了，其他的保留了，就变成这样一个词，这里边你这是我演示的是字母，你画的换成汉字，甚至其他国家的文字其实都是一样的道理，都是可以的。

然后我们整个算法的检测的目标是以字符为单位的，不是以字节为单位的，所以说你是一个字母也好，还是一个汉字也罢，没有关系都一样。

好，过滤敏感词大概的工作的流程我就给大家介绍完了，下面我给大家通过代码把这个过程给你实现出来。然后的话我们看一下用代码来实现是怎么样一种感觉，好。

那么我就打开我的Idea，然后我们要开发的是一个能够复用的工具，因为我们不止在一个很多功能都要用，你发帖子也好，发评论也好，其实都要用到，那么我就要写一个工具，但我在写这个工具之前，我先把敏感词给它定义好，这个敏感词我们可以把它定义在数据库里，也可以把它定义到一个文件里，我们这个项目我就把它定义到文件里，这样我在resource之下新建一个用来存放敏感词的文件，叫三ctf敏感words，敏感词点txt，然后我在这里随便写几个词，我就不写的那么多了，写几个是那么个意思就可以了，比如说赌博是敏感词，嫖娼是敏感词，吸毒是假设就开票也是，完了我就写4个，反正他每1个词独占一行，你可以无限的写下去，那么这就是我们敏感词的来源，我们就要判断我们的一段文字里有没有这些词当中任意一个，如果有就要替换成星，这是我们这个算法的要求。

好了，接下来我就写一个工具，然后把刚才的算法给它实现了，那么这个工具我把它写到YouTube包下，新建一个类，这个类我叫三CT built就敏感词过滤器，三c点q的build，然后这个工具为了复印方便，我给它托管的容器由容器来管理，然后或者加一个component。因为其实在各个层次你都可以用，它不只是说是某一层专用的，是一个通用的这么工具。

然后刚才我PPT上也说了，我们要开发这个算法怎么办？大概是三步，第一步你得先把这个数据结构定义好，对吧？得把这个数据结构定义好。第二步你要把它初始化，这个数据初始化出来。其实就是刚才画的图，我们根据我们既有的敏感词的文件，把这个数初始化好在内存里，然后第三步就是做这个动作做这个过程。好咱们一步一步来实现。

首先我要把前缀数的结构定义好，这里我们可以写一个独立的类，或者我们也可以写一个内部类，因为这个结构只是在这个工具里用，其他的地方基本上不会用到比较特殊，所以我定一个内部类就不允许外界访问了private class，然后我们定义前缀是我们定义树形结构，其实主要是定义这个节点与节点的关系，所以说这个类型是trie是前缀数的意思，Tree no就前缀数的节点，所以我当前定义的数据的结构其实是描述前缀数的某一个节点的，然后这个节点我们需要描述什么？需要描述它是不是敏感词，它有没有标记差对吧？这是一个要描述的方面。这里我写不出是吧？我们要描述的是关键词结束的标识。我给他一个布尔值，因为你要么是就要就否，然后布尔值我叫is key work。是不是一个等于默认等于放肆，你是不是一个敏感词，或者说你是不是一个单词的结尾，如果是单词的结尾就是敏感词，就这个意思。

好了，那么当然了外界要访问他，你得给他提供一个盖特赛的方法，我给他生成盖特赛的，除了这个以外，你还要描述什么？树形结构，你要描述某一个节点的孩子对吧？

描述某一个节点的孩子，因为任何节点他都有可能孩子，但也有可能没有，我们要描述某一个阶段的孩子，这个好描述，就再加一个成员，这里我写个注释，就是我要描述的是子节点，当前节点的子节点，那么一个节点他的孩子可能是多个，所以我用一个map对它进行一个封装，map的key，我用这个character就是字符，你这个子节点对应的是哪个字符，然后的话value区no，就当前前缀数节点的类型，这样的话我就能够通过当前节点找到它，直接点对应的是哪个字符知道，直接点的数据结构。

也知道这个意思。

好属性我给它取名叫 sad not子节点默认的给它初始化，成为一个空的哈希。好了，这里我给它写个注释写完整一点，这个key就是我们 Map的key，它是下级的节点的字符，是下级字符，然后 y6是下级节点，这样便于我们去使用，一会我们用的时候你可能就有体会了。好了。

那么当然了你这个子节点默认是空的，你什么时候往里装数据呢？你什么时候获取它，你得给它提供往里装数据和获取它的办法对吧？所以我这再加两个方法，这就不生成了，因为它不是简单的site，我需要提供一个添加子节点的方法，当然这个方法得是公有的对吧？Public然后爱的。

3note你添加一个子节点，这个t cat是一个字符，然后 Value是 tree，node就叫node，你要加到哪里去，加到 sup no这里去对吧？所以说也很简单，就是 some knows点put，然后 k是c value是no的，这就可以了，很方便，然后再来一个，那么你加完以后将来肯定是要获取要用的对吧？

所以再提供一个获取，使节点的办法也是有的。获取指点，通过key取外流，那么取到的是y的是去no的，那方法名我就要get300，no的，然后t character我就直接的return map点get key好，这就可以了，这就行了。就是说这个前缀数其实它的结构并不复杂，我们主要是做好标志，并且维护好上下级的关系就可以了。这个写个注释说这个是什么？前追溯我们所有的算法，基于这么一个简单的树形的结构。

好了，那么这一步完成以后，我们下一个环节就是我们要根据敏感词就是文件，然后来初始化前缀数中的数据，下面我把初始化的代码接着写一下。

好前置数我把它放到后面去了，我在前面写这个结果放到后面去，然后首先我在组件当中先实例化一个logo，因为我们在做处理过程中，如果说有的地方需要打日志，我好提前准备好 Private时代里边山东lover sensitive filter their class。

好，另外我在定义一个常量，这个常量一会要用到，就是说当我们检测到这个字符串中有一个敏感词以后，我们需要把敏感词替换成什么样的符号？我把它做成一个常量，将来如果说好改的话，要改的话，我好在这个地方统一改，这个叫产量一般都大写，replace问题替换成什么？简单点三个星，完了就是注释，这是我们要替换符号。

接下来因为我们要初始化这个数对吧？肯定得先从根开始初始化的初始化起对吧？所以我首先先把根节点初始化好，根据点比较简单，它是空的什么都没有，只要你有这个对象就可以，所以我先初始化一个根节点，吹no的名字叫root，no根节点的意思等于no，是no的。

好，就这样。

那么接下来我就要根据刚才的配置文件，就这个文件，根据它里面的数据去构造这个树形，那显然不是说一个属性能搞定的，我们得需要写一套逻辑，得有一个方法，这个方法显然应该是在程序启动的时候，或者说在你初次调这个工具的时候，它自动就把它初始化好比较好，它也不需要初始化多次，只要一次就可以了，所以说你只要首次访问它时初始化一次就可以了。

因此这样我在这个方法里写加一个他盈利我给类加一个初始化的方法，然后在盈利的方法之前加上Post，construct不知道大家还记不记得注解，那么这个注解表示说这是一个初始化方法，那么当容器实例化这个病以后，在调用它的构造器之后，这个方法就会被自动的调用，那么这个并什么时候被初始化，在服务启动的时候就初始化了，所以在服务启动的时候，这个方法就被调用，这个树形结构就构造好了。

启动完以后用户去访问的时候，我们就可以拿它用来去对敏感词做筛选了，所以这个时机是比较合理的在这里写。

好，那么我们想初始化这个数，首先你得把这个文件里的字符给它读出来，对吧？你要读读的方式有很多，我可以用这样的方式叫diss点get class，任意对象点get class，loader，我要获取类加载器，那么类加载器它是从类路径下去加载资源，所谓的类路径指的就是这个克拉克类克拉斯目录之下，我们的程序一编译，所有的代码都会编译到classic之下，包括配置文件，所以我就从这里边去读这个配置文件。

好，get class loader，然后点get，它有一个方法get results as使用。那么只要你写上这个文件在 class 下存放的路径就可以了，当然你看我编译一下 CTRL f9，编译完之后你看他说get classic下他还没有，他有的时候就是这样的，你简单的CTRL f9他没有因为他没有怎么说，它是一个txt，所以说有的时候它这个快件它不会默认编译这个东西，那么这种情况我可以把编译代码口令一下，然后我再重新编译一下，然后点微文插件重新编译一下，然后再看这个to，get这回就有了。

好，所以这个小问题你要注意一下，我们直接CTRL f9它这种非class文件有的时候你新建的它不会变异，如果已经有了没有问题好了，那么 get resource as stream就是 class下读那个文件，我们刚才也看到了，文件编以后就在classes之下，所以它没有附目录，你直接写文件名在这就可以了。

所以我写 sensitive words点儿txt，我们这句话所加载到的是一个字节流 input，string。好，这是一个资金流，当然了资金流将来肯定是要关闭，所以什么时候关闭应该在find类当中关闭对吧？所以我最好给它加上try，catch我可以这样写 try。

那么我在try里边去开启或者是创建某一个对象，那么它会自动的编译时自动的给我加上finally，然后把它关闭掉，这样我就省点事，当然你出来完以后你还得 catch一下，catch我就catch，因为这是io操作，io操作好，那么如果开头异常，我应该记个日志 log点儿说加载敏感词文件失败，然后带上异常的信息就完了。

好，我们得到的是一个字节流，我们从字节当中去读文字不太方便，最好把它转成字符流也好转，你把一个字节流转成字符流，你可以拗一个input，席俊瑞德这是一个转化流对吧？

你把 input stream传进去，它会给你转成一个reader，然后我们直接使用 reader也不是很方便，最好我们用的是一个缓冲流，它读取数据来的话会效率比较高，所以我们再把这个字符流转化为缓冲流，再转一层你就可以8分的reader叫reader等于new8幅的reader，然后把刚才的字符流传给他，最终我们得到的是一个缓冲流，这样效率高一点。

当然我也把它写在try里，最终的话自动在find当中关掉。

在try的内部我就需要通过 reader去读取每一个敏感词。

读的也比较简单，首先首先定一个变量叫key word，每次读取到的词我都存到这个变量里，然后我就用在while循环中去读 while key word它等于 reader点read，因为刚才我们看了配置文件里是一行一个敏感词，你读一行读一行复制给key word，如果说它不等于空，就表示我读到了，读到以后怎么办？

我应该把敏感词添加到前缀数对象中去，那这个逻辑就比较复杂一点，我需要我最好能把它封装为一个独立的方法去调用这里，我先这样，我先写第四点。At key word，我先把这个方法规定好，叫这个名字，然后我需要把 key word传给他，他把 key word帮我挂到树中去就可以了。

好，下面我就来写这个方法，把一个敏感词添加到前缀数对象当中去，写注释，将一个敏感词添加到前缀数当中去，那么这个方法是内部使用的，也不需要工友，private就可以，方法名刚才我规定的叫爱的key word，他的参数就是梯位，那么实际上实现的逻辑就是比如说我这个ABC，你要把它搞成这个样子，把它做成竖的上下级的这样一个关系就可以了。

当然我们目前有的节点只是一个 root节点对吧？那么我要根据这个单词去创建它的一个逐级创建，它的下级节点，那就创建。首先我先创建一个吹no的取名叫time，no，它是一个临时节点，然后相当于一个指针，默认的就指向根。

然后的话我们根据这个单词拆解每一个字符不断的指向下一个字符，那就不断的去构造树的下一级，就是这个意思。好，我默认它指向根就是root，node默认它等于根，然后那么我们要往根下挂节点，得看这个单词里有什么对吧？你得遍历，你得遍历这个单词中的字符也好，遍历i等于0，i小于key word，但是佳佳。

好，那么每次遍历我们能得到很容易得到一个字符 k word点又at，对吧？很容易得到一个字符，那么得到一个字符以后，我们就需要把这个字符挂到当前节点的下面，因为你默认是根变了的，第一个字符就挂到它下面，你再变一个字符就挂到a的下面，对吧？以此类推。

好，那么你把一个字符挂到一个节点的下面怎么办？其实我们就调刚才我们写的方法，就当前节点是temp node点get some node。我们试图去找一下它有没有子节点，因为可能是之前挂过同样的字符了，我们找一下有没有，如果有就不用新建，如果没有我们就需要新建一个节点，就这个意思，所以我试图去获取一个子节点，那么判断一下 Sup node它是不是空的，那么如果三不弄的是空的怎么办？你要初始化在这里，你要初始化子节点，如果不是空的你就直接用就可以了。

那么这里初始化子节点就是sub node等于new train node。另外你得把这个子节点挂到他的父亲之下，挂到当前节点之下，当前节点是谁？不就是temp node对吧？它相当于是个指针，Temp node，点的三个note，然后当前的字符是c对吧？然后当前的子节点是三个note把它挂好。

好，到这儿的话，我们要么找到了一个已存在的子节点，要么就新建了一个子节点，反正到这儿这个子节点已经有了，而且这个子节点对应的就是c就这个字符当前这个字符，那么这个阶段处理好，我们再进入下一个循环，再进行下一层节点的处理对吧？你怎么处理你可以这样，我们让指针指向子节点，调整指针就可以了。

指向子节点，然后进入下一轮循环就可以了。我就可以写time，node就当前节点它等于子节点，下次循环的话就不进入下一层了，很容易。

好，最后你还得当某一个单词它循环完以后，它循环到最后一个字符以后，最后一个字符你要打个标记，所以这里你要设置结束的标识得判断一下，不是每次都设置，如果你的l走到了最后一个位置，Key word点last减一，你找到最后一个位置，这个时候你就要给它打一个标志，标志就是time。

Node，点，sat，keyword。And结束了。结束。表示这是一个敏感词。好，这就可以了，经过这套逻辑我们就把提味挂到了树里。好了，二这是第二步。我们根据敏感词去初始化这个前缀数也就这个图。好了，那么最后一步就是实现了检索的过程，主要是利用这么三个指针的这么一个对应关系去检索字符串里有没有敏感词，当然这一步是最关键的，我们接下来再把最后一步给它实现了。

方法就应该是公有的了，因为这个方法是被外界调用去解决它的问题的，对吧？它应该是个公有的方法。Public这个方法我返回死菌，返回的是谁呢？就这个结果你记录的结果你传给他的是一个可能含有敏感词的字符串，它返回的是替换掉以后的字符串，所以返回的是死菌，然后方法名我叫filter过滤。

这个方法的参数是一个文本txt，这样我把注释写得详尽一点，因为毕竟它是为被外界所调用的过滤敏感词参数是待过滤，不对待过滤的文本返回来的是过滤后的文本，要写得清楚一点。

好，然后首先我们得怀疑一下这个参数你有没有值，你万一传个空值进来，我就没有必要去判断了，对吧？所以我就看空一下子俊you tells点儿意思 block参数，那么如果参数是空的，我就直接return now不管了。

好，那么如果参数不为空表示确实是一个字符串，这时候我就要去过滤了，我过滤的时候依赖三个指针对吧？我需要声明三个变量，指针一指向的是树指向树指向树的节点，这个比较特殊，我需要声明一个去弄，还叫time node，因为这个节点它会不断的向下移，它默认应该是指向根的。一开始它指向这个根的，然后我们随着检测它会指向下级，下级对吧？是这么一个过程。

所以默认指向根指针一，接下来指针二，指针二和指针三它指向的是字符串的首位第零个位置对吧？指针二我取名叫应该是个整数，它是一个索引，Begin when是0。好，再来一个指针三也是一个整数，索引我叫production，等于也是0。还有我们还要有一个变量记录最终的结果，对吧？这个变量因为它要不断的追加这个字符，所以你最好用变成字符串，你不要用死菌效率低，我们通常把这个叫这结果就注释，通常我们都用使用build把它初始化好，使用build可以了，下面我就要开始去运行这个算法了。

这个算法整个的过程，你去检测这个字符串从头到尾的检测，我们可以以指针二为目标，从0到尾，其实也可以以指针三为目标，从0到尾，而且指针三它会优先到位，比如说你检测到比如说指针二检测到这个位置的时候，发现比如说检查这个位置，假设这个是a他疑似敏感词可能指令三就再往下移一步，可能会出现这种情况，所以指令三会先结束。

所以我们利用指针三去做循环的话，可能是它会效率会更高一点，否则用质量二的话可能会多循环几次。

好，我这样写我们要做一个遍历，就是当production指针三小于test点nice的时候，那么指针三没到结尾的时候，就意味着只要它在中途，我们就需要继续遍历，直到它到结尾，我们就停止遍历，就这个意思对吧？

因为它比指针二有可能优先到达结尾，所以利用它去作为一个判断的终止条件，效率更高一点，好。

所以就这样写了，写完以后每次遍历我们肯定是能得到这里面的——一个字符还是又c等于text点又at，我们通过这个project就能得到当前的某一个字符，它不就指向某一个字符就得到这个字符对吧？

得到这个字符以后就要对这个字符做判断，当然我们可以先不着急判断它是不是敏感词，或者说是不是疑似敏感词，我们先做另外一件事，刚才我画图是没说，这里要补充一下，什么事我们要跳过符号，什么叫符号？

就是说现在上网的人也多了，他也变得越来越聪明了，他也知道这个网站会对敏感词做过滤，所以他为了规避过滤的机制，他可能不去直接发什么开票这样的词，比如你直接发开票很容易就被识别，这是敏感词，他可能会这样在这个字当中穿插一些特殊的符号，我觉得打出特殊的符号，表情符号再加个五角星就加一个得了，你看我在开票中间穿插着加上五角星，那么如果你写的算法不严谨的话，你可能会连续因为连续的开票俩字是敏感词，但中间隔了一个符号，你可能不认为它是敏感词，可能会绕开一些不严谨的算法。

所以这种情况我们也要考虑怎么考虑。这个还得再多说两句，我再简单的再画个图，那么针对刚才的情况，像这种情况你像开票，如果说我们要去构造就开票这两字，我们要去构造授信结构的话会是什么样？

我简单化，大概是这样，开票是我们的敏感词，我把它竖起来，就表示一个树形结构，默认的话指针是指向根结点指向这两字之上，对不对？

好，然后我们实际用户输入的这个词是这样的，中间带符号的，它是这样，比如说我们用这样的敏感词，这样的树形结构去判断这里边是不是包含敏感词，要规避特殊符号的情况，那得怎么规避？

但是首先你注意，比如说我二指向正，三也指向正，就有一种情况我们指向了这个特殊符号，并且同时这个树形结构它刚好是指向的是根，这个时候怎么处理，其实我们就把它略过就行了。对吧？因为我们当前树形结构指向的是根，就是说它不在疑似的怀疑的范围之内，我们把它略过就可以了，保留它就可以了。不用去把它删掉怎么着，就留着它进入下一个环节，到这个字就可以了。

好，然后你到开的时候，你一检测说这个顺序会宽的，我这里有个开疑似敏感词，因为它不是结尾，这个时候我们只能再往下移，它移到了星这个位置对吧？移到星这个位置这时候怎么办？这时候因为开是疑似敏感词，你星过掉再看下一个，忽略它过掉再看下一个，但二保持不动，因为它可能出现在敏感词之中，我们是再往下走，三往下走它不动，这时候你看这个票有没有票？

有发现敏感词，所以说特殊符号在一开头的情况这边处于根的位置是一种处理方式，就它俩都往前走，否则它处于它在中间的情况，然后它不载入的这个位置，那么只有三往下走，大概是这个意思。

好了，我们再回到这个程序里，我们所谓的跳过符号就是跳过五角星这样的符号，避免它干扰我们正确的判断。

当然你要跳过符号的话，我们最好把跳过符号的逻辑稍微给它封装一下，我得再单独写一个小方法，我在底下加一个判断是否为符号，私有的就可以，就这个类自己用，然后账号名叫is symbol。条件或者说参数就是一个字符，就判断这个字符是不是符号。好，我就直接这样写了，就很简单。

我利用又YouTube工具点is ask，然后阿尔法number c这个方法是判断这个字符是不是合法的字符，是不是普通的字符，如果你ABC123普通字符它返回数，如果你是特殊字符符号它就返回false，所以我加一个取个反，如果它是特殊符号，返回符号取个反就变成数就是特殊符号这样另外的话还要再写再写一句话，我写完再解释就C还得小于0x2180或者c还要大于0x9fsf好，这两个数是什么？其实0x2180到就是02180到9fff这个范围其实是东亚的文字范围，东亚的文字东亚文字范围，我们是不认为它是符号的，你

你这个c在东亚的文字范围之外，我们才认为它是特殊符号，因为东亚的文字其实也是正常的文字，只不过它和西方的文字就是区别比较大而已。

这里我写个注释从这儿到这儿是东亚文字范围，包括中文、日文、韩文，可能在西方人眼里文字也挺特殊的，其实但不是特殊符号是正常的，所以你得在文字范围之外才有可能是特殊符号，大概是这个意思。

好，我们再回到刚才的逻辑当中去，我说了这里我第一步要跳过那个符号，我判断一下首先你是不是特殊符号，如果是的话，我还得看一下我这个时候我那个数，我指针是指向了是根还是不是根，因为它否则的处理方式不一样，对吧？那么弱指针一处于根节点，这个时候刚才的这种情况，这种情况糟了，重画一个把擦了就这种情况。

它此时它是处于根的，这种情况指针三也是要指向这儿，这时候我们不就说了，把它略过就可以了，对吧？若指针一处于根节点，那么我们就将此符号计入结果，这个符号我留着我不把它过滤掉，因为它不是特殊符号，然后让指针二向下走一步，因为它过去了，指针二向下走一步，但指针它也会走。

好，我就这里我得判断一下，因为是指针一组为根节点，那就得它得等于root，no，这个时候就是它处于根节点，这个时候我要把它计入结果使用，点判断c另外的话只能二往下走，就是begin佳佳就可以了，而指针三我在外面加，为什么在外面加？

因为刚才我们也推演了，如果是另外一种情况，就是指针二在这儿，指针三在这儿，发现了特殊符号，这个时候指针二不动，指针三往下走，你会发现指针二有的时候走有时候不走，但指针三一定会走，对吧？指针二什么时候不走？

什么时候走这个时候其他时候他不走，然后只能这样无论如何都要走，所以他在外面写这个逻辑有点绕，你自己课后自己画一画，自己推一下，把这个情况都考虑到就好了。好，那么这里写上注释，无论符号在开头或中间，那么指针三都向下走一步。

好了，这个逻辑就处理完了，处理完以后，因为我们这一步是跳过符号，跳过符号以后后续的业务就不用执行了，到此为止，所以我这里写一个continue进入下一轮循环，这个符号就处理完了。

好，那么符号处理完以后，我们这回那就如果走到这能就说明了这个字符不是符号，你就要对它进行检测了。

好，检测就是检测检查下级节点，因为我们默认处于根节点，我们第一个检测就是根的下级，然后在下级的下级对吧？所以检查下级节点，你要检查下级节点，你得取下级节点对吧？Time no的点，get356的c取到下级节点，然后再把它付给time node，当前节点就变成了下级节点，我们就检测它了。检测它有几种情况，一种是他们弄的等于no，什么意思，就下级没有节点，我下级没有节点。

刚才的在此之前的他们弄的，我们没有看到他是敏感词，然后他又没有下级，那就证明当前这种情况不存在敏感词，对吧？当年是什么情况？就是以begin开头的字符串不是敏感词，以begin开头的字符串不是敏感词，这个时候我就可以把Begin所指向的字符看一下图，就可以把begin指向的字符记录下来。

好，我就记录了使用build判的，然后text点叉，艾特北根，把北根的位置记录下来对吧？怎么记录下来，然后就进入下一个位置，进入下一个位置你就得给你得加，对吧？Begin处理完了，他下一个位置是不是还不知道？加一下，然后这个时候position得和begin保持一致对吧？Position等于begin。我一句话就把它搞定了，当然这两句话也可以对吧？好，然后再把树形结构的指针归位，指针重新指向根节点，我们再进入下一轮的判断，我们就谈奋斗的，就等于如果听到了好，这是一种情况，然后再来还有一种情况也有可能是这样的，就是temper no的。

第二意思p word and这个是最好教这个方法Is key word and比方说我当前取到的下级的点，它已经有标志了，就说明我找到了一个敏感词。我发现了敏感词，这个敏感词是比更开头或者是结尾对吧？所以我需要将begin到production这一段的字符串，这一段的字符串替换掉。

好替换掉我往死忠哥这里存的是常量，对吧？Replacement就三个星就完了。然后与此同时我需要让这个值进入下一个位置，这回不是北给你进入下一个位置，因为从begin到production是敏感词，应该是让production进入下一个位置对吧？所以你应该加production，那begin得和它重合，此时所以说begin等于加proceed。

好，然后此时在重新指向根节点，跟这句话是一样的。

好，这是一种情况，还有一种情况，否则否则就是什么？我指针比如说指针二指向它，指针三指向它，然后指向的是b我在检测的途中没有发现没有检测完，也没有发现这个符号，我需要根据继续往下执行继续检查，那就只能再往下移动一下就可以了，别的都不用动。

所以那就是检查下一个字符，继续检查下一个字符就行了。

好，那么经过这个循环以后，我们基本上能走一轮，但不是严格走一轮，这块要注意，不是严格走一轮，因为有这么情况，指针二指向他，然后发现疑似，然后指针三往下走一步，指针三就到头了，发现不是，这个时候指针二和指针三，这两个字母需要把它记不起来，它不是敏感词，有这种情况出现，所以这个时候你循环结束了，我们在循环内部还没有把它记录下来，对吧？

我们没有考虑这种情况怎么办？你在循环之外补一句话，我们就将最后一批字符计入结果，三提前到终点，二没到终点的时候，这个时候你像那几个字母又不是敏感词的时候，把它单独记一下，那就时间标点点儿判的txt。第二，杀不死菌，这不是一个我从begin截取到最后记不起来。好，最终把使用标准透视图卷进入结果返回就可以了。

好了，那么反正就根据刚才我们所推演的流程，我们用代码把它给实现了，实现完以后这个程序能不能正常运行，我们得进行一个仔细测试。

所以下面我就在测试的包下新建一个测试类，新的测试类我叫 sensitive tests，那么它里面的注解我copy随便找一个类copy，好，然后我想测的是 Filter对吧？

所以我把 filter把它注入进来，三c还有filter注入进来，这样以后我写一个测试方法测一下，方法名我就叫test sensitivity。

好，首先我得写一个带有敏感词的字符串，咱们看一下，看看能不能把它过滤好，先写个字符串，这个里我就瞎编了，我就瞎写了，说这里面可以赌博，可以嫖娼，可以吸毒，可以开票，说白了我就把那几个敏感词都写到这句话里了，然后完了接下来我就用这个工具去过滤它三c to filter点，filter把text存入它会返回过滤后的文本，我们把这个文本打印看一下，看它对不对？这个逻辑很简单了，那就执行。

看一下执行结果。对了，你看4个敏感词都被换成了星，但这是1种情况，这种情况的话帖子的人还比较初级的，它里面还不知道放特殊符号。

我再来一个，我把这句话里边放上特殊符号，还是星我还得点这块，但是你不放五角星，你放什么都行。好，我用这个芯把它分割开来，看你这回这个算法能行不？当然这一类好了，我们再对它过滤一下，然后再打印一下，看一下能不能执行，看结果这也可以。

你会发现敏感词的开头和结尾的符号被保留了，但它中间的符号被pass掉了，那么我们认为这赌博就是一个敏感词，就把它变成一个行星，这是ok的合理的好了，那么这样的话我们就把过滤敏感词的组件就开发完了，那么下次课我们去开发发布帖子的功能的时候，这个就可以用得上了。

好，那么我想这个代码其实并没有那么难，并不没有那么复杂，其实主要的是你要能够明白算法它背后的一些逻辑，这个逻辑有了以后，那么这个代码就比较容易能够写得出来，因为有这么逻辑去指导，所以大家课后我觉得你建议你不只是写代码，你也自己画一画，自己把它推演明白，然后的话再去写就会轻松很多了。好，这次课我们就演示到这里，我们下次课再见。

## 3.2 发布帖子

这次课我们来实现发布帖子的功能，那么在实现这个功能的时候，我们需要用到异步请求，那么所谓的异步请求就是当前我的网页不刷新，然后我还要访问服务器，那么服务器会给我返回一些结果，这个结果不是网页，那么我通过这个结果中提炼出来的数据，我去对网页做一个局部的刷新，通常是给一个提示，更改一些样式之类的，好，这样的方式就叫异步请求。

那么实现异步请求的技术叫access，就是异步通信技术。所以首先我们需要对 Access这项技术做一个了解。

Jack它是一组单词这样一组单词的缩写，那么翻译过来就是异步的JavaScript与xml所以你根据命名来看，它里边并没有包含什么新的技术，它不是一门新的技术，它只是一个新的术语。换句话说，我们只是利用老的技术，然后产生了一种新的玩法，这个叫access。

然后像我刚才所说的就使用access，那么网页能够增量将增量更新呈现在页面上，也就是说当前页面不刷新，然后访问服务器给我返回一些，结果我再把这个结果增量的更新到页面上，大概就是这个意思。

然后 Access中的 x代表的是xml但是现在其实我们要求服务器返回xml的时候几乎没有了，因为我们得到了xml以后不好解析，它结构比较复杂，目前的话我们都是要求服务器返回杰森格式的字符串，它比 xm简单多了，方便多了，然后任何语言都可以解析这种格式的字符串，所以说很方便。

好了，总体来说 ajax他的手册你可以从某z了官方网站查到这个地址，我列在这儿了，那么科科网如果你有时间的话，可以自己去看一看，课上的话我们就不去看了，那么我这些关于Jack的一些总结提炼都是从手册中提炼出来的。另外一会我们演示的案例也是根据这个手册提炼而来的。

好，那么我们一会演示的内容主要两部分，一个就是我们先写一个 Jack请求的小例子，大家体会一下依附请求是怎么去处理的。当然我们发异步请求可以用原生的js也可以用 Gs框架就carry，那肯定是框架比较容易，所以说我们就用carry。

另外我们的项目前端页面是由做的，它就基于这块，所以我们每一个网页上就已经引入了几块，所以说我们用几块也会非常方便在项目里。这个事例你了解以后，我们就利用这种方式去开发发布帖子的功能，这是我们这次课的一个安排。接下来我就给大家演示一下使用解块发送异步请求的一个示例。

好，我打开这个idea，那么要想开发示例，首先我想这样，我在community utu之下，我想加几个方法，是处理杰森字符串的方法，因为我们服务器要给浏览器反馈杰森字符串对吧？

现在我们没有这样的工具不方便，那么实际上我们当前的项目中已经依赖了一些杰森相关的包，就是不一样，他把我们引入进来了，叫Jackson，但是我们平时开发的时候可能更习惯于用另外的一个叫fast节选，就是它的效率更高一点，性能更好一点，所以说这样我就再引一个包 Fast Jason，然后我们用包下的API去处理基层相关的就转换好，我就打开浏览器，我已经提前打开了，这个网站我搜一下包叫fast Jason。

那么你看第一个就是它的使用数是遥遥领先的，然后这个是阿里巴巴开发出来的一个包很好用的，然后我点一下 fast的节选好，然后选一个最新的版本。好，把这个代码拷过来，把它粘贴到 home点xml零。稍等好了，这个包就依赖好了以后，接下来我就利用这个 API去开发几个工具方法，一会好用，这个方法是什么？其实就是我们平时给服务器向浏览器返回的阶层数据往往需要包含几部分内容，我们就把这几部分内容整合到一起。

第一个就是我们通常会给浏览器返回一个编号，比如零代表什么意思？一代表什么意思？编码代表什么意思？返回一个编码。

第二个很有可能会给浏览器返回一个提示信息是成功还是失败，还是怎么样。

第三个也可能会返回一些业务数据，这个业务数据不同的功能就不一样，所以根据这样的一种诉求，我这样是提炼这个方法，首先它得是公有的，然后静态的方便调用，这个方法最终返回的是一个Jason格式的字符串，所以是个string，然后方法名我叫get Jason时间获取Jason字符串，然后你需要给我传入的是扣的编码，然后再来一个message，提示信息，再来一个map，Map当中就是封装，你的业务数据不同的功能就不一样，这样的话就比较灵活一点。



好，那么调用者把这三个条件给我，我把它封装成一个接生对象，然后把这个阶层对象转换成字符串，我们就得到了一个阶层格式的字符串。

好，首先我需要把这个参数封装到阶层对象里，阶层对象我们利用 fast计算API是这样的，它叫结算不对，好，就是 com的阿里巴巴 fast节省就它了，我上面的变量名就叫节省等于new，节省不对，然后我把传入的参数装到对象里去，节省点put扣的，值是扣的，然后节省点put message。值也是message。

Ok先把这两个值装进去，那么map我需要把它打散了，把每一个建制队单独装到对象里头去，所以我要先对map做一个判断，它y为空了对不对？先判断一下。

如果说它不为空，我需要便利 Map，它便利map有三种方式，一个是你便利它的key，一个是你便利它的value，一个是你便利它的 key value，我这里便利他的key，遍地k你就要获取map点key set key的集合，每次标的得到一个key，我就节省点put，我就把这个key存入 Jason，那么它的值是它对应的value就是MAC点get t ok好，那么经过了这一段代码，继承对象它的里面就封装了，你传给我的这些数据，最终我把它转成字符串，那就是杰森格式的字符串，瑞森杰森点儿too Jason死菌这就行了，我们在调用的时候，有些时候可能就没有什么提示，只有一个编号，有些时候可能就没有业务数据，只有编号，或者是只有提示，就这个编号一定是有，然后提示或者是这个业务数据可能没有，因此鉴于这种情况，我再加两个方法，就对这个方法重载一下，便于调用。

好，再来一个 get范围跟它一样，存在参数不同，比如说有一种情况是只有这两个参数，就是code和message，这个就很好办了，我们就可以调它是吧？To，return get就是传入的是扣的message。然后 map没有，全是文档好，再来一个，有些时候可能连 message也没有，只有一个扣，就这种这样的message也是no，这样写就方便了，那么这个写完以后可能是大家对杰森 API不熟悉，所以说我们写一个命运方法给他测一下，测完以后你就对他印象就直观了，我就因为这个工具类比较简单，它也不需要由容器来管理，所以我就在这里直接写一个命运方法进行测试了，这样。

我先创建一个麦，就模拟一点业务数据。

好new map，然后 map点put，比如说我装一个name叫张三 map点put，装一个ed假设，他是25岁好，然后我就调用 get杰森斯顿这个方法，比如传入的客户的是0，我传入的提示消息是ok，没有问题，我传入的业务数据就是一个map，然后我把得到的结果打印出来非常简单，我执行一下命看一看结果对不对？

好，你看我们最终得到了一个json格式的字符串，其中 message值为ok，扣的值为0 name值为张三，h值为25，所以说我们很容易就能得到这样一个格式的数据把它传给浏览器，浏览器把它转化为GS对象，然后可以得到每一个key所对应的值也很方便，这样就很容易实现一个前后端的交互了。

好了，有了这个工具以后，下面我就开始给大家演示 a jacks，它的示例我们怎么发送一个异步请求，我需要在CTRL那里先加一个方法来处理这样的请求，是一个小事例，所以说我在阿尔法CTRL里写代码，我在最后的位置来写个注释，这是 jacks实例。

首先我声明这个方法的访问路径，pass等于就叫 jacks简单一点事例，然后请求的方式我设为post，就是通常情况下很多时候是页面是要给服务器通过异步的方式提交一些数据，然后我们保存完以后给浏览器返回一个简单的提示，它一提示就完了，那就是我们一会要开发的，发布帖子的功能其实就是这样的，所以说我这里就用的是 post请求，然后因为是异步请求，我们不像浏览器返回网页，我们返回的是字符串，所以你要加上注解 risk，boss body。

好，我再接着说明写这个方法，返回字符串，方法名随便取叫test DX然后 POS的请求浏览器要向我提交数据，比如说我希望他就提交一个内容，提交一个位置，然后我去处理，当然我这里是模拟就不处理了，我只把它打印出来，只要能接触到怎么处理都是次要的。

然后最终我要向浏览器返回一个屈深圳服换，我们就可以利用刚才所装的工具了，community Ocho点get Jackson get Jackson string，这里我没有不需要返回业务数据，我只要返回一个编号00代表成功了，然后给一个提示消息说操作成功就完了，我就不返回 map。

好了，那么 CTRL这个方法我就完成了，完成以后我们需要有一个网页，网页上面还得写一段解决代码，然后来访问这个好我要新建一个网页，为了方便这样，我就写一个静态的网页，我在这个时代的一个之下，在html里写一个静态网页，这样的话我这里就不用加方法去访问网页，静态网页可以直接访问吗？

好，我就利用一个html，这个文件名我叫 jacks demo这样看起来比较直观，放大一点，这个标题改成叫 ajax好，这个网页我把它搞得简单一点，写一个段落，然后段落里写一个按钮，只要我一点这个按钮就向服务器发送一个异步请求就行了，tap等于8000，然后 y的按钮名字叫发送，我一点就发送一步请求，然后点的时候我要触发单击事件，在这个事件里我要调send的方法，所以接下来我们的重点就是来实现散的方法，那么实现这个方法我需要用几块儿，首先我得在这里引入几块儿，当然我们项目中原有的页面不用引入，它默认就引入了，怎么引入几块儿，咱们随便找一个页面去看。

比如说首页搜一下index，点html再找到底部，这句话引入9块2，通过解决儿官网引入解决儿，不是本地的。好，我就把这句话靠过来，我当前页面就引入了几块儿，然后我再写上 square的标签儿，在这里面写实现 send方法也说明这个方法 send。好，那么用且夸瑞去发送一个if请求非常简单，到了点儿，如果你想发get请求就点get，如果你想发post请求就post非常容易。

当然其实解决儿还有一个万能的方法，就是到了点儿和jacks这个方法的话就是你可以配置任何的参数，你是get请求POS请求什么请求都可以通过这个方法来配，当然它也最复杂。

那么其实我刚才所说的get post是一个简化之后的方法，我们就用简化之后的到了点pose，那么这个方法它常用的参数有三个，为了看起来比较舒服，我通常会要换行三个参数，第一个参数是你要访问的路径，我要访问谁，前面那天然后阿尔法the jacks这是我的访问路径，逗号隔开。

第二个参数就是你要向这个服务器提交的数据，这里我们是要写一个json，就是写一个怎么说 js对象，说明一个js对象其实它也是json格式，js对象的直接量其实就是Jason格式，我需要向服务器提交的是name对吧？Name，然后比如说值是张三，不要向服务器提交年龄age，它的值是23，这就可以了。

最后我们还需要声明一个回调函数 function是匿名的，没有名字。

那么当这个服务器给浏览器做出响应以后，那么浏览器得到响应以后，它会掉这个方法，并且它会把服务器返回的数据传给这个方法，这个data是什么？

Data就是服务器返回给浏览器的数据，当然它是个字符串，好，我要证明一下，怎么证明很容易，我在控制台把 data打印出来，我打他的太 TS有个函数叫type of data就可以看出来这个data它是不是死菌，但是另外我再把 data整个内容打印出来，那么它是字符串，我需要把这个字符串转换成GS对象，这样我才方便用GS去调，怎么转呢？

用解决也很方便，就到了点pass，Jason你把这个字符串传进去，那么它会把这个字符串转换成一个 js对象，我再把它赋给data，经过这句话的执行，data就变成了js对象，我就可以调用它里面的属性了。这里我这样我它的类型再打印一遍，看看有没有变好。

然后我去访问它，访问它，访问这个对象data点，这对象里有什么，它里边不是有扣的有message对吧？我就data点扣的再来一行，得特点卖c好了，总之是一个简单的小实例，而且解决儿发送异步请求的这个格式是固定的，你哪怕把它背下来也没有关系，好，写完以后咱们测试一下，我把服务启动一下，Sorry搞错了。我现在我选错东西了，我应该选 community，application不是YouTube，我说怎么半天不动呢？

好再执行一遍。好执行完以后我就打开浏览器，首先我要访问静态页面，Local host8080community静态页面放到了html路径下对吧？然后页面的名字叫a Jack's demo点html，我就看到这个页面了，它只有一个按钮，好先不着急先点右键点警察，我把控制台打出来看他报错说没有。图标，我们确实页面上没有引入图标，不管它，因为是个小例子，我就点发送了，点发送你看他反馈的结果，首先我输出的我打印的是什么？是返回的数据的类型，string，字符串没问题。

第二个我输出的是整个数据，它是一个字符串，message是什么？扣的是什么？没有问题对吧？接下来我又把这个字符串转成了对象，再打印它的类型就是对象object，然后我data点扣的就等于0，data点message就等于操作成功有没有问题？

好，那么利用这样的方式，你看我们这个页面不管是说路径也好，还是说页面的内容也好，是没有刷新的，保持住了，然后我们在这个情况下访问的服务器得到了一些结果，就这样了。

好，那么这个小事例搞定以后，接下来我们就可以利用这个内容去实现发布帖子的功能了。

好，下面我们就去开发这个功能，这个功能的开发还是那么几步，我们还是从数据访问层开始写起数据访问层，首先我们得有增加一个方法，就是我们能够插入帖子的数据对吧？得增加这么一个方法。

好，我就打开这个 Discuss post map这个接口当中之前我们声明了查询有关的方法，现在我们需要追加一个追加一个增加帖子的方法，这个方法因为是增加返回的是增加的行数int，然后方法名我叫int discus post，那么你插入数据时要传入的是实体 discourse，post，好，这就声明完了就这么一个方法。

下面我们再去实现这个方法也比较简单，我需要打开 discourse，post，map。

好，那么我要在这里就是在写一个音色的语句，然后把刚才的说明的方法实现了，这样音色的语句我也在前面定义一个 circle，然后把这个音色它的字段把它在这儿声明出来，其实insert的字段往往就是比查询字段少一个ID而已，我就可以复制粘贴，但你 ID改一下叫音sir的feels。

Ok，接下来我就要来写 insert的语句，写一个音色的标签，ID等于刚才的方法名，避免写错，就copy音色的，然后返回的不是返回的是参数，因为返回的肯定是一个整数参数，private type就等于实体 copy一下。好，然后我们在这里写 circle就可以了。Insert into discuss post。

然后括号是你要插入的那些字段，我们可以就include引入这些个字段叫insert，讲错地方了，insert，然后我就需要给这些个字段的赋值，这个音色的feels和select feels它不是其实差不多的是吧？那么负值的话，Select它没有负值，我就只能一个手写了。

没关系手写，但是顺序不要搞错了，首先第一个是user ID第二个是标题抬头，然后是content的是内容，正文再来就是类型，帖子的类型，然后是帖子的状态，然后是帖子的创建时间，接下来是这个帖子的评论的数量，他们的抗体，最后是帖子的分数，好了，这个DNA我就写完了，写完以后其实最好是测一下这里，我上课上的我就不去测了，节约点时间，你课后的话因为你写的代码相对少一点，你最好多测一测，测好以后再往下进行。

这里我就偷个懒。

当然如果说一会我们开发完以后，在执行的时候报错，我们再去看 circle，其实也行也可以。好了，数据访问能写完以后，接下来就写业务层面对吧？业务层我们需要提供一个能够保存这个帖子的业务方法，就在这里就需要把我们上次课所讲的过滤敏感词就能用上，因为你添加的帖子的数据，它里边可能写一个大段的文字，可能就会包含一些非法的词汇对吧？就需要做这样的处理。

好，那么接下来我就看有没有已经有了 Discourse，post service，所以说我们直接在里面追加一个方法就可以了。好，我就追加一个方法，返回int，方法名叫at this cast post参数discourse post。首先我先对参数做一个判断，如果说post的值是空的，那我要抛异常了，我就不能往下继续了，在这里给一个提示说参数不能为空。

好，这是第一步对参数的判断。如果说参数不为空，接下来我们就需要对 Post当中的数据做敏感词，相关的过滤 discourse post，它里边的哪些数据需要过滤，我们看一下这个表，看一下表。

Id这个数字肯定不需要这样的，就需要后面是数字还有这个时间了，这些不需要只有title和content需要做处理。

做处理的话，我们一个是过滤敏感词，还有一个事儿，其实你最好也把这个标签也给它去掉，比如说他填写的一个标题，他是这样写的，带了个标签里边的话可能是写的是什么呢？

我们最好把这个标签小于号大于号也给它替换掉，不然的话这个标签显示到网页上的时候，可能会表现出元素，表现出一个标签的特点，比如它里边可能注入了一些对叶脉有损伤的破坏的一些内容，可能有影响，所以说我们不希望我们只希望浏览器认为这是一个普通的文字，而不希望他认为这是一个标签，这个意思，所以说你要对这个标签也做一个处理，这个处理比较容易，是不用的，工具，我们所依赖的

工具其实就有这个功能，我给大家写一下。

首先 post点sat先处理开头写注释，这是先转译，先转译的是html标记怎么转移我们Diao这样一个工具叫html utilize，这是 spring mvc给我们提供的一个工具，然后它有一个方法叫htmlsk你把内容传入进来，如果这里边带了标签小于号大于号，它会做一个转义转成转义字符的，好我就post点get开头，这样的话我就把原有的抬头进行一个转移，当然你如果没有标签它就不处理，同理，还有 content也需要做一样的处理，html ugs, SK, 然后 post点get content好。那么转译完标签以后，接下来还要做一层就过滤敏感词，那么过滤敏感词就要用到我们上次跟他所写的工具了，对吧？

所以我们需要把工具注入进来，那个工具叫三ctf beautiful。那么注入进来以后，我们再做过滤敏感词的处理，pose点set开头，然后三CT filter，点filter，post点get开头。好，同样的处理 post，site content好了，那么该转移的也转移了，敏感词我们也过滤了，那么最后我们就可以去实现插入数据这一步了，直接调迈克就能实现了，我就return discourse的迈克点，insert，discard post，我的参数名叫post，好了，这样就完成了，完成以后下一步就是要开发视图层，那视图层一个是我们需要写 controller，对吧？

第一个第二个对页面我们需要做一些处理，但是我们这次处理的页面不是说随便配个表单，它是一个异步的方式，我们需要写点这个js代码。好，接下来我们就写 CTRL，首先我打开，你看我们之前是做了首页，首页是查询的帖子是在home control里做的，这样我新建一个control，在这个control里来实现增加帖子的功能。新建一个类 this cast post很多了，那么以后再有和帖子相关的业务我们都写在这里来。

好，首先我需要在类上写上CTRL的注解，标记一下，另外我给你俩加上一个与开始卖品配好一个路径给他。这个录音楼写的简短一点就叫discard，简写了，然后那么我们要在这个内容要实现增加帖子的功能，你肯定需要依赖discus post service对吧？我们就把它注入进来。好，朱书记来下面我就可以去处理增加帖子的请求了，注意这是一个异步请求。

放大一点在这放大一点，首先声明它的访问路径，爱的这些路径叫discourse at就很直观了，然后因为你是增加数据，你要浏览器要提交过来很多数据，所以说提交的方式请求的方式应该是post，另外我们给它返回的是字符串，不是网页，所以说加上request包点，那么这个方法public，返回的类型是死菌，然后这个方法的名字我叫ed discuss。

Post，那么页面上传入的是什么？其实页面上只传入两个值，页面上只传入标题和内容，那么帖子类型默认都是普通的帖子状态，都是默认的状态，然后创建时间就是当前时间，这都是自动生成的，默认你新增的帖子肯定没有评论对吧？

默认它也没有分数，所以说浏览器只能给我们传这两个值，ID生成的，那么优质ID就是当前的用户发的帖子，说到这我们还得再注入一个内容，因为你也得获取当地用户对吧？

得通过 Host后者去获取，所以我们还得补充一个猪肉叫 house偷的，好了，那么参数刚才我说了，页面上只需要传入一个标题，另外一个content就可以了。

好，那么首先你发帖的前提是你得登录，你要没登录是不允许发帖的，这样我先尝试从后者里去取优点，如果优者都取不到的话，你后续也不用继续了就有问题了。

好我获取user我判断一下，那么如果说user是空的，就表示说你还没登陆，这种情况怎么办？我就给页面反馈一个提示，注意是异步的是阶层格式的数据，我可以利用它，没有那些有q的工具点get接生词卷给他返回一个403的编码，403意味着没有权限，403代表的是没有权限的意思。

好，然后这里再给个提示说你还没有登录好，那么如果说程序执行到这，假如说他已经登录了，他登录了我们就需要调service去保存这个帖子，当然你在保存之前你需要构造一个实体出来对吧？

我就创建一个POS的对象，ID自动生成，那么user ID我们需要传入的是当前的用户的ID，标题是传入的标题内容，就是传入的内容，另外后面还有类型默认是普通帖子0，状态默认是0，所以说这两个字段不用传也行，因为它默认整数就是0对吧？你这个时间得初始化一下。这些数据默认都是0，还有时间需要处理一下好。我就 post their site create time。然后 New date。Ok，对象构造好以后，我就调用service的爱的方法把它存进去就完了。

好，最后我就return虽然给他一个反馈，我们默认如果说程序执行到这，我就认为成功了，对了，我就给他返回一个正确的提示 community get精神0，表示一个正确的状态，然后提示消息说发布成功，完了有人可能会想说你永远都是成功的，这里就不有可能没有可能这报错什么的，那么如果说程序在运行的过程中报错，那么将来我们会统一处理，我们就不在这一个处理。

所以说这里我写个注释就是报错的情况，将来统一处理，所以说我们认为程序正常，不会报错，而报错是我们在另外的地方单独处理就好了。

Ctrl写完以后，接下来我们就要处理页面了，这个页面在什么地方实现发布帖子，我们之前其实也看过，这里我访问一下首页，在首页上点我要发布点这个按钮，然后再发布帖子，所以我们需要看一下首页是什么样，搜一下index，html首页，那么我找到发布帖子的那个地方，它在头部不在头部里，在这个内容里，在弹出框在这个部位。

那么是这样的，你看我们74行这有个按钮，这个按钮就叫我要发布，那么当你点按钮的时候，这个框就会弹出，然后你在这里输入一些信息给你看一下，就这样，目前默认已经实现了，页面上默认就已经实现了，这个动作已经由GS实现了，我们不用管，然后我们在框里填数据，填的是标题和正文，这两个框中的值我们待会要取到，取的时候通过ID去取，比较方便注意取完以后，那么当用户点发布的时候，这个数据就要发送给服务器，所以说那点发布的时候发布按钮他就得调一个方法对吧？

其实这个方法也已经自动的注册好了，页面上就已经带了。在哪？是在当前页面所依赖的Gs？

李好。咱们可以看一下GS，index的GS里我们找一下js文件，搜一下index点GS你看他这里写了一段代码，说到了function在页面加载完以后，那么获取到按钮，发布按钮，然后给它定义了单击事件，单击时调这个方法，那么这个方法的动作已经定义好了，然后他首先把输入数据的框这个框给隐藏了，因为你就发数据了对吧？发数据以后这个框就不用写上隐藏了，隐藏以后那么接下来他又把一个提示框显示出来了，这个提示框过了两秒又隐藏掉，还有个提示，那这个提示框找一下在这提示框在这里会显示一些提示的内容，当然这个提示的内容我们需要根据服务器返回的结果做一个动态的处理。

好，所以说我们页面上需要用到这些内容，但这些内容不是说我们去直接改这个html，而是我们去处理js。好，下面我就给大家写一下GS，我们再回到index，点GS，在这儿主要是写这个方法。

首先当我点发布的时候，那么我就把刚才填数据对话框隐藏掉不用动，然后不要急于去显示提示框，你得向服务器发送消息，它返回结果以后再再提示对吧？

所以这个是后显示的，我在这中间我需要做一件事就要发异步请求了，发异步请求之前，我得先获取到标题和内容，不好传这个内容对吧？我获取这个标题和内容，好那就得用到ID了，我现在打开的东西太多了，我先关掉。不用的，要不然碍事，我们看一下ID是什么，叫这个名字。

拷过来不用改，我首先取的是标题，开头等于用这个写carry ID选择器，选中的是文本框点Val得到的是框里的值对吧？同理 content也是这样获取的，也是ID选择器，看一下它的ID是什么叫这个名字。

Message、text、点位，好了，取到这个值以后，接下来我就可以发送异步的请求了，这个请求是一个POS请求，因为我要提交数据对吧？那么怎么发送刚才我们不是也演示了吗？到了点pose，然后是三个条件，第一个是访问路径，访问路径是项目名我写上常量，context、pass加上discus，然后艾特这是我们要访问的路径。

然后第二个是你要传入的数据标题和内容，我就开头冒号开头值，然后逗号，content，冒号content的值，最后再加一个回调函数来处理返回的结果，那么这个返回的结果返回的是字符串，我需要把它转成对象对吧？

所以我还是那么写 data，等于dollar plus Jason，那么得到了对象以后，那么我会得到一个状态以及一个提示消息，我先不管三七二十一把这个提示消息显示到这提示框的ID我需要用到这里，我先写一个注释说在提示框当中显示返回的消息，我利用ID获取提示框修改它的内容，是利用txt的方法改它里面的文本，它的内容应该是data点message改了消息以后注意提示框默认可是不显示的，你接下来需要它显示出来，其实就是这段代码先显示，然后过两秒自动隐藏，你不要让用户去点麻烦，所以我把这段代码粘过来，改一下缩进这里写一个注释。

这里的意思是在提示观众显示反馈消息以后，这里面我要显示提示框显示以后，那么两秒后自动隐藏提示框，是这样一个逻辑。然后还有一个如果说我们返回的结果是ok的，表示我已经数据已经添加完了，添加完以后怎么办？我要把当前的页面刷新一下。

好，看到这个数据，如果说你添加失败了，报错了我就不刷新了，但是我给了提示，不管你成功还是失败，我这里都给了提示，但只有成功的时候，我们刷新页面，就这个意思。

在这刷新页面得判断一下是不是成功的，那就判断data点code它是不是0=0就表示成功，刷新页面我们用GS很容易实现，那就是windo点、location、点遗漏，重新加载当前页面，我们就能看到最新的数据了。好了，到这儿我们这段前端的代码就写完了，写完以后咱们测试一下，看看能不能行。这样我重新把这个项目的启动一下，重新部署重新启动。

因为有的时候你的改动比较多，尤其是涉及到了新改的新加了一些静态资源，前端资源的时候最好重启一下，因为静态资源它有可能有的时候它就编译的不到位，你要注意一下，好，完成以后我就打开浏览器刷新一下。目前还有一个问题，我要发布，你看我没登陆，按理说它不应该显示出来对吧？你得我登陆了以后才能显示，所以我需要对这个按钮做一个登录与否的判定。

这个之前没有做，我再打开首页，然后找到我要发布的按钮，就在74行这个简单，就是说你判断一下当前用户有没有登录，你判断一下 log news是不是别闹就可以了。我这就写th if等于log in use，如果说它不等于not，我才显示这个按钮，等于我就不显示了，好，Ctrl f九重新编译，打开浏览器刷新没了好。

为了要验证能不能发布帖子，我需要登录一下，我就因为我之前注册的账号这个是1还是l我不确定。

换一个。

好，登录成功以后看到了这个按钮，我就点我要发布，然后写标题，随便写我就写叉，然后发布提示发布成功等两秒页面刷新看到了一个叉叉叉，当然你现在点点进去还不行，因为我现在一个帖子的详情页页面还没做，但是我们在列表上能够看到叉叉叉这个帖子表示ok了，另外你也可以看数据库，你查一下这个数据也没有问题，对吧？

好，再测一个，我刚才添加的是一个没有任何敏感信息的数据，我再来一个有敏感信息的数据给你看一下，比如说这个标题，我加上标签，然后的话随便写一句话，alert，完了标题上带了这个标签了，然后正文里我带一些非法更非法的信息，也带上标签，这标签我就不带我带上一个 h—可以开票加QQ，然后什么大概就这么一段话，我这里有开票是一个敏感词。

好了，然后我就发布了点发布也成功了，然后你看它显示的是正常的内容，我们看不出来它有没有转译，我们可以看数据库看一下，再重新查一下这个数据，你看这个标题的内容，是不是他把小于号转成了and lt对吧？

把大于号转成了and GT这样的话在页面上显示的时候就不会出现，他以为这是一个标签的情况，这只是一个普通的文本，然后正文可以怎么样？

被过滤了，被替换成了星，比方说我们敏感词过滤的工具也起了作用。

好了，到这儿我们就把发布帖子的功能完成了，总之在这个功能里我们学会了使用 Jack发异步请求，我们也运用了我们所封装的过滤敏感词的工具，也成功了。好，这次课我们就演示到这里，咱们下次课再见。

## 3.3 帖子详情

20-帖子详情.mp4

这次课我们来实现帖子详情的功能，在首页的帖子列表页面上，你可以随便的选一个帖子，然后点它的标题，我们就可以打开一个这样显示帖子详细信息的页面，然后在这里把帖子的详细内容显示完整，这就是帖子详情的功能，那么这个功能开发起来就相对容易一些，因为我们利用之前所学的内容就能把它实现出来，这里就不用再添加什么新的知识点了。

然后我们还是按照之前我们所总结的步骤套路去开发。

那么首先我们开发数据访问层，加一个就是增加帖子的这么一个查看帖子方法，然后我们在业务层也增加一个查询方法，最后是表现层我们除了在CTRL那里处理查询请求以外，那么我们还需要在首页上在每一个帖子的标题上增加访问详情页的链接就加个链接就而已。

另外我们这个详情页的模板，那么我们需要在模板上处理静态资源的访问路径，并且我们让黑的区域能够复用。最后在正文上我们要显示标题，作者发布时间正文等等这些内容，这就是我给大家总结的一个这个功能的开发流程，接下来我们就来演示一下这个功能实现的过程。

好我打开了我的开发工具，按照刚才所总结的，首先我们开发数据访问层，我需要在map里咱们帖子的map已经有了，我需要在map里再增加一个新的方法，这个方法是查询帖子的详情，其实就是根据帖子的ID，然后查询出一个帖子的详细信息而已。

那么这个方法也很容易声明出来，它返回的是一个帖子，所以返回的是discuss post，实体类，方法名我叫做select this cast post by ID。我们根据组件的来查询，所以参数就是ID了。好，这个方法我声明完了，实现它也非常容易，我需要打开对应的配置文件找一下。之前我们已经创建过了这个文件在这也很方便，我只需要在这里再加一个谁来的标签，对吧？他的ID刚才我复制了就是那个方法名，然后他的返回的数据的类型，它的返回的数据类型你需要进行一个声明，那么通过resolve tap进行声明，result type等于discuss post。

好，然后我们就来写查询的语句也非常简单，其实我们可以参考之前来写select什么from这部分是一样对吧？

把它可以拷过来，然后我们在此基础上需要加一个条件，这个条件是ID，所以说很简单Where ID等于参数，ID好了，这个方法我就实现完了，这一层的开发就做完了，做完以后下面我们开发就可以开发业务层，我们在业务层中也增加一个查询的方法，那么我就找到业务组件，discuss post service。

在这里面我增加一个方法，也是根据ID查询帖子，然后查询的时候，这里也不需要做什么敏感词的过滤，也没有什么额外的处理，反正基本上就是一步就搞定了，所以非常简单，把它实现一下。Public discuss。Post。方法名我给它定义成叫find。Discard post，白ID参数，帖子ID。实现起来非常容易，就直接return map的方法就可以了。Discuss post，map点，select by ID，把ID传入就完了，这里就没有额外的业务的处理好。你看数据访问层也写完了，非常容易，然后最后就可以处理表现层的逻辑。

首先我需要在帖子的Ctrl那里增加一个方法能够处理查询的请求，这个也容易，这就是一个比较普通的get请求。当然你查询的时候需要根据ID查，你把ID传进来，我们考虑到这一点就可以了。

首先我先通过声明一下这个方法的访问路径，pass等于帖子详情，所以这个路径我叫detail，然后我通常这种根据ID查某数据，我们都习惯于把Id参数拼到路径里，所以我要求dq的下一级就是ID这里我写的更明确一点是discuss post ID。

好，另外我再声明一下，这个方法的请求的方式是request，method点get。

好，然后当然这个方法我们需要返回模板，就是dq discards dq点html所以说你就不需要写response body什么的，然后这个方法是public返回词句模板路径，方法名我叫get this cast post我写错了，this cast post ok，那么在这个方法上我们需要加一个参数，好能够接收到变量对吧？

那么我们在路径当中取一个值一个动态的值，我们需要用到注解叫pass variable对吧？好把key拿进来，然后我声明这个变量，变量的话你叫什么名字都可以，在这里我也叫discus post ID另外我们需要查询到结果，将这个结果你要发给模板，我可以通过model对象来携带相关的数据对吧？好这个参数就够了。

那么接下来我们就在这个方法之内处理这个请求，处理的逻辑也非常简单，就把帖子查到就可以了。我就查询这个帖子，那我就调用这个discuss post service，Find discuss post by ID然后把discs post idea传给他，我就得到了我想要的的数据，这是帖子的数据，我需要把它传给模板，就model点艾特，然后给参数取个名字，为了在模板上我们引用这个数据方便，我给它取名字简短一点就叫post。

好完了，但是你要注意这里我们还需要做进一步的处理，做啥处理，咱们看一下这个数据看这个表，这里我已经把这个表里的数据查到了，这个表里的数据其他的字段都没有问题，像什么抬头 count那样，你就直接显示就可以了，对吧？然后时间你就格式化一下显示就可以了。

那么用户的ID你要做处理，因为我们在页面上要显示帖子的作者肯定不是希望显示ID对不对？他肯定是希望显示用户的头像用户的名字这样的信息是吧？我这个表里得到的是ID怎么办？那么有两种办法，一种办法我们可以在 Map在这儿实现查询的时候，写一个关联查询，关联到user表，然后把user表里的数据也查到，查到以后，但是这个结果就需要做一个特殊的处理，这个叫关联查询。

那么 my business是支持的，就是说你可以做相关的处理，这是一种方式。

然后另外还有一种方式，我们也可以在这里我先查出了这个帖子的数据，那里里边不是有个ID吗？我根据ID再调user service，把 User查到我通过model再把 user发送给这个模板，那模板得到了，帖子也得到了，user也是ok的。

当然了，那么这两种方式前者我们通过关联查询肯定是效率更高一点，因为——下我就查到了两张表的数据，但是这样做的坏处或者说有缺点就是说我这个查询方法可能就会怎么说有一些冗余或者有一些偶合，因为可能有的业务当中我并不需要展现用户的这些信息可能不需要有的需要，所以说你可能你要么要提供两个方法，如果是你只提供一个方法的话，它就有一些数据的冗余。

第二种方式没有冗余，比较明确，但是他需要查两次效率稍微低一点，我课上我还会采用第二种方式，因为这样比较简单比较方便，其一其二效率问题我们怎么解决，后面我们会讲radius，我们可以把这种有效率的地方，有效有效率影响的数据，将来缓存到rides里，我这一步查询就直接通过readiness，通过内存取到了这个值，效率也是非常的高的，几乎就没有什么影响的。

所以说这样的话我们就既能够让代码变得简洁，然后的话又不会溶于一些数据，然后又能解决性能的问题，这是我们通常的一个做法。

所以说我这里现在我单独查询一下 user，当然对于说有同学想知道我怎么去做关联查询，这个你可以去买这次的手册里去看一看它有好几种办法，你都可以了解一下，好了回到 CTRL了，下一步我要查这个帖子的，其实就是作者这样写错了，作者那也很简单，就通过user service来查，我看这里我还没有注入user service，这里我需要注入一下，好，已经注入进来了，回到刚才的位置，那就user servis fanned by Dee，然后你把帖子的里边的优质ID传给他，我们就能得到帖子的作者了，然后把作者传给这个模板就可以了。

好，这两份数据就是我们模板上需要的，这两个数据我就已经准备好了。

当然其实从完整的功能角度来说，其实你并没有完全写完，因为什么？我们帖子的详情页面，我相信你也看过静态页面，它上面不只是有帖子的什么标题正文，它下面还有帖子的回复，对吧？

但我们现在还没有做这个回复的功能，所以这里我们先不去查询这个帖子的回复相关的内容，那么等将来我们把这个帖子回复开发完以后，这个地方还需要做一些补充好，到这我就可以返回模板的路径，我就是return return模板放到了site，目录下它的名字叫discus gun detail点extent，mail，Discourse，detail，好，这样就可以了。

那么这一步完成以后，剩下最后一步就是去处理那个模板，刚才我PPT上也说了，我们p我们那个模板要处理两个地方，一个是首页上的列表，每一个帖子的标题上面要加一个链接，能链到链接能够访问到这个方法，其实我们再去处理详情页面里边的数据的展现，所以接下来我首先在这打开首页，然后我们找到这个帖子的列表就在这了，119行，然后帖子列表里边，我要找到的是帖子的标题，这个就是帖子标题开通对吧？

这是一个超链接，原先我给它写成了井号，现在我需要把它改成一个能够访问到帖子详情页面的路径，涉及到路径，我们通常都是这样去写th hef里边我可能要带一些变量，或者说我要用到 timeless的特殊语法帮我便捷的处理路径，我们通常都是这样写，括号这样写的好处之前我们也讲了，它能够帮我们补充上自动带上项目的访问路径，所以我们那一级就省略了，无论你将来是否改项目名，这个地方都不用动。



好，然后这个路径我们是这样的，是 `discus`，对吧？`Disk us post control`的路径是它，然后详情页下级是`detail`，`detail`的下级是帖子的ID，这是一个变量，那么如果你在这个路这里头，你既要包含一段固定的值，又要配上一个变量，你不能直接这样写。怎么写之前我们也说了，你需要前后加竖线，竖线里边可以包含的是一个常量以及变量的组合。好，所以后面下一级我要拼一个变量，这个变量应该是帖子的ID，你看帖子的开头是这么取，他的ID跟这个格式不是一样的，对吧？`Map`点`post`点ID好，这样就可以。

我们一点这个链接就能链到详情页上去。那么最后一步就是去处理详情页里面的数据的展现，所以我打开详情页。好，这个页面是我们第一次去编辑，所以我们需要从头开始。首先我需要在前面先声明一下，这是一个`time leave`模板好。

其次我们要把这个项目中的这些静态资源的路径进行一个整理，这个地方我要加上艾特，同理这个也是好当然了，那么在尾部还有一些`js`，我们也需要做一个处理，这样我把回帖先折叠起来，因为目前我们还不会去关注这个地方，回帖我们还没有实现，尾部也不用把它折叠起来，然后 `Gs`文件我们需要做路径的处理。

好，那么路径处理完以后，那么`hide`也需要处理，因为`hide`是可以复用的，那么这里我就写 `th`有`case index`吗？好，`hide`，好，让 `hide`得以复用，最后我们就集中精力把帖子的信息把它展现出来。咱们一点一点来看，首先的话这个是帖子内容，里边是帖子详情，而这个详情分为三部分，这里注释都写得很清楚了。

标题一部分，作者一部分，正文一部分对吧？这三部分我们看哪些地方是需要动态的值去替换的，首先标题的部分，这个就是标题对吧？这里目前写死了需要替换，这是第一个地方。

所以我需要在`space`上写上`th`冒号，这里最好用`utx`因为用户在发帖的时候，这个标题当中可能会带有标签，我们用`you text`可以把这个标签正常的显示成标签的样子。如果你不用`u text`，因为它那个标签我们存的时候不是转译了，它就会显示转译的字符就不好看了。

好，然后这里标题刚才我 `Ctrl`这个对象叫`post`对吧？所以这里面你要写`post`点，标题开头 `ok`，后面这三个按钮我们先不管，先不用处理再往后看，然后是作者的这一部分，那么作者首先他有一个头像，这头像目前写死了，我们需要替换，这里我就写`pH`冒号，`c`好把它替换成咱们不是传过来作者变量名叫`user`对吧？对。就是`user`点 `high`的URL好，这就行了。

另外很显然这地方是作者的用户名，你也需要做一个替换。`Th`，冒号，`Utecht`，等于`user`点，`user name`。然后作者的下面是有一个帖子的赞的数量和回帖的数量，这两个地方我们先不去管不去管，因为什么？我们现在这个点赞回帖还没有实现，等我们实现到这个功能的时候，这个地方我们再来再来处理，还有这地方还有一个发布于某时间，这里要显示的是帖子的发布时间，这里也是要处理，实际上它不可能有标签，所以说就`text`就可以了。

那么这个时间我们需要进行格式化，显示我们习惯的格式，那么要用到`Format`对吧？`Format`，然后`Format`当中我们需要获取帖子的发布时间，就创建时间 `post create time`对吧？另外你要指定这个时间的格式，`杠mm杠ddhhmmss`好了，这个时间也处理完了，最后就剩正文了。那么正文比较简单，反正一个`div`里边显示一个大段的文字，那就`th`冒号又`txt`等于这是`post`点，内容框填。

好，关于帖子的信息我们在这就处理好以后，接下来我们就统一做一个测试，那么我把这个服务启动一下，好启动完毕以后我打开浏览器，我先访问一下首页，翻一下首页，然后我随便找个帖子来看，比如说我看然后你把鼠标放上去的时候，你可以观察一下浏览器的左下角这个地方，能够看到它当前的路径是什么。

首先我们看这个路径的格式没有问题，那个数据应该也对，像ID是234，你看这个是282281是变化的，应该就没有问题对吧？我就点一个点玄学帖点了以后，那么你看这个是帖子的标题没问题对吧？然后这部分用户的名字就是AAA用户发布于某个时间没问题，当然这俩数据是假的先不管，然后这是帖子的正文没有问题，这是用户的头像也没问题。

下面回帖是假的，我们暂时先不管，以后再处理，这是一个，我们再看另外一个，随便找一个，这个我上次课发了一个帖子瞎写的是吧？那也可以点进去看一下，你看它这个标题也是能正常显示成标签对吧？用户名CC没有问题，发布于某个时间没问题，内容做了敏感词过滤也没有问题。好了，那么这个帖子详情的功能我们就很容易就把它开发完了，那么这次课我们就写到这里，咱们下次课再见。

## 3.4 事务管理

---

### 21-事务管理.mp4

这节课我们来学习事务管理，因为我们下一个要开发的功能是添加评论，那么在这个功能里面我们要用到事务管理的这方面的内容，所以我们提前先介绍一下，那么关于事务管理这个话题，我想大家应该并不陌生，因为你在学数据库的时候肯定会学到这方面的内容，但是为了避免有些同学就对这个内容有点生疏了遗忘了，所以我们先对这个内容进行一个回顾，首先说什么是事物，事物就是指由一步或多步把数据库操作序列组成的逻辑执行单元，这一系列操作要么全执行，要么？

全部放弃，全都不执行。什么意思？就是说我们在执行数据库的操作的时候，其实是以某一个业务为单元的，那么一项业务可能是要访问数据库多次，那么我们要保证这项业务的完整性，如果说你这项业务所涉及的数据库操作全都成功了，你就提交都成功。如果中途有一个失败的报错了，你要全都回滚全都放弃，我们是以业务为单元去让数据库执行是否有效做这样一个判定。

那么从学术的角度来说，事物它包含几个特性，就4个特性，原子性、一致性、隔离性和持久性。其中其实原子性和一致性他们阐述的内容差不多就是说事物是一个整体，是一个不可再分的最小的整体执行体，而且事物执行的结果需要让数据从一个一致性状态变为另外的一个一致性状态。

那么所谓的一致性状态就是说，你这个数据需要满足数据库相关的约束，是唯一性约束还是非空约束等等满足，你在没有改变之前满足改变之后依然满足，这叫一致性。另外还有持久性指的是数据事物一旦提交了，那么你对数据所做的任何改变要永久的记录下来，你要存到像硬盘这样的永久的存储器当中，而不是存到内存里。

然后隔离性指的是针对并发的操作在并发的时候，那么独者程每个线程都执行一个事物，那么他们之间其实应该不互相干扰，应该彼此隔离，否则就会出问题。

指的是对并发的一些处理的方案。总而言之事物的这4个特性，原子性、一致性、持久性是比较容易理解的，那么我们就对他们就不多说了，但这个合理性是很值得我们去探讨的。

那么你面试的时候，面试官其实也比较关注这个话题，他很值得去聊，所以我们对隔离性就多做一些介绍，下面我们就看一下隔离性，刚才也说了，事物的隔离性是针对病房而言的，那么我们换一个角度，就是说如果在多线程的环境下，我没有去做事物的隔离会怎么样？

当然我再强调一下，就是我们所开发的服务器程序，它是一个多线程的环境，每一个浏览器在访问服务器的时候，服务器就会创建一个线程去处理它的请求，然后如果说在这次请求当中你要访问数据库，那么就可能会产生一个事物的操作。

所以说其实我们开发的服务器程序，它是独者程环境，是多事物并发的一个场景，那就很有可能说我多个用户同时访问你网站的同一个功能，可能会同时去访问同一条数据，也就是说多个事物同时访问同一份数据的情况就会出现。那么如果你不去做隔离性的处理，就可能会出现一些问题，因为你同时改对吧？以谁为准这个就会出问题。

那么常见的并发的这种问题主要是两大类，一类更新的问题，一类是读的问题。那么第一类问题就是更新，关于更新有两种，一种叫第一类丢失更新，一种叫第二类丢失更新。关于读的错误，一种叫脏读，一种叫不可重复读，一种叫换读。那么这5种常见的异常错误我一会会详细的讲解，反正我们先有个印象，那么我们所谓的事物的隔离性其实就是为了解决这些问题，然后在解决问题的时候，它是分层级去解决的，什么叫分层级去解决？

就是说它分为几个级别解决的程度不同，你比如说read on committee，它是一个最低的级别，然后希尔莱斯堡是一个最高的级别，由低到高，其实这个级别它能解决所有的问题，你可能就会要产生疑问了，既然它能解决所有的问题，还要其他的三个级别干啥是这样的，它能够解决所有的问题不假，但是它解决的问题是有代价的，它需要对数据枷锁，而枷锁会降低数据库处理的性能，会降低数据库的性能，使得性能下降得很快，但我们网站又希望提高性能那就矛盾了，所以它又提供了其他的三个级别，那么我们实际上在开发的时候，我们应该选一个一般都不会选这个级别，因为它性能太低了，我们会选中间的一些级别。

那么以什么为准？就第一个它既能满足我业务的需要，保证我业务的安全性，当然不是100%的保证，比如说百分之九十几的保证，但是很多时候就能被接受，在这个前提下我们追求一个更高的性能，大概是这么一个路子。

当然了，你说我到底怎么选呢？如果你想知道我到底应该怎么选，你肯定要先知道这些个异常代表什么意思，然后每一种级别能够解决什么问题吧？然后你再遇到一个需求的时候，他去分析哪个是我能接受的，它性能就是在我们接受的范围内是最高的，是吧？大概是这么个意思。

所以接下来我给大家介绍一下这些常见的并发的异常分别代表什么含义，这4个级别都能解决哪些问题。好，下面我们就来看这个常见的并发异常这个话题，首先我们看第一类丢失更新的问题，我对它做了一个概括，什么叫第一类丢失更新？它指的是某一个事物的回滚，将导致另外一个事物已更新的数据丢失了，这个叫第一类丢失更新。

然后右侧我给一个表格，对整个的过程做了一个推演，我以两个事物为例，一个简单的数据为例，对这件事做一个推演，后面的其他的问题也是类似于这样的方式去推。

然后第一列是时刻T—时刻t2时刻，这个时间向前推移，但是你要有这样意识，t1到t6，总的的时间间隔也会非常的短，因为计算机的执行的速率非常的快，可能整个t1到t6也不过一毫秒甚至更短的一个时间就完成了。

但是我把它区分的仔细一点，t1到t6，比如说在第一个时刻事物一抢到了CPU的时钟，它先读取了一个数据，假设读的字段是n读到的10，在t2:事物二抢到了这个时钟，它去读也读这个数据，读到的数据也是10，也就是说几乎他们同时去读这个数据，几乎同时得到的都是10，因为t1和t2间隔是非常小的，然后在t3时刻十五二首先对这个数据做了改变，把它改成了9，比如说它减了一个值，减了1改成了9，然后在下一个时刻他提交了对于t2这个流程没有问题，这就是把数量减一，最终提交了他也是成功的ok的。

然后紧接着在t5:，十五一抢到了十中，然后他也去改这个数据改成了11，它是一个加的操作，但是改完以后，后面某些程序发生了异常，于是t6:他回滚了，那回滚以后他原来读到的值不是10吗？所以他一回滚就回滚成了10，那么你看我们单独看事物一这个流程没问题对吧？一开始读到10改成11，然后回滚成10没问题，看事物二也没有问题，一开始10减成减1×9，commit为9也没问题。

但是你要知道，那么这t1到t6个时间间隔极短，他们几乎同时做了这样的操作，然后得到的结果响应给浏览器而响应的的时间，因为网络的延迟，需要一秒钟，所以说网络延迟大远远大于事物执行的时机，所以就在两用户看来，这两个行为几乎同时发生的。

那么这就出现了问题，对于第一个用户来说他觉得没问题，因为我原来读的这个数是10，然后我尝试把它加一失败了还是10没有问题对吧？而对于用户二他就不能接受了，因为他得到的数据是10，他尝试去修改也没有报错改成功了，但是最终的结果是什么？

最终结果还是10他的9没有体现，所以你在去看一下我总结的这句话，就是某一个事物的回滚，它的回滚导致另外一个事物已经更新的数据没了，你回滚是你回滚，但是你把人家的数据给整没了，这就是第一类丢失更新的问题。如果你不去做事物的隔离，就会有这样的问题。

好了，那么这一类问题了解完以后，那么后面的问题就都好理解了，因为都是这种形式去给你讲解。

我们再看第二类丢失更新，这个其实刚才类似只不过他是这样解释的，是某一个事物的提交，导致另外一个事务一更新的数据丢失了，刚才才是回滚，这个是提交而已，还是像刚才的场景事务一先读读到10，然后是万，这三步操作读成10-1为9，提交以后，然后在t5: 15 1+1成功了。

T60他没报错，他提交了11没问题，但是响应到浏览器的时候，那么用户一用户二用户一能接受，反正我原来10加了11没问题，用户二不能接受，因为什么呢？我是10减了1变成9也没报错，但是最终结果变成了11对吧？这就是问题。

所以第一个事物的提交导致第二个事物它已更新的数据没了，这是第二类丢失更新。

好，再看还有一个问题叫脏读指的是某一个事物它读取了另外一个事物没有提交的数据，还是看这个表格这样几个时刻在t一时刻是v读到的数据n等于10，t2时刻他把这个n加一改为11，然后t三时刻失误二去读n读到的是它加完以后的值11，而t40克它回滚了。

因为可能在此之前有人有这个问题发生，它回滚变成10，但是对于用户二，他读到的结果反馈的结果是11，但实际已经变成了10，他得到一个错误的结果，一个脏数据这叫脏读。好，再看这个叫不可重复读，指的是某一个事物，它对同一个数据前后读取结果不一致，我一个事物读取同一个数据，前后间隔很短的时间范围内，读到的结果却不一致。

那么这样推演t一时刻事物一读到值10，t2时刻事物二读到的也是10，然后第三十个15亿把它改成11，第四把它提交了，对于15亿没有问题，反正数据提交了成功了11，但在t5时刻，那么事物二又读了一下这个数据，得到结果是新的结果，11你看在t二时刻t5:，那么事物二先后两次得到结果不一样，那么这两个结果分别参与运算，那么不就矛盾了。如果说这个时间间隔t2和t5间隔很长的话，还说得通，那么它们之间的间隔是很短的，小于一毫秒的这个就不能被人接受，在这么短时间范围内你就矛盾了，这就是问题。

那么这叫不可重复读，在很短的时间间隔内，我读同一份数据不同了。

好，再看最后一个是换读，换读其实和刚才的不可重复读，从这个定义上看有点相似，但其实它角度不同，换读指的是某一个事物，对同一个表前后查询到的行数不一致，不可重复读是对同一条数据查到的结果不一致，换读是行数不一致，换句话讲，换读指的是我查询多行，数据导致不一致，而不可重复读是查一条数据不一致，他们的角度不一样。

好，我们再推演一下，比如说t一时t一时刻，事务二做了一个查询，条件的是ID小于10，比如说他查到了三条数据，ID的分别是123，就这样假设我们目前的数据库里ID小于10的就这三条数据，123，然后在t二时刻失误，一增加了一条新的数据，比如说数据ID等于4t3:他把它提交了，提交以后此时的表里就有了4条数据对吧？ID小于10就1234，那么在t4的时刻，事物二又做了同样的一个查询，ID小于10就发现了有4条数据的，那么在t一到t4:，这两个间隔很短的时间范围内，他先后查两次，得到行数却不一样，这也出现了矛盾。

好了，那么以上是我对常见的并发的异常做了一个介绍，那么当然了可能是你一时异常的情况比较多，你一时记不住，你课后的话对Ppt对照着这段视频你反复的咀嚼一下，反复的琢磨一下，把它吃透了，因为需要你面试的时候能够把这几种异常能够给面试官讲清楚，表示说你对这个事物理解事物的隔离性理解得非常的到位，所以说这个一定要记住。

好了，那么针对这些问题，如果你不处理这些问题都会发生，而我们利用4种隔离级别去处理，那么不同的级别就会呈现不同的问题。

好，所以这里我又列了一个表格，第一列是隔离级别由低到高排列，后面的是每一列是一种问题，然后这里面标的y和n表示说这个级别它会不会出现这个问题，会出现我就标红了。那么你会发现就第一者第一个级别read on committee的叫读取未提交的数据，这种隔离级别是最低的。在这种隔离级别下所有的问题都会出现，换句话说他就什么都没处理，他根本就没处理。

那么所以如果你选择这种隔离级别，基本上在并发的环境下，数据的安全性是没法保障的，所以基本上我们很少极少情况下会用他的，当然也不排除有些极端情况会用，比如说某些业务当中我对数据的安全性完整性要求不高，非常不高，你也可以选它，因为它的效率最高对吧？它越不安全效率就越高，但总之用的很少。

再看到最后这个级别序列化，希尔莱斯堡这个级别，那么所有的问题都不会出现，那么相应的代价是它要加锁，那么它的性能是最低的，所以我们平时用它也用的比较少，因为性能差，当然也不排除比如说你是一些银行的金融类的业务，那么对数据的安全性要求是极高的，那么你选择哪怕性能低，我也要选这个也是有可能的。

那么对于我们普通的互联网的需求来说，通常中间两个是比较适合我们的，那么中间两个它都解决了一部分问题，你比如说repeatable read，重复读，允许你重复读，那么它解决了4个问题，只有1个换读，没有解决，其实在绝大部分的应用当中，换读是能够被接受的，为什么？

因为换读不是查询多行数据，往往是在我们在统计一些内容的时候会出现幻图，比如说我统计一下我当前有多少个帖子，过一会我再统计，或者我统计当天有多少个用户一会再统计，那么可能会前后不一致出现这种情况，这个我们可以一个是通过我们的业务去规避，比如说我把统计都放在后半夜，凌晨三四点钟，这个时候几乎没有在线用户了，很难会出现有很小的概率会出现换流的问题，对吧？

再一个即便出现了，反正我是后台统计，那么影响的是我自己，那不行我再统计一遍，对吧？

反正一般情况下是容易被能被接受的，然后 read committed，它解决了两个问题，这叫读取已提交的数据，那么后三个问题会出现，那么很多互联网的应用，它可能对数据的完整性要求没有那么高，那么对性的要求就很高的时候，这个时候你可以用所以我们在日后的开发中基本上集中焦点是放在这两者当中的，那么如果安全性要求高一点，通常用不那么高，一般用当然了，到底应该选择哪一个，那么你在工作中最好和你的同事和你的leader去探讨一下，然后根据你的业务去探讨一下，不同的业务应该做出不同的选择，是这样的。

好了，那么了解了隔离级别以后，我们需要再去对它做一个更深层次的理解，就是说数据库保障事物的机制它是怎么实现的？其实有两种实现方式，一种叫悲观锁，一种叫乐观锁。那么数据库它自带的实现机制就都是悲观锁，而乐观锁其实默认没有，我们需要自己去实现。那么所谓的悲观锁指的是他看待问题非常的悲观，他认为如果并发就一定会有问题，既然它一定会有问题，我就提前要对数据加锁，否则就不行了。

而它加锁有两种锁，一种叫共享锁，s锁也就是说式微对某个数据加了共享锁以后，其他事物只能对这个数据也加共享锁，但不能加排他锁，那么他加了共享，所以后他能读到这个数据，但它不能改这样的而排他所叫x所，就是说是a对某个数据加了排他，所以后那么其他事物对这个数据就不能加任何的锁，那就是说其他事物就不能读也不能改，是这样的。

好，所以悲观所分为这两种类型，而乐观所我们需要自己去实现，那么乐观所的意思是我们总是很乐观地看待问题，我们总认为即使病发了，通常也不会有问题。

我先不管我该读读，该做什么操作做什么操作该计算，然后当我真的计算完了，我需要更改数据的时候，这个时候我就看一下数据变没变，如果数据变了，那就证明我处理的过程中有人改了，这个时候我应该放弃我的这次操作，否则没人改我就提交我这个数据，是这样的话，既能保证效率比较高一点，但出现问题的时候，我又可以做一个规避，那么我们怎么去识别这个数据变没变，通常是给这个表加上一个字段，或者是版本号，或者是时间戳，反正能够识别这个数据变没变就可以了。

然后在任何人更新数据的时候，你更新数据之前都要看一下版本号或时间变没变，因为你要更新数据，之前我们往往是要把数据先读到然后去改它里面的某些字段改完以后我要更新的时候，我就看一下，这个时候版本号变没变，如果版本号变了我就不更新了，我给用户一个提示，这次操作失败你再来一次，如果没变那我就可以更新，我就更新，更新完以后更新的同时，版本号要加一，或者是时间说要更新为当前的时间，就是这么一个逻辑。

好了，大家对实行机制有一个大概的了解，能够和面试官就是聊到一块去就好了。当然如果说你想对实行机制有一个更深的理解，你可以再找一些文章自己去研究一下，可以做一个更深层次的研究。

好，那么这是以上事务管理的相关内容，我们在开发的时候又怎么办？我们开发的时候其实 spring 对事务管理提供了支持，提供了非常好的支持。就是 spring 它有一个模块叫 spring data access。那么你从 Spring 官方手册上有一张是讲这个内容你可以去看，然后这个内容当中的第一个小节就是叫 transaction management，就是事务管理，可见他对事务管理多重视这个东西多重要。

而且 spring 事务管理也恰恰是 spring 一个引以为豪的这么一个技术点，就是说无论你底层你选的是什么数据库，你是 my circle，你是 oracle，你是 DB2，你是 space circle server 等等，甚至你底层选的是 ready，是包括 DB no circle 数据库也没问题。

那么 spring 在对任何数据库做事务管理的时候，那么它的 API 都是统一的，对我们来说都是透明的，我们就一套 API 能够管理所有的数据库的事务，非常的牛。

那么具体来说它管理事务有两种办法，一种是叫声明式事务，就是我们不需要写逻辑，不需要写一套逻辑一套代码，我们只需要在配置文件里做一些配置，或者是在 xml 里做配置，或者是通过注解，在方法上做配置就搞定，就加点配置就可以了，这个非常简单，我们这个课程中基本上就用这种方式。

还有一种是编程式事物，就是说我们需要写一段代码，写一段逻辑去解决问题也不难，那么需要用到的是 transaction，template 的类去解决问题。那么有同学会说既然它这么简单，还有学的必要吗？我也会给你演示，为什么呢？因为可能是你工作的时候你有一个业务逻辑，它非常的复杂，它可能涉及到了比如说 10 步数据库的操作，而我们需要控制事物，只是其中的中间的 5 步，其他的不需要控制，这是一个方法，十步操作。

如果你用这种方式的话，你把注解加到方法前，整个方法都被控制了，不是我们想要的对吧？我只想控制局部，这个时候编程式事务就会起作用，所以这两者我们都需要掌握好了，关于事务管理，我就给大家大概的过了一遍介绍了一下。下面就是演示了，我就给大家演示一下 Spring 这两种事务管理的方案到底是怎么去做的。

好，我打开了开发工具，然后这里只是一个演示，我们还没有正式的在项目运用，所以我们写一个 demo，而这个 demo 我们是模拟说某一个业务，所以说我们这个代码需要写在 service 业务层，我就写着阿尔法 service 这里边，然后这个业务比如说我就杜撰一个业务可能不是真实存在的，比如说我要为这样一个需求，我首先要注册一个用户，我就要增加一个 user 对吧？

第二步我要自动的系统自动的给用户发一个帖，就是新人报道，这也是增一个增加的行为，增加一个帖子，所以你看这一个业务包含了两个增加的操作，我认为这是一个完整的业务，要保证它的事务性，我就以杜撰的业务为原型给他演示这两种方式。

我既然要在 service 里访问 user 和帖子对吧？所以我希望把 user 和帖子的 map 注进来，首先是 user map，然后是帖子 discuss post map 好注入以后我就编写两个方法去演示这两种业务的这两种事物的管理的方式。第一个方法，public 最终返回一个值，比如返回一个 object，然后方法名我叫 CF— Cf—，这个返回值我就直接写上了，return ok 就完了。

好，然后我先把这个方法内部的逻辑先写出来，然后我们再说怎么去管理事物，逻辑就是先增加用户，再增加帖子对吧？我写个注释，新增帖子就是以用户的身份发一个帖子，我就直接杜撰一个数据优点 6 悠着，然后悠着点 site you 在 name 随便写一个阿尔法，然后有点 sad。加密用的盐。我要生成我就看没有 Youtube，二结束了。

第二 generate uuid 然后截取一下 sup stream，然后从 0~5，然后有点点 set，password 密码是加密以后的密码，community、YouTube、点、MD5，假设用户输入的密码很简单，123，那么我需要给它加上盐，然后再加密存入数据库，然后悠着点 set EMAIL 邮箱，比如说是阿尔法艾特 QQ.com，还有 user 点 site，either you are 给他写一个路径，这个路径就写一个优客网，自带的 image 点，然后 q 点点，com head，然后比如说 99t 点 PNG 好还有一个创建时间 site，create time 我就当前时间 new，完了那么其他的占有字段默认就可以了。

然后我就调用user map，然后点insert，把用户插入到库里，第一步就完成了，再写第二步，第二步增加一个帖子也是类似杜撰一个数据 discourse post等于new discuss post，然后 post点sat，优质ID，应该是用户的ID，当然这个用户我们没有给他赛特ID，不过没关系。

那么正常情况下他插入完数据以后，那么这个买卖点会向数据库要 ID，然后得到ID以后它会塞到这个对象里，所以映射了以后user就有ID了，那么我就user点get ID，当然我们还得检查一下，user map里边，我们在insert的时候是不是写的句话写了这句话，那么当他插入完数据以后，生成的ID就会付给属性，这就没问题了。

然后再来然后是post点set，开头标题我叫hello，再来 pose点sat content这里就写新人报道。好了，最后 post site create time。当前时间好，然后就调用map把这个数据插入到库里，好，这个也就完成了。完成以后，假设我后面还有一些逻辑，在执行这些逻辑当的过程当中报错了，这种人为造个错，因为如果不报错的话，正常情况下反正你不管理事物，这些数据都能插进去，我们怎么看这个事物有没有效呢？

就是说我这报个错，我看他能不能回滚回去，这样能看出来这个事物有没有有效。好，我就随便找个错，Inter点儿ly6off ABC意思是我要把 ABC这个字符串儿转为整数，肯定转不了，肯定会报错。

好，那么如果说我们就这样写，去执行会怎么样？数据依然会插入进去，因为你现在没有对他管理事务，那么他每次执行dml语句，执行完以后马上就屏蔽他，马上就生效，不会回滚的。所以我们需要在这个方法之前，通过注解的方式管理事物，让它成为一个整体，如果任何地方报错，我们都要回滚回去。

那么我们在使用当中通过注解的方式去做事务管理非常简单，就在号码上加一个注解叫 transaction就可以了，这样的话他会选择一个默认的隔离的方式，如果我们想手动指定你需要加参数，隔离性我们现在叫参数叫 Isolation等于我们调用那个类as lesson点常量。

比如说我选择效率较高的read，committed就是我们刚才讲的第二个隔离级别，读取已提交的数据，然后我们在配置事物的时候，其实很多时候还会连带再配一个内容，这个内容叫什么？叫事物的传播机制。我先写上，然后的话再解释顺便解释一下。那么 preparation传播机制，那么传播机制要引用的也是一个常量，这常量定义在preparation这个类当中，然后我把它指定为这个值 Requires的。

好，那么你可以看一下常量这里有好几个值，什么requires的？Suppose？Request new，never nasty等等，一共有7个值，那么这7个值我就不挨个解释了，不要啰嗦，我挑几个常见的解释一下，大家对它有一个必要的了解就可以这个解释，我在前面写个注释进行一个解释。

我们最常用的是三有三个，一个就是瑞卡的，好，第二个是 request new，第三个是 nest。

那么首先说一下什么叫传播机制是这样的，你想我在写这个业务方法的时候，是不是有可能我会调用另外的一个组件的业务方法对吧？就是业务方法a可能会调用业务方法b而这两个业务方法a和b它都可能加上这注解，都可能去管理事物，你a调b的时候，我掉了b b的事物应该以谁为准？以b我自身的机制为准，还是以a的机制为准，还是一个什么新的方式为准，对不对？因为这需要这涉及到一个两个事物交叉在一起的问题。

那么所以说事物传播机制指的是解决这种交叉的问题。

好，那么先说瑞克尔，你说这个事儿的时候 A调b你想想 b的情况，瑞克指的是它支持当前事物，那么你去看看手册也好，还是看类的注释也好，类如果你下载源码也有注释的，它有注释的，就是一块的。

好，你可以自己翻译一下，然后的话这里我就直接跟你解释了，它指的是支持当前事物，那什么叫当前事物 a到b对于b来说a就是当前事物说当前事物不直观，你可以把它理解为外部事物a调b b b被a调用了，对于b来说a就是外部事物，我们就这样说就是外部事物就调用我调用者，调用我的外部事务，支持外部事务。

那么如果这个不存在，外部事物不存在，那就创建新事物，你看这个单词叫瑞克尔需要他一定需要事物，如果说a调我a有a本来有事物，我就按a的事物来对吧？如果a没有事物，我就创建一个新的事物，按照我的来就大概是这个意思。

好，然后瑞克26它指的是创建一个新的事物，并且暂停当前事务，或者说暂停外部事务，a到b无视a的事物，你有没有我不管有我也给你暂停掉，我永远都传承一个新事物，按照我的自己的方式去执行，这是比如plus6，它需要一个新事物，然后 nest解释一下，它指的是如果当前存在事物，也就是外部事物，如果存在外部事物，则嵌套在该事物中执行，什么叫嵌套在外部事务中执行呢？

A调BA有事物我就嵌到这a的事物里执行，这里指的是就b在执行的时候，这个事物在执行的时候，它是有独立的提交和维稳的，它有独立的提交和维稳，就是我没侵占你 a的受理执行，但是我却有独立的提交和回滚，这个意思。好，那么如果外部事物不存在，否则它就和这个一样了，如果不存在它就和 request一样的。

好了，这是我们对这个事物传播机制的一个解释。通常的话我们可能用 Repair的或者是replace new多一点。具体来说用哪一个你了解了这个事物的大概的传播机制的意思以后，也是你得根据你的业务去判定你在实际的工作当中遇到了这种情况，你要和同事探讨一下，我这种情况下，我选择哪个是比较合理的。

好了，这个方法我就完成了，完成以后咱们测一下看它有没有效，我就写一个测试代码，这样我新建一个测试类，这个类我叫transaction test，然后它前面需要加上那几个注解，我把它拷过来，然后因为我要执行的是阿尔法service里的代码，所以我要把阿尔法service注入进来，那么注入进来以后，下面我就可以写一个测试方法测一下 test cfe方法非常简单，就调一下就完了，因为方法内部这个数据我都造好了。

点CF1，它会返回一个object，我们看一下我们能不能得到这个结果，把它打印出来看一下，OB记好，然后我就执行这个方法，执行但是在执行的过程中一定会报错，因为我不是造了个错对吧？本来是要转成这个数字，但是我给他传的是一个字符串转不了，它有个明确的提示。

好，我更关心的是什么呢？这个数据到底存没存进去？我们从库里查一下，我先查user，然后倒叙，因为我们新的数据会在最后我按照it道具来查，你看没有阿尔法说明没插入成功说明回滚了对吧？再看这个帖子也是倒叙查，也没有新人报道的帖子对吧？没有。

好，那就意味着什么？我们写的入园是有效的，当这个程序执行的过程中发生异常的时候，它完整的回归了，它保证了业务的完整性，成功了。好，这只是一种方式，刚才不是说了吗？还有另外一种方式，编程是事物，下面我再给大家演示一下，另外一种方式，还是基于这个业务来演示，还是这两步操作。那么我就再声明一个新的方法，也是返回update，方法名叫CF二，然后我们在这个方法里我们要用编程式事物，我们需要注入一个bean Bean就叫纯Jackson template。

B是 spring自动去创建，并且自动装备到容器里头，我们直接注入就可以用。然后利用 B我们可以利用它执行circle，它执行的circle就能保证事务性，就能保证这4个特性，我就注入它 transaction template好注入进来。

那么我们接着写 CF二这个方法，首先我们用并去管理事务的时候，我怎么去像刚才一样去声明隔离级别，怎么去声明传播机制，很简单，我们就直接调 template，然后点sat是类似来我然后这个时候我需要调一个类，它里面的常量对它进行一个设置，这个类是纯Jackson，纯Jackson definition点儿，比如说我还要用隔离级别 read committed，这里应该 read committed，你可以把它放大一点。

看完整一点，他没错。就这样去设置。那么设置传播机制也是类似的行为，类似的方式， Site， application，然后 be heaven。那么这里边我们也是需要这个类调用它的一个常量去进行设置点，然后是propagation，我还是用 you can't provocation you can't跟刚才一样。

好了，然后我们怎么去利用它执行SQL，保证这个事物很简单，我就这样直接调用它的一个方法，这个方法叫做sq的，然后这个方法需要我们传一个接口进去，这个接口叫纯Jackson call back，是一个回调接口，我可以用匿名的方式来实现，就直接new transaction call back， new transaction call back。

好，这里面我们可以声明它里边返回的这个地方用范型指定我期望最终返回的数据是什么类型，因为我们整个方法是object，所以这里就object就完了。当然了你这里遛完以后，我们需要实现接口所带的方法，我 out加insert实现方法。



好，那么这个接口它有一个方法需要我们实现，角度in transaction，这个方法返回的类型和范型是一致的。

然后这个方法的参数，这个方法首先说它是一个回调方法，它是由他不累的底层自动调，但这个方法的逻辑是由我们提供的，我们需要在这里面实现我们想要的逻辑，然后由他们那边去调，他在调的时候会对这个方法进行一个事物的管理，而且这个参数也是由它来传进来的，但现在我们用不上它，我就把刚才的这两步这两步再把它代码拷过来，省点事直接粘到这里来，然后最终也是一样，我业主也退了。

Ok整个方法执行完以后，这个返回的值将被这个template返回，所以就返回回去了。

好，这个数据我改一下，别和刚才一样，我就不叫阿尔法不叫贝塔，然后密码还是123，这个没问题，QQ邮箱贝塔，然后头像999当天时间没问题，帖子又在ID是这样的，然后这个标题比如改成叫你好，然后正文叫我是新人，其他的不变，错误还在就这样就可以了，不管怎么说固定的格式你就固定这么写就可以。

好下面我们对它进行一个测试，再打开测试类，这个逻辑和刚才一样，你也可以粘贴复制粘贴 test，CF二调用 CF二。

好，然后执行一下这个方法还是报错，我们还是关心的是数据的情况，我们先看优质表查询没有增加成功对吧？看帖子查询也没成功，那么就意味着我们刚才的程序它也是成功的回滚了。比方说确实这段代码的事物被成功的管理了，也就证明了这样的代码管理事务是没有问题的。

好了，总之这两种方式平时一般情况下我们优先选第一种简单，那么如果说你的业务逻辑比较复杂，而我们恰恰只是想管理中间某一小部分的事物，你可以用第二种方式。好，那么事务管理我们就介绍完了，那么下次课我们就会对它进行一个运用。这次课我们先讲到这里，咱们下次课再见。

## 3.5 显示评论

### 22-显示评论.mp4

这节课我们来实现显示评论的功能，就是在这个帖子的详情页面下面，显示出帖子所有的相关的评论，然后下次课我们再去实现增加评论的功能，然后再增加评论的时候再用上我们上次跟他所学的事务管理方面的内容。

这样安排的目的是因为我们在开发增加评论的时候，是基于显示评论会比较方便一点，那么显示评论功能并不复杂，然后我们在开发功能的时候，也不需要用到什么新的知识点，我们还是按照以前老的套路先来开发数据访问层，然后开发业务层，然后开发表现层，那么这里面每一层大概要做什么事，这里我都做了一个大致的这样一个列举。

我们就先从数据访问层开始写起，那么我们在开发数据访问层之前，我想我们先对这个表做一个了解，会比较好，所以下面我给大家介绍一下评论这张表。我已经打开这张表查到了一些数据，然后这个表的第一列 ID没什么可说的组建。

第二列是user ID，用户ID就是说这个评论是谁发布的，那么是对应的用户的ID。

然后第三第四类是itttititidintit是实体，这代表什么意思呢？是这样的，就是我们发布评论，我可以对帖子发表评论，你看这个页面我可以在最下方对这个帖子发表评论，我也可以针对某一个帖子的评论去发表评论，所以我们评论的目标是有变化的。

当然我们这个项目只是做了牛客网的讨论区的功能，牛客网还有其他的，比如说看课程的功能，你可能还会对课程进行评论，还有题库，你可能会对题进行评论，对吧？你甚至可以对某个用户进行评论，所以评论的目标是有很多种类型的，我们在开发的时候，你不能说我评论帖子写一个代码，对吧？

评论再来一套代码，那样就太low了，而是应该把这些评论的业务我们进行一个抽象，然后的话我们用一套业务解决所有的评论的问题，我们想设计一张表能够解决所有的评论的问题，是这么一个目的。

所以我们这里涉及的是这两个字段，一个是实体类型，其实就是评论的目标的类别，比如说一代表帖子，二代表评论，3代表用户，4代表题，5代表课程，类似的，然后 nttid就是你这一个类型的目标，具体是哪个目标？比如说具体是ID为228的帖子，还是229的帖子是这样的。

然后后面还有一个target，ID是什么情况给大家介绍一下，咱们看页面静态页面，我在评论的时候可能是我对某个评论进行评论，在这进行回复就可以了，或者我也可以这样，比如说我一点这个回复，你看这里提示回复谁谁，所以这里也是评论也是对评论的评论，但是这个评论是有指向性的，它指向某个人，他不是普通的评论，他还要指向某个人。

这个时候我们用target ID来记录指向的人，然后 Content就好理解了，是评论的内容，然后是状态，状态零代表的是正常的，那一代表的是它被删除了，被禁用了不可用的。

然后后面是创建的时间好了，总之只要你理解，其实这三个字段其他的就容易理解。

然后我们评论表里面它已经带了一些数据，这是我之前我在开发的过程中自己造的一些模拟数据，我们就在数据基础上做查询，已经现成数据有了，所以我们是可以去开发查询的功能的。

好，我们要开发查询功能，首先我们得把这个表的实体类型写好，对不对？那么我就打开这个idea，然后我在安推包下新建一个实体类叫comment，然后给它增加一些属性，首先是ID，然后是uzid反正就和字段的——对应就可以了，这里我就不进行详细的解释了，然后是it太太 mttid target ID然后是内容状态，抓紧时间好，然后我要生成盖特赛的方法，另外再生成一个to stream好。

这个实体类我就写完了，写完以后接下来我就可以开发数据访问组件map了，那么这是一个新的表，我需要新建一个map，那么我就在dao报价创建一个接口，叫做 comment map。好放大一点，那么给它写上一个注解，map。好，我们查询评论它是要支持分页的，你看这要分页查询的，因为某一个帖子可能有很多的评论，对不对？那么分页查询我们需要提供两个方法，一个是我们要查询某一页有多少个数据，另外一个我们还要提供一个方法，查询就是一共有多少条数据，我们好计算总的页数好。

所以我需要声明两个与查询有关的方法，第一个方法返回的是list。Comment有comment。方法名我叫c let。Comments by mtt就是我查询我根据什么来查询，我是根据实体来查询，我要查询帖子的评论还是评论的评论，还是课程的评论，对吧？很显然这个实体是你一个绕不开的条件，所以我声明条件是mttk还有 nttid实体包含两个条件。另外为了分页，我们需要增加分页的条件，一个是offset，另外一个为每页显示的行数的限制。

好，另外我再声明一个方法用来查询这个数据的条目数可以反馈整数，Select can't by under it，而这个条件还是under it，我把这两个条件 copy过来，好，那么 map这两个方法我就声明好了，下面我们逐一的把它们给实现了。

对于数据访问组件我还是利用配置文件 xml来实现，我可以随便找一个配置文件 c CTRL v复制粘贴，然后改一改，这个名字我改成叫comment点x32，它里面的内容我先给它删掉了，我要重写，然后当然name space里边的值你需要做处理，你要与comment map做匹配好。

然后当我们查询的时候，我先定义一个查询的字段，为了方便以后我们多次查询的时候可以复用，那么ID我叫select fields，好，这里面我写上我要查询的字段，那就所有的字段，ID，user ID，Entity type Entity ID，然后是target ID，content，standards，create time。

Ok，就这么多字段，然后接下来我就来实现那两个查询的语句。它的ID我copy一下，ID等于它我们查询要声明返回的类型是集合中封装的，是comment实体类，然后我就来写上。查询语句很简单了，就是select，然后这些字段我可以引入它10万，他们的表加上一个条件，Standards等于0，斯特等等于0表示这个数据是有效的，对不对？

所以你要加一个有效的判定。

然后除此以外，还要加上另外的两个条件，就是与安贴贴有关的安贴贴太以及 Entity idea，最后我们需要分页之前我们给它排个序，我们按照时间正序排列，也就是说你先评论的在前面一楼，后评论在后面二楼以此类推，然后加上一个分页的条件 offset，limit这个查询，某一页的数据我就写完了。

然后接下来我们再实现这个就是查询一共有多少条数据，ID等于它那么这个方法返回的是整数就可以了。那么select com ID条数不让他们然后 where条件其实就是这一部分，这里边因为我们只差一个整数，你就不需要排序分页什么的，对吧？但其他的条件需要保留。

好了，那么这两个查询我就写完了，写完以后这个数据访问层就开发完了，那么接下来我们就来写业务层的话，查询其实没有什么复杂的业务，就是直接调这个方法把它数据查到就行了，比较容易，那么我在12位是包下咱们新建一个业务组件，comment service我需要对它进行声明加上一个注解，然后把我不需要的这个内容注入进来，我当然就需要刚才的 map对吧？

他们 map把它注入进来，其他的组件暂时不需要，我就暂时不注入了。

然后这里边我需要提供两个业务方法，一个是查询某一页数据反复集合，他们方法没有叫饭的comment by Andy。那么参数就是an TT type，an ttid还有分页 outset以及里面那么实现了我就直接return调用 map的方法，一步就搞定了，这个就不需要什么额外的处理。Undertake under the ID offset limit。

Ok，同理另外一个方法也是非常简单的实现，这个方法我叫饭的comment，那么需要你传入的是anti tat以及mtatid也是我就直接return tonight，点select，count by anti就可以了，把条件传入即可好。

那么很快我们把这个业务层面就实现完了，实现完以后接下来我们就要处理表现层的话，一个是我 control要处理对应的请求对吧？

第二个是页面我需要展现数据。好，你看这个页面是帖子详情页面，那么我们访问帖子详情页面的请求是在迪斯卡斯POS的肯抽里写的对吧？所以我们现在要做的事情是对 discourse post control做一个修改，补充一些内容，不用新创建一个control了。

好，所以我就打开之前我们写好的迪斯卡斯普斯克扣了，然后这个是我们查询帖子详情数据的方法，我们就在这里补充一些内容，然后因为我们现在要查询的是这个评论对吧？

所以你需要依赖于评论的service，我们需要把最新的comment service注入进来。

所以我在前面补充一个内容，comment service把它注入进来以后，我们这次查询不单单要查询帖子数据，还要查询评论，而且评论的数据还要支持分页，分页的话我们知道我们之前不是需要传入一个配置，这个配置需要接收整理分页的条件对不对？所以把它生成好。

然后你注意只要是实体类型一个招聘，那么我们声明在这个条件当中作为一个参数的话，那么最终是不是mvc都会把并存到 model里，所以我们在页面上通过model就可以获取配置，大家要记得规则，然后这个方法当中之前我们实现了查帖子以及帖子的作者，对吧？

尽量我们要补充内容就是查评论的信息，首先我们需要把评论的分页信息做一个完善，这个我查评论的分页信息把它补充一下，我现在配着点site里面，我先设置一下，我每页显示5条评论，因为我模拟的测试数据数据不是很多，你要是每页显示10条还不够它显示的，所以我们也没显示5条，能够看到分页的效果。

除此以外再设置配置点site，pass它的路径，路径就是当前的路径，就是这个路径对吧？我就写好 detail之前，还有这部分别忘了，然后得听我的下面还有一个帖子的ID，我们这个是一个动态的条件需要拼上去对吧？

我就拼上帖子ID，帖子ID这个好拼的这参数里有，然后配置点set，rose你要告诉他一共有多少条评论数据，一共有多少条数据，它好能算出来总的页数，评论的数据我们可以从评论表里统计出来，或者我们也可以从帖子表里查出来，你看帖子这里不是有一个字段叫common count，对不对？

这个comment com是我们冗余把每个帖子对应的评论存了一下，为什么这么做？

我之前在讲帖子边表示也说过这样做是为了我们查帖子相关的内容是方便，这样的话他效率比较高，对吧？不然的话你每次查评论，每次查帖子还要查他的评论，效率比较低，然后你看比较新的数据，这个数据都是有的是对的。所以这里我就不去查评论了，我就直接从帖子里取。所以 Set rose是post，get comment，count。好了，那么设置完分页的信息以后，接下来我就可以分页查询了，对不对？我就可

以调 Comment service了。

点儿 find commons by anti。

然后首先我要查的是帖子的数据，你要指定帖子的类型，帖子类型比如说是一然后帖子ID是多少，帖子ID就是之前我们查到的帖子的ID，当然后面还有附件条件，比如说 page the get offset，而且错了，get offset，配点get limit当然这样写的，我们把实体类型写成一写死了。

这样不太好，因为将来可能类型会更多，最好把它可以配置或者说让它装成常量，将来好改，这样我把它维护到长城接口里去，我搜一下接口，community constant。

好，我在这个接口里增加两个常量，写上注释，一个是实体类型，具体来说是帖子，int实体类型itt太，帖子它等于一，好再来一个，第二个是实体类型，评论。

好，这个是comment好了，那么这个得等于二了，换一个值，当然可能以后还会有更多的值，但现在我们就先不处理，我们先处理这两两种类型。

好，那么当前的 control为了使用接口常量方面我最好实现接口，实现完以后，那么刚才这个地方我就引用接口中的常量 anti tape，post也是一好，这样分析查询我会得到一个集合集合的名字，我叫common list。

好这个方法有点长，我中间断开一下太长了。

好，这样我就得到了当前帖子的所有的评论。然后评论里我们还需要就这样还不行，我们需要补充一些内容，补充什么内容？你看这个评论数据，评论的数据里它有iduzid对吧？它有uzid像它get ID你像这样的ID你得把它转化为就是优质对象，我们好得到的的优质内码头像等等这个内容对吧？不然你显示ID不太合适，我们还是用老办法去处理我们便利集合，每次遍历我们就构造一个map，然后把 Comment存进去，把我们依赖的数据查到也存进去就好了，对吧？

好这里我就list，然后 might用map对我们要展现的数据做一个统一的封装。这个名字我取名叫comment。Vo list。什么叫vo就是 view object就是显示的对象，Map其实是我们用来显示的一个对象，所以就不叫 vo list。那么让他做一个区分，然后等于new a released。

好，实例化完以后我要遍历刚才的集合，但是我先判断一下，如果说集合非空，我就可以对它进行遍历，每次遍历我会得到一个 comment，我遍历了comment list每次遍历我需要创建一个map对吧？用来封装我要呈现给页面的数据，这个map我取名叫comment，vo等于6巴西map显示对象，然后我这里面存一些存什么，首先把 Comment存进去对吧？那就是comment vo点put，我把comment先存进去，另外我要把评论的作者就是优点存进去，我就comment vo倒容易点put user，那么 user的值我可以调用user service就犯的方法去查，而他的ID就是当前评论的 user ID对吧？

好我们页面来展现评论的时候，它相关的用户的信息也很容易就能得到，这样比较方便。好了，这是关于帖子的评论，你注意，不只是帖子有评论，评论也有评论，每一条评论也可能有多个评论对吧？所以我们在遍历的时候，我们得到每一个评论，还要把他的评论也要查到，也要返回给页面好显示对吧？这里有点绕，就是说你帖子也有评论，评论也有评论，我们给它取个名字，这样的话容易区分，不然的话我们在命名时都很麻烦，我这里写个注释，就是评论普通的评论，评论我们称给帖子的评论我们管它叫评论，然后我们管另外一种评论叫回复，就是给评论的，评论我们叫回复，这样的话好一点，所以刚才我们查到的是什么？

我们查到的是评论的列表，就是给帖子的评论的列表，然后 View历史的这是什么？就是评论的 Vo vo就v object显示对象的列表，然后这里因为这个地方代码比较长，所以说我把这个注释写的稍微详细一点，不然的话课后的话你看就很晕。这个map是什么？Map就是一个评论的vo这里我们是往 vo里添加了一个评论，然后又往没有里面添加了一个评论的作者。

好，这是我们刚才所做的事情，把它理清楚，接下来我要做的事情是什么呢？我要查询回复就评论的评论来查询回复他，回复也是多个，所以是个列表，这个也好查，我就还是调comment service，然后还是调find comments by and点去查好，为了断个句不断先写，然后再断。

这次我们查的是评论的评论，就是我们查的目标是评论的目标是评论，所以这个地方 anti对tag你要写成这个 comment，然后 ID应该写的是评论的ID，评论的ID不就是comment点get ID是吧？

然后每个评论的评论它通常不会太多，即便是多我们也不做分页了，你在这里做分页就恶心的对吧？

一个页面你套着两层分页就很恶心了，所以这里就不分页了，有多少查多少，但是我们所提供的方法是支持分页的这块怎么办也好办，反正我们从0开始，从第零行开始查，然后这里面我指定一个条件是 integer，max value最大值有多少条查多少条是这个意思就不分页了，相当于我们得到但是你给我断一下，我们得到的依然是一个集合 list，comment。

名字取得直观一点，不然一会乱。名字我叫回复 Replay list。

好回复列表有了以后那也是回复列表，这里边也得有user ID，你也要带上user，我们也需要做类似的处理，是吧？所以我們也需要再说明一个回复的vo列表，回复的vo列表，那么这也是一个类似的里边它存的是卖的名字，那叫 replay vo list。You are released。我需要遍历集合去构造 Vo list，对吧？我就还得做一个判断，如果replace list不等于空，我就遍历集合 comment。贝拉门就别叫comment了，别和刚才重复，我叫replay回复，然后 replay list。

好，每次遍历我们得到的数据，我要把它装到 vo里，我得先实例化一个vo也就是一个map，VPN vo=6by released，new的是希麦克搞错了，然后我首先往 map中存的是回复存一个replay，replay vo点put名字我叫replay直观一点。当你数据多的时候最好把这个名字取得很直观，不然的话容易混淆。

然后其次我要存的是回复的作者那就是replay Vo点tote user。那也是需要现查一下you there's service find大ID这个ID应该是回复的优质ID。好了，回复这个数据我也就补充好了，但其实还是有所欠缺的，欠缺什么呢？他盖的ID需要处理，刚才我们帖子的评论为什么没有处理他盖的呢？因为帖子的评论，他没有这种指向型的这种回复，他刚才的指发生在哪呢？只发生在回复里，它不会发生在评论里，明白吧？

所以说我们在这儿需要处理 target，ID就是回复的目标好了，回复的目标首先我需要查到回复的目标的优点，我得判断一下，先判断他有没有这个目标，有可能没有，有可能是你这种普通的回复他没有目标对吧？只要你点这个回复的时候就错了，只要你点回复的时候它才有目标。所以这没有这两种情况你要判断主要是判断的。

他get ID等于0，这里写个注释就是回复的目标，我就replay点get他是get ID，看一看它等不等于0，有没有值，如果是0那就没有目标，我就不去获取它，如果有目标我要获取有着service点饭的白d这个ID应该就是他get idea，对吧？我这样查询得到的就是目标用户是一个优点，他的名字我叫他get。

好，然后我需要在 Vo里把 target也带上，就replay vo点put。Target。Target。好了，到这儿我就终于把回复的数据把它补充好了，补充好以后最后你不要忘了，我们需要把 map vo装到集合里，对不对？装在一个集合里，但是它没有值，对不对？没有。好，这是关于回复的一个处理完以后再跳到循环之外，我们还需要做另外的一项处理。什么处理？

你这个循环结束以后，你是把回复处理完了，然后的话其实就是把整个 com的vo构造好了，你要注意你这个回复应该装到comment vo里，不然的话回复装的没装进去它就丢了对吧？我们 com的vo是返回给这个页面，通过它获取到要展现的一切数据，所以你在 if之外，在遍历之外，你要把集合装进去那就comment vo点put取个名字叫 replace。复数，然后装的是这个东西。

好，到这儿，我已经把 comment每一次的comment vo构造好了，它里边不只是包含了评论，也包含了回复，列表相关的数据。好，那么除此以外，我们其实还要再补充一个内容在这个地方，在 Video里再补充一个内容，补充什么？我们看页面就是我们每一个这个回复，每一个评论它有几个回复，这块有一个数据有两个文字写成二，有一个回复写成一，当然写错了，前面写错了，所以说你这个数量怎么来呢？对吧？

所以说你给它页面返回的不只是回复的列表，还应该包含回复的数量，因此我们应该在这补充一下，往 View里装上一个回复的数量，这个数量我们可以查到，我就查一下，这里写注释，回复数量，调用 comment service的 fund，comment com的方法对吧？把 an TD type设置为comment，因为你是回复的回复的数量 comment。

然后 ittid是每一个回复的，ID每一个评论的ID your comment点get，然后我把查询的结果用变量记录下来 replay on，然后把它存到 vo里头去。点put我叫瑞贝卡 replay com。

好，最终也不要忘了 Vo你要把每一个vo应该装到集合里对吧？目前这个技术还没有使用，你看它有波浪线有个提示对吧？

好，我们最终还得再补充一句，就是往集合里添加 vo，看我们的vo好了，到这儿我们终于把这个数据查好了，这块有点绕，大家在开发这段代码的时候，你要怎么说要有点耐心，然后一边对照着静态页面去琢磨这个数据的关系，然后一边慢慢的一步一步把每一步的变量写清楚，把每一步的逻辑理清楚，然后再去写，其实理顺了，还比较清楚的，完成以后，最终我们在整个循环之外，我们得到最终的结果要放到model里，传给这个模板是model点艾特艾特取个名字是叫comments多条评论，那么我们存进去的应该是 comment，没有历史好了，到这儿我们就把啃臭了的逻辑写完了，写完以后接下来我们就需要处理模板了，这个模板首先我们先打开首页，index我记得首页上好像也有一个评论数量，我之前没有处理，这里我们现在开发到功能评论的功能，我们需要把它处理一下。

好，我们先找一下哪个地方有这个评论的数量，其实就在这了，帖子不帖子列表，那么在帖子的列表里，在这个位置有一个点赞的数量，还有一个回帖的数量对吧？

其实就是回帖就是评论这个7目前是写死的，我们需要把它改成一个动态的值，我们只改7，所以我最好能在7外面套一个标签，在标签上写他们语法，来获取的数据，我就外面套一个死蛋。

好，然后 span之上为了取变量我就th test等于这个可以直接从帖子对象中获取，我们怎么去帖子，你看这里map点post去帖子对吧？那就map点post取到了帖子，然后他们的 count就行了。

好，除此以外，剩下的内容主要是去处理帖子详情页面，关于评论的展现，我再打开帖子详情页面这个页面模板。

好，这个页面比较复杂，我们先看一下静态页面，那么静态页面它给我们默认了带了两个评论，一个是一个是很显然我们需要便利评论，然后在遍历中去输出，因此我们只要保留一项内容就可以了，第二项删掉，以第一项为模板去输出就可以了。

所以首先我先把第二个给它删掉。

好，这是回帖，前面是回帖数量，先不管，然后后面是回帖列表，然后是第一条回帖，后面应该还有第二条，就第一个li还有第二个li我们找一下第二条回帖，第二条回帖就不要了。

好，这只是前端的给我们的一个事例，现在把它删掉，删掉以后，基本结构我们调整好了，我们就从上往下去一个去理一下，然后看哪一个需要把它改成动态的值就去改好。

首先是第一个地方是这个地方，一当前的帖子一共有多少条回帖，这个数据你需要改，那就改一下 th冒号txt等于这个数量我们可以直接从帖子里取，当前页面怎么取帖子的数据，帖子详情里应该有。Post post找到哪儿去了，怎么找不着了，在这儿这里就写post.comment, com, ok完成了。

然后再看这是第一个内容，再看第二个内容，往后找。第二个内容在哪里？第二个内容就是处理这个帖子的展现了，这里是第一个帖子，我们这是以此为模板去应该去循环它了，对吧？这里就注释就去掉，就不是第一个帖子了，就是要循环它了。

所以我需要在 Li上写上循环的语句，有th h等于每次循环我们得到一个什么？你看 comments它是comment view list。每次循环我们会得到一个comment view就是一个map是吧？

所以我们得到的是comment，vo变量名直观一点，我叫c vo cvo就不要简单的叫卖，不然一会去不开了，然后我们便利的是他们集合，这是第二个地方，我们需要便利，然后再往后看，这应该是第三个需要改的地方，这个地方是显示的是评论的作者，这个头像你需要动态的来取对吧？那就th冒号，把这个路径改成动态的值。我们需要从 Cvo中取 user，就cvo点user点，either you are就可以了。

好，这是这一个内容，再往下看。

第四个内容应该是这儿，这显示的是用户的名字对吧？我们目前是写死的，我们需要把它改成变量，贴成冒号，改成又txt了，等于 cvo点user，点user name，用户名，好，然后下一项内容其实不太容易发现是什么这是啥看页面就是一井二井，但是这个写错了，这应该是二井表示一楼二楼，这应该是一个递增的数字，怎么处理呢？

其实我可以这样，我根据这个循环，只要我们知道循环第几次，这就是几就可以了。

然后你还要考虑什么呢？还要考虑我们当前的是分页的，比如第一页它循环5次，第二页也循环5次，第二页的话你就不能是这个数据不能是1~5了，第二页它相当于应该是6~10，因为有第一页数据为基础对吧？

所以这个地方我们得用上一页之前的那一页的具体的多少行，加上当前的循环的次数才可以，重点是循环次数我们怎么获得，其实 timeline给我们提供了获得的方式是一个隐含的方式，这里我直接给大家演示一下，当然我需要改变一的值，而不是井号，我在一外面的套上点标签，套上个 byebye。

好，l标签已经默认倾斜的样式已经去掉了，所以说它就没有任何样式不会有影响的。好th冒号x等于这是一个变量，那么这样我配置点offset配置的offset是什么？是当前页的起始行，我们是自动计算出来的对吧？那么它需要加上什么？加上当前的循环次数，你循环了几就加几，这就是几楼是吧？循环次数是这样的，你这个变量叫什么？我就写什么，后面还要加上一个state，就是说timeline会给我们提供一个隐含的状态对象，这个状态对象的名字固定的形式就是循环变量名，后面加上state，这是隐含的对象的名字，我们可以通过它获取循环次数，.com循环到了第几次，你这样一加这个楼数就算出来了，是这样。

好，这是这个地方的处理，大家要注意。

然后再往下看，这说这开课是不是有点晚，这是一个评论的具体的内容。我们写th冒号，u text等于 cvo.comment点，讨论条件取到了评分的内容。好再往下，这里这不有一个时间，这是一个评论的发布的时间，这个时间我们需要渠道 Th冒号，text等于但是时间你需要格式化，对此点 format，然后 cvo.comment点 crate Thai时间格式喂。好，这是时间，再往下看。

下面这个站目前我们还没有开发站的功能，先不管，然后是回复需要处理，处理回复的就是回复的数量，你要把它取出来显示在这里。

二目前我需要在二外面套上个标签，然后给它写上动态的值，这样我套上一个还是一个l吧，然后 th text等于cvo点什么呢？咱们回到CTRL那里看一下，com的vivo不是放了一个回复的数量对吧？就它了，CV这样复制一下，cvo点.Replay.Count好。然后再往下看，那到这儿其实我们这个屏就普通的对帖子的评论的内容就已经处理好了，以后后面是回复，是对当前评论的一个回复，这是多条回复，这是第一条，后面有第二条，第三条以此类推。

那么同样的处理方式，我们没有必要留这么多个li我们只要留在这一个基础上循环就可以了，对吧？所以我觉得还是留第一个，然后把第二条删了，它只有两个，后面回复输入框你要保留，把第二条回复给删了，不用了，好，注释也不要了，我们主要是要对ln做一个处理，那样的循环便利显示a的内容，我需要在l之上加上一尺一尺，等于我们这次循环的是什么东西看一下。我们内层循环的其实循环的是里边缝对吧？

我们循环的内容其实是 Replace realist而replay realist的名字叫replace对吧？我们要循环的是replace，所以我这里有取名取变量名叫rvo replay vo然后到了括号，括号里边写成CV o是要从CV中取的，CV o点 replace遍历它，然后遍历的时候我们里边要显示什么东西，再往下看。

这个地方 div上显示的是回复的用户，还有回复的内容，回复的用户以及内容，但是你要注意这个回复的用户其实它有两两种情况，咱们看一下这个页面有两种情况，一种情况就是一个用户名，然后一个内容完了，一种情况是谁回复给谁，有目标的情况，这个地方怎么处理呢？

我们可以这样，就是这个地方的展现我们可以做判断，如果说你大概的有值，我就拼上谁回复谁，如果你target没有值，我就拼上，我就只拼谁怎么样就完了，所以这个地方做需要做判断，这样我可以在space上下判断，th冒号if等于比如说rvo点target等于now它是空的，这个时候表示说没有目标，没有目标我就不用去聘上谁回复谁，我就直接一个名字就完了，那么就在这拼上这个名字就可以了，th冒号text等于等于这个是rvo点用户点用户名，完了然后我们再做一个判断，把拷过来，再做个判断，你得这样写，如果他get不等于闹对不对？

他不等于闹，不等于闹的时候，你要配上某人回复给某人，是吧？

还保留前面，我们需要再加一个东西，一个什么，其实它的结构和下面这个是类似的，我手写 class它有class，class等于它也是txt音符，你可以复制粘贴好，然后！里边要显示一个人名，当然我们页面上其实他给的例子是显示这名字没法读了，我就先以他为例这个人，然后他进行回复，所以后面你要写回复俩字，他回复给他，然后这个名字得是动态的是吧？

所以我们也需要在这上写上th，冒号，txt，等于rvo点，user点user name，他回复给这个目标地，这个地方要改一改，要改成目标，你应该是rvo点，目标是他get点user内容，某人回复给某个目标，这样还行。好，当然了后面这个地方你要写上回复的内容是吧？Th冒号you text等于rvo点 replay点content好了，这个地方搞定了，然后再来再往后看。

接下来就是这里了，是这个地方就是一个回复的时间，我好取 pH冒号txt等于当然需要格式化，对此这样format，这个时间从lvo获取lvo点 replay点create time，然后给上一个时间的格式。

好，那么这个回复的时间处理完了，再往后是暂这个暂我们不用去处理。

再往后是这个回复，这个回复是需要处理的，回复当然它上面没有显示数量，需要处理什么呢？这个地方你要注意，我给你看一下静态页面，这个回复这个回复当你点我再刷新一下或回复，当你点这个回复链接的时候它会怎么样？

它会展开一个框点，这个会展开这个框，你注意为什么不点点，它不展开这个框就回复链接和框，它俩之间是匹配的，它俩之间是map match的它和它是匹配的，它们之间是通过一个ID进行匹配的，这个地方需要注意。

然后现在我们一般不是写死的，我们是循环输出每一个评论对吧？它和它的匹配关系需要去维护的。

好了，你看我们之前链接上是这样写的，是井号回复01它对应的是谁？是Div的ID这里是框所在的位置，它和它匹配。然后现在我们是在这个li下循环输出的，你每次这个不就写死了吗？不就写死了吗？你写死的话那就都叫同一个ID，他就没法去一一匹配了对吧？所以我们在匹配的时候Id你应该动态的生成，然后它匹配动态的ID生成也好办，每次循环后面比如我回复后面加一个序列号，循环的次数不就可以了吗？是吧？

所以这块要额外做一个处理。

好，鉴于 idea我需要动态的生成，我就前面加个thid等于这里属于是变量了，当然它不只是全是变量，是一个前面加一个前缀回复固定的值，后面加一个变量，我这样写双竖线回复，然后加个横线，然后后面变量，这个变量是循环次数，循环次数怎么获得和刚才我们讲的一样，就是你循环变量 rvo是吧？

循环变量rvo然后 state这是它隐含的对象固定的名字，然后.com循环次数得到了，所以我这个ID就变成什么呢？

就变成了恢复-1，恢复-2恢复-3每一个就不一样，这儿就要和他去匹配，所以这个地方也需要去动态的去拼，而不能是写死的。

动态拼怎么拼和刚才这个地方一样，th冒号，href等于竖线，竖线前面不动，我加个横线，然后的话后面怎么取和刚才一样，lvo state.com，好了，回复这个位置我们也处理完了，那么再往下看，下面这个表单我们就框输入框我们先不管，我们下节课开发添加评论的时候再去做处理，先不管。



好，后面还有个分页，分页需要处理，但分页的逻辑我们可以复用，首页的分页的逻辑我们不用重新写了，th冒号，replace等于index，冒号首页我记得当时取的名字叫kj nation确认一下。

还没有，其实我们首页的分页的逻辑，那么其他的页面是可以复用的，然后我们可以复用逻辑，然后我们要复用的话，我们在这个位置还需要加一行代码，那和hi的处理方式是一样的，加上th冒号，然后的话fragment等于相当于给这个区域取个名字，取名我就取这个名字就可以了。

好了，你这儿取好名字以后，那么我们在这儿就可以通过首页这个页面的名字以及这个区域的名字对它进行复用了。最后回帖输入这个地方也不用处理，我们讲增加的时候再做处理。

好了，到这儿我们终于把这个页面的逻辑处理完了，有点费劲挺费劲的，为什么费劲？其实不是他逻辑有多难，主要是我们这个页面本来有很多代码，你要是对这个页面的结构不了解，对这个标签不熟悉的话，是吧？你就没法去拆解去做处理，所以你需要首先对这个页面结构有一个了解，哪些去掉要哪些不要你要搞清楚，然后的话一点一点去找到那些需要改变的地方才比较方便。

好了，处理完以后咱们做一个测试，我把这个应用启动一下，但是启动之前它会自动的去重新编译，好启动以后我就打开浏览器，我再新打开一个网页先访问首页，扫描完以后，那么随便你访问首页以后，你看这个数据就变了，回帖数量就变了，12 13就动态改变，而不是写成一个固定的值了是吧？

然后我随便找一个数据去看，比如说点它去看，前面的内容我们先不看了，这块来看一下回帖期，回帖期这不12，好这个问题我们一会再说，我们先看一看后面的内容。

你看后面这12条规定说明后面这个没问题，但这有问题，这是不是我没处理，咱们回去看一下，可能是我忘处理了这个地方好，我再打开Detail。

那个地方是在回帖详情里的一个地方，回帖详情里我找找有一个叫回帖在这，在作者里这有个7我刚处理了，好我还得把它补充一下，我需要在7前面套上一个标签，他说吧，然后它的处理方式和这个地方其实是一样的，你就写th冒号text等于直接从帖子里取就可以了。

Comment com好，这处理完以后，CTRL f九重新编译，然后我打开浏览器刷一下，这就对了。

再看看首页这回就一致了，刚才这个地方有点别扭，然后我们看回帖一共12条回帖，然后一楼2楼、3楼、4楼、5楼，好，每1页显示5条数据，12条回帖。3页先看分页对不对？下一页。6楼7楼到10楼。再下一页。然后11楼12楼。没错，分页没有问题对吧？没问题。然后看数据，你看这第一条回复好吧，你赢了。

然后时间有回复没有回复0，这是2，当然这个数据有点随意，有点随意，好，我们看一看有没有更好的数据，我看一下玄学帖，这个数据比较好，因为他评论是有回复的，你看然后它一共是13条数据，也是3页每页5条，然后你看这个评论先看这个时间还是这个时间，然后这个是对评论的回复，然后你好，这里没有谁回复给谁是吗？没有，然后再找再看一下。

这个帖子是回复量是最多的，37个回帖。

然后的话你看它里面各种形式就有了，然后它有一个自己做评论，然后的话别人回复他顶你，然后的话他又回复谁谢谢，然后他又回复他其实怎么样，然后回复的数量评论一共有三个，回复这里是三没有问题。

那么这个比较复杂的情况，我们看了没有问题，那么其他的应该也是ok的，我们就不挨个去看了。

好了，那么到这儿我们终于把这个显示评论搞定了，其实这个功能并不复杂，主要的时间都花在调这个页面上，然后的话那么大家在开发的时候也是很耐心的把这个页面调好，其实问题就不大。那么这次课我们就先讲到这里，咱们下次课再见。

## 3.6 添加评论

这次课我们来实现添加评论的功能，这个功能也不复杂，然后我们还是按照这个数据层，然后业务层表现层顺序来开发就可以了。然后值得注意的是我们在处理增加评论的业务的时候，这里边需要用到咱们之前所学的事务管理的内容，因为我们增加评论以后，要连带着把帖子那张表里的评论数量加以修改，更新一下数量，所以说在这一个业务方法里需要执行两次dml操作，那么我们要保证业务的完整性，所以说要对这个业务方法进行一个事务管理好了，咱们就一步一步来开发。

首先我们来写这个数据层，我们在数据层首先先把增加评论的方法写好，另外我们把修改帖子评论数量的方法也给它写出来就可以了。

好，那么我打开这个idea，然后我们先找到 comment map，然后我在接口当中在新增一个方法是增加评论的方法，那么它返回的是整数行数，然后方法名我叫insert comment，那么需要你传入实体类 Comment，说明完以后我们打开它对应的配置文件，然后把 circle实现就可以了，配置文件我搜一下 comment，map、xml，这样我还是在一开头把新增的字段把它列出来，在这说明。

当然新增的字段和查询的字段其实就差一个主键，稍微改一下就可以了。

Insert feels，然后把 ID去掉，好，然后我们就写一个insert语句，把刚才那个方法也实现了，insert，然后ID call一下insert comment，然后新增的时候我们需要声明参数的类型，它是一个实体类型，parameter type等于comment。好，然后这里边我们把 insert语句写好，insert into hormone。那么对应的字段我们依赖刚才的 circle就好了，include refid等于它有栅栏问题，整错弄错地方了。

我再拷贝一下，好。粘过来，然后再万64里面我们写上注入的各个参数，第一个是优质ID，然后下一个是安贴一贴态度然后是inttid然后是target，ID，那么大家在写字段也好，还是参数的时候一定要注意大小写，然后是康婷婷再来CD以及克瑞特太好了，那么这个语句就写完了，写完以后就刚才说了还差一步，就是数据访问层，差的是我们还需要更新的帖子的评论的数量。

所以我们打开 discuss post map给它增加一个新的方法。We can soldier discuss post labour。好，那么其实我之前在介绍帖子表的时候就说过这个事，咱们再回顾一下，我们看一下这个帖子，这张表，这张表后面不是有一个comment，com字段对吧？那么这个字段它是冗余的，把这个帖子的数量除了一下，然后的话为了查询时方便，不然的话你每次看帖子，那么这个数量都要从评论表里再查一遍，有点慢。

所以这样容易存在是为了查询是速度快，所以这个数量什么时候更新，就是你增加，评论以后就要更新它好。

因此我在 map接口里加一个能够更新数量的方法，这个方法反馈整数就是你更新的行数，方法名我叫update。Comment com就评论数量，当然你更新的条件你是根据ID来更新的，通常都是这样，就叫ID帖子ID然后更新的是评论数量，这个写的清楚一点，comment。

好，然后我们需要打开接口所对应的配置文件，然后把这个方法实现掉，也是搜一下，这个是discus post map。好，那么我在最后这个位置把这个地方实现一下，放好没有先复制好。Update。然后 ID等于它因为参数是普通的数据类型，你可以不用去声明，返回值肯定就是整数，也不用声明。那么这个语句很简单了，就是update这张表，然后 Site comment，count字段等于我们指定的参数，comment，count，然后需要有一个条件是ID等于参数ID。

好，到这儿我们数据访问层就完成了，完成以后我们根据数据访问层来实现业务层的逻辑，我们就找到对应的业务组件，这样我首先在帖子的业务组件里加一个更新的方法，就更新数量，然后我们在开发评论的时候可以依赖于这个业务组件，那么这个业务组件它不只是可以依赖于它对应的 map，也可以依赖于其他的service，所以我打开 post map然后加一个方法，他们来it update，comment，count。

参数，一个是ID，一个是comment，count。

那么这个时间就非常简单的，就一句话，这里边也不需要什么额外的处理，我就直接返回discus post map点update comment com就可以了，但是你要把对应的参数传进去好了，service写完以后，接下来我们就可以开发我们这个功能的重点了，但是你看这写完以后，他这画了个波浪线报错，在我刚才编辑代码的过程中，可能代码没写完的时候，他报了错，写完以后他没有切换过来，我试着按一下CTRL f9，然后

重编一下试试。

如果还不行这还不行，你可以这样，你可以把整个代码推出a把它全剪切掉，然后再可能再把它恢复一下，这样有的时候就好了。

总之工具它有的时候偶尔也会有一些小的问题，但这个问题没有太大的影响，那么稍微处理一下就可以了。

好。接下来我就打开这个 comment service，然后我要在这里处理增加评论的业务，这是我们这次课的核心了重点了。那么增加业务方法，我返回的是整数和行数方法名，我叫爱的comment，参数就是实体comment。

好，暂且不管返回值，那么这个方法因为里边包含两次dml操作，所以我希望对它进行事务管理，让它保证这两次操作在一个事物范围之内，要么全成功，要么都失败。

那么做事务管理咱们之前讲过，我们通过spring进行事务管理，有两种方式，一种是声明式事务，只要做配置就可以，一种是编程式事务，我们当前的整个方法是在一个事物范围之内，并不是局部，所以我们用这个声明是事物加个注解就搞定了。

所以我在方法之前加上纯@Transactional的注解，然后通常我们都要声明它的隔离级别 isolation等于引用一个常量isolation点，我这里就选用@Transaction的 committed这个级别。

另外还要声明同时还要声明传播的机制， propagation等于propagation点为@Transaction的。

那么如果你在这你把隔离级别传播机制它所代表的含义你忘了你就回过头，然后我们前一次课去回顾一下，好，然后我们就来实现这套业务，这套业务的实现非常简单，首先我要增加评论，那么对于页面传入的实体，我们有必要把内容进行一次过滤，包括 html标签的过滤，还包括敏感词的过滤，因为它里边可能包含一些特殊的内容，如果不处理的话可能就会有问题了。

好，既然要使用敏感词规律，我需要把敏感词规律的组件注入进来。@Transactional优点把它注入进来，那么这样我首先先对先把这个参数判断一下，万一这个参数是空的怎么办对不对？保证严谨一点，如果参数是空的我就抛异常，给个提示说参数不能为空。

好，那么如果参数不为空，我要对评论中的内容进行一次过滤，那就comment的site，content把过滤后的内容传进来，首先我要过滤的是标签，那么利用这个工具 HtmlUtils点htmlSanitize然后传入的是过滤之前的内容。

好，同样的我们再过滤一下敏感词了， site content，然后是StringUtils点filter。 Comment点get content。

好，那么内容过滤完以后，这回我才能够放心的把它存到库里，我就调用康的麦克的音色的方法对它进行一次插入的操作，但是最终我会返回， insert插入的行数其实成功的就一行，然后在我们增加完评论的时候这写个注解，这是添加评论，添加完评论以后我们还需要做的另外一件事就是更新评论数量，但是你注意我们的评论它的针对的目标不一样，我可以评论给帖子，可以评论给评论，可以评论给其他的内容，而只有评论给帖子的时候，对吧，你才要更新帖子的评论数量，所以这里要做个判断，我们这里写的详细一点，更新的是帖子的评论数量。

好这个判断，那么如果 Comment点get Entity它等于其实它等于一的时候，那么等于一的话，我忘记了我之前有没有常量，看一下 community constant，我们上节课写的常量其实就是等于它了，对吧？

为了引用它方便，我也把刚才的 service实现这个接口，那么如果这个类型是帖子，我们就更新帖子数量，更新帖子数量，首先你要查到这个数量，我们可以调康尼迈克，查到最新的数量，它里边不是带了一个方法叫select count by anti的对吧？

首先我们需要传入的是类型anti type。你就可以传type， comment点get under type其实就是它了，就是帖子，然后另外还要传一个 Intid那就是comment点getIdIntid这个对象里都有。

好，那么利用这两个参数我们能查到实体它对应的评论的数量对吧？Count。好，然后查到数量以后，我要把它更新到帖子里，我这里需要把迪斯卡斯pose的map注入，service注入进来调用它。

好注入进来，然后回到刚才的位置，我调用discourse的service，然后我们刚刚写的update和我们的课程方法，那么它是根据帖子ID，帖子ID是谁？他因为当前类型是帖子对吧？所以这个ID就是帖子ID，就它根据帖子ID把数量更新了，更新为最新的值，这就可以了。好，那么这个业务方法我们就实现完了也不复杂。业务层实现完以后，最后处理表现层的话，首先你要在CTRL那里处理了浏览器的请求对吧？

其次你肯定是要对这个页面做出一些改进，我们首先处理CTRL Ctrl的请求，那么这个功能我们可以单独创建一个CTRL，然后我就在包下新建一个controller controller我叫comment controller，然后给它加上注解。另外我声明一下，Curto了他的访问路径，comment。好，另外我在这里要加上一个方法来处理增加的请求，那么我希望请求的路径是这样的，这里我放大一点，然后声明请求的路径是叫爱的添加对不对？然后还要注意这个还不行，咱们看一下静态页面，比如说我进入到一个帖子，然后我要给帖子添加评论，当我点回帖评论完以后，那么我们应该去哪？

其实还应该回到就帖子详情页面，你回到这个页面以后能够看到这个帖子不是更好吗对不对？

而不是去别的地方，你能够回到这个页面，这是帖子详情页面，那么我们帖子详情页面的路径，咱们真实的路径当中是有一集是带上了帖子的ID的，所以我们最终要澄清像的地方需要用到帖子ID，因此为了能够得到这个帖子ID，我也希望咱们这个添加的路径里，你把帖在地给我传过来，就这次请求说完了，你要把帖子在地传过来，我希望你通过路径把它传过来，我好后面好用。

好，所以我在这里定义一个参数，帖子ID，discus，post ID，另外再声明一下请求方式 request，这是一个增加提交的数据就是post，然后返回死卷，方法名我叫爱的 comment。好，首先这个方法之内需要得到这个参数对吧？你需要用到 pass variable注解，得到参数。好，另外因为我们本次请求是提交数据进来，提交的是什么？提交的是评论相关的数据对吧？他不提交页面上会提交什么？页面上其实主要就是提交内容。

除此以外还要提交另外两个隐含的东西，一个是你评论给哪种类型，哪个ID就实体类型实体ID这个学校也还传过来的，所以页面上需要传输多个条件进来，我就声明一个实体来接收这些数据，这样比较方便。

然后实体中接收的是三个值，可能还不完整，我们再补充一下就可以了。

好，然后我们在实现这个方法的时候，你注意那肯定是要用到我们刚才所写的 Service对吧？我把这个service要注进来，好我把service注入进来，然后我们在给 comment补充数据的时候，你要补充uzid你要补充状态，你要补充创建时间，悠哉ID从哪来增加？

一个评论肯定是当前的用户增加的对吧？你应该得到当前的用户的ID，所以我为了得到当前用户ID，我把 House的后者给它注入进来，好了，这样就行了，我们接着就来实现这个逻辑。

刚才说了 comment，它三只传三个条件进来，我需要补充一下，那就是comment点sat又在ID，我从house头这里获取，当然有可能用户没登录，没登录你这样写就会报错，因为你get user是空的，但是没有关系。

我后面会做统一的处理，就是说后面会做统一的异常的处理，所以说像这种问题也问题不大，不只是异常的处理，我们后面还会做这个统一的就是权限的认证，如果你没登录你就访问不了它，所以通过这种方式也可以规避好赛特优质ID，然后 comment再来再site，standards，状态默认给它来个0，表示的是有效的，然后 comment点，site，create time。

那就是当前时间 new date。

好，补充完了这些内容以后，我就可以添加了就调 service点爱的comment，把它添加进去就可以了。然后最终刚才我也说了，我们需要跳回到详情页面对吧？那就重定向 read write，然后从镜像到帖子详情页面，它的路径是这样，diss Cass detail，然后后面还要加上贴在ID贴在d这里有。

好那么CTRL咱们就写完了，那么最后一步就是处理这个模板了，模板其实帖子详情页面模板，因为我们是详情页面上去做评论的，这样我搜一下详情页面，它叫diss Cass, detail, 听见没有？

好，然后详情页面上其实有三个地方可以进行评论，一个是最底下大框中做的评论是针对这个帖子的，我们先把它搞定，他在比较靠后的位置往后长，在这在福特之前，然后分页之后，在这，然后这里面有一个表单，我们要提交表单，所以你需要对表单加以声明，提交方式为post，然后你要指定提交的路径，x这里为什么x前加这个th，因为我们这个路径增加，帖子路径当中包含一个动态的参数对吧？

所以我需要拼一下，我加个双输线，一个固定的值拼成一个变量，它的路径是comment，l的后面带上的是帖子的ID，我当前页面获取帖子ID怎么取呢？上前面找找，因为当前是帖子详情页对吧？我们这里可以很容易获取到帖子的数据，你看这POS的点什么？Pose就是铁的对象对吧？好这就好办了。

回到刚才的地方，这儿我要获取帖子ID就是 post点ID，好路径写完以后，那么接下来我们在处理这个框，这是一个文本域比较大的框，你要给它取个名字加上name，name等于我们要和 comment实体当中的属性相对应对吧？我们要把传给内容的字段，所以说你看name等于 content。

除此以外我们还需要传入两个隐含的条件，就是你评论给谁给哪个实体，而实体包含类型和ID，这两个我们需要隐含传，你传隐含的数据可以用，就是隐藏框 Type等于黑的，框的作用其实和文本框一样，就是他看不见，然后 name等于先传int cat，那么隐藏框的质量需要你通过挖掘指定，这个类型是回复给帖子的固定就是伊朗。

好再来一个，我们还要隐藏着传安推ID，这个ID就是个变量，你这里就得加th，贴成冒号，后面需要写一个变量，帖子ID帖子ID不就是它对吧？好了，给帖子进行评论，这个表单我们就处理好了。

再往上看，是前面的还有两种类型的框，咱们就看这一种类型是给一个评论，进行评论，但是他没有指向某一个人，再一个是你点这个回复，他又弹出一个框，这个框是给评论，评论，但是评论是指向某一个人的，他俩的区别其实就是差在这个人上，我们先处理。

没有这个人的往前找，这个是他然后我们静态页面上在实现的时候，他没有给我们写成表单，我们只能自己补充了，所以我在这儿补充放，我们这里最后有个缩进好看一点，form master的也得等于post，然后他提交的路径其实和刚才一样，我可以copy，这样不容易错。

Common的艾特，然后 pose点ID没问题。好，另外这个表达里有一个框，这里是输入内容的，你需要给它加一个内容，而这个内容也等于空调。

好，除此以外，当然这个表单里也需要另外的两个隐含的数据，所以你也需要两个黑的，我把它掏过来，把它放到后面放到这儿，一个是itt type，这个是回复给评论的，所以 Type是二，然后 ID就应该是评论的ID就不是帖子ID了，这里评论ID怎么取呢？咱们看一看我当前 li他在哪儿？他是在 u他是在 ul之下的，在 ul之下的。

然后那么在 ul li里，我们获取评论的，获取评论是通过cvo来获取。好，回到刚才的位置，就是这个地方你应该写成cvo然后点这是一个卖法，.comment，获取到 comment对象，然后点ID这就行了。

然后最后要注意这个回复按钮，它默认给的是一个button，不是一个submit，所以你要把它改成submit，提交按钮好了，那么给评论做一个普通的评论，我们也搞定了，剩下的还是最后一个表单回复给固定的某个人，好，我们再往前找，其实就是在这儿，你看他回复给谁谁谁就在这儿，当然他这里也忘了写form，我自己补充一下，然后缩进一下 form，上面的配置和刚才的一样，这个没有什么变化，然后里边的框它的name也是 name等于content。

这个button啥都没谈，当然他也需要另外的几个黑灯对吧？

我把这两个黑灯拷过来，好拷过来，当前他回复的也是评论的也是评论，所以y6是2，没有问题，然后评论的那个题评论的是评论评论的ID是多少也是这个值，没有问题，然后这个时候他需要就指向某个人，所以说你这个时候要给后端传一个什么传，你看一下这个表传。

他get ID目标的ID，用户目标用户的ID，所以你还要再加一个黑灯，然后把这个目标用户的ID写进来，type等于黑的，然后 name等于他改的ID，然后他的value应该是个变量，等于好，这里面我们应该回复的目标是谁？

回复的目标应该是当前评论的作者，那么当前这个评论他是谁呢？它其实是 rvo你看当前 li它是rvo它是个回复它的作者。

好，所以这里面我应该这样写， rvo点它的作者是 user，然后点 ID，好，然后最后再稍微注意一下，就这个地方这个文本框它有一个place holder就是一个提示，提示你回复给某某某，这里面他写死了，回复给寒江雪应该是回复给一个变量对吧？

所以这块我们还得拼一下比较好， th冒号，然后的话这里边要拼的话，我就加上这个竖线，然后寒江雪把它去掉，我们要把它换成一个变量，换成那就是换成的是回复给刚才评论的作者其实就是lvo点user，但是你这里要显示的是悠着点悠着内容。好了，到这儿我们这三个表单就配好了，配好以后咱们去测试一下，看看能不能正常的评论，下面我把这个服务启动一下，好。

启动完以后我打开浏览器，然后先访问一下首页，然后没有问题，我需要登录一下登录，我用这个CC登录，登录以后，比如说我要回复这个帖子，就回复换一个比如回复，回复我在这就随便写一个，你好好回帖，回帖完以后没报错，然后跳回了这个页面，然后你看回帖刚才应该是16变成了17，表示ok，但是你回复的帖子它因为它它楼层比较靠后，我们是一楼在前，所以它在后面你需要到最后一天才能看得到，你好，没有问题。

好，这样我再做一个回复一个，另外就是给给评论做一个回复，那么这里有写hello， hello，你看这个hello就有了，好，那么屏障以后我可以切换一个账号，我利用这个牛客这个账号去看一下，看一下以他的角度来看这个数据会是什么样，我退出一下。

然后患者好牛客，他的密码我猜一下是123我有点忘了， nvw7，对了，然后看一下这个事物，这样的没问题对吧？

这样我回到首页，他不是对我这个评论给我做了一个回复，我回复给这个人看一下，我回复给他，我不在这回复就回复给他，说8月回复，回复完以后你会看这是牛客回复给CC， how are you？

好，所以我们分别在这个地方，然后在这个地方以及这个地方做回复，它所表现的形式都是ok的，比如说我们评论就成功了。好了，那么这个功能我们就开发完了，咱们这节课就演示到这里，我们下节课再见。

## 3.7 私信列表

### 24-私信列表.mp4

这次课咱们来开发私信列表的功能，那么我们在开发功能的时候，也并不需要用到什么新的技术点了，但是这个功能还是它的业务还是比较值得我们去探讨去琢磨的。

那么首先我先来说一下业务，私信就是我们黑的部分不是有一个消息的一个链接，然后你点进去以后就能看到这样一个界面，这里我就不给大家看静态页面了，因为静态页面上它的数据是静态的，这里边都是一样的数据反而容易误导你，让你在理解业务的时候有偏差，我这里是一个做好的功能截取的数据，这个可能就更准确一点。

首先我们我们说的私信列表指的是朋友私信这个列表，并不是指的是系统通知，那系统通知我们以后再说，然后我们在开发这个功能的时候，首先我们要开发这个列表，这个列表是显示多条会话，这里面首先你要理解什么是会话，比如说我和CC之间往来有很多次通信，那么我和他之间的通信，所有的通信加起来就叫绘画，所以绘画是针对人的，我和CC的对话叫绘画，我和BBB的对话叫绘画，我和lhh的对话叫绘画，所以这里列举的是我和某一个人某一些人的对话信息。

因此我和 CC对过话，这里就出现CC，我和BB对过话，这里就出现BB以此类推。

然后我和CC的会话它里面其实包含了往来的多条消息，所以你点进去以后能够看到多条消息，所以我们这里也不只是要开发私信列表，私信列表其实指的是一个会话列表，是以人为维度的，然后我们点到某一个会话里能够看到私信详情，而私信详情是你和这个人之间详细的对话，所以说大家首先要理解绘画和就是详情的一个关系，我们会话列表上是显示了我和某人之间的对话，然后的话这里面显示的消息显示哪个消息显示的是我和他之间最新的一条对话就最后的一条对话显示在这里，是这么处理的。

当然了我们在显示的时候可能还要显示我总的未读消息数量在这。那我和CC的对话当中有几条未读在这儿和他有几条未读在这儿，要未读消息要显示出来。

再一个这个消息里还要显示，比如说会话包含几条往来的具体的消息，这里就共几条，它这里其实这个提示不是特别准确，他说共9条规划，其实这就是说这个规划共包含9条消息，就再概括一下我们所开发的私信列表，它所展现的是我和某些人之间的绘画，我和任何一个人之间多次对话为一个会话，所以说这里边你能看到的应该是不同的人，一定是不同的人。

针对某一个会话我点进去才能够看到我和这个人更详细的私信详细的对话。好了，那么像这样的一个功能，我们这个表怎么设计的，我们值得去看一下。咱们先看一下这个表的设计，你看我已经把这个表打开了，这个表叫message，然后首先是ID不用说了，然后是from ID，就是消息的发送，人的ID。To，ID消息的接收者ID。这个很好理解，用户的两个ID谁发给谁，然后先看这个内容。不说了，状态状态零表示未读，表示已读二表示删除，然后这是创建时间。

然后这里边还有一个字段叫conversation idea就是会话的idea，用来标识会话的，因为我们要查询会话列表，其实当然你没有这个字段，我们就是利用 from ID加to ID组合在一起也可以查到，但是不方便，那么有了这个字段以后会方便，我这里给大家说一下这个字段是怎么一个设计。

比如说我有一个用户，他的ID是111，还有一个用户他的ID是112，比如说111和112他们之间的会话，我就会话的ID，我就用就将这两个ID拼起来，中间以横线连接，但连的时候你注意，那有可能这个消息是111发给112的，也有可能是112发给111的，我这个ID如果说我就是直接按照自然的方式来拼的话，可能会出现两种情况，一种就是这样的，按照这个顺序，而这个是但其实你看无论是这种情况还是这种情况，都是111和112两个人之间的对话，其实都是他是一个会话，这两个人之间无论谁发给谁，它都是一个绘画，但如果你是这么拼的话，分两种情况来拼给人，感觉这是两个绘画，我希望无论是这种情况还是这种情况，那么我们批完以后批完会话ID以后是一样的，所以我们有一个规则，就是把小的ID拼到前面去，而大的ID拼到后面去，这样的话无论是这种情况还是这种情况，最终的拼的都是这样的。

所以说我们设计会话idea的目的其实是为了将来我们在查询绘画有关的数据的时候，我们以绘画为条件进行一个查询，筛选的时候，方便大家要注意好了，当然了这个数据其实还是冗余的，那么这个数据完全可以从from ID和to ID推演出来，等有了它以后，我们在查询时效率会好一点。

好了这个表说完以后，下面我们就来开始开发了。开发的时候这样，刚才我这个PPT上也写了，我们一共是两个功能，我们这次课都把它搞定，一个是查询信息列表，会话列表，一个是私信，详情就是点进去看到详细的往来的私信。

我们在开发的时候，我在开发数据访问层时，我把这两个功能需要的方法都一起给它写完，咱们一起做一测试，然后业务方法我们也把它一起写完。

我们在开发视图的时候，我们就分开来开发，我们先把这个列表相关的开发完，测试过，我们再开发详情这样去处理一下。

好，我就打开这个idea下面我就要开始写代码，首先我需要给这个表写一个实体类对吧？我在安贴包下新建一个类叫message，好，然后根据刚才表的结构，我增加了如下的属性，第一个是ID，然后是from ID，然后是to ID，然后是会话ID，这是一个字符串，因为中间配了横线，它不可能是数字叫conversation ID，然后是content，内容。

好，再来是状态，最后还有创建的时间。那么写完这些属性以后，我生成对应的get和set，另外我再生成一个to spin方法，好了，那么实体类就写好了。

下面我来写 Map，把数据访问层的逻辑一次性的都写完，我抄你一个接口，这个接口我叫mic，迈克这里我放大大一点，然后加上麦克注解，这里边其实我一共需要定义好几个方法，一共是1234、五。

对，应该是5个方法，咱们对着页面看也容易看得出来，看看页面点消息，因为这是1个列表，而且是知识分页，所以你要提供一个查询这一页数据的方法，你要提供一个查询总行数的方法对不对？这是两个。

然后你点内容进去以后，看详情它也是个列表，对吧？所以说你也需要提供2个方法，那就4个除此以外你还要查未读消息数量，而未读消息数量有两种，一种是针对某个人某个会话的未读消息数量，还有一个是针对所有会话的未读消息数量，就针对某个人针对我当前这个人的未读消息数量，但这2个其实可以合成1个来做，所以加起来一共是5个方法。

好了，我把这5个方法就是给大家理一理，然后把在这里做1个声明。首先第一个我要做的事情是查询当前用户的会话。没错了，会话列表查询的是会话列表，而每个会话列表里边就每个会话我们显示什么呢？我们显示的其实就是一个消息，但是这一个会话包含多个消息，我们显示哪一个针对每个会话只返回就是一条最新的私信，每个会话只有一个私信，最新的就可以了。

好，所以我最终返回的是一个集合多条数据，集合中的封装的是私信 Message，那么这个方法名我取名叫select conversations，因为他查的其实不是私信他，查的是针对绘画玫瑰画最新的事情，所以我取名跟逻辑相匹配。

我查的是某个用户的规划，所以你这个筛选团队可能是要有用户的优质ID，然后为了支持分页，你要加上奥赛的加上这个里面好，这是第一个方法的声明，好。

再来第二个，第二个就方便了查询当前用户的会话数量，说白了就是你把前一个方法以这个条件查询，最多能查到多少条数据得到，好，我们查询得到的是一个整数，那方法名我叫 select commission，count。参数user，ID。好，再来，接下来我们需要做这样一件事，查询某个绘画所包含的私信列表，就是在开发详情页面时需要用到这个方法。

好了，所以我得到的是一个集合中的封装的是和这个人多条私信麦穗，然后方法名我叫select letters，letter是私信的意思，然后我们这个查询是根据什么条件来查？是根据会话来查的，你和某一个人的会话来查的，而会话的条件是康复是一声ID是吧？

所以我们以他们是一些ID为条件做查询，然后支持分页加上outset limit。好了，再往后与这个方法匹配的还得有一个查询数量的方法对吧？我需要查询某个绘画所包含的私信数量，返回的是整数，那么方法名我叫select。Lighter count参数还是康复c是ID的太长了，我copy一下。好了，那么有了这4个方法以外，我们再查一下未读的消息数量，查询未读私信的数量。好，那么 select letter。Unread。未读Count。那么你查的话肯定是查当前的用户的未读数量，所以你可能优质ID是要带上，你不能查别人的，然后你如果是只以它为条件的话，你会查到用户所有的未读数量，对吧？

但是我们还有这种情况，这个地方是查用户所有的未读数量，而这个地方是查他某一个会话的未读数量，是这个人某一个会话的维度数量，所以说我们再加一个条件，就是会话ID，然后你注意会话ID我们把它作为一个动态的条件去拼，而不是一定会有你传了我就拼上去，你不传我就不拼，这样的话这个方法能够实现两种业务。

那么这5个方法我就声明完了，声明完以后我们需要对它进行1个实现，我需要写配置文件，那么我在map之下新建一个配置文件，随便找个文件拷贝一下，然后粘贴进行修改message map。好，那么改完以后我把它的内容删了，一会重新写，这里面需要改一下 map的名字。好了，目前我们所实现的都是查询的逻辑，所以这里边我首先把要查询的字段进行一个统一的定义，后面好复印 circle，ID等于select fields。

然后把所有的查询的字段都列在这儿。Id十万。Id to ID康，谢谢ID。框天。内容 Students以及 create time。好了，那么首先我们要实现第一个方法，就是它这个方法实现的时候稍微有点绕，不过也不是很难。我给大家解释一下，我先把这个结构写好，See like ID等于它，然后返回的类型是 message。好，那么 circle怎么写？这样我到这个工具里写一写给大家看一下，首先我们说我要查的是会话，然后其



实每个会话就是要查我和某个人的最后一个这个对话，所以说其实我和某个人有多少次对话，那么就应该产生多少条数据，是这样的。

好，首先我们来看一下这个表里我先做个筛选，我们先看一下这个表里的数据都有什么，谁来？行吧，12。卖水的，然后外是这点你要先加上不等于2，因为等于2就是删除了，我们不要二。好，我们看一下有效的数据是多少，然后再来and还要加一个from，ID不等于一，等于一代表什么意思？你看后面它有特殊的数据，等于——代表是系统用户，那么系统用户他给你发的消息这就是通知，这就不是私信了，所以你要把这种通知的数据排除掉，剩下才是私信。

好，我们看一下这数据有多少，其实并不是很多。

好，这里边你看12它之间有反复的有多次这种会话，我们去重以后有多少个，其实我们需要统计去重以后多少，就说白了康复c是ID，去虫以后有多少个，在这种情况下，那就是有多少条数据了。

你可以这样你可以去查一下diss shift，然后 Com c是ID，我查一下看看一共就这么多数据，这是多少个数据一下？

一个23456789 10、11 12 13 14，好像是14个，一共是14个，或者你可以这样count一下。

14个，所以说我们查到的结果应该是14个好了，但是我们怎么去写这个circle，因为我不只是说只要康c是ID，我要的是每次绘画我要的是最后最新的一套数据，一条完整的数据，我怎么得到每个绘画最新的一套数据，一数据它肯定是ID最大或者是时间最后对吧？

所以你可以这样，你可以我这先这么写谁来？Max ID我查最大的ID，当然你查最大的ID的ID，你得利用会话进行分组，对吧？你要不入办一下格若克拜，他说这是ID，然后你查就会得到14个ID，它是每一个规划最后一条数据，我们利用作为子查询，然后去对ID进行过滤，最后得到的数据就是我们想要的了，所以你这个时候理解了，其他就好办。

好，我把这段视频都拷一下，粘贴在这儿，但是我需要再稍微修饰一下。Select max ID from where它等于不等于2？它不等于1，然后不是会败注意，这里边我们还需要加上筛选条件，条件是优质ID对吧？一直加上去，这还没加，然后我这样加 and注意优质ID可能你当前用户可能出现在哪，你可能是发信的人也可能是收信人，所以你就在ID可能等于他或者是他对吧？

你要写一个或者的关系，这地方比较绕，from ID等于user ID或者这个to ID等于user ID。

好，这样的话这段这段代码才算是完整完成完成以后，那么我指的得了ID还不行，我以它为子查询，然后对外层查询做条件筛选，我就可以得到完整的数据外层，我这样写select，然后 include他不让卖c喂ID印对吧？这段话就起到了对于Id的筛选就可以了。

好，最后我需要在查询，结果里我要奥特曼一下奥特曼ID倒序，好，然后在支持分页就limit offset，然后 limit。好，那么第一个方法就写完了。第二个是查询数量其实也是可以和类似的，只不过它不要分页而已，我再把第二个方法也写一下。Id等于它然后第二个是它数量，它的返回的类型就是整数。

第二个我也需要以这样的类似的这样一个条件去筛选，对吧？我把它拷过来，然后当然也是以它为子查询了，我这样写外层查询，谁来看？

看着我看着谁，我一会内政大学我就要取个别名，这个别名，然后内政大学我给它取一个就子查询我取一个名字叫m吧，m点max ID服装子查询，把刚才代码贴过来缩进一下，当然这个地方给它取个别名，要不不好引用 as max ID啊，然后整个子查询为了好引用加个m所以我是从子查询里统计最大ID的数量，也就是绘画的数量有它以后就容易了。

好，再来其他两个就简单了，就比简单多了，好。

再来select ID等于他，然后返回的类型还是message。我这个就不用嵌套了，Select include他有这样的你搞错了，他然后让message，我这个方法是查询某一个会话之下包含的消息数量，所以你应该把会话条件拼进来，还要支持分页，当然也应该加上这样的条件，就是不等于二，不等于一，这个是要加上去的。我就写了where standards不等于2and12，ID不等于1，按的conversation ID等于conversation

ID, 然后 Out there by ID desc然后还得分页里面的outside里面的。

好了, 那么查询就完成了。下一个也很简单, 查数量, 当然它的返回的类型要声明一下是整数。好, 那么查数量就count一下就可以了。Select count. Id from message。条件的和刚才的上一个circle是类似的, 它不等于二, 它不等于一, 然后它是ccid等于它, 所以我直接粘过来, 这就完了。

最后还有一个查询未读的私信的数量, 它的条件是又在ID和com CS ID, 而后一个条件是动态拼的, 不是不一定有的。好, 最后再把它也实现一下, ID等于它然后返回的也是整数, select can't. Id. 不让 message然后 where是未读的消息数量未读, 那么就是状态你得等于零才表示未读。当然另外弗朗麦蒂依然要在这来, From ID依然要不等于一, 这也加上去。

然后你注意未读是我当前用户的, 未读那一定是别人发消息给我的, 所以说你就在ID拼进来, 应该拼到哪上去, 应该拼到to ID上去, 一定是别人发给我, to, 我的对吧? 是这样的, 所以你要按的 to ID等于参数又在ID, 这个条件一定要有, 然后 come see是ID, 如果有我们就拼, 如果没有就算了, 所以要用if来拼。

If test等于 com c是ID, 如果它不等于空, 不等于我就把这个条件拼上去, 就按的他ID等于他。

好了, 到这儿我终于把这些个方法实现完了, 我们这次实现写的方法比较多, 再一个有些方法还比较啰嗦, 有点绕, 所以说对不对? 不好说, 最好我们要测一下。

接下来我就写一个测试测试代码测一下, 这样我就在 map test这里测, 我就不新建类了, 那么我把新的map注进来, 好, 我重新写一个方法, 我就一个方法把所有的刚才的那几个方法都撤了, 就不写那么多了, test select。

Let us咱们逐个去测, 首先我要调用message, map点儿, 说第一个select conversations。好, 我要查谁, 我要查看一下。这样我重写一下谁来。行不让 message然后把这两条加上去。我查111这个人的信息, 因为111这个人的消息比较多, 是吧? 它比较多, 所以我传的 Uzid是111, 然后分页我查第一页, 比如10条, 那就01010个绘画。好, 然后这样我大点20咱不是说了一共14条数据对吧? 我这样写的话, 直接把这14条数据都看到比较好, 然后我就 list message等于它那么查到数据以后, 我把它遍历一下, 然后打印出来看看。

好, 每次输出我打印这个message, 好, 我先执行一下, 先不写别的了, 先执行一下。好, 你看最终的结果多少条数据? 123456789 10 11 12 13、14, ok, 没有问题。然后其实就是所有人这里我造的数据是这样的, 所有人其实都和111通信了, 所以你按照111查, 是把所有的数据其实都查到了, 这样比较方便, 就没搞得那么复杂。

然后你看这里的数据或者是from里有111对吧? 或者是to里有111, 那都是和111有关的对话, 都是他的绘画就这样。

接下来我们再测第二部分数量, Message, map, 点, select conversation, count还是111, 我会得到一个数量, 好, 一会我一起执行, 我先把另外两个方法也写出来, 接下来我们再测一下查询某一个会话之下的私信 select, 我要查哪一个会话, 我查这个就111和112, 所以在会话的ID应该是111下划线112对吧? 然后零10给他一个分页条件的限制, 我得到一个集合, 然后同样的我把集合来输出一下, 我可以 copy。

好, 然后再来我再查一下集合的数量是多少, 我们用另外一个方法可以直接查到卖c的map点see light lighter, 然后 CTRL ccid是它count等于它把它打印出来看看, 最后再测一下我查到的查的未读数量对不对? 好, 说到未读数量, 我们得看一下我这儿到底有多少未读数量, 因为未读数量比较少。Select seeing from the message where status等于0, 然后 and from ID不等于一。看一下, 你看vih的数量一共没多少, 其中规划1131它有两个, 我就查这个。我就a message map select letter, unread count。这个to, ID是1131131, 看他这个人的位置数量, 而绘画是他。好, 然后抗体等于未读数量等于它, 然后我再打出来看一下。

好，这几个方法都写完以后，我统一执行一下，看看它对不对？然后大家看前面这个是绘画的列表，刚才我们看过了不看绘画的数量14没有问题，1112之间的会话的详细的私信有这么多，一共有12345678条，后面查到数量也是8条没有问题。

然后最后未读消息数量131的是2。好，在这里面我们确认一下111和112它之间是8条私信吗？咱们可以这里查一下。再加个条件就不等于0了，这个就是说不等于2，and看我们先生ID等于11112我验证一下刚才我查到的数据对不对？不然错了就不知道。你看一共是12345678。没错。

好了，经过这个测试发现我写的代码ok，没有问题，那么下面我们就可以进行下一个环节就可以开发业务层，好下面我去开发业务组件，这样我在思尔维斯包下再新建一个类，因为这是一个新的功能，这个类叫message service。

好，然后我给它加一个注解service，这几个查询业务其实怎么说很简单，我们直接调 Map把数据查到就可以了，这中间不用做什么特殊的处理，所以我们就定义5个方法，直接调 map把它实现就可以。

当然首先我需要把 map注进来好，然后第一个方法是查询会话犯的consist这块没有引入一下，然后参数一个是优质ID，一个是offset，一个是里面实际上很简单就一句话，my smile点谁来看看CS，好这就完了，同样道理，另外4个方法也是这样简单的实现我就不啰嗦了，直接把它写出来。

好了，这几个应用方法我就写完了，这里没有什么逻辑我就不说了。现在业务层写完了，最后就是开发表现层，那么表现层刚才我们也说了，我们需要有两个功能，一个是会话列表，湿气列表，一个是点进去私信详情，都是分页显示一些数据，而这个数据其实数据并不多，比如显示一个用户名头像对吧？一个内容还有一个数量，时间差不多就完了，然后一循环就这么点东西就可以了。

然后分页我们还是可以复用首页的逻辑，所以说就很方便，所以其实看起来好像两个功能很多，其实很少。

好，那么接下来我写CTRL处理私信列表的请求先处理我在CTRL的目录下新建一个类取名叫message，很臭了大一点，然后前面加上CTRL的注解，好，然后这个类我就不给它取别名了，当然你取不取都行，我就不取了。

然后首先声明一个方法，处理私信列表请求，那么先声明了它的访问路径，访问路径是let类似的私信列表对吧？

然后请求方式是get，因为是查询方法我叫get later list，然后没有什么参数进来，为了向页面传数据，我需要声明 model，另外为了支持分页，我要加上配置组件，那么要实现这个查询，首先我需要把 service注入进来是吧？

得用它message service，另外我再把 house hold住进来，因为你要查什么，你要查的是当前用户的私信，当前用户不得通过它获取对吧？了好。

这里面首先我要查的是私信列表，会话列表，我首先需要对分页的组件做一个设置，我先设置一下分页的信息配置点site limit我设它为5，因为我的数据并不多，一页显示5条，这样还能看出分页的效果。

然后配置点site，设置一下访路径就是当前的路径，letter，list，然后 page点side，rows设置一下一共会有多少条数据，这个得查了。Message service点。Find conversation你要查询当前会话的数量，这里面我们需要传递uzid uzid我可以通过 house偷的获得，这样我在此之前先把user获取到，因为后面可能还会用到，提前先获取一下。

好复用好，这里我就可以写user点，get ID分页信息处理完以后，那么接下来我就可以查询绘画列表了，也就是一句话就搞定了，message service点算一个conversation，传入user，点get ID传入user ID还要传入分页，消息配置点get offset配置的get limit。最终我会得到这样一个集合的名字，我叫conversation please。好，这个有点长，最好换个行。好，得到数据以后你要注意这个数据还不够，因为什么？我们页面上不只要就是这个消息，绘画这个消息我们还要显示什么？

还要显示就这个内容就是数量，这还要显示每一次绘画未读的数量，还要显示绘画里边一共包含几条私信，就是他还要帮额外的一些内容，所以你需要把额外的内容也查到，然后做一个包装。

那就这样还是按照以前的方法来处理，声明一个集合，里边的封装的是map，我把多个数据都存到map里面好，那么这个集合我取名叫看了c损失，那么我需要遍历这个列表，然后去构造 Map，如果它不等于我就去遍历。

好，每次遍历我需要新建一个map来重构这个数据，首先我先把卖c之前存进去，我就卖掉点兔子这个名字我得取消恢复c型绘画和我们业务相匹配这样好。

理解好。记然后还要传什么？还要传的是每个会话里不是包含一个未读消息数量的，对吧？你要把未读消息数量查到。

好，我就map点put未读消息数量，按read count，然后调用 message service去查询 find，然后是later and read count，然后你要传入优质ID，另外还要传入会话ID，那个好办，从麦这里取 message点 get consisted，除了这个以外，我们还要在页面上显示会话里包含几条消息。这个也要查，我在之前补充在前面写。Map点put，let come私信，数量也是查。Message service，find letter count。那么需要传入会话ID，message点get consist ID。好，那么除了这个以外，其实还要显示什么呢？还要在每一个消息之前显示用户的头像。

会话有两个人 from有to显示谁一定是显示和我当前用户与之相对的那个人的，比如说当前用户是a我这里一定显示的是b对吧？你显示当前用户的头像就没有意思了，所以说显示的是与当前用户对话的一另外的一个人的头像，所以我们需要把这个目标找到，然后的话把它查出来。

首先我要判断这个目标它到底是谁，他或者是from中的一员或者是to中的一员，这样判定也很容易 to get ID目标的ID等于我们判断当前用户的ID它是否等于message点get from ID如果当前用户是这个消息的发起者，那目标就是接收人对吧？

所以说 to get ID此时就等于什么？就应该等于message点get to ID是吧？否则就是 Usid它等于to ID我的目标应该是from，ID有点绕，你稍微自己琢磨一下。

好，我得到了目标ID以后，目标这个对象就很容易查到了，查到以后我就要把它放到map里面，map点put，名字我取叫它get，这个得用user service来查，我还没有注入一下，注入以后回到刚才的位置，user service点儿饭的，白的把它盖的ID传入，我就得到了大概的对象也存进去了，最终我得到的 MAC 我需要把它放到集合里，别忘了。

好，那么经过循环以后，最终我们得到了集合中的数据，我需要把它放到model里，传给模板，给它取个名字就叫conversations得了。好了，这个列表数据就有了以后还少点东西，还少点什么看一下。

咱们刚才查的未读消息数量是整个页面上有一个未读消息数量是它，你得查一下，所以我们把它也查到，这得单查，所以再来就是查询未读消息数量，这回查询你就不要带 com system ID了，是查整个用户的所有的先生个变量叫light and red。

Count等于 message service点。Find。Let every account把 uzid传进去，第二个参数传入空不用了。那么得到的数据我们也要传给模板，好处显示名字就叫let I read counting。好，这就完了，最后我们就返回模板的路径就可以了。Return这个页面是叫什么？看一下。这个页面是叫它later，然后后面有详情页面叫later detail，所以这里我要写的是sat later。好了，那么 Control处理请求我就写完了，接下来我们就来去处理这个模板。

当然这个模板我们首先要在首页上加上链接链到那去，所以我先打开首页index html，然后找到head的部分，找到这个消息，这个地方这儿我们需要给它加一个链接，改一下这个链接，然后刚才我的链接的名字叫lighter类似的，好这个路径就写完了，写完以后接下来我就可以去处理 Leiter的html。

当然首先我需要先声明，这是一个timely模版，然后这些静态资源的引入我需要注意一下处理，最后的GS也需要处理。好处理完以后再往前看，这个页面比较长，这里边都是模拟的数据，这么多重复的数据，我们可以把其他的a删掉，留一个以此来循环就可以了，我顺便就把多余的ei给删了。因为碍事。好，先到这留一个可以了。

然后这次课我们开发的是查询，而不是发送私信，所以弹出框先不管，提示框先不管，这都是发送私信是要做的。然后前面的头部需要复用首页的头部对吧？所以我们需要写上提示为case, index还等，然后这个head我也可以折叠起来了。好，那么接下来我们就处理了正文的内容，咱们来看，首先我们处理这个地方，这个地方是哪？就首先你得能点能练就，其实就是练到当前这个消息页面对吧？能练过来。

其次未读数量要处理一下这儿，首先把链接给它改一下，有点长，我把它回个行，要不一瓶都看不下了，这样一下。

然后这个地方我改一下链接thhe和刚才首页上链接是一样的，Later list，然后这里边18里边的显示的数据，你需要去动态的处理，th冒号，text等于是未读消息数量是哪个变量它然后你要注意，如果说未读消息数是0，你就不用显示个0在这了，对吧？没有必要，所以如果是它等于0的时候就别显示，不等于00:00它才显示。

所以这里我再加一个条件，th冒号if等于它不等于0是否在显示这个地方就处理完了，剩下我就是处理这个列表了，处理li就这个li我们肯定是要做循环处理的，我们要循环的刚才的CTRL的发送过来的集合，所以这里我需要写thh每次循环的变量名我叫map，然后我们需要遍历的是谁呢？是它循环了，然后再循环内部我要显示的这些数据要改成动态的值。

首先这个地方是什么是未读数量我要把它先处理一下。好，我就写th冒号text等于map点，未读数量。I read count。当然这个也是跟刚才未读数量也是类似的，就是说如果说你未读数量是0，你也没有必要显示对吧？显示它就不好看。所以加个判断就当这个值不等于0的时候才显示，否则我就不显示这个数据了。

好，然后再往下，这个是一个图片是用户的头像，这里我们需要显示目标用户的头像，目标用户是谁？他get MAC点他get对吧？然后点Hide UIM再往后这儿，你要显示目标用户的名字，th冒号u text等于matte点。Get点。有点好。然后再往后要显示就是消息发布的时间，会话发布的时间在这里写th text时间的需要格式化，diss点format，然后是map点看了，谢谢。

点create，然后指定消息格式，好这是时间。然后再往下这个链接是链到详情，页面我们先不写，先不管它，然后就链接先不管，但是这个内容我们需要先处理一下，这个内容，绘画的内容。最后一条消息的内容，我这样写th冒号，you text等于Map。

点最好copy一下，因为这个单词有点长容易写错。Map、.composition，点内容、content。好，再往下再往下就这儿共几条绘画，这个几你得是一个变量对吧？不能写死，所以我在5前面加上个I吧，然后在a上声明它应该是几，这里应该是map点，lighter com是会话所包含的私信的数量对吧？好，那么这个地方Li里的数据就处理完了，处理完以后最后就说分页，分页就简单我们直接复用index，页面的分页就可以了。

所以在这里写th冒号，replace等于index。

冒号这个名字好了，到这儿我会话列表就写完了，写完以后咱们测一下，我把这个项目重新启动一下，启动以后我打开浏览器先访问一下首页，好，然后我点这个消息，但是这个消息上这个数量咱们还不先不处理，以后再说先点什么都没有，什么都没有，应该是我当前登录人有问题，就这个人他可能没有和别人发生过对话，我换一个人换一个人，我换成鄂AA AA它应该就是ID为111的用户，我看一下确认一下是不是对用户的会话是特别多的，最多的AAA的密码还是AA然后登录以后点消息。

好，你看这个数据基本上是数据大概是对的，你看但是头像明显头像不对，这个头像我估计是哪写错了，我们看一下检查一下头像在哪呢？这儿这里是这个符号写错了，不是at这个是dollar。

我直接引用一个变量路径，然后作为src的值，所以这里不是艾特好，这个地方改完以后重新编译一下，然后再打开这个页面刷新一下，好，这头像就ok了。

然后你仔细再看一看其他的数据，然后你看这是和BBB的会话和lhh的会话，AA的会话，yyy的会话，www的会话没有问题，然后它的内容比较简单了，也都各不一样，头像也各不一样，然后当前用户只和他有一个未读的消息也没有问题，然后后面每一个会话里边包含多少条数据，这里也是动态的，也没有问题。

然后分页我们看下一页，可以下一页没问题对吧？首页没问题。好，当然了这里面我就不去详细的去验证说8条数据对不对？然后的话一条未读消息对不对？你可以自己在开发完以后自己去查询一下表，自己做一个判定。好，目前我们就把私信列表的功能就搞定了。那么最后我们再去花点时间把私信详情搞定就可以了。好，那么私信详情所需要的这两个查询方法我们已经写完了，不管是数据访问层还是业务层我们都完成了。

所以我们实现这个功能直接从表现层开始写就可以。那么首先我在master control里再加一个方法来处理的，这次请求加个方法。

我先声明一下这个方法的访问路径，这个方法它的路径我声明为叫later detail，然后因为你是点中一个会话，然后的话查看会话包含的详细的往来消息，所以说你这次请求当中需要传入的一个会话的ID，我希望在路径当中就包含会话ID。

好这个参数我叫composition ID，然后因为这是一次查询，所以说查询请求方式我用get好，然后声明一下这个方法名我叫get lighter。

Detail首先在这个方法里我就要先获取路径中的参数，那么还是使用pass外包，我们得到的是一个字符串，然后我们在显示详情的时候也是显示多条消息，所以说也是要支持分页的，所以说我把配置加上。

另外为了向页面传数据加上model，好了，那方法的声明完了实现的话其实并不难，首先我先对分页的信息做一个设置，配置点set，limit我还设置了它也显示5条数据，然后配置点set，pass就是当前的路径，Later detail当然detail后面需要加上参数，然后再设置行数，这个行数需要查 message service点儿饭的来着抗体，然后把 com ccid传进去就可以了，那么有了分页的信息以后，我们就可以做分页查询了，那么我就调用message service点find典范的letters。

当然你需要把会话ID存进去，然后再把分页的相关的参数存进去，我们最终会得到一个集合，里边封装了一页的消息，那么这个名字我叫 later this，好，我们当前得到的就是一个信息列表，写注释得到以后，那么还没完，我们也需要对这个数据做一个补充，因为我们在页面上显示的时候，你像 user你要显示它的邮政名头像对吧？不能是ID。

那么这个message里面有两个uzid一个是from，一个是to，我们要显示要哪一个还是都要，咱们可以到页面上看一下，看进程页面就可以了，消息进去。那么这里其实谁是发信人，这里就显示谁的头像显示谁的名字，所以说不管是哪条记录都显示的是 from user是这样——条。

所以说我们需要补充的是 from you的相关的数据，这样我再声明一个集合里面的存放map给它取个名字叫lettuce，好，然后我对莱特利斯的做一个判断，如果它非空我就对它进行一个便利，然后每次循环，首先我要先实例化一个map，准备用来封装数据的，然后首先我先把私信先放进去，给它取的名字就叫light，和我们的业务保持一致，另外我再把 from ID转换成from user做一个查询就可以了。

它的名字我叫福旺user，然后我调用user service进行查询，find user for ID这个ID应该是从message里取 get from ID好，完成以后不要忘了把 map放到集合里，所以let us点艾特，好了，集合数据就有了。那么我们在循环以外需要把集合发送给模板，所以说我要model点艾特，然后 Let us写错了。

好，除此以外还要补充一个内容，补充什么？我们打开刚才这个信息列表看一下，在列表之前有这样的一个东西来自谁的私信，其实这是来来自谁谁的，是当前登录用户与之对话的目标的人的名字。

所以说我们还得查到和当前的登录用户对话的那个人是谁，然后把它显示到这里，我们称这个人为目标，这个目标其实和我们之前的方法取的方式差不多，就是说我们看一下当前用户他是from ID吗？

如果是那目标就是to ID否则就是from ID但是我们这已经遍历完了，我们在编辑之外需要这样一个数据，你可以其实随便从集合里取一条数据，比如第一条数据，然后以此来判断可以不取这集合中的数据，比如说我用康乐西山ID来判断也可以，因为它不就是两个ID拼在一起的，对吧？我们看一下斜线横线，哪一端的ID和当前的登录用户不一样，那就是目标对不对？这也可以我从这里取取的逻辑我简单封装一个私有方法，然后一会好用，后来一个方法最终返回的是直接返回的是一个user，然后我叫get later。

好get就私信的目标，我需要你给我传入的是康复新ID。好那么首先我要把康复CS就做一个拆分，它中间是用一个横线相连，我就对它做一个拆分，拆分以后我会得到两个字符串，其实就是一个两个ID字符串，当然了那一会我们查的时候，我需要用到的是整数，所以说这样我把数组里边的每一个字符串转成整数，第一个叫第零，那就等于 integer pass into ideas对吧？

以此类推，再来一个 ID11，那么这两个ID有了以后，我们和当前的登录用户做一个判断就好了。Host holder，然后点get you的点get ID。那么如果Id它等于ID0的话，就是写错了少了一个，我说怎么这么别扭呢？那么如果它等于ID0的话，那就说明目标是ID一，我就返回ide对应的用户 use the service点，find by idid好，否则我就返回另外一个数据对应的用户就可以了。

好了，这个方法的封装以后，回到刚才的代码里，我们进行了下一步操作，就是我要查询私信的目标，然后要把它发送给模板上的好处显示我就直接猫都有点艾特，然后这个变量名我叫条带目标，这个数据调那个方法直接获取就可以了。

就get lighter to get，然后把康c加ID传入就行了。

好，那么最终数据处理完以后，我们要返回模板，那个模板叫做site，然后叫let it kill，好，那么 CTRL我们就写完了，写完以后接下来我们就可以开发这个模板，首先我们先在会话列表里先做一个处理，因为你要加上链接才能链到详情页面对吧？你要把链接做好，链接是在哪？在列表里在这个是私信的内容，私信的正文，那么我们点正文就能链到详情页面去，目前的话它这个路径是不对的，我们去改一下。

好，我把它链到，但是我们这个链接里边还包含一个变量对不对？所以说你得拼给我写个双竖线，然后的话是light，然后是 detail，然后是一个变量，这个变量我们需要取到康ccid咱们看一下怎么取呢？其实也好取，你看这个数据 Map点conversation点什么对吧？Conversation其实是一个message，它里边就有conversation，ID是吧？

所以说你就可以这样 map点看看星星，然后看看谁先ID好了，这个链接处理好以后，就可以链到详情页面，下面我们就可以去编辑详情页面了，我打开这个页面我搜一下，lighter得跳，这个页面我们从来没有编辑过，所以说要从头开始。

首先我先把这个模板说明好，好，另外把静态资源引入路径改对了，忘了背了括号补充一下，好。然后这个header你需要用首页的header去替换，替换完就可以了。至于这个内容我们一会再改，我们先往下看，尾部不用处理，这些静态资源都需要改，这是需要改的。

好，下面我们就开始改正文了。从前往后过，首先在这个地方就是你要把这个名字改成目标的那个人的名字对吧？这个容易就是改个名字而已。Thu text等于目标是 to get点user name。

这就行了，然后弹出框这里不用显示什么消息，这是增加天发送私信的时候用的，我们暂时先不用，然后是私信列表，然后私信列表里边也是有多条数据，每一条这里是一个I然后静态页面上给我们模拟的一些数据，我们可以把多余的In的删掉，保留一个就可以了。

好，我保留了一个li，然后以此为模板进行遍历。

我在ls上写上th冒号，each等于我们便利的集合，它里面封装的是卖的，所以便利什么我就卖，这样比较直观，然后集合叫let us好，每次遍历里边数据有一些地方要改，首先用户头像要改，刚才我们也说了用户要显示的是 from就是消息的发送的人的相关内容，所以说我们要显示的是from相关的信息。

那么就是map点 from you'd刚才我们查到了是吧？然后 Ida URL行了，再往下看。然后这个是 from的用户名，也就是thu text map点from user点username。好，然后再看，这有个时间应该是这个消息发布的时间需要格式化，th text等于格式化一下，对此 format，然后时间的获取是 map点 later。Create time。格式指定一下。好，再往后看。然后这就是正文you text等于点 later点内容 content。

好了，那么这个消息的展示的内容这就设置好了，设置好以后最后还差个分页，而分页的逻辑我们可以复用首页的逻辑，这个就简单了，它等于index冒号，然后是这个名字，好了，到这儿我们详情页面也就处理完了，处理完以后行不行，咱们再试一下，我重新编译一下 CTRL s9服务重启的，然后我们打开浏览器看一下，当然目前我所处的是会话列表页面我刷一下，刷以后看这个链接对不对？

比如说你看放到这上去看，左下角的链接没有问题，我点进去看，这有8条规划，8条消息，然后第一页123455条消息，来自BBB的私信，当前用户不是AAA对吧？目标的私信。然后左侧显示的是发消息的人的头像和他的名字没有问题，然后下一页没问题。当然这里还有返回是吧？点一下返回不对，只有个返回它这个链接。不对，这是我们需要改一下才可以的。

好，下面我们来改一下返回的链接，好，我再回到这个工具，然后找到返回的位置，他之前写的Location点h—f等于一个静态页面，静态页面没有问题，但是我们现在不是静态页面了就不对了。

这里我们需要改成我们当前的路径，就是莱特，然后 detail当然其实你也可以改成什么？

History点back，但是 history点back它是靠历史回去的，我希望他不走历史记录，而是每次他真的去重新访问一下会话列表，然后这样的话好可以刷新一下，因为我后面还会把已读的状态设置好，比如说你私信里有未读的，你点进来就变成已读了，我希望他能刷一下，如果你直接后退的话可能就刷不过来。

这个地方因为直接写不太好写，这里头直接写这里不太好写，因为我们现在路径变得复杂了，而且我们路径当中还要包含这个项目名，而项目里面我们把它做成了一个常量，你在这里不方便写那个东西，我这样我就直接写一个方法叫back，然后我在下面写一段js来实现它就可以了。

当然 GS是很简单的在这儿，好我来实现后退的方法，其实就是做一个很简单的跳转，就一句话location点href等于然后项目的访问路径，context pass加上上一线，双引号，然后是let list我们回到会话列表页面。

好，写完以后重新编译，再看我在这刷新一下，然后点一下就回去了。好了，到这儿我们就把私信详情也开发完了，整个私信列表私信详情，这两个查询我们就完成了。好，这次课我们就演示到这里，咱们下次课再见。

## 3.8 发送列表

25-发送列表.mp4

这节课我们来实现发送私信的功能，那么发送私信的功能我们需要用异步请求的方式来实现。除此以外，我们在完善一下私信详情页面，当我们进入到详情页面以后，如果详情页面上包含了未读的消息，但是这个时候你进来了，你已经看到了，我们需要自动的把它设置为已读，要再补充一下这么两个功能。

当然我们对PPT说这两个功能并不生动，我给大家看一下静态页面，咱们通过静态页面来直观的了解一下这个工程到底怎么回事。

首先发送私信你是在消息页面，然后点发私信，弹出个框，然后你填写的发给谁，填写用户名，然后填写内容点发送，然后用异步请求的方式发给服务器，然后服务器返回了结果以后，判断是成功还是失败，我们最终给一个提示就好了。

除此以外，我们在进入到详情页面的时候，这个地方也是可以给某人私信，就给当前你对话的目标私信你点的时候，这地方要自动带上当前对话的目标，那个人的名字，然后的话内容你去填就可以了，然后点发送完成。

所以这个地方和刚才那个地方，其实这两个地方的逻辑是一样的，完全是一样的，我们可以分成一套代码，只不过我们在这个页面上点这个按钮的时候，它只是默认带了一个用户名而已，这个好处理。

再有当我们进入到私信详情页面的时候，那么如果这里包含了已读的未读的消息，你要把它改为已读。

好，这样我们先来处理添加发送私信的功能，首先我们还是从数据访问层开始写起，我们需要在数据访问层写一个增加的方法，对吧？我就打开 Message map，然后补充一个方法，增加那么增加返回的是整数，然后的方法名insert message。



好，然后参数就是mess，把实体写个注释，这是新增一个消息，除此以外，一会儿我们还要做就是把未读的消息设置为已读的这么——一种状态的这么一个逻辑，这个显然就是要更改这个消息的状态，把它由零改为一，然后我们在私信页面一下能看到多条消息，所以你可能一下需要改多个私信，所以说这里我定义一个修改状态的方法，不对的 status，那么这个参数这个修改的条件应该是多个，ID不是一个，所以这里我给一个集合，inter，然后爱丽丝另外你要把这个状态传进来，你要把它改成什么状态？

传进来。

好，这个方法是修改消息的状态，当然你不只是可以利用它设置已读，你也可以利用它设置，删除它都可以。那么定义了这两个方法以后，我们就来实现它，我打开配置文件搜一下 message map。好，那么要实现增加的逻辑，我把 insert字段也把它提出来，在这里写一下，把ID去掉就可以了。

然后我就来实现这个方法，insert message在后面补充了，ID等于它，然后这个参数是message，那么增加的时候 ID是自增长，我声明 ID字段对应的属性是谁这样生成，完 ID以后会回填对象中的属性，这样我们通过这个对象就能得到刚刚生成的 ID。

好，下面我就写 insert语句了，很简单了。

我需要引用刚才的 circle。好了，然后 Virus给它赋值分别是from ID to ID，他们申请ID，然后是 content，接下来是状态。最后是创建时间，好了增加的逻辑就完成了。

好，接下来我们再来处理修改的逻辑，ID我copy一下，省得写错了。好update，然后它的参数多个我就不声明了，返回的一定是整数，不用声明，然后不对it sat。standards等于standard的参数，然后 where你注意它传入的ID是多个，所以你就不能ID等于了，你肯定得ID in。

In，然后多个我们知道in的语法是in，括号里边有多个ID中间逗号隔开，这里怎么写？因为参数给我的是一个集合，买卖特色它允许我们通过一个叫fall意识的标签去遍历集合，然后拼成一个括号里边带有多条件，中间逗号隔开的形式怎么拼？我一写你你就能看得懂很简单的 For each。然后 collection你要写什么？就是集合你要遍历谁？

遍历的是爱丽丝参数集合，那么每次遍历我们需要声明一个就是变量，用来指代每次我们获取的值需要写一个属性叫 item等于ID，每次我们得到一个ID，我们就是要把 ID拼到括号里去，拼的时候我们以左边的括号开头，右边的括号结尾，中间以逗号隔开，这个也需要声明的，那可以也很好写，open表示说以主括号开头，然后 separate，逗号表示中间以逗号隔开，close右括号表示以右括号结尾，这就可以了。

好。然后 for each中间我这样写并号ID，那意思是我最终在这个循环当中我要拼的是什么？我拼的是每次得到的一个循环变量，那就是把 Id拼到这俩括号之间，以逗号隔开，这样一个形式非常简单。

好了，这样的话咱们数据访问层就写完了，写完以后我们就可以接下来写业务层也不难，能够打开之前我们创建好的业务组件，message service，我需要先增加一个方法，用来就是添加一条消息。

爱的。

message，参数是message，这个没什么好说的，唯一需要注意的是你在添加之前你要对消息的内容进行过滤，避免它有敏感词有标记就影响别人阅读是吧？

我需要把过滤敏感词的工具注入进来。

对，好了。

我就对内容进行过滤了，很简单了，message点，site，content，然后的话先过滤标签，html点，然后是麦穗点，get content。好，然后再过滤敏感词，message点get content。那么都过滤完以后，我就直接调用map去做插入就可以了。好这就完成了。除此以外我们在实现再补充一个业务，读这个消息，把消息变成已读。

好，我写一个方法来解决这项业务这个方法。我叫read message读取消息，因为读取消息最终是改变状态，所以我返回的是改变的行数。我支持一次多多条消息，所以你可以传入一个集合，ids实际上也非常简单，就一句话，message my project update，然后把ideas传入，读是把它状态改为已读，已读状态就是一就可以了。

好了，那么业务层面也写完了，最后就是处理数据访问层，接下来我就打开这个麦子就肯抽了，我在后面再追加一个处理请求的方法，这个方法先声明它的路径是let send可以了，然后请求的方式这个得是post，因为这是要提交数据进来。

另外刚才我们也说了这个请求是异步的，所以说你要加上注解 response body，好，我把这个方法声明一下，send light。

然后那么页面上那个表单需要传两个数据进来，一个是你要发私信给谁，而且传提交的不是接受人的ID，他是接受人用户名，那就to, name。另外传的是私信的内容，content，这两个值传进来就可以了。

好，那么我们要发私信，首先我们要构造私信的数据，而这个数据这两个值有了，其他值还没有，但这个值不行，这个值是接收者的用户名，我们需要的是ID，我需要通过用户名查询用户，然后再得到ID，所以我需要调user service的find by user name这样的方法。

User service点find，但目前还没有，这个之前还没写，我们就补充一下很简单了，我打开user service写错了，user service我在后面补充一个方法是一个查询方法，他查到的是一个user，是吧？

然后方法名我叫find user by name，根据用户名来查，显然你传入的是用户名，实现非常简单，Map里我们早已经把这个方法已经加上来了，user map点select by name。

好。

有了这个方法以后我们可以继续了。回到刚才的这个地方，那就是user service点find user by name。然后你把to name传进去，我就得到了我对话的目标了条，get得到以后判断一下，万一用户不存在对吧？判断一下，如果他盖的是空的，我们就直接给他返回一个提示，就不往下进行了，返回一个Json数据，肯定内推YouTube点get杰森 string。给他返回一个—代表错误，然后给他一个提示，说目标用户不存在。

好，那么如果目标用户存在，我们可以继续可以利用当前的数据去构造，要插入的对象就是message。

好，那么我们挨个的构造它的数据 Message点site from ID那肯定是当前登录的用户发的消息，对吧？所以我们要从当前的用户中去取，然后 message点site to ID给谁 to get ID是吧？好，然后还有个字段是会话ID，会话ID之前我讲过它是这两个对话的人，ID拼起来中间有横线拼起来，但是要求就是ID小的在前，大的在后，所以我们需要判断一下 message点get from ID它是否小于message点get to ID？

那么如果是的话就把from拼在前面，我就这样写。

Message点set看了ccid然后是message点get from ID加上别忘了下划线，加上message点get to ID，好，这样拼一下，否则就反过来拼就可以了，我把这句话 copy一下，然后把这两者顺序的颠倒一下，或者直接就改。To ID from ID。

好，可以了。再往后那就是message点，site，内容，content，内容，然后 Message、点、site，status你不设置也行，因为默认值0默认就是0，然后可以set，create time当前时间好了。那么这些都设置好以后，我们就可以做插入了。你就调message，service点ID就可以了。好，最终如果没有报错，我们就给页面返回一个状态0就可以了。

0就行了。

当然如果报错，我们将来统一处理异常，统一解决这个问题。好这个请求也处理完了，处理完以后，那么接下来我们就可以写这个页面的逻辑了，那么我们打开这个页面这个页面，我们看一下这个按钮怎么回事，搜一下这个页面。

叫light点html好，那么在当前的页面上我找到发送私信的按钮，从这儿你看不出来什么，因为什么点这个按钮，它所做的事情是由GS来实现的，所以说我们得看看当前页面引入了什么GS，GS里怎么写的，其实主要看GS。好，往后看。Global GS显然不是应该是light点GS好，我们就看一下light点GS里有啥？你看light点GS里的它对这是发送按钮，单击事件做了定义，单击时调这个方法，这个方法里先把对话框隐藏掉，然后把提示框显示出来，两秒以后再把提示框关了，这个逻辑没有问题，就是你点这个我说错了，你点刚才的发私信按钮，其实它是弹出这个框，然后你在框里填完数据以后点发送时，然后才会调这个方法，这是对发送按钮做的一个事件的定义。

那么发送的时候它把弹出框先关掉，然后再把提示框显示，然后再自动关闭，是这样的，所以整个的逻辑我们应该在哪写？应该是在这个中间对吧？先把框关掉，然后我去向服务端发送数据，然后服务端给我反馈结果以后，我再去做提示，在这里面写。

好，这个写就很简单了，我们从页面上获取接受人以及内容，先把这两数据获取到 Where to, name 等于。

我看一下那个人怎么取，看这个框发给他，它的ID是它可以通过ID获取的ID选择器，然后文本框的值点Val同理在获取内容，看内容，框的ID是他也是ID选择器，然后它的值点Val好，然后我们就异步的发一个post的请求，很简单三个参数，首先的话是访问路径康txt。pass框态的pass，然后后面的路径是light，然后是散的，接下来我们需要声明我们要传的数据的参数，一个是to name，它的值就是to name，再一个是content，它的值就叫content，最后是处理服务端返回的结果。

接受一个数据data，这个data是一个普通的字符串，当然它满足杰森的格式，我们需要把它转化为GS对象，就用解卡瑞的purse杰森。

来解决。

把它传入进去，转为监视对象就可以了。

好，那么有了这个对象以后，我们需要对返回的结果进行判断，如果data点code等于0，比如说成功了对吧？

发送了，我们给个提示，提示到提示框里，这个是提示框其实就到这里了，提示框的ID是hint body，我就通过ID获取获取提示框里的内容，普通的元素不是表单元素，我们取内容点text，然后给个提示说发送成功，否则发送失败了，发送失败我们也是给一个提示，还是这句话，但这里面你要把具体失败的原因都写上，带他点卖c好，那么无论是成功了还是失败了，最终的话我都希望把页面刷新一下，刷新页面的逻辑就这了。

先把提示框显示出来，然后过两秒自动把它隐藏掉，隐藏掉以后当前页面刷新一下location点，对漏的承载当前页面，好这就行了，然后还要注意一下注意什么？我们在私信详情页面上点给他私信的时候，这个地方会默认带上用户名，但现在我们没有处理过对吧？

我得处理一下，所以我们再打开私信详情页面，找到那个地方给它处理一下。弹出框里是发私信发给某某，目前的话这是写死的，我们需要把它改成一个变量对吧？改成一个变量。好，我就写上th冒号，我们应该取谁？应该取我们对话的目标，对吧？我要把私信的发给对话的目标，对话的目标怎么取？你看之前我们不是取过吗？他get点you the name对吧？就这么取一模一样的。这个地方就th点value，然后等于也是Target点you的name。

好了，到这儿我们发送c相关的逻辑就已经写完了，咱们测一下，我把这个服务启动一下。

好，启动完以后咱们试一下，我打开浏览器，然后访问一下，首页这已经登录了这已经登录了，登录的账号是AAA我就看一下AAA的点这个消息，然后我在这发私信，比如说我要发给CC，然后的话发的内容是你好CC好，然后点发送提示发送成功了，然后这样我和CC之间就多了一条会话，但是这个是由CC来读的，我是默认就是已读的，因为是我发的，对于CC来说他需要读，读完以后才能改这个状态，然后你进去看这个也没问题。

然后比如说我再发一个，比如我在这在这我再给他私信看这行不行，给他私信。Cc有了。

然后说。

你在吗？

发送。

然后它刷新了当前页面，所以又多了一条数据，你在吗？没问题，然后是倒叙。可以的。好，Cc我先不看了，因为CC现在看的话一定是未读的状态，我们需要再把刚才所说的另外的一个环节补充一下。当CC看这个数据的时候，当他看到的时候把它改为已读这个站点处理，其实很好处理。

我再回到开发工具里，我们只需要在CTRL那里，在这儿你看 get later detail不就是处理访问私信详情页面的请求吗？对吧？在这次请求里你就给页面返回了几条数据，那么如果这个数据中包含未读的话，此时你就可以把它改为已读了，因为用户打开这个页面，他看到这个结果以后他就是读了，对吧？他就是了。

好，所以说我们需要做的事情是再补充一步，你要把私信列表里头的未读的消息提取出来，然后把它自动的设置为已读就行了。

好，如何从集合里提取出未读的消息，我们可以再单独补充一个方法，在一个方法里面把它实现了。好，我加一个方法 Private，当然我得到的未读的消息我要的是一个集合多个，然后要的是一个 ID整数就可以了。那方法名我叫get light。Ideas。那么需要你传入私信列表，later list，然后我从中去筛选。好，那么这样我首先实例化一下集合，准备往里存数据，list爱丽丝等于new和released，最后我就return的是爱丽丝。

好，然后我需要看集合里有没有我想要的数据，我需要对它做判断，如果它不为空，我对它进行一个遍历。

好，每次遍历我看一看它里边没有我想要的数据，这里你注意，你不只是要看，不只是要看他的状态，你也要看一下我当前的用户他是不是接收者，因为只有当前的用户是接收者的身份的时候，他才有可能是他才是一个独的身份对吧？

他才是做一个独的操作，他才可能会把数据变为已读。如果你是发送者对吧？你本身就是读到你发的消息了。

好，我判断一下，如果 House的后者叫get user叫get ID，它等于 message点get to ID，我是一个接收者的身份，另外最好还要看一下当前的 message，我是接受者的身份，而且这个消息它确实处于未读的状态 and message点get standards等于0，这个时候我就需要对它做处理，我就需要把它加到集合里面去。

Id是点艾特，message点get ID。

好这样就行了，反正经过方法的处理以后，我们就得到了集合当中未读的消息的ID，我们在这次请求当中把它自动的设置为已读就可以了，我们直接调那个方法就行了，在调之前先把 ID得到 IDs等于get letter IDs然后你需要把 let list传进来，把它传进来就可以了，不用聚合的集合这个数，这个没必要，我们只要用这个就可以了。

好得到ID以后去更改它的数据，当然更改之前判断一下，ID点is安排，如果说它非空，你要为空就没有必要去更改了，如果是非空的，我们就卖c就service，点read message，然后把爱丽丝传入就行了。好，那么处理完以后重新编译，已经重启了，我打开浏览器我先刷一下，我换个人刚才不是发信给CC了吗？Cc有两条未读的消息对吧？我登录一下CC的账号。

好，然后看一下有两条未读消息在一次会话里，两条，然后我点一下，好，这就读了，这就读了，读了以后返回。

再看未读的状态就没有了，就说明刚才我们设置已经ok了。

然后你看我清空一下控制台，我再点一下会话，然后回到控制台，你可以看一下打印的circle，你看还有没有不对的，你仔细一看就只有查询，没有update说明，这次已经是已读状态了，所以说这句话没有被执行，所以我们这个逻辑还是ok的。好了，这次课我们就把发送私信开发完了，我们这次课就演示到这里，咱们下次课再见。

## 3.9 统一异常处理

26-统一异常处理.mp4

这次课咱们来学习如何统一处理异常，那么要想统一处理异常，我们先来回顾一下我们服务端三层架构，咱们之前讲过服务端分为三层，表现层，业务层还有数据层，然后浏览器发送的请求一律发给表现层，然后表现层调用业务层调用数据层。

那么你可以想象一下，比如说数据层这儿出异常了会怎么办？对吧？

他会抛出去抛给谁，抛给他的调用者业务层，那业务层把这异常再往上抛，会抛给表现层，所以整个这三层架构其实表现层是冲在最前方的，它是一个顶层的层次，所以无论是数据层还是业务层，无论是哪个层次的异常，最终都会有汇集到这个表现层到这里来。

因此只要我们抓住表现层，我们对表现层统一的捕获异常，处理异常，就能整个的处理系统中所有的异常，所以我们统一处理异常主要是针对表现层的，那么我们如何对表现层处理异常呢？其实spring boot也好，还是spring也好，它都给我们提供了一些现成的方案。

你比如说 spring boot他给我们提供的方案就很直接，你只要在我们项目的某一个特定的路径下，加上对应特定状态错误状态的页面，那么他在发现这种错误的时候，自动的就跳转到那个页面就很方便。

当然了，其实我们在浏览器我们的服务器在给浏览器响应的时候，错误的情况是有很多的，但我们项目中最常见的就是一个就是404找不到资源，一个是500服务端报错，这样我先不去说 spring的方案，我先把 spring boot简洁的方案先给大家演示一下，我们先看一下。

好，那么我们这个项目中其实已经有了自带了出页面它在哪儿在这儿？艾瑞里这两个页面就是错误页面，一个是404找不到资源，一个是500报错，那么我们需要把它放到特定的位置下，那么它才会自动的生效，放哪去？就是把 era、目录以及这样的页面的名字放到templates目录下就可以了，而且我就直接拖拽就把他拽到这个位置来，Ok拽过来就这样就行了，你注意这个错误页面所存放的路径一定要叫air，然后这个文件的名字一定是错误状态，作为文件的名字404 500这样的就这样就可以了。

然后如果说我们现在程序中有问题，它能够起作用。这样404好说我们随便瞎写个路径，就404 500的话我造个错，这样我在 message control这里访问，会画列表的时候，我人为造个错，而且inter点y的二ABC，你试图把一个字符串ABC字符串转成整数，肯定会报错的。

好，我们看一下目前这两个能不能成功，我把这个项目启动一下，启动好以后，那么我打开浏览器访问一下，先访问首页，没有问题，然后我瞎写个路径，比如首页加个ABC，没有这个路径，咱们项目中对吧？

没法去响应你回车，其实它就跳到了促页面，但是为什么没有显示对，是因为咱们这个错误页面，我刚才没有对它做模板的配置，这个得配一下。

忘记了配一下好。

首先我先声明 time leave，好，然后静态资源要配一下，不然样式不对，然后 header复用。好再看这个错误页面非常简单，中间就有一个内容区域，里边就摆了一张图片而已，这个图片路径你要把它处理一下，其他的什么都没有，给一个错误提示而已，好，这个地方也要处理好了，404页面处理完了500也是类似的，我把它也快速的处理一下，这句话copy一下。

好，然后手写黑的，我copy然后也是一个图片，处理一下尾部没有任何处理的东西，这要处理一下好，处理完以后我重新编译一下，编译好之后打开浏览器，我再刷一下这个路径，刷新，你看因为这个资源不存在，所以它自动的就跳到了就404出一面，然后给了这样一个提示，避免说报一个一个非常乱的让人无法识别的错误消息，这样的话比较好看一点。

好，这样我再回到首页，我再去访问这个消息，我一方面消息肯定会报错对吧？看看会怎么样点。这个消息本来是应该报错报一大堆错误提示英文的，但现在它跳转到了500点html对吧？这样体验就好多了，即便是错误，那么也是要给用户一个清晰直观的友好的提示，而不是说一大堆像乱码一样的东西。

好，这是它 Spring boot它自动的给我们做的一个处理，但是这个处理还并不是100%符合我们的预期，你像404这个倒ok了，你找不到资源就是给你一个提示，也没有什么更好的办法。

500就不一样，500就意味着我服务端报错了，我报错的时候我是不是得最好记个日志啥的，好以后好分析对吧？

所以你直接就跳转页面，这只是表面上的一个处理，我们内在需要记录日志还没有处理，那么要想我们做记录日志这样的行为怎么办？那么spring它有统一处理的方式，那么spring它给我们提供了这样一个注解叫controller，but web控制器的通知，注解用来干嘛？它用来修饰，主要是用来修饰controller，它是controller全局配置类，说白了注解是用来对controller来做全局的配置的，Ctrl了我们全局统一的配置以后，那么无论是哪个CTRL了哪个方法报错，我们都能统一捕捉到，统一处理很方便。

好，那么我们在这样的配置类里需要可以对CTRL做如下三种全局配置，一个是可以处理这个异常，一个是可以绑定数据，一个是可以绑定参数，那么如果你想处理异常，需要用到注解，写一个方法，在方法上加这个主角。

如果你想绑定数据写一个方法加这个主角，如果你想绑定参数写个方法加这个主角，所以说你还要结合着另外三个主角去实现了这三个功能。

这个我一会演示，我先说这俩，简单说一下，什么叫绑定数据呢？比如说我肯抽了有很多请求，要请求当中要用到同一个数据，给这个模板，我需要往帽子里装同一个数据，那么我可以利用注解加一个方法，然后给这个猫的统一绑定这样的参数，所以说它的作用是给model绑定统一的参数给所有的CTRL用，那这个data band的作用是这样的，就是说咱们页面向服务器传参，那么会被自动的做转换，会被自动转换，你看我们传的配置里边的参数，它自动转为配置对象的属性，他凭什么做到这一点的？

其实是spring mvc它底层有很多内置了很多参数转换器，那么你传的参数它去判断这个参数应该转成什么，它自动做了判断，自动调了参数转换器，万一说我们程序当中默认的参数转换器不够用，你有一个全新的类型，特殊的类型需要特殊的处理，那你可以制定一个转换器，然后用data banner把它注册上，是这样的，当然那么这两个注解目前我们没有需求要用，所以我课上也就不掩饰了。

然后如果日后你工作时需要用到的话，你去查查手册，它们的使用也很简单，而且他们俩的使用其实和exception handler大同小异。

好，我接下来重点是演示就是说我们利用它统一处理所有control的可能发生异常。好，我就打开我的开发工具，首先这样，首先我在home control的加上一个请求，为什么？因为你想我服务器我controller发生了异常以后，我统一处理，我记了日志，处理完以后我干嘛去，我得去到500那个页面，这个时候因为我们人为处理的，我们需要手动的把它重定向过去，所以我需要提前把500那个页面访问请求的访问给它配一下，增加一下请求的处理，我在 Home control里加一下很简单了，我就写上一个request，买品声明一下，它的路径防路径，因为是错误的页面，所以我叫艾尔。

然后 master等于 request，第二get获取错误页面。这里边其实没啥，就是get air配置就完了，然后你直接返回页面的模板的路径就可以了，它就是然后500就行了。那么除了在我们一会要写的配置类里要用到这个东西以后以外，将来其他的地方可能也会用到这个请求，所以我们提前写好是为了以后的复用。好，接下来我就要利用刚才所说的这个东西，Ctrl的y声明一个CTRL的全局配置类，然后所有control的异常做统一的处理。

那么我需要新建一个通知类，或者你理解为CTRL的配置类都可以。那么我这样我在CTRL的包下新建一个包就叫advice通知，然后我在包下新建一个类叫exception，but west那么这个类你需要利用刚才所说的注解加以声明，CTRL or the west。

好，这样的话 spring容器会组件会扫描所有的包范围太大了，我们通常需要进行一个限制，让它范围小一点，可以这样写，@RequestMapping就是注解等于可以等于多个注解，但你写一个也可以，等于其实我就一个 controller。点儿好像写错了。对他 @RequestMapping了class。

这样写的意思是组件注解他只去扫描带有CTRL的注解的那些包，说白了就是CTRL的组件了那些组件了对吧？所有的带有CTRL的注解的组件都能被扫描到，都能统一处理。

好，然后我们在这个类当中需要加一个方法，然后要处理所有的错误的情况，这个方法这样写方法之前需要加一个加一个注解，加注解，比方说这个方法是处理所有异常的方法，那就写上exception，handler，然后你括号里要写啥，你要处理哪些异常，可以写大括号，里边写多个数据，你可以处理多个异常。

我这样写 exception、点、class，exception是所有异常的父类，比如说所有异常我都用它来处理就完了，这个方法必须是公有的，然后没有反馈值，方法名随便取，方法名无所谓，我叫handle，处理异常，这个方法上其实可以带很多参数，你可以去 spring手册上去查，太多了，咱们也没有必要挨个去看也记不住。但是通常比较常用的就三个，一般这三差不多就够了。

一个就是exception就是CTRL当中发生的异常，当发生异常时它会把异常传过来，我们可以处理这个异常。然后处理完以后，在处理过程中我们可能会用到request，response去处理请求和响应。所以我再把 request声明一下，再把 response声明一下，有了这三个参数基本上就能解决绝大部分的问题，当然我这里要干什么？我这里不是要记日志对吧？我还得把日志逐渐的实例化一下，突然位置期待 final，然后 get log当前类点class。

好，那么当这个方法被调用的时候，就说明肯定control发生了异常了，我们没有什么好说的，就把这个异常记到日志里就可以了，我就log点air，然后 e点get message。然后这里面最好前面加一个前缀，说明一下这是怎么回事。说服务器发生异常，后面是异常的一个概括，这只是一个概括。

我希望把异常非常详细的站的信息都记录下来，这个也好办，我们需要便利站的信息。一点get stack trace。这是一个数组，每次我们得到的是一个stack trace element。Stack。Trace element。好，遍历这个数组每次得到这样一个对象，每一个element就是记录了一条异常的信息，而且它是有序的要把它记录下来，就是element点to stream就行了。

好，那么记完日志以后，我们需要给浏览器一个响应，我们要重定向到刚才的错误页面，但是你注意这个地方还要再额外再处理一下，因为什么页面访问浏览器访问这个服务器，它可能是普通的请求，希望你返回网页，你重新降到500没有问题，那么浏览器访问服务器也有可能是异步的请求，他希望你返回一个json对吧？

这个时候你重新上到这就没有意义了，他要的是json，不是html对吧？所以你要区分来处理。

因此我们这就要判断一下我们这个请求是普通请求还是异步请求，这个怎么判断？这里我介绍一下大家把作为一个固定的就是技巧或者经验记录下来就可以了，你可以通过request来判断，因为我们判断请求的方式 Request，第二hide，我们从请求的消息头里去尝试获取一个值，get header k是这样的，是x，这个一定不能写错了。

Request ed requested with请求的方式是以什么方式来请求？

返回的是一个死病，这个死病我叫我按照 k来取名 x request with，那么如果说返回值它等于这样一个值，xml HTTP request一定不能写错， xmlhttp request，如果是这样一个值就表示说这是一个异步请求，因为你看这个请求它是以 xml的形式来访问的，就是他希望你返回的是xml，我们知道只有Eva请求他希望你返回xml，当然你可以返回json其他的数据对吧？

否则普通请求期待的是你返回HTTP，不是html好，所以我就判断一下，如果说它等于变量，就说明这是一个异步，请求这个时候我就要响应一个这么差就response点， site, content, type, 这里有写 application 嘎？

其实你可以写杰森，那么如果你写的是json，那么我们向浏览器返回一个字符串，浏览器会自动把它转成json对象，也可以这样写 play，比如说我们向浏览器返回的是一个普通的字符串，但是这个字符串可以是阶层格式，那么浏览器得到以后需要人为的将这个字符串转换为 JS对象就是用到了点plus json那个方法，这样我们所有的应用当中，我们所有的请求当中都返回一个普通的字符串，但是这个格式我们确认是json就可以了，我们这种方式，然后我们手动转，然后写上又site，等于utf-8声明，字数集是有贴吧好支持中文。

好，然后我要获取输出流相关输出一个字符串，它就print writer， writer等于response点get right。好，当然了这句话我们需要处理异常，这个异常你可以好，直接抛出去，然后我就向外输出内容， right？点right？ 输出一个阶层字符串，这个我们可以调用community uq那工具得到这个字符串，因为报错了，返回的扣的是一，再给一个提示说服务器异常，好就行了。

好，这是一种情况，否则普通请求我就要重定向，到抽页面我就为sparks点send redirect，然后我通过request点get context pass，获取项目的访问路径，然后加上这一级路径这个就可以了。

好完成以后咱们就可以做测试了，有人说这就可以做测试了吗？是的。我们用组件的好处是什么呢？我们不需要对任何一个controller去处理，不用在任何一个controller controller上加代码，就能够解决他们的问题，这就是spring的强大之处。

好了，下面我们测一下，但首先我这个程序中得有这个错误，刚才 Mac这里有一个错误，这是普通的请求，异步请求当中我也造一个错，这样我们好看一下它的一个效果，异步请求不是发私信是异步的对吧？

在这里我也造一个错，还是那么写 y62ABC，好，然后我重新把这个程序编译一下，它重启了，重启以后我打开浏览器，然后访问首页没问题，你访问比如说个人主页我们还没实现，所以他报错了，没有这个路径 html，比如说访问账号设置，这也没问题对吧？

就是访问正常的，没有报错的都没问题，下面我报个错，我点这个消息点消息，报错，然后他有没有记日志，你可以看一下控制台，你看控制台级别日志有了，然后对应的配置文件里日志文件里也有，我就不细看了。

好，下面我再测试一下发私信，不过消息这个页面我打不开了，这样我把这个消息页面那个错误去掉。好能打开它去掉，然后再重新编译。

好，回到这个页面上点首页，点消息这回可以了，然后这里我要发私信了，比如说我要发给AA内容随便写，然后点发送服务器上报了个错，那服务器有没有记日志也有， Lg的日志没有问题。好，那么经过测试发现这种方式非常的简单好用，这就是spring它的强大之处。当然你最终别忘了把这个错误去掉，恢复成正常的状态。好了，那么这次课我们统一处理异常就搞定了，咱们下次课再见。

## 3.10 统一记录日志

27-统一记录日志.mp4

这次课我们来实现统一记录日志这样一个功能。那么有的同学可能会想，我能不能利用上次课我们所学的控制器的通知来统一处理，你注意控制器通知它是在控制器发生异常的时候，我们可以统一的处理。而我这里所说的统一记录日志并不是说一定是发生异常，平时不发生异常我也要记录日志，控制器通知可能就不太管用了。

再有同学可能会想我能不能利用拦截器对吧？



你可以拦截所有的请求，但是拦截器也是它是针对控制器的一个处理，那么我们记日志可能未必一定是要针对控制器，记日志也可能是针对业务组件，甚至可能是针对数据访问层，而我们这次课的需求恰恰是针对业务组件，你看咱们现在不是已经开发了几个模块，包括帖子模块，评论模块、消息模块等等，而将来我们这个系统越来越完善，这个模块会越来越多，那么就会有越来越多的service。

比如说我有这样一个需求，我想对所有的service所有的方法即日志好，那么既然通知或者是拦截器解决不了这个问题，我们用传统的方式也可以解决，可能有的同学可能也很容易会想到，那也很好办。我把记录日志的代码分成了一个组件里，然后的话我在不同的service的方法里去调就可以了。

比如说你要在方法一开始记日志，你就在写的业务方法时就先记日志，然后处理业务，当然这样肯定是可以解决问题的。

不过这种解决问题的方式比较原始，它有非常大的弊端，它的弊端在哪里？我们这个方法是业务组建的方法，它的主要的功能是用来处理业务的，而在处理业务的同时，我业务代码业务需求的前面或某一个地方，我却需要去加入记录日志这样的需求，而记录日志其实它不是业务需求，它属于什么需求，它属于系统需求。

我们系统当中很多功能都具备的同一类需求，我们把它归纳为系统需求，所以我们在这个业务方法里偶合了就系统需求系统需求、业务需求偶合在一起了，这样是很有坏处的，你可以这样想，比如说将来有一天我的系统需求要发生变化，比如说我不想在前面记日志，我想在后面接着治，你想我们这个程序中会有多少个业务组件，有多少个业务方法，你要改多少遍对吧？

或者说我不要在前面或后面记日志，我要在这个抛异常的时候记日志，或者是我要在某些方法中记日志，某些不记日志，一旦这个业务发生变化，那么你的改动是非常大的。好，这就是问题。

那么有没有一种办法能够很好地解决这个问题？有。而解决问题的最好的办法就是我们将这个系统需求，像记录日这样的系统需求拆分出去单独去实现，而不是硬编码到业务方法里什么样的方式能做到这一点，咱们看新的技术叫aop，但是Aop我早在介绍spring的时候就提过，那么我们先来了解一下它的概念，aop它是这样一组单词的缩写叫aspect oriented programming，那么翻译过来就是面向方面编程，那么也有人把它翻译成面向切面编程都无所谓了，不过我们从表面上其实很难理解这个方面切面是什么意思，这种辩证方式代表什么意思？

确实就这项技术它是一个比较抽象的概念，我们学aop其实很多的篇幅主要就是理解这个概念，理解aop的术语，理解它能解决什么问题，而至于说AP的使用，那个代码确实非常简单的，所以我们这次课我们需要花一点时间把aop的概念给大家说清楚。

然后接着说aop它其实就是一种编程思想，那么这种思想是面向方面的编程思想，它是对面向对象op思想的一个补充，那么有了这种补充以后，我们在开发的时候就可以进一步提高的编程效率了。

好，我们再回到刚才的业务当中去，你想我们用普通的op思想放到一个b点去调用，去解决那个问题，有不好的地方对吧？那么我们aop就能很好的解决。所以说它是aop的一个补充，你注意AP和op它是互补的，它并不是一个就是竞争的一种状态。

好了，那么如果我们采用aop思想解决问题，可以做到这样的一种程度，看这个图，比如说我们程序中有多个业务模块，而每一个业务模块都有相同的这个系统需求，比如说我要对他们统一记日志，我要对他们统一做权限检查，我要对他们统一做事务管理，那么我们使用aop就可以统一解决这个问题。

而我们使用aop解决系统需求的时候，我们只需要单独定义一个组件，这个组件和业务组件不发生任何直接的关系，你不用在业务组件里去调它。我们额外的定义了一个系统组件，那么把这些个所有的业务组件的通用的逻辑都封装到那里面去。

所以如果画图来看的话，就好像这样一种感觉，就是我这个组件它横跨了多个业务组件，它横向的扩展了业务组件的一些需求，横着切了一刀，所以说那么从直观的感官上给它命名，这叫面向方面，编程就是我们编程的角度是面向这样的一个横切的组件的。

好了，当然这样的话也并不能够让你彻底理解 Aop的概念，下面我们再往下细看，你要想理解aop首先你要了解aop的相关的术语，它的术语还挺多的，我们——来介绍，首先我们要想处理的那些业务组件，那一批业务组建的系统需求，这一批业务组建，一个的业务组建是我们处理的的需求的目标，这个目标我们叫套盖，叫目标对象，而我们采用aop解决问题，不是在目标对象上直接写代码，而是我们需要把代码单独封装到一个组件里。

我们封装业务需求的组件叫aspect就叫方面，或者说就叫切面，我们整个的编程的过程完全是针对 aspect来编程的，所以叫面向方面，编程 aspect里写什么咱们一会再说，把这个系统的逻辑封装到这里，这个地方怎么能知道，所以说我们就需要利用这个框架做一个植入的行为，把这个方面组件的代码植入到目标对象里，所以这有一个植入。

当然这是框架给我们提供的功能，而植入可以分为不同的时机，不同的框架，它采用不同的实现方式，那么可以在编译时植入，如果你在编译时植入，就需要使用特殊的编译器，或者可以在装载类的时候植入，你就需要使用特殊的类装载器，也可以在运行时置入，那么就需要在运行时为目标对象生成代理。所以不同的置入方式有不同的要求。

好，总体来说支柱方式越原始，编译式支柱它越原始，它的好处就是程序在运行的时候代码已经制好了，运行时速度快，但是因为你织的时机比较早，你在编译时植入可能很多运行时的条件你还不知道，可能有些特殊情况，你可能还怎么说处理的没有那么精细，而运行式植入它的好处是我程序已经运行起来了，我这个时候所有的条件我都知道，你想植入什么代码我都能解决，它比较灵活，但是缺点是我程序一边要运行，你一边还要植入代码对吧？

限制入网效率是比这种情况要低的，但是各有利弊，然后你注意还有一个概念叫连接点drop point，drop point指的是什么？你植入代码植入到目标目标对象里，你植入到哪个位置？其实目标对象可以有很多地方允许你植入代码，比如说我植入到属性上，比如说我植入到构造器里，我植入到静态块里，我植入到一个成员方法里，就是目标对象上，允许你有可能让你植入代码的那个位置，我们称之为连接点装part。

好了，最后我们再来看一下这个方面组件里都要做什么事情，其实主要做两件事儿，第一件事儿我们要声明point cut，声明切点，那么这个声明切点是用表达式来声明，什么意思呢？就是说是你目标对象有这么多地方可以允许我植入代码，我到底要植入哪些对象的哪些位置呢？对吧？你不可能说所有位置我都要植一遍，肯定是我有选择的去植入。

那么part card就是声明，我具体来说要把代码植入到哪些对象的哪些位置，然后是还有一方面的代码叫or the west通知，那通知就是实现了我具体的系统逻辑我到底要做什么，而且呢我做的具体的位置在哪里？

我现在我知道了，比如说我要把代码植入到成员方法里，我是植入到成人方法的一开头，还是结束的位置，还是抛异常的地方，还是有返回值的地方对不对？

所以说这个通知解决的是具体植入的逻辑。

好了，以上包括目标对象、连接点、支柱方面，切点，通知这么多概念，我就介绍完了，但是我想你初次接触 Aop这些概念可能一下会非常的绕，因为这个概念太多了，而LP本身又是一个抽象的东西是吧？可能你一时记不住，然后甚至有些混乱，这样我们再稍微理一下，如果理完了你还没有记住，你把这段这个视频你再听一遍，把它搞清楚了，我们再进行下一个环节。

好，首先我们程序当中的已经开发好了，处理业务逻辑的一个并，我们称之为目标对象，这好理解。

第二个，目标对象上有很多地方可以被植入代码，那么它的那些能够被植入代码的地方，我们统称为连接点。那么aop解决统一处理这些系统需求的方式，是将代码定义到一个额外的并叫方面组件里。

那么这个方面组件在程序运行的时候或之前就需要被框架植入到某些连接点之上，那么方便组件里我们需要通过point card切点，说明我代码到底要植入到哪些对象的哪些位置，我们需要再写上通知列的方法来声明我这个方面组件到底要处理什么样的逻辑。

好了，以上就是aop的这么几个相关的术语，那么除了了解aop的术语以外，下面我们再去看一下 aop 我们到底怎么实现，因为aop它是一种编程思想，对吧？我们怎么实现这种思想呢？其实实现 Aop的思想的技术框架有很多的，这里我没有列举全，因为没必要我们也不会都用。

然后这里我只列举了两项，一个叫aspectj，一个就是 spring AOP，好，那么 aspectj它是语言级的实现，什么叫语言级的实现？就是说它是一门新的语言，当然它是基于Java语言扩展而来的一门新的语言，然后他利用了这门新的语言定义了aop语法，所以说它是一门新的语言，专门解决aop的问题。

很明显它的功能会非常强大，aop所有的问题它都能解决，这是它的优势，但是它的缺点就是我们要用这项技术还得再学一门语言对吧？这个代价比较高，所以说有利也有弊，然后AspectJ它是在编译期植入代码比较早的时候植入代码，然后它里边有一个专门的编译器用来生成遵守Java字节码规范的类，这是AspectJ，然后AspectJ支持所有的连接点，你想在哪植入这个程序都是可以的，它功能非常强大。

然后第二个要说的是spring aop它是aop的一种具体的实现，那么spring aop它是使用纯Java实现的，它的好处就是说你不用去学一门新的语言，我们就用注解就能搞定这个问题，这就很方便就能上手。然后所以他也不需要专门的编译过程，编译器也不需要什么特殊的什么类加载器，这些都不需要。

然后是不是AOP它是在运行时植入代码的，那么是通过代理的方式植入的，代理我们一会还会再说，但是它只支持方法类型的连接点，就是说你只能将方便组件中的代码植入到目标对象的方法里，其他的地方你植入不了，它比较有局限性，但是你注意我们平时在处理这方面的逻辑是我们绝大部分情况至少是80%以上的情况，我们都要植入到目标对象的方法里。

这只是理论上，其实实际上我们所做的项目其实100%都是要知道放哪里。所以说其实是这样的，是不是AOP它是采用了一种灵活的方式，因为它是运行时植入，运行时植入我们能知道运行时的一切上下文的条件，所以说一切逻辑我都好实现。

而它植入的位置只支持一种就是最常见的方式，就是说它采用了一种最灵活的方式，解决了一种对常见的问题，他追求的是一个性价比，那么所以是不是AOP它不是一个全面的解决方案，而是一个性价比最高的解决方案。

有人会想到万一说AOP有他解决不了的问题的时候怎么办？你可以再去用AspectJ，这个时候 spring它是支持对AspectJ的支持的集成的，所以你看spring在这个方面它追求的是一个高的性价比，而不是一个全面的解决方案，是这样的，所以这是很有效的。

好了，那么了解了aop的实现以后，我们再看最后再说一下， spring AOP它在运行时植入程序采用的是代理的方式，代理是怎么回事？其实代理有两种方式，一种叫jdk动态代理是jdk自带的一种代理方式，一种叫CGLIB，动态代理是第三方的一个技术。

那么jdk动态代理是Java提供的，代理技术自带的，那么它有一个要求，所谓就是所谓的代理，就是我们对某一个对象给它生成一个代理对象来代替它，你调的时候去调代理对象，而是不调原始对象，我们植入代码就是植入到代理对象里，那么我们使用spring的应用，我们所调用的b都是通过容器去调的，那么我们通过容器去调目标对象的时候，然后他一看这个目标对象上有AOP的作用，你就需要他就知道我需要调代理对象，他就找到代理对象，调代理对象就可以了，而代理对象上又有植入的代码就能解决问题了。

那么我们采用jdk代理，它有一个要求，目标对象必须得有接口我才能带，在代理的时候是生成接口的一个实现类，它自动生成一个接口的实现类，然后代替了原实现类，是这样的。

所以spring它默认是采用这种方式，然后在接口的代理的实例中植入代码，但是如果说你这个目标组件没有接口，你像我们写的service没有接口，这时候怎么办呢？这时候就不能用jdk动态代理了，就需要用CGLIB动态代理，而CGLIB动态代理它是采用底层的字节码技术，那么在程序运行时，它会创建一个子类的代理实例，就是说你不是没有接口吗？

我可以创建目标对象的子类，我用子类来代理原来的对象，我们知道正好是支持这个多态的，那么我们可以用子类型来代替父类型，这是没有问题的。

所以当你的目标对象没有接口的时候，那么是不容易，LP就会用这种方式，这种方式也被spring的集成进来了，它会在子类的实例中植入代码。

好了，以上就是我们关于 aop的一些概念和一些实现的基本原则的一个介绍。下面我们再写一个小例子，然后来体会一下 Aop它到底是怎么回事。然后了解了它的语法以后，我们再去实现我们最终的目标，统一记录业务层的日志。好。那么我打开开发工具，然后我要统一处理，比如说业务层的某一项通用的需求，咱们先不说什么需求了，反正就是解决他们统一的需求。

我们看 aop怎么来解决，首先我在包下新建一个包，专门来存放方便组件，那么这个包我就叫aspect，方面然后我先创建一个事例，我先写一个类，实力我叫阿尔法s back，这里没有什么业务需求，反正就随便写点代码，什么体会一下 spring LP它的语法。

然后首先我需要把这个方面组件声明为一个b然后让容器来管理它，所以我需要加上一个component的注解，它不属于特定的某一层，另外还需要加一个新的注解，叫aspect表示，这是一个方面组件，它可不是一个普通的组件。

然后刚才我们说了，这个方面组件你要定义什么？要定义两个内容，第一个缺点，你要定义针对我项目中的b我要把代码植入到具体的哪些病人的哪些位置。好，那么定义这个切点非常简单，首先我就说明一个方法，这个方法没有返回值，方法名你随便取，这个方法名我就叫power in the cut，方法里不用写任何逻辑，这个切点我们是通过一个注解在方法上加一个注解来定义的，这注解就叫point cut。

好，然后在括号里我们需要写一个表达式来描述哪些b哪些方法是我要处理的目标我先写好，然后的话再跟你解释。

好了，好，其实这个很容易理解，你看这就是一个固定的关键字，然后的话星代表的方法的返回值，就是说新人表示什么反馈这都行，我不关注反馈值，然后 com点 Q点.community点service就是报名表示service包下的星，所有的业务组件，再点星表示所有的方法，就是所有的service组件里边的所有的方法，括号里边点点表示说所有的参数，那就是说所有的业务组建的所有的方法，所有的参数，所有的反馈值，那么我都要处理，是这样的，这样的一个表达式就匹配了所有的业务水平，所有的方法就这个意思。

当然你也可以比如说这不写某不写星，而且一个具体的返回值，这里不写两点，而且一个具体的参数来明确筛选出某一些特定的方法也是可以的。

或者比如说这个地方方法名你可以选爱的心，比方说以爱的开头的方法也可以，总之这个表达是很灵活，我们利用它能够很快速的筛选出一切你想要的连接点。当然我们现在选中一切就可以了。那么有了这个切点以后，下面我们就可以来定义这个通知了，然后利用这个通知来明确的解决一些问题。

然后通知分为好几类，我可以在连接点的开始做什么事情，结束时做什么事情，返回值返回的数据以后做什么事情，还有抛异常的时候做什么事情，还有连接点前后，同时都要做什么事情，就是说它有好几类，有你看1234有5类5类通知，而这5种通知我们是用5种注解来实现的方式大同小异，我挨个给大家实现一下，但是我的实现很简单，就是打印一句话，如果你想在连接点的一开始，开头的地方记日志，你就写注解before，然后你这里要说明哪些连接点呢？

其实就是一个切点，所以说我们需要这样写，把这个方法写到这儿来，比如说我要以此为切点，我逻辑是针是针对这个缺点，针对这些连接点有效的，好，然后我说明这个方法，这个方法不需要反馈值，方法名你随便取，我为了直观的还要比放，这里我就随便的打印一些内容。

好，你在前面植入代码是这样的，后面的话和它类似，我就copy一下，如果你想在后面记日志，你这个地方就不要写before，写成after也是缺点，方法名我改成叫after，写错了，after。

好，那么还有一种情况就是说比如说我想在有了返回值以后再处理一些逻辑，我这块可以这样写。After returning这会写after returning，那也有可能你希望在抛异常的时候植入代码，你可以这样，那就是after studio。

好，然后还有一种特殊情况是我既想在前面植入逻辑，又想在后面植入逻辑，但前后都植入逻辑，我可以用单写叫run，然后切点也是point cut。

好，那么这种情况我们就需要有返回值，返回一个object，然后还得有个参数，它的方法名忘了写了，这个就叫around，它得有个参数叫proceed专判，这个参数就是连接点，就是我目标植入的部位代表部位，方法参数名叫张胖子，好，然后我的声明抛出异常好了，写错了stress。

好，那么这个怎么去记在前面，这个处理逻辑又在后面，同时处理逻辑其实主要是利用这个参数，连接点指代的是我程序植入的部位，我可以调用这个参数，然后点proceed，不是点 proceed，那点proceed其实就是调目标对象被处理的方法的逻辑，就是说说白了就是调我们要处理的目标组件的方法，这句话就是调用目标组件的方法，然后当然目标组件可能有反馈值，你可以这样反馈一下，然后我就把目标组件的反馈值为确认掉，那么因为程序在执行的时候会执行当前的会执行一个代理对象，这个逻辑被植入到代理对象里，用来代替原始对象，而我们这段制度的逻辑是我利用它去调用原始对象的方法，这样的话原始对象方法被调了，我在这句话之前做什么事情，这句话之后做什么事情不都是可以的吗？

这样我在这句话射雕目标组件前后各加一段代码就解决了这样的前后同时处理一些逻辑的问题，是这样的。好，这里我写什么？

Around的地方，然后这边我写a round，after，然后我们程序执行的时候，我们可以看一下这些个标志它在什么时候输出出来的。好了，这就写完了，写完以后目前这些代码就针对所有的service就都有效了，而我对service是一行代码都不用改，你可见这个死不是LP它有多强大。

好，写完以后咱们来测一下，我把这个项目启动一下，启动完以后我把控制台清空一下，这样我一会看控制台内容比较清楚一点，那么首先我访问一下首页，你看不耽误它的使用，不影响它的使用。

然后我们看一下控制台输出了多次这样的这样的标志，为什么？因为咱们首页上不是查询了帖子的列表，而每一个帖子我们又查了优字对吧？所以说我们调了多次业务组件，而针对每一个业务组件，我们这个方法都会被触发，所以说这里有多次这很正常。然后你看首先 around before环绕开始，around，后面还有around，after。环绕结束被调用了是吧？然后前置通知被调用了，后置通知after被调用了，还有返回以后after returning被调用了，这些调用的顺序我们也能够看得出来它是这样一个顺序，然后后面有第二次这样的逻辑。

第二次循环我们就不去看第二次了，但是这里边没有输出这个东西，为什么？因为咱们程序没报错，如果报错的话就会输出了，这个可以你自己试一下，我就不去照这个错了，因为其他的都好使，这个也没有问题，好。

所以你看这个LP虽然不好理解，但是很好用，咱们就写这么简单的代码，就能够对这么多业务组间就统一处理他们的业务逻辑非常的方便。

好了，有了这么好用的一种方式以后，接下来我们就处理我们最终最开始所说的那项需求，我就要把所有的业务组件对他们的记录日志，而我什么时候记录日志，我希望在这个程序的就是在业务组间调用的一开始进入制就是用比方的方式，然后记录日志的格式是这样的，就是说用户某某不是用户名，而是IP地址，因为可能有的用户没登录，用户某某在某个时刻访问了某功能某方法，就是这样的——一句话的提示。

好，这样我把实例把它注掉，因为我不想每次都输出这些内容，举一个事例明白就可以了。好，那么为了处理正式的业务逻辑，我需要再单独写一个类，那么这个类我取名叫叫service log aspect。

好，那么根据刚才我们所学的事例，我知道我需要在这加上可能的注解，需要加上 aspect注解，然后因为你要记日志，所以说你要实例化 Log。好了，然后我们首先要声明的是缺点，那么缺点我们可以和刚才 Copy一下，因为一样还是所有的业务组件，所有的业务方法我都要去处理好。

然后我要用前置通知在一开头置入程序，所以比放然后朋友你看，那么这个方法我写上怕被不好意思，比方好刚才我说了，我要声明的，我要记了日志这个格式我再来写一下，是这样的，是用户，然后比如说一个方框某某是一个IP地址，比如说1.2.3.4，假设是这样的，好，然后用户某某然后在什么时间叉叉时间，然后访问了什么功能，比如说com点，然后q点，community点 service什么方法，就是这

样一个格式好。

那么你要想记录这样一个格式的日志，首先用户的IP怎么获取呢？其实用户的IP我们可以通过request获取，但是我们在这个方向里怎么去获取request对象，你可能不能简单的说，我在这里说明一个request，对象不能这样做，我们可以利用一个工具类叫做request。Context。Holder。Request。Content它有一个静态方法叫get。Request h表示。那么这个方法我们可以把它转型成一个它的子类型得到的值可以返回转换一个子类型，这个子类型叫so累的request，h表示。

好，这需要强转，因为它返回的默认的类型叫request，我是把它转成子类型这样功能多一点。好，然后通过 h表示我们就可以得到request，对象点get request。那么它的类型就是HTTP12类，对case。好了，那么有了request以后，这个IP地址就能得到了，很容易得到IP等于request点。Get remote。Host好，这就可以了，但是除了IP以外，咱们还得拼这个时间，时间好办，我们就6个date，然后格式化一下就好了。我说明一个变量叫now，我需要把格式化的后的日期字符串复制给他，你有simple的format，然后指定格式，然后点format new，对。

好了，最后还有你要访问的是某个类，某个方法哪个类，哪个方法我怎么知道呢？

这就需要加一个参数了，其实除了环绕通知以外的通知也可以加连接点的参数，那么他们这个方法上加连接点的参数，这个名字就叫John point，那么连接点指代的是你程序植入的目标，组件你要调用的方法，它指代那个方法我们通过它就能得到我们要调的将要调的是哪个类哪个方法，怎么获取是这样的。

Join point点get signature，然后再点get Q带q然后 txt得到的是类名，我们还要拼上方法名，你可以这样加上一个点，再加上因为你中间不是拼了个点，对吧？再加上这个方法名，而方法名的获取方式也是类似的。是John point，点get是个nature，然后点get name这就是方法内容。这样的话我就得到了这个目标的类型名和方法名，我就要 target。

好了，那么准备好了这些条件以后就很好拼了，就log点，这个级别我叫infer，因为没有报错log点Info，然后的话这样我为了拼这个字符串，我用使用点放卖的方法去拼。首先我先写上一个字符串的格式，是用户某某用占位符代替，然后在占位符某时间访问了占位符某方法。

好，那么这三个占位符我需要用不同的变量去替代，第一个是IP，第二个是第三个是target，好，那么这就完成了，就这么简单，完成以后咱们再做一个测试，我把这个项目重新编译一下，然后它已经重启了，我清空一下，然后再打开浏览器，刷新一下首页，看控制台，看他有没有记日志，应该有用户某某某。

当然这个IP因为我那个页面上写的是logo house，所以说logo house它转成IP就是这个样子。如果你不喜欢这样看，你也可以这样，你可以把 Logo house改成127.0.0.1，这也可以飞车127.0.0.1它和logo house的是等价的。

再看你看用户 IP地址在某一个时间然后访问了，因为我们在查询帖子列表的时候也顺带查询了用户，访问了user service点，find idea对吧？

没有问题，这是针对首页其他的功能可不可以咱们也试一下，比如说我登录一下好，那么登录以后我访问这个消息，好，然后再看你看用户某某某，然后在用户某某在某时间访问了消息组建的 find later，everyday count方法，这是find use by ID这是find later count，这是find conversations。

看到了就是说我们在访问任何一个业务组件的方法的时候，那么通知这个方法都被触发了，我们就统一处理了问题，而这种方式又很简单。

好了，总之 AP怎么使用是很容易掌握的，但是他的这个理念，它的实现一些机制，它的一些术语，一些概念不容易掌握。课后的话大家多花点时间把这个概念再好好再理一理，再好好再琢磨，然后代码你很容易就能学会。好。这节课我们就演示到这里，咱们下次课再见。

# 第4章 Redis，一站式高性能存储方案

## 4.1 Redis入门

28-Redis入门.mp4

这节课我们来学习第四章的内容，这一章我们主要学readiness，redis它可以对我们应用程序当中的很多功能，提升它的性能，所以我们学完redis以后可以利用它来开发我们应用当中对性能要求较高的功能。除此以外，我们学完以后还可以利用它重构一部分我们现有的功能，这样的话也能够提升这部分的功能的性能。

那么我们第一节来学习redis入门相关的内容，主要是大家要了解什么是redis，然后他怎么去安装，然后呢安装以后基本的使用的方式，首先我们先对redis的概念做一个介绍，redis它是一款基于键值对的no sql数据库，然后它的值支持多种数据库数据结构，我们先看这句话，就是说先说 no sql，no sql，数据库是什么数据库？

它是指关系型数据库之外的数据库的统称，反正除了像mysql oracle这种关系型数据库以外的其他的数据库都统称为no sql数据库，这个no sql它是这样一种单词的缩写，叫 not only sql不只是sql。为什么？因为其他的类型的数据库它数据存储数据的方式有很多，它不只是靠sql去存储数据，有的是按照建制队的方式存，有的是按照列的方式来存，有的是按照文档，有的是按照图形，反正这个形式是各异的，那么redis它是按照键值队的方式存储数据的，那么它来存数据的时候，键值队的key都是string，而value的value这个值是支持多种数据结构的，然后它的值支持的数据结构主要有这么几种，当然还有其他的我们课程中主要是先来学习这么几种，分别是字符串然后列表集合以及有序集合，那么这些比较常用的基本的值的类型，我们一会会给大家做演示，后面课程当中基本上大部分我们都会用到。

除此以外，那么redis它为什么能够以一个很高的性能的方式去存数据，是因为它把所有的数据都存放到内存里了，我们知道我们和内存打交道，我们向内存中存取数据的速度要远胜于硬盘，速度比较快，所以它速度惊人。那么但是我们一直把数据存在redis里，万一这个服务器关了对吧？这个数据不就没了么？你也不用担心，在它把数据存到内存的同时，它还可以将内存中的数据然后的话或者是快照或者是日志的形式也可以存在硬盘上，从而保证数据的安全性，这样的话它就性能和安全性都具备了。

那么快照的形式其实也叫rdb的形式，那么这种形式就是说它直接把当前内存中的数据，然后原原本本的直接存在硬盘上，那么这种形式的优点是它数据的体积小，反正你内存中有啥我就存在硬盘上什么。

然后如果说我们想从硬盘上恢复，数据到内存里速度很快，但是它的缺点是因为你是一下子把整个内存中的数据存进硬盘上，那么它在做这件事情的时候比较耗时，而且在做存储的时候它会产生阻塞，如果这个时候你还你还在处理其他的业务的时候，会对其他的业务有影响，会影响性能，所以说以快照的形式往硬盘上传数据不适合实时去做，它适合比如说几个小时做一次，做一个数据的备份比较合适，然后以日志的形式存数据到硬盘上，这个叫aof这种方式。

那么它是每次你每执行一个redis命令，它就把这个命令存下来，它一致的方式把命令存下来，存一个命令，就一句话这个是非常快的，所以说它可以做到实时的存，但是它存这个命令是以不断的追加的形式存的，它比较体积比较大，比较占磁盘的空间。

另外的话如果你想从磁盘里硬盘里把数据恢复到内存里，它是把你存的那些个命令从头到尾再跑一遍，所以说它恢复的速度比较慢，但是好在它实时性好，这两种方式你不见得说我必须得用其中的某一种可以两者结合在一起用，是这样的。

那么redis它是一个非常好用的工具，一个是它性能非常好，第二个是它支持的数据类型非常的完善。

第三个它的使用上非常的简单，特别容易学，那么这么好用的工具它都适合做什么？这里我列举了它常用的使用场景，第一个它最适合做缓存，比如说有些数据我们访问的非常频繁，那么你存到硬盘上它不就慢了对吧？我缓存到内存里，从内存中取就快，这种功能我们一律可以用redis来搞定。

第二个是这个排行榜，比如说我们做一个热帖排行，那么排出来前10个或者是前30个热门的帖子，这个热门的帖子是访问非常频繁的，除了热门的帖子之外，其实访问量就比较少了。

既然他访问这么频繁，我们把热门的帖子那么进行一个缓存，你来访问的时候效率不就高了。

第三个是计数器，比如说模仿帖子，那么这个帖子的浏览量就要加一，你访问浏览量又加一，每个人访问他浏览量都要加一，这个操作是非常频繁的一个热门的帖子，很多人会去看，数以万计甚至几十万的人去看对吧？

所以说你去计数这个数据，如果存到硬盘上，那么这个性能又比较低了，而存到redis存到内存里，性能OK的，另外它也广泛地应用于社交网络的一些功能当中去，比如说你点踩关注你像这样的功能，那么访问是非常频繁的，比如说你发了一个热门的帖子，很多人会给你点赞，那么很多人会对这个帖子进行关注，对吧？我们把它这样的数据存到内存里，效率也高。

再一个你就是一个帖子点赞，其实就是一个数量。那么这样的数据你存到一张关系数据库的表里，其实它也不方便存，就存一个数据对吧？你还建一张表，其实也很麻烦，所以这种方式我们以建制队的方式存在内存里就非常方便。最后那么Redis还能可以用来做这个消息队列。

当然了redis它不是专门做消息队列的工具，所以说我们在课程中不会去演示这个功能，但是就是说如果你的项目对这个消息队列要求的不是特别高，Redis通常也能够满足你的需求，如果你要求特别高的话，你可以用更专业的消息队列的服务器，我们后面讲的卡夫卡就是这方面的佼佼者。

好了，那么我们大概对Redis做了一个大致的介绍，下面我们看一下它的官网，它的官网是redis.io。好，我打开一下官网看一下，有点慢，大家稍等一下，打开以后，那么如果你要看他的手册，你点documentation这里边它针对一些功能或者是某些方面做了一些归类，这里可以去看。

但是其实Redis是非常简单，你不看这些内容也可以，Redis是我们学主要学什么，其实就是学他那几种数据类型，你只要会用那几种数据类型，你会往Redis里存这些类型的数据就OK了。

而你操作这个Redis去存各种类型的数据，其实就是用各种命令，所以说其实你去学会它的命令基本上就会了。

那么与命令有关的手册，你可以看这个地方come on，好我点一下，然后这里边他给你做了分类，这里默认是所有的你可以看，比如说string相关的命令，我这个情况有点慢，大家稍等太慢了，我就跟你大概说一下，你点string这里边会做一个过滤，过滤出只有string相关的命令，然后以此类推，这个是集合，这个是有序集合，这个是列表，这个是就这几个比较常用，其他的我们后面也会涉及，但现在我们先不管，总之命令是非常多的，那么我在课上不会说把每一个命令给你演示一遍，太啰嗦了，那么大家就是我课上给你演示一些基本的最常见的最常用的，那么课后或者是你以后的工作当中，你可以多去查一查手册，看一看它还有什么功能，这个是你自己要自己去查的。

好，然后我们要安装redis，那么当然在官网上你可以去下载你点下载的，可以下载，但是下载的话是这个地方是正式版，你点但是下载的是redis.tar.gz这样的，包这个文件是针对Linux系统的，而redis官网它没有提供针对windows的这个安装包，不过好在有redis-windows，它是非常流行的一个工具，那么引起了微软的重视，他一看既然你自己不提供安装包，我给你做一个，所以微软它给redis做了一个windows系统下的安装包，那么下载的路径是在这get up，那么我已经把这个网址打开了，打开以后我们往下看在下面有它的一些介绍，我们可以点release配置，点开以后这里边他列举了各个版本，那么最上面的这个是最初的版本，3.2.100，然后你可以点一下3.2.100，下面有一个这个列表，列举了他所有的版本，所有的安装包，然后一个是MSI就是安装程序，一个是ZIP就是解压缩压缩包，这两个是源码包我们不需要，而这两者我选择的是安装程序选的是它，然后我已经提前下载好了，大家看一下就是这个文件。

好，下面我就把工具给大家安装一下，很简单了，基本上就是一路默认就可以下一步勾选同意，然后下一步这里边你可以改变一个它存放的位置，改一下我把它放到d盘work目录下，然后在work目录下我想新建一个文件夹叫redis。



好就放到 d盘work reads目录下，然后下一步你注意这个是它的端口，当然你这里可以改，但是我们没有必要改就默认就可以了。然后下一步安装很快就安装好了，这是一个非常就是简约小巧的工具，安装完以后，那么我打开第一盘work看一下，这不有rise对吧？

就这然后它这个目录也很简洁，你看里边的话它连一个文件夹都没有，就是一堆文件，其中这个是配置文件，然后其他的有一些你像 Reds杠cla是它的客户端文件，当然我们访问它的话通常会通过命令行的方式来访问，那么为了能够容易直接访问这样的工具，我们最好是把安装路径配到环境变量里，这样的话我们民航就能够直接访问到这些工具了，我配一下高级系统设置环境变量，然后 pass，新建一项粘过来，这就可以了。

好，配完以后，那么我来访问一下，come on，打开命令行，然后的话我现在要启用的是rise的客户端，好访问这个readiness，然后你readiness安装完以后，它服务就自动启动了，我们通过客户端来访问它，radius杠CRA你看输出这句话表示我已经连上了。

127.0.0.1本地端口是6379，然后 rides这里我给大家演示它的一些常用命令，你把基本的使用方式会用了就行了。

那么reds这个数据库它默认了，其实内置了16个库，这16个库它没有名字，它是用索引来区分，所以分别是0~15，那么默认它选中第零个你可以切换，比如说select一，这就切换到第一个库，select二切换到第二个库，我再select0切换回第零个库，我就在这里就给大家做一个就是示例做一个演示。

然后比如说如果你演示过以后，库里有一些演示的数据，你将来不想要了，你可以刷新一下，把它刷没了，你可以flash DB刷新库，刷完以后库里就啥都没有了，当然我现在也什么都没有，因为我还没有往里加东西。

好，接下来我给大家演示一下刚才我们 PPT上所列举的内容，5种值的类型，我们都怎么去用？

我先演示的是死菌类型，我怎么往库里增加这个值为死菌类型的数据，怎么把它取到，怎么对它做操作，给它演示一下，非常简单，我们要往库里添加死菌类型的数据，就直接site后面它这一串文字是一个提示，这是一个提示，他意思是你接下来你要设置的是key，因为它是py6结构，你得先声明这个k那么这个k你像我们这个关系型数据库，如果是我们 key或者说我们表有两个单词构成，我们会中间加一个下划线对吧？

而这个rise它提倡这样的规则，如果你有两个单词中间相连的话，用冒号好，比如说我 key要取个名字，我要叫 test count，数量你就这样写test冒号count，这是一个key。

然后指定它的value，比如说value是一，后面其实还可以指定他超时的时间，这里我就不写了，后面带方括号的表示可以省略回车，ok表示这个数据我就已经存好了，我已经 key和 value一对对key value存到了 radios里，那么其中的 value它是以字符串的形式来存的，怎么取到存是site取那是get，get后面写上 t你看取到的是一字符串，然后它还支持对这个值进行一些计算，比如说我要让这个值加一，比如说我存的是一个浏览量，我让他加一怎么办？

也很简单，incr就是inc增加的一个单词的缩写 t我让 t对应的值加一，加完以后变成二减一就是decr decadent减法的缩写，好减完以后变成一。简单，这就是死菌类型的数据的使用方式。

好，下面我们再来看一下希我要存希类型的数据是这样的，这命令以h开头，h set存数据 key比如说我要叫test user，我要存user，而这个值它不是一个值，因为它是希这个值也是p value，feel the value 好。

User比如说我有ID，它的值是一，我就这样写，存好了，然后我再存一个h site，test，user还是往 key里存数据，user还有另外的一个属性，user name张三也好了，存完以后我怎么取其实很容易取。

H get就是取然后写上 key test，user，而这个k我们得到的是一个希，你到底要希中的哪个值，你再把哈希的k再写上去，比如说我要取ID，ID是一，接下来 h get test user。

然后比如说我要取user name张三，就是这样一种形式。

好，再来下面我们再演示一下 list就是列表，那么 ready是他对列表的设计比较灵活，这么说它的列表是一个它这个列表是一个横向，你可以把它看成一个横向的容器，那么我们往这个容器里存数据，支持从左边往里装，也支持从右边往里装，取的时候支持从左边取，也支持从右边取。

那么如果说你是左进右出的话，那其实就是队列，如果你是左进左出的话，那就是栈。

所以说你用这个列表可以做出做出队列和栈这样的功能。

举个例子，比如说我要往类似的存数据，按照左进右出的方式来存，比如说我存的是ID分别是101102103，我先存的是101左进的101，存进去了，然后再存102，你注意是左进你就要在左边加上102，然后103左进103是这样，所以最终出版的是这样一个对于列表里的第零个位置就是103，第一个位置是102，第二个位置是101，如果你又出的话，首先出的是101，然后是102103是这样，你可以把它想象成一个横向的容器这样。

好，具体怎么去操作给大家演示一下，我就按照刚才我所说的左进右出，首先我从左侧往里存数据，是l表示list toss左侧近，然后 key test我要存的是比如是ID，我叫ID，然后101其实你也可以直接把另外的多个这里可以写多个值，我就直接102103那意思是我分别存101102103，那顺序和刚才我说举的这个例子就是这样的，它是先101在102在103注意好，然后回车出来，三个数据进去，出完以后我们可以看一下 len那是看列表里边的长度，我看它有多少个数据？

三个。然后再看l index，就是我要看某一个索引所对应的值。下面你把 key写好，test爱丽丝比如说我要看索引为0的值103对吧？1030，好，再来。l index test ideas。比如说2101最后一个，好，那么还有我还可以说我不只是看某一个位置的值，我想看某一批位置的值，或者说从索引几到几的范围内的值也可以。

l range rang然后 Test爱丽丝，比如说我从0开始到2，一共就三个就0~2，你看分别就是103102，101很直观了。

好，然后我还可以让他从右侧出兑，就是弹出一个一个值，你就rr表示right，l表示left，刚才我说错了，不是表示list left，这个r表示right，r然后 test ads表示从右侧出一个值，它弹出的是101，他们的右侧对吧？好，再来 are pop test。爱丽丝102。以此类推。

好，这是列表，然后我们再看集合 site，那么集合的特点就是列表是有序的，有序的，然后集合是无序的，而且集合中的内容值是不能重复的，这是集合的一个就是要素。

我们怎么往集合里存数据很容易是这样si s代表的是site添加，你还要写一个key，比如说我要存的是老师，我写test，teachers，然后你可以后面写多个老师的名字，我就随便写 aaabbbccc dddeee写5个，回车我就往集合里加入了5条数据，然后我可以通过这样一个命令 s card test，teachers这句话可以是统计这个集合中有多少个元素，5个没有问题，然后我还可以通过这样的1个命令，s这句话的意思是从集合中随机的弹出一个元素test，teachers回车，你看它弹出的是CCC，s pop test。

Teachers这会弹出的是意义。

好，那么有时候这样这有什么有用这个的话，我们可以利用这样的命令去实现抽奖对吧？那么你把这个人或者是人的编号存到 site里，然后它不能重复，又是无序的，就相当于我们一个抽奖箱一样对吧？然后你每次弹出随机的一个用户，他就中奖了，非常简单。好，然后还有一个命令是s numbers，这句话是查看集合中还有多少个元素，你看弹出两个之外还剩三个没有问题。

然后最后一个要演示的类型是sorted set有序的集合。

刚才这个集合是无序的，如果说我希望集合有序，你可以用有序集合，有序集合其实它是这样的，它是跟每一个存的值就是附加了一个分数，然后他按照这个分数去排序，它提供了一个能够按分数排序的这么一个功能，那么有序集合，我们是这样操作，以z开头爱的添加，然后比如说我要添加学生test16.4。

好，然后你后面要写什么呢？后面你要写这个省可以省略的我就不写了，你要写分数，然后以及和对应的值，然后可以写多个，比如说我就写了10，分数是10AAA20BBB，30CCC，40DD，五十一——就随便写了5个人，然后他们的分数分别是abcde回车加完了，加完以后其实有序集合的命令和集合有点相似，我可以通过z card来统计一下它里边有多少个数据？

5个，然后我可以这样z是杠，是查询就是某一个值它的分数是多少？Strong的分数你要写上key test，16.4，然后写上一个值，比如说我要查 CCC的分数，CCC30没问题。好，然后再来我可以这样，z rank，z rank是返回某一个值的排名，它排第几默认是由小到大的 test students，比如说还是CCC它排第二名，你看这个排名其实要求达到就是这样一个顺序，从左到右对吧？012没问题。好，再来还有一个g range范围，取某一个范围内的数据test，students，比如说我写02，意思它是按照由小到大排序，然后取0到第二个值那是谁？

Abc A是0，b是1，c是2，就这样。好了，你看总之 rise它的命令很简单，反正我们利用这么几条命令就可以操作不同类型的数据，只不过你可能一开始接触命令，因为有点好几套，你记不住那多写一写就记住了，然后多去查一查手册，还有额外的其他的命令，我没有演示到的，你也自己去看一看，自己试一试。

好，那么除了这些命令以外，我们再说一套命令是全局的命令，是对所有的数据都有效的命令。首先你要知道 Key星意思是查一下我当天的库里一共有多少个key，星代表一切，所有的key都列出来了，所以它针对所有的类型都有效，这个也比较常用，然后你也可以这样kiss，比如说test星，意思是我要看一下我库里test开头的key有多少个，其实我们都是以test开头的，就查到了。

总之 case命令主要是查看库里的 key相关的内容的。

好，接下来还有一个命令叫tat，是看某一个t它的值的类型，比如说我看一下test，冒号user看一下 key它的值的类型是什么。没错，还有一个exist是看一下某个key是否存在，比如说我看一下 test，user不存在一表存在，好，这样我把它删了，删的话是给del指定一个k把这个k删掉，删掉以后它的值就没了，我把 test用这删掉，删掉以后我再判断 exists test。

User要表示00表示force没有了。

好，那么最后还有一个命令比较常用，是我们可以设置某一个key，它的过期时间，它到了这个时间自动销毁就删除了，比如说我们登录功能打开登录页面，它不是有一个验证码，验证码你存到库里，其实过一分钟删掉就可以了。

那东西不用存很久，所以说像这样的需求就可以用到这个功能。

我给你演示一下，xkr过期，然后你指定一个k我指定的是16.4，然后指定秒多少秒过期有10秒。好，写完以后kiss星，你看目前19等值，还有我再来kiss星，还有好我多等一会儿，等他超过10秒，应该差不多我再执行一遍，你看就没了，只要超过10秒，它就自动的把这个数据删掉了，好了。

那么以上是我们 rise入门相关的内容，你理解rise是个什么东西，另外了解它的每一种数据类型，然后我们怎么去做基本的操作。当然课后你要把 reds安装好，自己多试一试，把这些命令用熟了。好了，那么这节课我们就演示到这里，咱们下次跟他再见。

## 4.2 Spring整合Redis

29-Spring整合Redis.mp4

这次课我们来学习，利用spring整合redis，那么关于整合的方案，我们从spring的官网，它有一个模块叫spring data，然后里边有关于整合redis相关的内容，你从这个手册里能找到这个答案。当然了这个手册比较长，那么我把这个手册中的内容做了一个提炼，然后呈现在这次课里呈现给你。

那么总体来说我们用spring整合readiness大致分为三步，那么第一步我们需要打包，而super boot他对Red spot做了一个处理，那么我们可以直接导它整合以后的 start data readies包，打完包以后第二步我们需要对riders进行配置，因为它也是一个数据库，所以说首先你要在配置文件里，也就是application点properties里配数据库的参数，其次我们需要写一个配置类来构造 spring rides它的一个核心的组件叫rides template。

那么这个组件是由spring提供给我们的，当然了其实 spring boot它也对ready是做了一个自动的配置，其实它已经配好了 readies template，不过它配的对象，因为我们知道瑞迪斯是q56结构的数据库，他在配的时候他把 key做成了object的类型，但是这样做的好处是更通用，但是其实我们一般用的时候我们都是用的都是死菌，所以说它配成了奥PK的，我们用起来不方便，因此我们需要重新配，这里是其实是做一个重新的配置。

那么配好了 release template以后，我们在写代码的时候都是用组件去访问radius。那么比如说你要访问 string类型的数据，那么就read this template点ops of reasons for value。然后你要访问希类型的数据，佛希for list, for sat, for this sat, 所以说通过这个方法名，你也就知道它大概代表的含义，总之这里边每一个这样的方法就返回一个对象，这个对象能够访问这种类型的数据就非常非常简单非常直观。

好了，那么接下来我就把这三个环节给大家演示一下，首先我们先来打包，那么我打开 maven reporter到com，然后搜一下包我就搜release。

好，那么收到以后我们要 spring boot data red is daughter，然后点进去，我选一个比较新的版本，然后把这段配置代码贴过来，粘贴到我们项目当中的 pom的xml。

好，实际上我们可以像这样的包厢springboard，它所整合的包或者是死不用说整合的包，其实很多时候我们可以不用去写 working，你可以把它去掉，为啥不用写word？

因为当前 poem它是继承于了某一个副poem，副泡沫中声明了很多包的常用的包的版本，所以我们纸泡沫可以不用去写这个版本，而且它负泡沫声明的这些版本互相它做个测试是兼容的，不容易出错，负泡沫声明什么咱们可以看一下，你看这就是副泡沫，然后这个是它的ID，你CTRL可以点进去，点进去以后它又有一个副框是它你再可以CTRL点进去，这里你可以看到它定义了很多关于各种炸包的版本，比如说我想看有没有rest，你看他就定义了 spring boots, start date readies他定了这个包，然后版本是2.1.5，那么这个版本它在这些包当中彼此是做过一个测试的，是互相兼容的，我们如果不写word的话，它用的就是这个版本。

好我就不写了，因为我们当前用的版本和其他的包，就这个体系当中的包它是比较匹配的，就这样就好了。那么打完包以后，第二个我们需要对数据库进行一个配置，我就打开 application, practice在后面做追加，我们当前的配的是release，这个ready你看一下，ready你可以搜一下。Rises out to confirmation这是死不认不得，对。

Rises做自动配置这样的BIM。

然后你会看到它自动配置的时候配的其实是类中的属性，是对类进行配置的 radius process，然后你会发现它配了 ready stop累的，然后 key是奥迪不好用，所以一会我们需要把它重配好了，回到配置文件就是我们配的所面向的类是这个类，然后这里边我就直接写我要配什么 spring，点readiness，点data, base。

就是我pay，我要用readiness中的哪个库，它不是一共有16个库对吧？我得选1个，我要用哪1个，你随便这里我选11好。然后 spring点rides，点cost，你要写上一房的库，它的 IP是多少？那本机 Host。

好，再来spring点rises part端口。那么我们安装的rids默认端口是6379。行了，当然其实还有更多的参数，我们不用挨个都配，其他的参数都有默认值，这三个是你一定要配的。好了，那配完以后按照刚才我PPT上所说的，下一步我们需要编写配置类，把 release他配的配好，那么我就在 can figure目录下新建一个类，然后这样 radios can figure对吧？这名字写得不好，写错了。

虽然说无关紧要，但是还是改一下，red is conveyed。

那么配置类我需要先写上consideration，好，然后再写上b我需要定义第三方的b那么这个方法声明为公有的，那么你要把哪个对象装配到容器当中，那么你就返回这个对象，它返回的是readiness，template。泛型它的key我希望的是使命，它的外流是object，然后这个方法名其实就是bean的名字，我叫red is timeless。

好，那么因为数据库我们访问数据库利用time play的访问数据库，那么这个time play的要想具备访问数据库的能力，他得能够创建连接，连接是由连接工厂创建的，所以说你需要把连接工厂注入进来，然后注入给他们类的，他才能够访问数据库。

那么你可以这样在这个方法之上，声明连接工厂叫reds connection。然后 factory red is connection，factor的名字我叫factory。那么当你在定义一个bean的时候，然后方法上说明了这样的参数，那么spring容器会自动的把并注入进来，这个并已经被容器装配了。好，那么在这个方法里我就要实例化，这个ready template6室内化完以后，我需要把工厂设置给 template， site，然后是connection，factory。

好，那么template有个工厂以后，它就具备了访问数据库的能力，我们配这个产品类的主要配什么？其实主要配的是序列化的方式，因为我们写的程序是Java程序，我们得到的数据是Java类型的数据，最终你要把这个数据存到 risk数据库里，那么你要指定一种序列化的方式，就是说或者说数据转换的方式。

好我这样社长，因为rides它是ky6结构的数据库，所以首先我要声明我要设置 T的序列化方式，除此以外我还要再设置普通的value的序列化方式，普通的，然后另外那么有一个外流比较特殊，y流本身是个希又分为py流，你这个时候还要这样设置希的t序列化方式，然后设置希的外流的序列化方式，基本上我们把这4项设置好就可以了。

好怎么设其实也很好，起 template。点set key，series设置key的序列化方式，序列化方式有一个类里边有了一个统一的定义，我们可以直接访问。类的叫radius theorizer Radius sterilizer，然后点儿使劲，那么这个方法它返回了一个能够序列化字符串的序列化器，你就指定为他就可以了，我们自己不用去创建这个东西。

好，然后再设置 value的 template点儿，site value zero，lever，然后还是去掉，相当于是一个工具类，然后点 value的话可以有普通的值，也可以有集合列表，你要把这样的数据需求化成什么样的方式呢？

通常我们建议是把它序列换成Jason，因为 Jason格式的数据它是结构化的，我们恢复回来也很好去读取，很好去识别，所以说节省的去的话，现在这里也有有点节省就可以了。

好再来，那我在template点site希写错了，希key zero letter。那又是key又是字符串，那你还用这个就可以。

好，然后 while yo tamper layed，点cide哈希Vai6 zero Liza，那么value可能是各种形式的数据，你还是把它转成Jason比较好，其实主要就是设置这么几个东西。那么设置完以后，为了让template当中的这些参数生效，你还要调一下 Template的点after property set就是当你设置完了，做完了一些设置以后，它要触发一下生效。

好，最终我们return template配置就搞定。好了，现在我们包也倒了对吧？数据库的参数也配了，然后Template我们也配好了，接下来我们就可以通过 ready template去访问readiness，下面我给大家写一个测试类，我们把对reds的各种类型的访问做一个演示，看一下reds他们类的怎么用。

好，我就创建一个新的测试类，我叫readiness test，当然这个类上也需要加上那么几个注解，我copy一下。

好，然后我们要访问 release，全是要利用刚才我们所配的 template，所以我们哪儿需要访问rides，就需要把 template注入到哪里去。好，我注入进来了，Red is template，然后下面我就写测试方法，首先我先给大家演示一下我们怎么访问这个字符串，以字符串为值的数据，写个字的方法，这个方法名我叫test使用。

好，首先我先声明一个 key，我在声明之前我们先看一下 ready库，先确认一下这里面有什么东西，我当前的项目指定的是利用的是11库是吧？所以我选中11 kiss星，你看这里啥都没有，目前什么都没有。好，然后待会我们出了数据以后我们再看，首先我要声明一个key，我先声明一个变量把它定义好，ready？

Skip等于。

这是测试数据，所以说我以test开头，然后 count还是数量，就和我们上次跟他演示基本的命令一样，我想存一个count。

好，然后那么存的方式就是利用 template，然后点ops，放 Value就是string类型的值，访问string类型的数据，然后我要存数据点sat，你想咱们命令不就叫site对吧？其实它方法和命令基本上是对应的，然后这是你要声明我要给哪个k赋值，值是多少，值是一，就这样。好，这是纯数据，我怎么取再演示一下，取出来以后把它打印出来。Read this template，然后点ops for value然后获取get你要写成key，red is key。好，再来，比如说我要把 count增加或者是减少，怎么办也很好做。Radius template点。

Ops for value点。Internet，那么我们所学的命令是incr是简写，然后它它对象里的方法是全称更直观一点。

好，然后 red SK好再来一行。那么如果是减少就是de decadent，好就演示这么几个，然后我就执行一下这个方法运行。好，那么执行完以后看一下控制台它的输出，我们看打印的是三三个获取的行为，第一个我取到的值是一没问题对吧？然后增加以后返回了新的值，变成了二对吧？减少以后新的值一没问题。然后这个时候你再看控制台，命令行 His星有这个值了。

好了，以上就是我们对string类型的数据的一个访问的方式，那么其他类型大同小异，我这里也快速的给大家演示一下，也很简单的。

那么下面我们再演示如何访问，希还是先声明一个key，test user，比如说我要存的是用户，然后我就这样哈，ID s template点ops骚哈希，然后往希中存值是put首先你要写上 k而你存的值本身也是一个它也分为ky6，你要写这个值的k你看HK就希t值的y6，类似再来一个，比如说我想存一个user name张三，好，这是存值，取值也非常简单，我就copy一下。

第二 get写成key，right？

Sk然后的话再写上value的key ID，同理这个我在获取邮政内容就可以了，写完以后执行。好，你看我最终取到的值是一张三，看命令行，kiss星太悠着，有了，没问题，好，再来。如何访问列表。

也。

是一样的，首先写个key，比如说我要存一组ID，然后我先存这个列表，从左边进或者从右边进都可以，这里我从左边进。那就read this template。点ops for list。点left toss，好，命令是l toss，这里是left toss，它就全称没有简写。

然后 Red is key，我要放一个101这个数据，同理你可以在102103放多少个都行，都是从左边往里放，放完以后我们再演示怎么去取，怎么去访问，还是瑞丽斯谈不累的点，ops for list。

然后的话，比如说size是获取当前列表中一共有多少个数据，它这个方法叫size和命令不统一，但是可能更直观一点。

好再来，比如说我还可以这样 index，然后零那意思是我要获取第零个位置的那个数据，也就是获取某一个索引所位置所对应的数据，好，再来还有 range按照范围取数据。

那么02意思是我要获取从0~2范围之内的数据，02是索引，然后还有弹出的方法，怎么让这个数据从列表中弹出去，对。也非常简单，我这样写 Ops for list点那就看你想从哪边弹了。

从左边弹就是left pop, 右边弹就是write, pop我就从左边弹了。Left pop, 然后 red SK好执行三次把它们都弹出去, 好, 这就写完了, 那么执行一下试试。好, 你看这个结果, 一开始那么统计的数据量是三没问题对吧? 然后第零个位置的数据是103, 因为左近对吧? 101先进去了最右边, 103最后进在最左边, 然后这个范围内的数据就是这样的数据。所有的数据谈的话, 我是从左边谈的, 103102101是这样顺序也没问题。

好, 那么接下来我们再接着演示集合, 老规矩还是先声明 key, 比如说我要纯teachers, 然后那么我们往集合中存数据非常简单, readiness template ops for sat, 然后的话纯数据 l的把 t写上去, 然后加上你要存的数据就随便写, 比如说刘备其实这块可以直接后面逗号加多个参数, 纯多个数据, 直接存多个多写几个, 好, 那么我写了这么几个人名, 然后这是存进去了, 怎么取怎么统计它里面的数据量什么的, 那是这样的, 把这句话call一下, 然后点size, 就是多少个数据, ready ski就可以了。

好, 再来复制一下, 比如说我要从这里弹出一个数据, 你就pop它不分左右, 随机弹出一个值。好, 再来比如说我要统计集合中一共有多少个数据, 所有的数据都是什么, 你可以这样 members, 然后ready ski非常简单, 好, 执行一下, 你看这个结果存完以后一共有5个值没问题, 对吧?

然后随机弹1个他弹的诸葛亮, 然后再去统计它里边一共有多少个人名, 就4个没有问题。

那么最后一个要演示的类型是有序的集合, test, sorted set, 还是先声明一个key, 比如说这回我要存的是学生16.4, 那么存的话就是red is template点ops for这三条 l的然后你要先写上 key, read this key, 那么有序集合它是存的时候要存分数。

第二个是存值, 比如说我要存的是唐僧这个人, 第三个要存他的分数, 就随便写了, 比如说80 80分, 这是一个, 再来一个悟空, 比如说90, 我再来八戒50, 然后再来沙僧, 比如说起始好最后一个白龙马60, 这样的话我就存了好几个人, 每个人都有一个分, 都有一个排名, 我们来看怎么去访问这些数据, 把这句话copy一下。

然后点z card, z card是统计这里边一共有多少个数据。好, 再来。10杠写出来。10杠。Scar是统计某一个人的分数, 比如说我要统计八戒, 查询八戒多少分, 好再来, 这个是比如还可以这样, rank是统计某一个人的排名, 比如说八戒的排名默认的话, 它是由小到大排序的, 如果你想由大到小排序, 你可以这样可以 reverse rank, 它就会由大到小按照倒序排列, 然后给你取一个排名, 好再来还可以这样问题。

认知是某一个排名范围内的数据, 我希望从0~2取前三名, 这样的话是由小到大取前三名, 如果你想由大到小取前三名, 也是像类似瑞沃斯 range, 它是按照由大到小倒序排名, 然后取前三名012。

好, 那么执行一下。好看。首先统计的数量是5。没错。然后统计8届的分数, 50没问题, 八戒的排名第四, 那么按照由大到小应该是悟空第一, 唐僧第二, 然后三, 第四, 八戒排第五, 但是它返回的是索引从0开始, 所以八戒就是4。没问题, 然后统计由大到小前三名, 分别是悟空、唐僧、沙僧。没问题, 对的。

好了, 当然了除了访问这5种类型以外, 那不是还有一些公共的命令可以访问 Key, 对吧? 这个也演示一下, 也有相关的API可以做到, 我直接写了比如说 radius, template。

第二我想删除一个key, delete, 我要删除这个 user, 好, 我们先确认一下这里有没有user? 有对吧? 现在我要删除它好, 然后再来我还可以这样做。

Read this template点has key, 相当于是 access命令, 判断某个key存不存在, 删除以后我就判断 key 还有没有。

好再来我们还可以针对某一个p的数据设置过期的时间我设置一下, write is template点x expire过期的, 然后你要指定一个k我就随便写了, test16.4, 我希望它在10秒, 但这个10秒的单位你要指定它默认好像是毫秒, 不管是什么, 我们可以通过一个工具类来指定叫time, unit这个类, 然后点seconds, 比方说我10它的单位是秒, 这样就清楚了。

好, 就演示这么几个, 至于说 kiss命令我们通常不会在程序里这样写, 这都是一般我们去数据库查看, 有什么东西偶尔用一下, 我们平时写代码几乎不会用到命令就不演示了。

好，我就运行以后，你看这个输出结果是false，没有 key了，确认一下 kiss星没有了，但此时还有 students，因为这个时候还没到10秒，我再来一次，随阵子没了，超过10秒了，那么常规的操作我们就演示完了，那么在刚才我演示这个代码的时候，可能有细心的同学会发现一个小问题，你比如说像这个方法，我在一套逻辑里可能是要访问 release多次，有可能每次访问的时候我都访问同一个key，每个API我都要把 key传进去，这做一件重复的事显然是比较麻烦的。

那么其实针对这样的逻辑，我们是有一定的简化的方案，我把这个方案也给大家做一个演示，这个方案也很简单，就是我们在一开始的时候，我们在创建访问瑞丽斯对象的时候，就把 t给它绑定进去，是可以产生这样一个绑定t的对象，然后我们利用这个对象就可以反复的访问同一个k如果有这种情况，你用这种方式解决可能会更舒服一点。

好，那么写个注释，这就是我们多次访问同一个k。

以绑定的形式来存在，我们再写一个新的测试方法。

这个方法我叫test bond绑定 operations绑定操作。好，其实我们将一个key绑定到一个对象上，所产生的对象就叫绑定对象，就叫棒的什么operations。

好，比如说我还是要访问 test，count key我先把 key先给它说明好，叫test。Count刚才我们访问过了，而且它里边还有一个值是一，好，那么如果说我想反复的访问它，我可以这样写，棒的，然后外流瑞森斯，如果你想访问其他类型的key，那么你可以这样棒的，比如说希 operations site of reasons，所以说中间的棒的什么office中间是你访问的数据类型的一个单词，这里我要访问的是 value of reasons。

那么这个变量我就operations，然后等于read this template点，然后棒的棒 value operations，如果是其他的类型，那就棒的什么？其他的operations。

好我把 key传进来，然后我利用这个对象去访问 release，访问就是 k访问的时候，它 API其实和之前的API是相似的，只不过不用传 key了，非常简单，比如说我要访问 com，我要将它累加，我就直接点 internet，你看我们之前访问的时候，之前累加的时候是要传入key的对吧？这样的，现在不需要了，因为提前已经绑定好了。

好我多加几次再来一次，好我又累加了5次，最终累加完以后，我再把这个数据得到再输出一下，那么获取的时候也是利用office，然后点get叫get就可以了，get的时候你也不用传输t因为已经提前绑定了，那么写完以后我来执行一下这个方法，你看最终的结果是6，那么之前我们增加过添加过这个数据，然后对它做个操作，但它最终的结果是一这次我们又给他累加了5次，最终结果是6，这是没问题的。

好，最后咱们再聊一个话题就是事物，那么因为reds它也是一个数据库，它也是支持事物的，但是它所支持的事物的机制其实不完全满足 acid4个特性，因为毕竟它不是关系型数据库，只有关系型数据库才严格满足这4个特点。

然而总体来说 release他的事务管理的是比较简单的，它的机制大概是这样的，就是说当我启用事物以后，当我再去执行一个 release命令的时候，它并不会立刻执行这个命令，而是把这个命令它会放到一个队列里，先存着，然后你在执行一个命令再放到队列里，直到你操作完了，你看你提交事务的时候，它会把队列中的命令——股脑发给 reds，服务器一起执行，它是这样一个机制。

所以这里面其实就有一个隐含的问题，你使用的时候一定要注意，那么因为他这个事物之内的命令不会立刻执行，而是提交时统一批量的执行。

所以说如果你在事物的过程中做了一个查询，这个查询不会立刻返回结果，也就是说你不要在事务中间去做查询，你要查你要么提前查，要么事务提交以后再查，所以说这一点大家一定要注意它可不像是关系数据库这一点你要注意。

那么我们在开发的时候写代码的时候，spring也是支持程式事务，还有声明式事务，当然也是声明式事务它更简单，然后只要做一些配置，然后加上存在性的注解就可以了。



但是因为它的事物刚才我说了，它有这样的一些问题存在，所以说我们通常不会用声明是事物，因为你声明是事物，它只能精确到一个方法，我要在这个方法加上存在智能，而方法内部，那么整个逻辑都是整个的事物范围，这个方法之内我就没法去查询了，对吧？

所以说通常我们都用编程式事物把事物的范围缩小，我比如说中间这三步，我需要管理事务，就把第三步写到相关的代码里，其他的地方你该怎么样就怎么样。好，所以鉴于这种情况，我只给大家演示编程声明式事务，因为用的少我们就不说了。

好写个注释，这个是编程是事物，再写一个测试方法。

这个方法我叫test转加起来，

演示的是这个事物，那么我们想处理事物其实也还好比较简单，那么也是调用red template，然后点 sq 的，那么这个方法内部需要你传一个接口的实例，我们可以做一个匿名实现ani筛选，然后叫call back session，call back接口里边带了一个方法叫sq的方法，这个方法会在 sq的方法调用的时候，它底层去自动去调的，调的时候它会把 operations就是执行命令的对象传进来，我们就利用对象来执行命令来管理事务。

当然这个方法返回一些数据，那么最终会返回给 sq的方法。所以说我们可以在这儿接受这个数据，它的类型是opx由你来定到底是什么东西。当然我得到这个值以后，我最终把它输出一下看看，而不是 return。我们看一下这个是个什么东西就行了。

好，下面我就在这个方法之内给大家演示一下这个事物，当然我首先需要提前定义一个key，我要定义一个key readiness，key这个key我叫test

tx。

TS就是casino的缩写，事物的意思，好。

然后那么我们利用这个对象有两个关键的方法，第一个方法叫做operations。

我把这个参数名写的简单一点， business点 market这句话是启用事物，启动事故以后你可以执行一些相关的操作，然后最终的return的时候你要return of recent点 exec这是提交事务。这中间我做一些简单的逻辑的处理，随便写点啥，我这样写。Of reasons there。Of ops For set，对，我要存的是集合的数据，我要往里添加数据，首先把 key写上，然后的话加一个直张三。

同理我再来，事物肯定里边是多次的点，ml操作对吧？李四王五。然后我说了我们在开启事务和提交事务之间，他是把命令放到队列里不会立刻执行，所以说你查询其实这个时候是没有效果的，咱们可以试一下。

Operations点 Ops for site，然后我要查一下它里面的数据 members， ready skit，看看里面能不能得到数据，这就完了，总之我们编程式事物就这两句话是关键，整个还有整个代码结构是一个关键，好写完以后我来执行一下这个方法。

那么大家看这个结果，首先我们在刚才这个方法内部做了一次查询，你看第一次输出就是一个查询，然后它打印出的结果是空的，确实什么都没有，所以说我们在rise管理事务的时候，中间一定不要做查询无效，注意，然后最终他确实执行了，执行完以后给我们返回了一个最终结果，这个结果你看这个结果包含的数据比较全面，首先它的三个一代表的是每一个命令，执行的时候，它影响了数据的行数或者数据的个数，每次都添加一个数据，———分别添加一个数据，没问题，然后最终他给我们显示出来集合中最终所有的数据都查出来了，张三李四王五也就没有问题。

好了，那么这次课我们就把 spring整合ready相关的这些API给大家做了演示，那么这次课我们就演示到这里，咱们下次课再见。

## 4.3 点赞

这次课我们来实现点赞的功能，那么点赞功能大家应该都非常熟悉，这是一个使用频率非常高的功能。那么如果一个大号他发了一篇文章，那么可能会吸引非常多的人给他点赞，所以说这个点赞功能我们开发的时候重点要考虑性能的问题，因为可能同时有很多人会给同一个帖子点赞，你要考虑到性能的问题。

因此像点赞这样产生的数据，我们最好把它存到 rise里，因为你把数据存到内存里性能会好很多，这是我们这个功能大概要注意的点，就是把数据存到readiness里来提升性能。

那么从这个功能上来说，我们的项目当中有两个地方可以支持点赞，一个是在这个帖子上你可以对帖子进行点赞，另外对于帖子的评论你也可以点赞，那么我们在点赞的时候，那么我点一下，如果他没有赞之前我没赞过，我点一下，那就是赞了，我已经赞了以后，我再点一下就把赞取消，是这样一个规则。

那么当我点完赞以后，功能开发完以后，我们还要做一些显示，那么首页上咱们不是有一个帖子列表，每一个帖子他一共被点了多少个赞，我们需要做一个显示。

另外我们帖子详情页面，然后也要统计帖子，他一共点了多少个赞，另外每一个评论有多少个赞，就是关于赞的数量的显示。

好了，那么接下来我们就来实现这个点赞的功能，那么我打开我的开发工具，然后按照我们之前长期开发养成的一个习惯，我们首先应该去写数据访问层，但是现在我们是要把数据存到 rides里，而存到 rides里，其实就好像我们操作一个map一样，非常简单对吧？

然后因此数据访问层我们就不单独去写了，因为它太简单了，我们直接写业务层，那么在处理业务的过程中，那么你需要存数据直接调 ready sample的往里存就可以了，当然取也是同样的。

好，然后那么我们在像ready存数据取数据操作数据的过程中，那么它是以 k为关键的，我们是面向它的 k去编程的，所以为了能够让 t反复的复用，然后的话比较容易记，那么我最好给 readiness写一个工具，专门的用来生成它 key，这样好复用，所以首先我先去写生成 red SK的工具，那么我在YouTube包下新建一个类，这个类我给它取名了叫。

ready tuto。

就是一个简单的工具，这个不需要由容器来管理，我们直接提供静态方法去访问就可以了。

然后我们的key都是中间是用冒号隔开的对吧？然后是由一个单词一个单词拼成的，那么我们在实际开发的时候，那么可能有些单词固定的，可能有些这个单词是你需要传入的，是动态的。

好，那么这样我首先先说明一个常量，就是用来拼 k的一部分的常量，时代的一个final，然后实际上我讲。

是贝类等于冒号。

那么我们所有的key在拼的时候，中间以冒号来分割各种单词，就用这个常量就可以了。

好，然后现在我们要准备要存这个帖子或者是评论的赞对吧？

那帖子评论我们可以一起存，我们管帖子和评论我们统称为实体，我们要存实体的赞，我希望实体的赞他以某一个前缀来开头 t我也说明一个常量，也是stream，然后我叫这是一个前缀，后面还需要加上具体的帖子，可能加上ID什么的，这个是前缀，所以我叫price下划线。

anti。

然后。

like。

实体的赞， like就喜欢赞，然后我希望是以这样的前缀 like。

冒号。

and当然后面你可以再加冒号加其他的更明确的条件，反正这个是前缀。

好，有了条件以后，接下来我再写一个静态方法，然后要求你传入一些变量，我把它拼到长江上面去，最终得到一个完整的k好。这里我写个注释，我要生成的是某个实体的站，这样一个k某个实体的。

站。

因为它需要被外界访问，所以这个得是公有的，然后是静态的，然后返回的是一个kiss菌方法名，我叫get。

Entity like he。

好，那么像这样一个t实体，你肯定得传入实体的相关的关键信息，而实体的关键信息不就是 tap和ID对吧？所以你应该把这个int太传进来。

ittid传进来，实体类型和实体的ID传进来，传进来以后我们把这个前缀以及这两个变量拼到一起，就形成了这样一个实体的key，然后这里我再说一下，我们以 key来存的时候我们怎么存呢？

或者说 key将来长成啥样，它是这样的，let安贴贴后面的话拼上 NTT太太，然后再拼上nttid最终拼成这样的一个形式，然后 k它所对应的值我用一个集合，而不是简单的一个整数。

当然我们存一个整数可以，你点了一次赞，整数就是一，你再点赞就是加一是可以的。

但是那么如果将来我们的需求发生变化，比如说我想看到到底谁给我占了，对吧？你这个时候存一个数字，可能这个需求就实现不了了。我这里我存的值用的是 site集合，而集合中我装的是什么呢？ User ID，谁给实体点的赞，我就把这个用户的ID存到集合里。那么当我们需要统计站的数量时，这个集合它会有统计数量的方法，对吧？然后如果将来我需要知道到底谁给我点了赞，那么这些每个用户ID我也能得到，所以我们这样存的话，那么能够最大程度地满足各种各样需求的变化，这个比较合适。

好了，既然我们了解了存的形式，那么这个地方我就这样写了，return我要return的是他，然后再加上失败类的，没有错了，加上冒号，然后再加上NTT type加上是贝类的，再加上安贴贴ID，那么我这样的方式不就是拼成了我想要的这个样子，对吧？

好了，那么我 key准备好了，准备 key的过程其实也是我们梳理我们要往release中如何存数据的这么一个过程。

那么梳理好以后，下面我们就可以开发我们业务组件了，那么这是一个新的业务，所以说我需要新建一个service，那么我在包下新建一个类叫lack。

service站，当然我需要首先加上servis注解。

除此以外我们要把数据存到readiness里，所以说你需要注入 readiness template。

好。

那么接下来我就来实现了这个点赞的这个方法业务方法，这个方法public不需要返回值，只要不报错，反正就是点赞成功了，然后这个方法名我就叫那么需要几个参数，首先我加上uzid是谁点的赞，其次我加上 Entity tap点赞的实体是谁安全的ID也是实体的一部分，方法说明完以后实现其实比较容易，首先我们需要拼出来，我要存的数据的key，反正你要点赞，我要往 ready这里存，我要拼上 key的名字，我叫安贴。

贴赖

t nttlkt然后等于red is key utu点get NTT like t那么你需要把NTT type ittid传入，那么我们会得到这样一个k然后有了 k以后你注意我刚刚开始说的时候说我们第一次点赞，那是如果我再次点赞，就是取消这个赞对吧？

所以你还得判断一下我当前用户点没点过赞对吧？那么如果点过了我要取消，如果没点过我要点就是我要存这个数据，具体怎么判断？因为我们最终存的是集合里，存的是uzid所以你判断一下 Uzid在不在集合里，就知道你点没点过赞了，对吧？这个很容易能够想到，所以我首先判断一下它在不在集合里，那么我就 ready stand play的点ops form sad for sad，然后可以调这个意思，it's number这个方法。

然后当然我需要传入 key NTT like key，然后传入你要判断的是哪一个值，是优质ID。好，那么这句话执行完以后我会得到一个布尔值，我先把它记录下来。布尔值就是is number。

好。

然后我判断一下。

如果说。

返回数，那就证明说我已经点过赞了，这个时候我就需要取消赞，那就是把uzid从集合中删除掉对吧？那也很容易 Ready，template。点ops for sat，然后点remove。Remove，然后 t是on TT来k删除的是uzid行了。

然后。

否则那就是uzid不在这个集合里，就是我没点过赞的，我要添加这个数据。

好。点爱的，然后把 t写上去，然后 uzid存进去，好了，这个就可以了。

好，这个点赞方法我就实现完了，但是你还要注意我们不只是要点赞，后续还要统计点赞的数量，并且我们在详情页上不只是要统计数量，我们还要显示点赞的状态，你是你没有赞这里就显示赞这个字，如果你已经赞过了，这里要显示已赞，所以说你要根据点赞状态，你是否点过赞来显示这个字，因此我们还需要把统计数量统计点赞的状态这个方法也写出来，好，为后面的功能做一个准备，这里我就一起写了。

那么我再增加一个方法就是查询实体点赞的数量，查询了某实体点赞的数量。

那么。

这个方法就返回的是浪。然后这个方法名儿我叫find an ticket like come。

好，因为你查的是某一个实体的数量，所以说你肯定是要把实体这两个值传进来。

好，那么传进来以后我们无非就是看 T对应的值，里边有几个u在ID有就是几个数量，所以也需要用 key把这个代码粘过来，粘过来以后我就直接return就行了，return read this template，点UPS，fall，sad。

然后统计数量 size，然后你把 k传进去，我们就能够统计到 k所对应的集合里边的数据的数量，这非常简单了。好，除了这个以外还要统计实体的点赞的状态，某个人给实体点没点过赞，就查询某人对某实体的点赞状态有没有点过赞。好，那么这个其实从我们的需求上来说，我们返回一个布尔值就可以了，对吧？数据点过放似就没点过，但是这里我返回的是整数，为什么呢？是因为将来还有可能，比如说将来业务扩展了，我们可能还不只是点赞，还可以点个彩，你布尔值就表现不出第三种状态，而我用整数的话，这三种状态就都能表现，对吧？

所以它更具备扩展性，我返回整数。

好，那么这个方法名儿我叫find anti like status我要查询的是点赞状态，是某人得把这个人要传进来，又做ID，然后某实体安贴的相关的条件你要传进来。好，那么传进来以后你判断是不是点过赞也很容易，其实就调刚才这个is number就可以了，对吧？当然首先你得有 Key，把 key copy以后我就 return read。

this。

template ok as for the foresight, 然后 is its member, 然后的话是 key Entity, leg, key, 然后 use ID当然 is number返回布尔值, 那么如果它返回true, 我就把它转成整数, 显示为一, 否则就是零。

也就是说如果我返回一表示说你点了赞, 返回0表示说你没点赞, 如果将来我们要开发点彩, 你可以返回-1, 这样就很清楚好了, 目前这个业务方法就这么几个方法我们就实现了, 实现以后接下来我们就来处理表现层, 表现层我们这个点赞是在帖子详情页面去点, 点的时候这个是异步请求, 整个页面不刷新, 然后点完以后根据返回的条件, 你动态的改变一下, 这个字就可以了, 或者改变一下赞的数量就可以了。

不用整体刷新, 页面效率高一点。

好这样我再新建一个来CTRL, 好, 然后这里边我要把莱克斯维斯注入进来, 因为我要调用这个点赞的业务对吧?

好把like service注入进来, 然后谁点赞是当前的用户点赞, 为了得到当前的用户, 我把 House偷的注入进来, 好注入这两个变, 然后我需要写一个能够处理义务请求的方法, 先说明这个这个方法的路径, 我就叫like, 然后 MR的等于post, 因为你点赞你要传入一些信息进来, 所以这个是post。

好, 然后因为是异步请求, 所以我需要再加上另外的注解 response, response body, 那么说明这个方法我返回死卷, 然后名字就叫当你点赞的时候, 你是针对某一个实体点的, 所以你肯定得把实体传进来, 那就是因 mtt太以及 NTT ID。

好了, 那么因为是当前的用户进行一个点赞, 所以我首先先获取当前用户。

好, 那么取到以后我们要点赞, 你就调这个like service, 当然有人说你是不是要判断一下用户, 他有没有登录, 其实我这里不判断也行, 那么我们不是用拦截器可以去判断某一个功能, 他有没有登录, 登录以后才能访问, 我们不是做了这样的事情, 所以您可以通过拦截一下, 不登录不让访问就可以了, 这个你自己去实现。

再一个那么将来其实我们会采用 spring security对于这样的方法进行一个重构, 对这样的对拦截器进行重构, 然后的话用 security安全模块统一管理, 这样的权限问题。所以说我先放我这里, 我就不用去判断这个变量它有没有值了。我们之前会有一个拦截。好, 那么下面我就要实现点赞了。

那。

就调 like service, 点like方法, 你把这个参数传进去就行了, 这就点赞了, 那么点赞完以后你要统计这个点赞的数量以及点赞的状态, 把它返回到页面上, 根据你的反馈值好去做数量和状态的一些显示。

那么这里我写上就是数量, 先获取数量是个long, 然后 Like count。

等于like service点,

find。

Count然后把这两个条件都传进去, 好, 这个数量就有了, 那么再来除了数量以外, 还有这个状态。

对。

状态是一个整数, 我叫like斯德尔斯, 那么我需要传输优质ID, 我需要传入就uzid得这样讲, 悠着点get ID然后 idtcmidt ID。

好。

那么最终我需要把这两个值把它统一的传给页面, 我需要把它封装一下, 传这样零碎的不好传, 我用 map封装一下。

然后 new。

希map就 map点put, 一个叫like count, 然后再来 map put like status。好, Map写注释。

这是返回的结果，现在点赞也点了，结果也有了，最终我就返回一个阶层格式的数据就好了，对吧？我就 community YouTube点get Jason string调这个方法，然后正确返回0提示我，就返回now，然后把 Map数据返回给页面。

好了，这个CTRL咱们也写完了，写完以后接下来就是去处理页面的逻辑了，而这个点赞我们刚才也说了，是在帖子详情页面去点的，所以我们需要打开帖子详情页面进行一个处理。

好，我搜一下，帖子详情页面是disc os，detail点html，然后我从头往下找一找这个页面比较长，咱们慢慢看，但实际上这个页面上有三个地方能点赞，一个是对帖子点赞，一个是对那两种类型的评论进行点赞，一共有三个地方有点赞，我们一点点找，好，那么很快我就找到了这个地方，赞这个地方是对帖子进行点赞，然后这里边我得处理一下超链接，我点了一个超链接的时候去访问服务器，那么因为是异步请求，所以我们需要提供一个GS方法来实现。

就点赞的提交请求的逻辑，这个地方我这样写，我把 hres井号改成java secret，冒号分号，比方说超链接，我不把它当做超链接来看，这里不是路径，这里我需要在超链接上写js，但是 js这里目前什么都没有。空的，好然后怎么办？我可以这样。

可立克我给它定一个单机事件，你点超链接超链接废了，但是这单机事件会触发的，说白了我就是把这个超链接当成一个按钮来用了，然后安克利克里边我要调一个GS方法，然后去发请求，那么这个方法当中因为我需要给他传一些条件进去，因此需要拼一些变量，所以说我在安克利克之前加上th为了拼变量，然后这个方法的名字是静态的，我就加双竖线方法名我叫烂，然后括号，然后以分号结束，这是整个方法的一个基本结构，那么这个方法我需要传几个参，首先我传个例子，类似代表的是当前你点这个节点就当前的超链接，为什么要传它？

因为我们整个页面上三有三个站点，这三个站中的任何一个他调的方法都是我们肯定不能写三个方法对吧？是复用同一个方法，我在这个方法里怎么区分你点的到底是哪个站，我是要区分的，因为你点了哪个站，我将来要改这个站后面的这个数对吧？

所以我得知道我把this传进去，就是为了知道当前你点的到底是哪个a这个意思。好，然后我们点赞你得传NTT tap nttid对吧？你当前是对帖子点赞，这个tap就是一逗号，mttid那就是帖子的ID对吧？那帖子ID这里写谁，你看这里取帖子是post，对吧？那去帖子ID我们就知道了，应该是这样写，post点ID。好这就可以了，我们一会要做的事情无非就是把这个like方法写一下就好了。

好，当然这里还得再稍微处理一下，就是说我调了这个方法，我向服务器发了请求，那么点了赞以后服务器会给我返回，数量我得改变一下这个值，他会给我返回状态，我得改变一下这个字对吧？

如果返回的状态是点了赞，这个字要变成已赞，返回的状态是没有赞，这个地方就只是显示赞，所以说为了便于说改这两个东西，我最好给这两个东西外面套上一个标签这样我好访问它，不然就不好访问了是吧？

这样这块有点长，我把它折个行，这样方便一点，然后赞11我也折个行，因为我需要加标签，我在账的外面再加一个b对不对？然后在11的外面我加一个a那么把它俩做一个区分，用标签我得到 A以后就很容易得到a的子节点b和l好，对他们加以修改。

好了，这是第一个站，我已经把这个单机事件的处理好了，另外的两个站你找到以后，那么处理方式是类似的。好，我往下找一下。

第二个站应该是在回帖列表当中有，然后第三个站是在回复里，这儿好这是第二个我把它处理一下，处理的方式也类似，这样我也换个行吧，那么我也是那样想知道吧？Secret冒号分号，然后这里边也要写上th，冒号 on，click等于然后 like括号，括号结束当前当前的节点传入this，当前我是对评论点赞它的类型是二对吧？

然后第三个参数是应该是评论的ID，那么评论数据怎么获得？你看前面参考一下 Cvo.comment对吧？你能够获取到创新时间，ID就是类似的就是点ID就可以了。好，然后赞这个字和一这个数我们也需要改，我在它外面也是套上b和l执行相同的逻辑，

把一外面套个a好，这是第二个赞。最后一个在这儿也做一个换行处理好，我就不啰嗦了，按照刚才的方式去做。

那么这个地方我们取的是回复的ID，而这个回复你看前面rvo点replay这不就是回复对吧？l他的苦于他们这样去idea。

那就是点ID。

就是replay，点ID就可以了。同理这地方也是做一样的处理，套上一个b套上一个a好了，那么三这三个超链接我都处理完了，处理完以后我需要写这个like方法。

我看一下我当前的页面它依赖了哪些js文件，我在哪里写比较合适，它只依赖了 global点js，Globo是全局的，任何网页都引用的，我们在这里也不太合适，所以我们就单独创建一个GS，copy一下，那么我创建一个新的js文件，因为当前是帖子详情页面 GS。

我叫。

discourse点GS好，那么接下来我就在static GS目录下新建一个js文件， discuss点js，那么在这个文件里我就需要写 like方法，这个方法很好写，function like。按照刚才我们所写，我们需要传入三个参，第一个参代表超链接，但是超链接我是把它当成按钮来用了，所以说我给它取名叫button按钮。

那么第二个参我们传进来的是实体的ID，安不是实体的类型，NTT太对吧？第三个是nttid那么都传进来以后，我们要向服务器发送一个异步请求，对这个数据加以处理，那就到了点，这是一个post，请求我要提交数据的，那么首先我需要写上路径康txt，加上这是我们这个功能的访问路径。

然后第二个是携带参数，首先携带intet type，好，然后再携带 mttid。

好，那么接下来我们处理返回的数据，data，对于返回的这个字符串，我们需要把它转成GS对象，到了点儿pass Jason data。

那么有了对象以后，我们做一个判断，如果data点儿扣的等于0，就比方说我这次请求是成功的。

否则那。

就是失败了，失败了我就直接弹出一个提示就完了。

那就right。

data点message。

当然我们失败的是统一处理的，我们通过控制器通知统一处理的成功的话，我们需要怎么做？我们需要改变，找一下站，我们需要改变站的这里面的数量，我们还要改这个字，是占还是已占？

好，你看我们不是传进来了当前你点的按钮吗？那么我就可以通过这个按钮得到它下级的 B标签和这个l标签它的子节点，然后去改变它的值是吧？我先改这个数量，我就到了8天，我先获取8天节点，然后获取它的子节点，children它的子节点的标签名是l显示数量的，然后我想在 l里改变它的值，改变它的内容点text，这个内容是什么？是我们data返回的数据，Data点like count就可以了是吧？

另外还要改那个状态，那也是类似的。8点点丘疹显示那个字的标签是b B的内容，text b的内容我们得看一下data点like status。如果like status—如果它等于一，比方说我赞成功了，我赞成功了，我这里要显示什么？我就要显示已赞，否则它不等于一，那就是没成功怎么办？我就显示。站就你可以去站好了，Js文件我也写完了，到这我们可以测一下，看一下我们到那个页面上点能不能起作用，我把这个项目重新启动一下。

好，服务器启动完以后咱们来测一下。

Look, host, 先访问首页，当然我这里需要登录一下，登录以后当然我们现在首页上赞的数量都是假的，我随便找一个帖子，比如说就玄学帖，这个也是假的，我先不管他，先我点一下，点赞以后他成功了，显示成已赞了对吧？

然后再点一下，再点一下以后就取消了，取消以后赞的数量都变成0了，我再点一下已赞。

好，那么比如说我再点一个评论，别点评论，这个没有反应，我们看一下是服务端报错了，还是说前端报错了，我们看一下。

首先服务端没报错，我们看一下前端的逻辑，看看页面上有没有报错，页面上报错了，那么它提示我，你看它有提示说cvo is not defined，cvo未定义可能是我调那个方法是条件写错了，我看一下 cvo这个是rvo前面这个是 CPU确实写错了，因为你这里取的是一个变量对吧？你得加上dollar，对吧？不然这样写的不对 Dollar，这rl是不是也写错了，看一下这也是写错了。

好忘记了，我重新编译一下，编译以后再打开页面，我刷新一下，刷新以后这个状态显示的就不对了，因为什么？我们只是点完以后动态的改它的值，但是我们访问详情页面默认它它就是显示一个固定的值，我们只要没点没触发它，这个值就不变，所以这块显示上还有问题，我们一会再处理。

然后这里我再点一下一个赞，点赞又变成已赞了，点赞表明可以了，那么数据对不对？我们也可以去Reads里看一下。

然后。

我先看一下kiss，我们那个点赞的key是以 like开头的，你看有了这么几个数据，包括你看 e开头的就是帖子对吧？二开头的这是这个评论没有问题，我们不是一共点了一个帖子的赞，我们一共点了三个评论的赞，对吧？

一共4条数据。没错好了，那么这个点赞实现了，现在欠缺的是首页上我赞的数量显示的不对是吧？然后帖子打开的时候，那我们查看的时候，数量和站的状态显示的不对，这里我们需要对这个显示进行处理。

首先我们先处理首页站的数量的显示，这样我要打开 home controller，因为我是在这里，显示的是这个帖子列表的，那么我们在查询帖子的过程中，我们可以在查询的过程中去查一下这个帖子有多少个赞。

我在这儿补充一个逻辑，在这儿加上查询站的数量，我需要把 like service注进来好，那么注入以后这个地方要查了long like。

count等于like。

service点。Find Entity。Like count。然后那么我们查的是帖子的账，所以这个tap应该是一，但我别写死一，我还是引用常量。

好，这个地方我就可以写成on TT type post对吧？第二个是实体ID，帖子ID帖子ID，帖子ID是多少？就是post点get ID对吧？赞的数量就有了以后我把它装到 Map里，页面上就能得到了是吧？就map点put like counter。好，那么完成以后这就可以了，我们打开首页，那么找到显示站的地方，然后把它处理一下，搜一下。Index.html。

好，我找一下，这是在帖子列表里，这儿回帖数量我们已经显示了，站的数量我们需要做处理，我希望在11外面套上一个标签，套上一个死蛋，然后死伴之上我要获取数据，那就th冒号，然后是txt等于一个变量，这个变量我从map里取对吧？Map。

第二let好。

写。

完以后编译，然后打开浏览器试一下，刷新一下你看这回所有的帖子它占的数量都是0，只有赞的数量是一，这个帖子不就我们给了一个赞对吧？这就对了。

好，最后还要处理一下帖子详情页面，它显示赞的数量以及状态，我们需要对加以处理，而帖子详情页面我们是在discuss post control里做的请求的处理。Detail请求对吧？好，这里因为你要查的是站的相关的消息，我还是要把max注入进来。好，我看一下它有没有实现接口实现了。好，我回到 Detail这个方法对它进行处理。那么首先我们要查这个帖子的赞，这里我查到了帖子的作者，帖子的作者，后面再



补充一个就是赞点赞有关的信息。

首先是来一个数量like。

count。

等于 like service点, find Entity like account。那么帖子实体是一, 然后 Id是帖子ID, 参数上有。好, 那么这个值我要把它传给页面, 我就model点ID like count。

好。

那么不只是有点赞数量, 还得有这个状态对吧? 我分开写点赞数量, 点赞状态就是。

我当前用户是否对这个帖子点过赞, 那我们得到的是一个整数。

我。

调用like service Find like find on the next datas去查询, 那么查询的时候我需要先传入用户的ID是当前的用户的ID, 当前用户我这里有没有house头的?

有。那就house偷点get use点刚才的, 然后帖子它的类型是, 然后帖子ID是好, 同样我需要把这个状态发给页面猫, 都有点艾特 like status, 那么这个地方我们在实现的时候还要考虑这样一点就是说这个帖子和帖子详情页面, 其实如果用户不登录他也能够访问, 如果没有登录的时候, 你是不是这个值就取不到, 这值就会报控制人异常, 你这样写。

所以说这个地方还要再做一层判断, 如果你没有登录, 那么点赞状态就返回0, 就比方说你没赞过, 你都登录都没登录, 肯定也就相当于你没赞过对吧? 你没有身份, 否则我们再去查询, 所以这块我们得稍微再判断一下, 我再处理一下。Household点get user, 如果它等于now表示说它就没登录, 我就直接返回0了, 否则我再去做这样一个查询, 好。这个地方做这样一个处理。严谨一点。

这是对于帖子的点赞的相关的数据的查询, 那么对于这个评论也需要做类似的处理, 所以我把这个代码拷贝了一下。然后看这是有评论的作者, 我们需要对评论增加这样的数据, 我在这做一个补充。好, 那么点赞数量复用这个变量, 所以说前面就不用声明这个类型了, like count, 然后等于like service点find的时候, 这个时候你是查评论, 是吧? 这个地方就不要写成post而写成comment, 然后你这里要传的是评论的ID, 评论的ID你可以从 comment里取对吧?

要get ID。好, 那么这个数量我要把它不是直接用猫的发送, 而是把它放到存到 vo comment vo里, 这地方要处理 comment vo, 然后点 put。

like count。

这是点赞的数量, 那么点赞的状态是类似的, 首先也是做一个判断, 那么没有登录返回0, 那么否则我就登录了我就去查, 那么查的时候又在ID没问题, 然后这个是评论的类型, 然后应该是评论的ID, 那么同样的这个状态, 我们是要把它装到vo里, comment will点put好了, 这是关于评论的点赞的信息, 那么还有回复, 回复的处理也就是类似的。

好, 那么我们在这查到了回复的数据作者以及回复的目标, 对吧? 那么后面再补充一个关于点赞就可以了。好, 那么我们再过一遍, 看看有什么地方需要改动。首先点赞数量 Find这个类型是 comment没有问题, 然后 ID是回复的, ID而回复是replay对吧? 是replay, 运费点开台d然后我们把 like count要放到 replay video里, 对吧? 不是comment vo了, 名字没问题。

好, 再看到点赞的状态, 也是做这样的判断, 然后这样的查询类型是comment, 然后 ID应该是 replay点get ID, 好, 那么这个状态也是放到replay video当中去是吧?

好了, 那么到这儿我们关于这三三三个地方, 这个点赞的数据就查到了, 都把它放到了相应的对象里头去。

那么最后我们再回到帖子详情页面，然后的话我们找到那三个点赞的地方，然后把它的默认就打开页面默认显示的数据给它设置好就可以了，我打开帖子详情页面，我们从头开始过一遍就那三个地方。

首先是帖子，这儿这个地方先看这个数量，先显示数量，数量你要显示的就是我们刚才CTRL的时候查的来个com变量，th冒号text等于你看我怎么返回的这个地方，我就直接存的是like count，是吧？所以你就写。

like come就行了。

好，然后关于赞，我们也需要做一个处理，th冒号，text等于我需要看状态。

like status。

它是否等于一，如果它等于一，我这里要显示。

你在对吧？

否则我就只显示在。

好了，那么。

点赞的数量和这个状态我就显示好了，那么另外两个位置它的处理方式和雷同了，我们再找一下第二个，这是第二个。

然后下面三第二个，上面我要写th冒号text。

等于我记得刚才我写的是我把 like com存到了vo里，而这个vo就是 cvo对吧？Cvo like count。好，那么赞的状态也是类似 th冒号 text。那么等于CV o点like。standards它等于一吗？等于一，显示已占。否则显示在好。这是第二个。

那么第三个在这儿处理方式和刚才是一样的，首先我们还是先处理数量，pH冒号，txt等于 Like count，我是从 replay队伍里取，从 rvo里取的对吧？

Rvo rvo点like com，

然后点赞的状态 th text等于 rvo like。Status一吗？等于一显示已在，否则显示在。

好了。

那么到这儿，这三个就是点赞的地方，我们把它打开页面以后的初始的值和状态也做了一个设置，下面我们重新编译，然后访问一下这个页面，看看有没有问题，好，我刷新一下首页，我当前是以CC的身份登录的，然后就这个帖子有赞我就点进去，已赞对吧？没有问题。

好这样我把它退出一下，退出以后点首页还点帖子，那么数量还是111没问题，但是因为我当前没有身份，所以这个状态显示的就是赞，而不是已赞，说明状态数量都是ok的。好了，那么这次课我们就把这个点赞的功能给它实现了，那么这个功能它的核心是我们对 readiness它的使用，巩固一下对release的使用。那么这次课我们就演示到这里，咱们下次课再见。

## 4.4 我收到的赞

31-我收到的赞.mp4

这次课我们来实现我收到的赞这样一个功能，就是我们在个人主页上要显示这个人他所获得的所有的赞的数量。当然我们现在通过点赞的数据是可以把这个数字统计出来的，不过你有点麻烦，那么因为我们现在的是显示某个人他所获得所有的赞，你要统计这个人发过多少个帖子，发过多少个评论，那么把这些帖子和评论的赞的数量把它加起来，这样会很麻烦。

所以我们一般采用另外一种方式，就是我们在用户点赞的时候，那么可以再增加一个维度把站的记录下来，我们可以再加一份数据，我们以用户为key，然后把这个数记一下，这个时候统计就非常方便了。

所以那么要想按照这个思路来解决问题，首先我们需要把之前我们所开发的点赞功能进行一个简单的重构，就是我们点赞的时候要以用户为t把赞的数量再记录一下。

当然每次记录这个只是数量而已，所以说我们只需要调用 in cream的方法加一就可以，当然如果你取消这样的就different减一就行，非常简单。

那么这件事完成以后重构完以后，那么我们就是来开发个人主页，然后个人主页上要显示用户的一些信息，然后要显示赞的数量，目前我们关注还没有开发，所以说关注我们先不用显示好了，那么大概了解这个思路以后，咱们现在就开始写这个程序，那么因为我们要在rids里承继一份数据，所以首先我打开ridsqq然后我再增加一个k加一个前缀，这个是以user为key，所以说我叫practice user来好这样一个前缀，然后我再提供一个方法，写个注释，这是某一个用户的赞。

Get user like t因为你是某一个用户的赞，所以说很显然你应该是传入的是uz ID对吧？当然我这样我在注释的把 t它的形式把它写得直观一点，是like user，然后的话是user ID，那么我们存的时候就直接存一个数，整数就可以好，然后我就retrain，然后这个前缀，然后再加上一个分割符是贝类，然后再加上优质ID，key就ok了，拼好了拼好以后，下面我们就来改造点赞的业务方法，like service，我搜一下 like service。

好，就这个方法。

那么这个方法因为我们需要再加一个维度加再加一个维度记录数量，那么就是我们这一个业务当中会连续执行两次更新的操作，所以整个业务应该保证事物性。

那么rise如何保证事物，咱们之前不也讲了，我们可以通过编程来解决问题，所以说这个代码我需要做一个重构，这样我就干脆把它注掉，因为这个代码结构差的比较多，把它注掉，然后的话重新写的时候参考一下之前的代码就可以了。

好，我就写ready template is cute，然后new session call back。因为我们当前的方法没有反馈值，所以说我这里就不处理返回了，这个参数名改的简单一点 vs。好了，然后在这个方法内部我们实现我们要实现的逻辑，首先我要把这两个k拼好，是吧？一个是以实体为k再一个是以you的问题把 K也先拼好。Ready? Qq点get user let t这个时候我就需要传入uzid但是当前我们这个不是 user ID，这个是点赞的那个人，而我这个是被赞的那个人的user，说白了应该是实体的拥有者，实体的作者，user。

当然了我们一个是可以通通过实体去查，但这个查又麻烦了，因为你得判断实体的类型，它是帖子还是评论，然后去查询。

另外你这里查的话又得访问数据库，本身我们这个ready是操作是性能比较高的，你一访问数据库它又把它的性能给拉低了，所以这样我就不查了，我要求你调这个方法的时候，再传一个参数进来，你把实体的作者传进来，加个参数mtt user ID。

那么我们在点赞的时候是在帖子详情页面上你点赞的时候，实体的uzid是很容易得到的，你传过来就很方便，加了这个参数以后，所以这个地方我get key。传入的就是 inttuzid好，t有了以后，下面的逻辑还得做这样一个判断，做这样一个判断。因为你要知道当前他点没点过赞，好进行下一步的处理。

我还是像之前的那样的写法，只不过这回是叫operations operations for site然后 is number把安贴贴来，key传入把 user ID传入，我们看一下当前用户有没有对实体点过赞，然后这块要注意的是你这是一个查询，你这个查询一定要放在事物的过程之外，因为之前我讲是不是讲过这个readiness事物比较特殊，如果你把查询写在事务范围之内，他不会立刻得到结果，因为他在事务过程当中，你执行的所有命令他没有立刻执行，而是把这些命令放到了队列里，当你提交事务时他统一提交，暂时性他比较特殊，好，所以提前先查询了，然后我们开启事务就是operations Martin对吧？

最后你执行事务，那么就是operations点exec那么在中间我们要执行两次修改的操作，当然我得先判断了is number，如果他已经站了我就要取消站，就operation死掉了。

Ops for site点取消删除数据，remove安贴贴来，然后又是ID。

好，另外你这是对实体为t的数据做处理，同时以优则为k也要做类似的处理ops放这个我们存的是普通的死菌，所以放y6，然后这个不是删，它是减一，数量 decrement减1，这个key是user like key，否则就说明没点过赞，我要点赞，点赞就是把这个数据加进去，所以call一下。

第一个还是实体为key，然后的话把u的ID加进去。第二个当然这个加得是爱它，第二个是以user为t t没问题，然后是增加所以是equipment。

好了，到这儿我们点赞的方法就重构好了，之前的这段代码不要了，处理完以后我们在 service里面再补充一个方法，因为你最终是要统计用户他被占的数量对吧？现在你是记了，你还得提供一个方法去查对不对？所以我在后面再追加一个方法，就是查询某个用户获得的赞的数量，我们直接返回一个整数就可以了。方法没有叫find you there like come。

好，那么参数就是uzid你把uzid传输就可以。首先我还是要拼 Key，这里我可以copy一下，这是uz ID拼k推完 t以后我就调用red established点ops for value，对吧？然后统计数量这是一个字符串，你查询的话就直接get，然后把 t传进去，那么我们会得到一个结果。当然它默认得到的是一个opat我们需要把它转为这个integer，这里需要转成integer。

好，转完以后最后返回数据返回的时候做个判断，因为有可能这个数据是now如果是我就显示为0，否则就显示数据的整数形式就可以了。那么service重构完以后，我们来重构表现层 CTRL以及页面的处理逻辑，我打开 like control了。

好，找到站的方法他已经报错了，那是因为我加了个参数，现在这个参数还没有对吧？好，我需要给这个点赞补充一个参数，目前这个方法还没有接受这个参数。我就加上一个int entertain，user ID。

好，当页面的把参数传进来，那么我就把它传给 Like方法这个点赞就搞定了，搞定以后这就可以了，那么但是我们需要对页面做处理，因为这个参数页面得传过来，现在还没有传对吧？那么我们点赞是在帖子详情页面点的对吧？

所以我们需要修改帖子详情页面，那么我搜一下详情页面的是disk，Cass，detail，html。

好，我记得一共有三个点赞的位置，然后调了三次那个方法，我们逐一给他做一个处理。好，首先这个地方是对帖子点赞，like，那么这里边我就需要给它加一个参数，你看当前是传入了实体类型，还有实体ID，我们需要补充的是实体的 user ID，那么实体这个post就是实体就是帖子对吧？那么user ID就很容易取，就是post有这种ID就行了。

好，这是第一处，那么我们再往下找，还有两处再往下看，这应该是第二个位置。Cvo对评论，然后后面是回复，没问题，这也需要加一个参数，这个是评论的优质ID。Cvo.comment是评论对吧？所以我就可以写cvo.comment，user ID。好，然后再看最后还有一个回复，在这儿再加一个参数，那就是rvo replay点user ID。

好了，那么你看这三个点赞的地方，我都把这个参数补充完了，但是你别忘了我们调的 like方法，也得支持传参数对吧？所以还得去重构这个方法，我们就打开对应的 GS我记得 GS是discs点GS，我给这个方法再加一个参数，叫做itt user ID。好，那么当你把参数传入以后，我需要传给服务端，我在这里加一个参数。

好了，那么其他的逻辑不变，主要是追加了这么一个参数，那么点赞的时候，我们就通过这个参数把用户把用户的他所得到的占的总的数量做了一个累加，当然最终我们是要在用户的主页上对这个数据做一个显示非常简单，因为那个页面的很数据很少，我们就把显示一起开发完以后我们再统一测试，那么对于个人主页它是属于用户的某一个功能，所以我不新建CTRL了，我把这个功能起到user controller之内，user controller里我在这里再追加一个方法，个人主页，好，那么这个方法名方法路径我叫做主页一般叫做profile，然后注意个人主页，一个是当前显示的是当前登录用户的主页，再一个你也可以查看别人的主页，因为你想我们在帖子列表上，每个帖子墙上不是有个用户头像，那个是可以点的，用户的详情里边的每一个评论也有头像，你私信也有头像，这些有头像的地方头像都能点到那个人的主页里，你是可以查看别人的主页的。所以个人主页不只是查看当前用户的主页，也是显示就是任意用户的主页，谁都可以访问的。

所以个人主页我希望你能传过来，你要看的那个人的ID，而不是当前用户，所以我希望这样我希望把用户ID拼到这个路径里，然后就叫优质ID，因为这是一个查询，所以说请求的方式是get好，然后把这个方法省略好，get profile配置。

首先我需要声明帕斯维尔堡注解，然后把 uzid得到从路径里解析出来。另外为了给页面的携带参数，我需要把 model加上去。好，下面我们首先把你要访问的用户查出来，这个简单了放 Id，优质ID为了避免有人恶意攻击，他特意传一个错的数据进来，然后反复去查攻击你，我们判断一下你 uxi定的是否存在，就 uz是否存在，那么如果不存在的话，我就抛异常了，那么该用户不存在。

好，那么如果没有报异常用户存在，那我就查询相关的数据。刚才我们PPT上也看到了，页面上要显示的是用户相关的基本信息，所以我要把用户的基本信息发给页面，很简单的猫都有点爱的爱去表达，然后优的把我们得到的一个结果传给一面。另外我们再去查一下这个点赞的用户的获赞的数量，那就得掉。Like service我需要把 service注入进来，还没有去处理。

好，那么有了这个service再回到刚才的方法里，我就like service，find user like com，你把 user的ID传给他，那么他就得到了这个数量，那么把这个写清楚一点叫like count，因为将来我们这个方法还得重构，还要处理关注数量等等对吧？做一个区分。我需要把这个数量都发给页面给猫都有点at like count。

好了，到这我们页面上目前所能够得到的数据就这些了，我就直接返回模板，指定模板site，然后那个页面名字叫professor点htma Professor，好，这个处理完以后，接下来我们就处理这个模板，首先我们打开首页，在首页上加上用户的访问路径，然后再去处理这个页面，先访问首页上有多个地方可以要处理，第一个是头部不有一个当前用户的个人主页的路径，这个地方你得改，这个地方要以当前用户把他的ID传进去。

好所以这里我写th冒号，那么整个路径是 user profile后面要带上一个变量，所以说为了拼变量，我前后需要加一个竖线，好，这个地方变量要写的是当前登录用户的ID，当天登录用户是谁？是login user对吧？老给你悠着点的地方了。

除此以外，那么在帖子列表里，每一个用户的头像，它外面都有一个超链接，这链接应该链到用户的主页上去，所以这个链接也得改，那么也得拼一个变量，所以我先把书先写好路径还是user profile，然后跟着一个ID，这个ID应该是帖子的发布人的 Id，帖子的发布人就应该是你看 map点优点对吧？

Map点优点，点ID，当然除了首页的这些位置以外，详情页对吧？然后私信那些地方都有头像，都应该加上这个链接，我不挨个去改了。因为这个没有什么难度，大家可以自己把它完善一下。我把这两个地方改完以后，能够能够做测试就可以了，能够演示这个效果就可以了。

好了，既然能亮过去了，最后我们就来处理了。这个是主页 Profile，profile html这个页面是我们第一次编辑，所以说稍微麻烦一点，你需要把这些基本的东西都写好。好先声明 timeless模板，然后这个地方要做处理，Heider要做处理，然后最后的 GSI要处理。好，这些细小的基本的细节都处理完以后，最后我们再来显示正文的数据，就在这了。

正文的数据这里边你像这个选项这三个可点的地方我们先不管，因为目前我的帖子我的回复咱们都还没实现，就不切换了，以后再说，然后这是个人信息的显示的位置，这个地方数据我们要加以显示。

首先这里要显示的是用户的头像你这里得改，我们传过来的是优的对吧？优这个点，high的URL头像。

好，然后 span这个地方是用户名，那就是悠着点悠着关注他，我们现在还没做，不管用户的注册的时间需要处理一下，格式化一下，然后是悠着点create time给一个格式。

好，这是用户的注册时间，后面的话显示的是他关注了多少人，他的关注者是多少，我们还没有处理，还没做先不管。然后是用户获得了多少个赞，这是我们要重点显示的内容，赞的数量传过来的就叫like count。

好，那么这个页面它显示的数据比较少，到这就可以了。好，那么这个功能我们就代码写完了，我们做一个测试，我把这个项目启动一下。

好，启动完以后这样，因为我们重构了点赞功能，所以说之前的点赞数据和我们当前的要展示的数据不一致了，我这样我把之前的数据给删了，咱们重来重做一个测试，不然有点乱，你看这里都是一些点赞的数据，之前的我把它删了这个好什么都没有了，然后我们访问一下，我需要做一个登录，我用CC登录。

好，那么登录以后，比如说我查看Aaa的帖子，看AA的帖子，然后我给帖子点个赞了，然后当然我取消一下也没问题吧？再赞一下，然后再给评论点个赞取消。可以的。没错，然后我想看一下Aaa目前他有收到过多少个赞，但你注意其实现在他应该收到一个赞，因为这个帖子是赞给他的，而这个评论我是赞给BBB BBB的李，红鹤的BBB的对吧？赞给别人的不是AA好，我这样我回首页，我点头像看AAA的主页获得一个赞。

好我希望再多一点，我这样我去这个帖子里有AAA的评论，我给评论加个赞，还有他的评论再加个赞，此时AAA应该拥有三个站，看一下三个站。没错，主页的头像名字也对。好当然我也可以看我当前登录用户我自己的主页，我点个人主页，这头像名字也对，占的目前是0。好，你看我们经过这样的一个改造和简单的这么追加了一个页面，统计我收到的赞的功能就实现了，我们这次课就演示到这里，咱们下次课再见。

## 4.5 关注、取消关注

32-关注、取消关注.mp4

这次课我们来实现关注和取消关注的功能。那么这样的功能非常的简单，我们平时在上网的时候经常会有这样的操作，大家都能理解。我们开发时就是这个需求，我们是其实关注的是两方面，一个是我们要实现关注和取关的功能，再一个我们还要在用户的这个主页上统计出他的关注数和粉丝数，统计出一些数量，这是两方面的功能。

然后我们开发功能的时候，这个关键是你要把这个关系搞清楚，我们在写代码的时候，尤其是存储的时候，那么我们很依赖这个关系，那么这里我说的很清楚了，就是说如果a关注了b那a就是b的粉丝，这很好理解对吧？B就是a的关注目标，这也很好理解，你注意我们在纯数据命名的时候，那么我们把粉丝习惯于叫follow，我们习惯于把这个目标叫follow，所以说这两个单词代表的含义你要理清楚。

然后第二个我关注的目标，其实我可以关注一个人，我也可以关注一个帖子，我也可以关注一个题目，我也可以关注一个课程，其实我可以关注很多东西，所以说我们不要把这个程序写死，我们在实现的时候也是做一层抽象，我们关注的目标把它抽象为实体安贴点。

好了，你像关注这样的功能，其实也是一个非常高频的功能，所以说那么我们在实现的时候也会把这个数据存到redis里，为了提高性能。

好了，下面我们就来实现这个功能，当然我们主要是分两大步，第一步搞定关注取关，第二步实现统计。因为我们要把数据存到redis，所以说我还是先从Key开始规划起，事实上你把key规划好了，那也就清楚了我将来数据是以什么形态来存，那么开发业务组件的时候这个就更清楚一点。

好，首先我先上面一个前缀。然后叫following是我们关注的目标，我这个q就叫sorry。另外我再声明一个前缀叫sorry，这个地方就叫follow，我上面两个前缀就意味着那么我要存两份数据，其实和我们点赞一样，一个数据是为了存我们这个业务相关的数据，一个是为了统计方便这里也是那么我点我关注的时候，我关注了某人，那么我把这个目标我的目标存下来，另外那么我关注的那个人，我就是被关注者，不是他就是被关注者，那么我就是他的粉丝，那么我以那个人为key，把我也存进去，把我周围他的粉丝存进去，这样的话从我的角度统计我的目标，很好统计，从他的角度来统计，他的粉丝也很好统计双方的一个相互的关系。

所以我们需要记两份数据，为了统计方便，好，接下来我就写两个产生拼key的方法。

第。

一个是某个用户关注的，注意他关注的是实体，不一定是什么不确定。

好，我写一下，我希望 key是什么形式，我希望是这样，因为你是关注的目标，所以说它是以 follow me 开头，然后的话冒号后面因为我们关注的是实体，所以说后面我跟的是 uziduzid表示谁关注的，谁关注的，然后关注的是哪个实体，把类型拼上去，key是这样构成的。

我存的时候 key对应的value我存一个有序的集合最sad，而这site当中我存的是ittid因为这site可以排序，那么它需要有个分数，我以当前时间的整数形式作为分数，那么为什么这样去设计？

首先你看是某个用户关注了某个实体，所以这里边一定要体现用户和实体的关系，你看这里用户有了实体也有对吧？

用户关注某个实体，那么我是按照分类对他从 t上都做了分类，你关注的是人哪还是帖子做了分类，而这个值是具体你关注的什么东西，所以这个值是一个实体的ID，为什么要以当前时间作为分数来排序？

因为可能有的时候我需要统计我关注了哪些东西，我把它列举出来，按照先后顺序列举出来，可能会有这样的需求，所以说我们这样去处理的话，能够应付更好的业务需求的变化。

好，既然是这样一个key和这样一个value，所以能够把这个方法写一下时代的一个试卷，然后我就要get following。

t。

那么很显然我这个参数需要传入的是因为你过道k我这个参数需要传入uzid和idd type，而it ID是我们具体在存数据的时候，那么得到的 t我们要提前传进来，好了，下面我就把 t拼一下，首先是前缀加上分隔符，然后加上按照我的要求是uzid然后加上分隔符，然后再加上安TT态度。

好了，

sorry关注的目标，key我们就这样去拼好了，数据形式我们也做了一个整理。

再来那还要记录一份数据，也是某个用户的拥有的粉丝。好，我希望 t的形式是这样的，follow粉丝，然后那是某个实体，不是用户，某个实体拥有的粉丝可能是一个帖子，可能是一个题目等等，这里面我得写上实体 itdti对吧？

然后冒号intt ID itt type和ittid能够唯一的标识一个实体，那么存的时候这个值依然是有序集合，是为了更好的统计，按照时间统计，集合里存的是什么呢？

拥有的粉丝，其实这个好理解粉丝就是人，所以这里面存的应该是uzid那么我也以这个时间作为分数来存，这样我就很容易统计出实体他的粉丝，他的粉丝都是在什么时候什么时间关注的，这样就比较方便。

好，那么把这个方法来写一下，就是get。

follow t。

那么这个参数就应该是type和ID对吧？我就return这个前缀，加上分隔符。加上 NTT太加上mttid好了，到这儿，我们这两个p它的拼接的方式就声明完了，那么当然这里边可能是你没有开发过这样的功能，那么在对 k的处理上可能没有经验，所以说一开始可能会觉得比较别扭，我希望你把就这一段我刚才所说的我所做的解释，你好好的琢磨一下，把怎么说 key的设计的方式最好能够把它理清楚。

再一个这两个方法名就是关注的目标粉丝，他们是非常的单词非常的接近的，很容易就混淆了，你把它也理清到底什么是目标，什么是粉丝理清清楚，follow跟follow的区别搞清楚。

好，那么理清以后，那么接下来我们就可以开发相关的关注以及取消关注的业务，那么这是一项新的业务，所以说我在包下新建一个类叫follow service。给它加上一个注解，另外因为我们要把数据存到res里，所以说我把res template注入进来。

好。

那么接下来我就来实现关注和取关的功能，这两个功能还是有一点点麻烦的地方，所以说我们别在一起写了，我们分成两个方法来写，这样直观一点，先写关注 follow。

那么 follow关注的时候你要传入用户ID实体，使用户关注了某个实体都要传进来。

好。

那么传进来以后，我们在存的时候，那么是要存两两份数据，一个是关注的目标，一个是粉丝，那一项业务有两次存储，所以说要保证这个事物，所以说我还是要这样写。

new session高度。

因为我们方法不需要有返回值，所以说我就不返回了。

首先我们先构造这两个key，一个是follow目标的key。我就write is key YouTube点get follow with key。那么你需要传输uzid还要传输itt type。

好，再来。那么在构造粉丝 follow up t ready, sqto点get follow key需要传入ittittid这写错了，那么有了key以后，下面我们就连续做两次存储的操作，因为你是关注的时候你要往里存一份数据对吧？我就当然我这里得先启用事物点Martin，最后 Operations点 ex EC对吧？

这中间我们做两次存储的操作，那么就是office点 UPS方有序集合 c set，那么往里添加数据艾特把 t加进来，我先添加就是关注的目标，关注的目标里边存的是什么？刚才我们不是总结了吗？它里面存的是实体ID，另外还要有一个分数，分数是当天时间实体ID Andy tid然后再来一个分数，当天时间我们把它转成毫秒数 Cs他们点 current time, minutes，好，这样就行了。

好，另外一个 operations还是 for this site。点ID，然后这个是follow key，follow key里边存的是什么呢？是优质ID。好，然后也需要存上当天时间作为分数以用来排序。好了，关注我们这个逻辑就已经实现了，也不复杂，反正要注意事物就是了。

好，那么接下来我们再把取消关注搞定，取消关注其实代码逻辑和它是类似的，无非就是这里是添加那个是删除而已，对不对？P都是一样的，所以我就copy一下，然后取消关注，我给它取名叫Ann Salo，还是这三个参数不变，还是要这样管理事务，还是这两个t不变。

然后启用事物以后，这个地方我不是添加数据，而是删除为目的。

那么。

remove的时候你要传入key，你要传入你要删哪个值，这个分数就不用了，因为你要删就不用指定分数了，这个也是 remove。

然后 t值分数不要了，好了，关注取关的业务逻辑我们也就实现了。

那么下面我们就可以开发这个视图层，首先我们需要在CTRL那里处理，关注取关这样的请求，因为是一个新的功能，所以说我需要新写一个CTRL，那么我在这儿新建一个类叫follow CTRL放大一点，然后加上注解 controller，那么这个类上我也不写这个路径了，我直接在方法上写路径。当然了因为我们是做关注，所以说你肯定是要把 Follow service注入进来对吧？注入以后，接下来我就要实现关注和取关的功能，我们也分两次请求来处理，把它拆解开。

注意这个功能我们在操作的时候是一个义务的，我们在页面上一点关注，整个页面不刷新，整个页面不刷新，然后我们做局部的刷新。好，所以说我请求的路径，我的写成叫follow，然后请求方式这是一个提交数据的过程，所以说是post，因为是异步的，所以说加上保底的。

好，那么你关注的时候，关注肯定是我当前登录用户去关注某一个实体，所以用户ID不用传，我们取当前登录用户就可以，而实体你要传进来，为了获取当前用户，所以说我还要把 Host注入进来，我首先要做的事情就是应该获取当前登录用户对吧？

为了后面使用方便，取到以后那就关注我就调follow service点儿 follow，然后把you这点get ID重去int 太怎么去it ID怎么去这就行了。



当然有人说我这个地方如果没登陆，你注意你应该把这个方法就是用拦截器做一个检查，强制它登录以后才能访问，我就不补充这个逻辑了，因为这个逻辑我们之前演示过，你自己把它补充完善就好了。好，那么关注完以后我就要给页面返回一些这个消息，那是异步的请求，所以我返回一个Jason get接近它，零表示成功，然后给个提示说已关注就完了。

好，这个关注就处理完了，处理完以后取消关注其实和他一样就调的方法不一样，但路径你得改一改。这里我改成安发了对吧？

然后方法名改成叫iPhone，然后取消关注的时候也是某人取消对某个实体的关注，某人是当前用户，实体传进来，参数不变，然后当前登录用户我们也是这样先获得，然后这个地方你要调的是Anh, Farlow的方法，然后按Farlow的时候也是把这三个参数传进去，他去处理逻辑，这就可以了。

然后这个地方来说已取消关注，好这个CTRL咱们也就实现完了，实现了以后，那么就是说我们其实完整的网站的业务应该是很多地方都能去关注，这里我就不挨个去写了，那么我们只演示一个地方，你能把它搞明白，其他地方大同小异，我们就演示一下一如何关注一个人，那么这个关注人是在那个人的主页上有个关注按钮点一下就可以了。

所以说我们需要去处理主页的关注的逻辑，我打开主页 profile。Dash temple。那么这个关注按钮很容易找到，关注它。

好，关注的按钮其实它所对应的事件已经由GS处理好了，这是我们给大家传静态页面的时候，静态页面就带了。

好，关注按钮的逻辑它是定义在profile点js里，我们打开js看一下，你看这个逻辑都有，这是关注按钮，单击事件调关注方法，然后这个方法里首先获取当前的按钮，然后判断了如果这个按钮的样式是这个应该是一个灰色的样式，那就表示我没关注，然后的话不这个表示这应该是一个蓝色的样式，好像表示说我可以关注我就去关注，如果否则它是一个灰色的样式表示说我不想关注了，我可以取消关注，灰色的时候我可以取消，我就取消关注。

所以它是通过样式做了一个直观的区别，是你能关注还是不能关注，然后做了不同的处理。

然后关注的时候它这里只是做了一个效果，它把这个字变了，变成已关注还是关注它，根据你关注的状态，然后把样式做了，改动这里面样式我们不用处理了，这个字的显示我们也不用处理了，我们需要做的是关注时和取消，关注时你把逻辑补充完善就行。

好，无非就是发一个异步请求对不对？那么这里做一个处理，首先是关注到了点pose，好，然后先写路径，context pass，然后是 follow，其次是参数，因为我当前关注的是人人实体，它的类型我看接口说不定一样好。

我们之前订了订阅的实体，只有帖子和评论，我把这个人也加上去，那人就往下排，三实体类型是用户，然后的话这个是tap。

u等于3，

所以这个地方回过来我们传的安贴贴太，我就写死成三，刚才写错了，冒号三，当然还得有安 nttd对吧？冒号得有一个值。

好，ID我们需要从html中获取，你看怎么取？目前我们页面上没有显示ID，没地方取，没地方取我可以自己做一个处理，页面上不需要显示的没关系，我比如说我在8天之前我加一个隐藏框，我在框里给他写上Id，然后我从这个框里取不就完了，对吧？

所以我这里写input。

type。

等于黑的，为了取方便我给他加一个ID，ID我就叫nttid当然你得给它一个值，这个值得是个变量，所以我写th往后value，然后y6这里面我们需要写的就是用户的ID，用户是谁不就是user优点点ID，然后我们点按钮触发了单击事件，在单击事件里我们得到当前的按钮，然后的话获取当前按钮的上一个节点就能得到这个数据，这是我们利用GS的逻辑去做一个处理。

好，所以回过头来这个地方我得这样写的。

到了括号button，这个button是当前的节点，就是按钮，到了button点它前一个节点Free的值为l这样就行了。然后最后处理返回结果，dater，我先把它转成GSD相转成了js对象，然后做一个判断，它的编码是不是0？如果是零表示成功，否则怎么样？好，那么如果是0表示成功，我们需要改样式这里我就不去这样改了，因为你不只是改样式，因为你改完样式以后，页面上将来关注的数量都得改。

如果说我们这样写一句话，改样式还得再写两句话改那俩数量通过GS也有点麻烦。

这里我说点事儿，因为这个前端也不是我们的重点，反正只要实现这个效果就可以了，我说点事儿，直接我就刷新页面了，利用Gs那就window location这样让页面重新刷一下，这个改样式就不用了，租掉了不要了，那么否则那肯定是错了，我们给个提示，把返回的一些提示信息打印出来就行了。

好，这是关注的逻辑，那么取消关注和类似的应该也不要了，取消关注，我把这个代码贴过来，然后我们稍作修改，这个路径应该改成iPhone对吧？然后anti type还是三，因为也是针对人的anti tid依然是这个值返回值data，依然这样处理。

那么如果是0，就说明你取消关注成功，我们就刷新页面，否则给个提示完了，好到这我们关注就搞定了。搞定以后咱们先做一个测试，看一下行不行，我启动一下这个项目。

好，启动完以后我打开浏览器，然后访问一下首页，我这已经登录好了，那么是CC比如说假设我想关注AAA那么我访问Aaa的首页，这里我可以点关注它，我点一下那点完以后没反应，为啥没反应？

首先他没报错，应该是没报错，没报错就应该是关注成功了，关注成功以后咱们不是刷新页面了，但刷新页面以后，我们这个数量还没有做相关的显示，所以它没动。

再一个我们关注它这个状态，我们没有去做特殊的处理，所以状态显示的不对，因此这个是我们一会要处理的地方，就关注数量，我们需要获取，关注状态我们需要去改变，但是数据对不对？我们可以查release，看一看有没有这个数，先这样看一下。

我查询一下这个case，你看follow we follow这个数据都有了，比如说我看这个follow follow151应该是CC的ID，我就z因为这是一个有序列表，range范围查询，然后k我写following不要写错，然后冒号151，冒号三，然后写上你要重，哪查到哪指的是索引，我要从0查到1-1表示最后一个作业回车，你看它里面只存了一个一个ID，就是111——是AAA的ID我关注的是AAA没有问题，数据就对了。

好了，那么既然关注我们搞定了，那么最后还要做一个数据的正常显示，当你访问用户的主页时，他的关注的数量你要显示对，关注状态你要显示对吧？那么为了显示这些数据，我们需要在Service里再补充一些逻辑，因为你要把数据查出来再补充一些逻辑，所以我再回到这个follow service，再增加一些查询方法。

首先我先补充一下我要查什么，我要查询关注目标，关注的实体的数量，查询某个用户关注的实体的数量，返回浪，然后find follow目标，count。

那么你需要传入uzid是谁的关注目标，还要传入类别，你是某一类的目标，而不是所有类的，因为你没有必要说你要统计关注了多少个帖子，关注了多少个用户把它俩混在一起去统计，这个没必要，你是分开的，你关注多少用户是一份数据，关注多少帖子是一份数据，肯定是分开的，统计放在一起没有意义。

好，那么这很简单了，首先我过到key发了为。

key等于red。

is key YouTube点get follow your key。把user ID传进去，把n TT type传进去，得到 k我就直接就能得到它里边的数据量，red is template点ops phone z site。那么统计z site中的数量是 z car。好，然后你把k传进去，数量就有了，好了，这是统计关注的实体的数量，我们还需要做一个统计，某一个实体粉丝数量查询实体的粉丝的数量，还是返回了 find follow。

然后 count查询实体的粉丝数量，你需要存入的是itt type。

Ittid传的是实体，我首先构造的还是一个t follow a key，read this key，YouTube，get follow a key and detail。Ndtid也是直接return的register template ops for ZZ然后还是 z卡把k传进去。好，那么除此以外还要做再加一个方法，因为我们在页面上要显示我当前用户是不是关注了这个目标，我要显示这个状态，当前用户有没有关注这个目标，所以你需要做一个查询，好判断出来这件事儿。

所以这里我再加一个方法。

就是查询当前用户是否。

以关注该实体。好，那么这个我返回一个布尔值，最终是一个判断返回布尔值，方法名我叫has，follow的是否已关注，加一个参数uzid表示赞成用户，然后你关注某实体，你还得传实体对吧？好，我们可以看一下我当前用户的关注目标里有没有实体就知道了，我就构造 key，follow为key，目标的key rides，YouTube点get发了vt然后又在ID传进去。

NTT type传进去。

好，那么有了 k以后，我们就很简单判断一下，red is template点ops for z sad。

然后其实有很多方法都可以判断，我可以调这个方法，随便找一个，比如说我查询一下分数，查询一下某一个数据，它的分数，你能查到就说明这里有对吧？你查不到就没有，这个就很容易 follow为key，这是key。

然后ittid我查 ittid的分数，如果说我就这样，如果说查到的结果不是空的，就表示说肯定就有这个值，他就已经关注了，如果是空的没有这个值，那就是没关注，就这么一判断就可以了。

好，那么有了三个查询方法以后，我们在主页上要显示出正确的数量和状态，而主页是通过user controller去访问的，对吧？

所以说我们还得去处理 user control这个逻辑。那么之前我们在这里查到了用户查到了点赞的数量，下面要补充一些内容，我们需要查询关注数量，就是这个人他关注了哪些实体这个数量，还有你还要查他的粉丝的数量，还要查他是否已关注我当天登录用户对这个用户是否有关关注？

那查询关注数量我就调发对对调follow service大家还没注入，好，把它注入进来，然后回到刚才的位置，我调follow service，然后关注数量是find follow account。

好，然后你要把 user ID传给他，还要把实体的类型传给他，那么实体的类型，因为我们当前是查询某一个用户，它的实体用户的实体其实就是三，但你这里不要写使你写成这个常量，我看一下有没有实现常量接口，没有实现一下。

好，回到刚才的这个位置，我就写NTT type，user23，好，那么这句话一执行，我就得到了一个关注目标的数量。

Follow me。Come。我希望把它传给模板。好，关注数量搞定，同样的再把粉丝数量查到，我就long follow account。

等于follow service。

点find。Follow account。

然后你要传的是实体类型，用户实体还是常量，然后用户实体ID就是优质ID。好同样我也要把这个数量传给模板，最后再查一下我当前的登录用户对这个用户是否已关注，那么这个产品你做个判断，所以我先把这个变量先声明好。Has

follow默认等于false。

然后这里我得判断一下我当前有没有用户登录，因为如果你没登录，你也可以通过首页看别人的空间对吧？没登录也可以，所以这里你要判断一下登没登录，那house偷得了get悠着，如果说他不等于闹，这个时候才有可能关注。

我就判断一下 Has followed就等于 follow service做查询点has followed，那么你需要把当前用户传进去，就是host holder，get user get ID然后当今用户关注谁，关注某一个实体，关注的还是用户类型是他实体ID是你这个空间的user ID好这样才行。

查完以后 has followed也需要成为一面好处状态的判断。

好这些数据我都传给了主页，最后一步我们就是打开主页，把数量状态给它处理。好我就打开 Profile，我先处理数量比较直观，首先就是当前的用户，他关注了几个人，那么这个地方我要显示的应该是 follow we count对吧？好，这里我显示了 follow we count。

其次当这个用户的关注者是几个人？关注者其实就是他的粉丝，那就是follow count完了，然后还要处理按钮的状态，8天的状态你得看一下我登录用户对这个用户是不是已关注，如果已关注你这里要显示已关注，如果没关注你才显示关注它这个按钮的这个名字，你要动态的改变，好我就这样我就th冒号，text等于这里做一个判断has这样我把它换个行，这个太长了，——瓶都摆不下。

Has。

follow。Pass follow。这个是帧吗？如果是帧表示已关注，我显示已关注。好，如果是假，我就显示关注它，好，这是对按钮的值做一个处理。

除此以外，大家还要注意另外一种情况，什么情况？我是登录用户，我访问的是我自己的空间，这个时候你就不应该显示这个按钮了，登录用户访问自己空间的时候不要显示按钮，不然的话他点一下自己关注自己怎么说，虽然不能说它有多错误，但是这个业务没有意义，自己关注自己没有意义，所以说这个案子我还得做个判断。

Th冒号if等于等于什么？如果当前登录用户和 User是一个用户，你就不要显示我这样写，如果love in user登录用户，假如它不等于now并且那么。

log in user。

点ID不等于。

user点ID。

好，我是这样写的逻辑，我们再重新解读一下，首先你登录用户得为空，就是说如果你登录用户为空表示你没登录，那也不显示你没登录的话显示它没有意义对吧？你不应该这样的操作，并且登录用户的ID不等于这个空间用户的ID，如果等于就表示说他是一个人就不应该显示是这样，所以这块还有点绕。

好到这这个就处理完了，下面我们再做一个测试，我把 CTRL f九重新编译一下。

好编译完以后我打开这个页面，然后重来我是访问首页，然后的话比如我点 AAA你看已经关注过了，然后他获得了三个赞，然后 AAA他没有关注任何人，但是他的粉丝有一个就是CC没有问题，比如说你看我这个CC我再点一下，再点一下就有点问题，什么问题？

就是我们按钮就是怎么说，我们点是点一下它关注，再点一下是取消关注，而关注和取消关注它的逻辑的切换是通过GS通过样式去处理的。

就是说如果它是这个样子，我就可以关注，如果它是另外的样式，我就可以取消，而我刚才样式没有处理，所以说它样式不切换就导致我点的时候就有问题，所以说还挺麻烦，样式还遗漏了，这样我们把样式做一个重新的梳理按钮，这是 class目前它是写死成button，音符，8t音符你也看到了它是一个蓝色其实代表的是你可以你刷新一下，蓝色代表的是就是说你可以去看一下这个逻辑，如果是8d音符表示说你

可以去关注它，而当前我们这个逻辑是不对的，好我这个地方需要需要动态的去拼，你已经关注了，我显示一个样式，你没关注我显示另外一个样式，好让样式能切换，让我们这个逻辑能动态起来。

好，我写th class，然后竖线这里面要加个变量，这个地方巴丁音符不能写死，这里我写死了，我应该是做个判断的是。

has followed,

然后那么如果已关注我就要显示某一个样式，如果未关注我显示另外一个样式，其实未关注应该显示大厅音符，如果已关注了，我应该显示谁显示他，这两样是之间切换好了，但是这个前端逻辑，你可能没有写过这个页面，所以说你这个样式不太熟悉。

这里就是说我直接告诉你，你把这个记住就好了，这块有这么一个隐含的规则。

好了，那么我再重新编译一下，再打开页面刷一下。

好，这回就对了，你看他已经关注了，比如说这个地方我已经关注了，它就变灰了，然后如果已关注，这个时候我再点一下，再点一下他就取消关注了，取消关注的时候，这时候我就可以再点关注，那这个时候你看他关注者是0，是吧？因为我取消关注了，他没有粉丝了，我在关注他，他粉丝是一我再取消关注他，粉丝是0再来一下。没错是吧？

好，然后另外我再访问我自己的主页，看看有没有什么问题，访问个人主页。你看我关注了一个人就是AA，我的关注者没有我没有粉丝，我登录自己的主页我看不到那个按钮不能点，应该是这样的。好了，目前我们就把关注取关以及相关的数量状态的显示就处理完了，我们这次课就演示到这里，咱们下次课再见。

## 4.6 关注列表、粉丝列表

33-关注列表、粉丝列表.mp4

这次课咱们来实现关注列表和粉丝列表，那么上次课咱们不是开发了关注了，对吧？那么我们把关注的数据已经做了一个保存，那么然后我们在某一个用户的主页上能够看到他关注了几个人，然后几个人关注了他，然后那两个数量，他关注了几个人，关注他那两个数量，其实是可以点进去以后各有一个列表，其实就是我们这次课程要开发的功能，那么现在的数据已经有了已经存好了，那么我们要把它做一个展现其实并不难，那么相对来说是比较容易的，然后我把我们要做的事情也做了一个归纳写到了PPT上，因为我们使用reds访问访问这个数据，在数据访问这一层比较简单，所以说我们通常就不去写 deo 了，我们直接写业务层，然后就访问ID，或者说把它当做一个是内存的组件来对待，我们做这个功能业务层显然我们需要查询对吧？

那就查询的某一个用户，他关注了哪些人，然后你要分页。

第二个你要查询某个用户，他有多少粉丝也是要分页，那就是说从我们上次课存的关注相关的那两组数据当中去查就可以了。然后查完以后业务层开发完以后，接下来我们需要做展现表现层，一个是我们要写CTRL了，我们需要处理这个查询关注的人，查询粉丝这样的请求。

其次我们需要编写这个模板，也是要把两种查询的模板配置好。总之这两个列表他们的展现形式极为相似，差不多，几乎就一样，所以说我们就一起并行的把这两个功能一起写出来。

好，那么在写代码之前，因为刚才我靠嘴说不是很直观，我打开精神页面给大家看一下这个功能，比如说我点用户打开了他的主页，这里不是有两个数量对吧？

然后比如说你点进去能够看到这个用户他关注了多少个人支持分页，后退一下，当然点这也能后退，然后你点这个就是看当前用户他的粉丝就关注这个人的人，就是他的粉丝也是支持枫叶的，总之当然这个地方也是可以切换这两个功能，他们的展现形式是一模一样，主要来说就是展现他关注的人，或者是关注他的人，要展现的是人的头像，人的名字，除此以外还要展现的关注的时间。

我们在开发的时候将这个时间做一个倒叙，有人说这个时间是什么呢？咱们上存到有序集合里不就是一个时间，其实那个分数就是那个时间，我们把它把那个分数取到把它再转化为时间，这时间就有了。然后另外那么当前的登录用户看到了你，关注了这么多人关注了你，我如果对他有兴趣，我也可以关注这个人。

所以说当前用户也可以关注这里面的某个人，所以说这里边我们还要判断当前用户与这些人的一个关注状态，从而去开发这个功能，但这个功能的开发其实和首页上和主页上，这个其实是类似的，这个代码我们是复用的，简单处理就可以复用好了，废话不说了，我们直接来写代码，那么刚才说了我们首先要开发业务层是吧？

所以说我把业务层 follow service 打开放大一点，在这里我要补充一个新的方法，然后写好注释这个方法我要查询某个用户关注的人，这是一个，除此以外还要做另外一件事，还要查询某用户的粉丝，就这两件事儿，咱们一个先来开发第一个咖啡，那么我们在集合里存的其实是一个你关注的人或者是你的粉丝的 ID，但我们传给页面的肯定不能只是 ID，我们需要有 user 还需要有关注的时间，所以说其实是一个 user 对象和关注时间的整合在一起的数据。

这样我返回一个集合，首先这是多条数据，然后集合中我通过 map 对这个数据进行一个包装好，这个方法名我叫做 follow with，那么你要查询的是某一个人的他关注的用户，注意我们在存的时候，其实我们不只是说存他关注的人，我们存的时候他关注的一个目标一个实体，但因为我们这个需求就是过于明确了，我们当前做的就是查询人，如果说你把实体都融在一起去做这个查询有点麻烦，你又得判断他是人，我要查人，他是这个帖子我要查帖子，他是评论我要查评论，我这块我就不那么写了，我存的时候按照实体来写这个地方我就需求明确一点，我只查人，如果将来说我们还有需求要查帖子，查关注的评论，我们再加方法去实现，这块就不揉在一起写了。

好，既然要查某个用户关注的人，你得把用户传进来对吧？User ID 传进来，然后我们要支持分页，所以说把分页的条件传进来。

好，那么具体实现的比较容易，首先我要拼 key，follow with key，咱们拼一下就可以 copy，follow 为 key 等于第一个我要传入 uzid 第二个因为我们明确当前是查关注的人，所以说这个态度我们就可以固定下来写个常量就可以了。

Mtt 好像没有引入常量接口，我就实现行了。

回到刚才的位置，这个地方就是 NTT type user 是三，好，有了 k 以后，下面我就要查询了，我要从集合中把这个数据查到，那我就 ready template。

然后第二，offset for reset 是有序的结合，然后查多条数据，你要按照范围的方式来查询 range。

Range 它默认是由小到大的分数来查询，但刚才说我希望它倒叙，我希望它由大到小，最新的时间是在前面的，你可以这样，你可以 reverse range 倒叙，好，然后我们需要把这个条件传进去，首先你 key 传进去，然后除此以外，我们在范围查询的时候，你要查你要传入两个索引，你要从哪儿查到哪儿，传入索引。

首先我要从奥赛的索引开始查奥赛的然后截止的索引其实是截止。如果是 my circle 分页，我们第二个条件就给一个当前页最大显示的数量 limit 就可以。但现在我们这要的不是 limit，我们要的是截止的索引，其实就是 offset 加上它，但是你要减一，因为这个奥赛的是包含的，所以你要减一这样才行。

好，查完以后我们就会得到一个集合，我们会得到一个集合，集合中的纯的是整数。

那么。

存的其实我们是我们关注的目标的 ID，它盖 ID 我给它取一个名字。

好了，那么查到以后我对这个数据做一个判断什么判断它是不是为空，如果为空的话，我就不用做进一步的封装处理了，否则我就去处理。如果说他 get ideas 等于空值对吧？我就直接 return now 就返回就完了，否则我就继续。那么如果不为空的，我们需要最终把 id 转换成更详细的数据存到集合里，这里我就先把集合实例化一下。

好，然后我要便利他get ideas，然后把每一个用户渠道把相关的数据渠道好了，那么渠道以后每个他改的ID我要查到了对应的用户，查到以后要封装到 map里，所以说我这里需要先实例化一下 map，好，那么然后我要把他get ID对应的用户查到，那我就调用user service，但目前设备还没有注入进来，我注入一下。

好注入以后咱们继续查询查 find user ID，然后把大概的ID传进去，我得到 user把它放到最终 map点put user那么除了 user咱们还要查关注的时间，其实就是有序集合中的分数，那也很好取，就read this template。

第二ops for g set，然后你要取一个分数，咱们之前不是演示过是调 score方法，当然你需要把 key传进来，然后你要取谁的分数，你把这个值传进来，那就是他get ID。

好，我要看看这个值它对应的分数是多少，我们会得到一个小小数，但我们知道这个小数其实我们当时存的是用这个时间毫秒数转化而来的，所以我们可以把它再还原回这个时间，我这样写map点put存进去，这是一个关注时间我叫follow time，这个时间我就new date。

然后 sky这是小数。

第二浪漫流把它转成了就可以了。最终 Map中包含了这两个值，我想要的，我把这个map在放到集合里就好了。

好最终我把历史的返回就行了。这是查询某个用户所关注的人，查他的粉丝其实也是大概的流程，我把另外的方法也写一下，但这个我们写的时候就不啰嗦了，我就直接写了，还是返回map，方法名这回我叫find follows。

好一样的参数，你也是要查某个人的，粉丝也要支持分页，也是这么几个条件。首先也是先拼 key，我还是copy。好，那么 key当中的tap固定为人的类型，所以还是它然后 ID是人的ID就是ID就行了。

那么有了t以后，我们还是像刚才那样去通过倒叙的方式把 ID查到，返回端还是一个site，我忘了我们之前在演示spring整个 ready的时候，我演示有序集合的时候，有没有说过我们调用这个位置或者是 reverse方法查到一个范围内的数据，并且他会做一个排序，排序的时候它返回的其实是一个set对象，写到这儿的人可能会会有疑问，返回的是site，这个site不是无序集合吗？

你注意默认的话GDP的时间确实是无序的，那么其实 release的 range，方法它返回的是 site这个接口不假，但它的时限内其实是它内置的，它自己实现的是一个有序的集合，所以这块它是做了一些处理好。这个变量里面我还叫 target Alice，然后read this template ops for z set，然后 Reverse range道具，把 follow key传进来，然后起始的索引，截止的索引，offset加上里面的建议。

好。

那么这个数据我还是做一个判断，如果说他get IDs等于我就对他其实和刚才这个是一样的，如果不为空，我们需要也是把最终结果封装为一个集合，整个代码结构和类似了，我就copy稍加修改就可以了。

好我实例化了一个集合和刚才一样，然后遍历 target Alice，在每次遍历的时候，我实例化一个map没有问题，然后要查 u字也是根据他刚才d来查没问题，然后把u字装到map里没问题，接下来也是要取这个分数，就是时间 ready，Stanley的ops符号计算 key不对，把它换成我们新的t传他get ID，然后也是通过 scar，我们得到 follow time存进去，然后添加到结构里返回，其实整个这两个方法都是一样的逻辑，只不过对 key处理不一样而已。

好，这个业务方法我们就写完了，写完以后开始处理数据访问层处理表现层，首先我们需要处理这个请求，我要打开 CTRL，然后根据当前的逻辑我需要添加两个方法。

好，我打开 follow controller搜一下，那么后面我加一个新的方法，先声明了他的访问路径，pass等于 follows我要查某一个用户他的关注的人，那是哪个用户，你要把用户ID传进来对吧？我也希望把 ID通过路径传进来，所以说我再加一个下级， user ID，因为这是一个查询，所以说请求方式为get就行了。

那么方法名方法说明一下时间这个方法名我叫get follow with。在这个方法之内，首先我就要得到路径当中的 ID对吧？然后好用它来做进一步的查询，我就需要用到帕斯维尔堡注解。

好，因为要分页，所以说我要声明一下配置对象，那么为了往页面模板传数据往底下是model，好，这些条件准备好以后，首先我们要把 u的查到， service还没有注入进来，得注入一下。

好回来。那么就user service， find user by ID把 User ID传进去，然后我们对又得出一个判断，万一他写路径特意写个错的怎么办，对不对？万一是个错的没有值判断一下多了一个埃尔法，如果说是空的我就报异常了，给一个错误页面的提示，说该用户不存在。

好，那么如果不报错没问题怎么办？我要把我要把你访问的优点传给页面，为什么要把这个又再传给页面，你看一下好。

我点你看我们不只是列表，这列表前面有一个标签能点的页签，页签上，谁关注的人，关注谁谁谁的人，就是你访的用户，名字要显示在这，所以我把用在床上，为了显示这个名字，好我就murder I的 attitude，然后的话悠着传进去。

好，那么有了user以后，接下来我要查询列表中的数据了，因为要支持分页，我对分页条件进行一些处理，进行设置site limit，比如说我每页显示5条数据，因为我的数据比较少，然后配置的set pass声明一下路径，路径不就是他，当然他得有下级，他的下级是优质ID，好我把优质ID拼上去，最后还要 site rose，我要查一下一共一共有多少行数据，从而计算的对总的页数做一个判断。

那么你要查询一共有多少行数据，我们可以调 follow service，然后点find。Follow we count方法之前我们实现过find。Follow we count，然后你要传入uzid另外还要传入这个类型是固定的，NTT我好像还没有实现这个接口是吧？好，这个类型是固定的。Itt type user，当然了我们最终查到的这个方法是不是它是一个它是浪，但是赛特rose我们这个方法要求传的是一个整数，这样我们强转一下，那么关于分页我就设置好了，设置好以后最后就要查询分页查询我要显示的数据了，我会得到一个类似的里面装的是MAC，好。

那么。

变量名我叫优则类似，因为我查到的是优的相关的数据，那么调用的是 follow service，点find。follow with，把user ID传进去，把分页条件传进去。

好，那么最终得到的数据我们还要做一个补充，补充什么页面上，刚才我一开始我也说了，我们还得判断当前用户登录用户是否能关注他对他的关注的状态，而目前我们的关注状态没有。这得补充一下。

我就做个判断， User list如果说他非空我就遍历，然后补充数据，我就放每次遍历得到 map，得到遍历的是user list，好，那么每次我要得到map以后，我要从 map当中把你要看到 user得到，然后的话去判断当前用户对 user关注的状态，然后把u字取到基本上我叫u别和前面的 u字冲突了，好u等于 map点 get，然后的话是user当然这里你要转型，然后我要判断当前用户对用户关注的状态是怎么样，这样我把这个方法单独封装一下，这样好复用，因为一会我们还有个方法要用的，突然问题玻璃返回布尔值s follow。那么你把你要判断的 uzid传进来，我们判断的是当前用户对它的关注状态，你先看一下当前用户那就household点get user判断他是不是登陆过是不是为not，如果说当前就没登陆，没有用户你就直接返回boss就行了。

因为如果没登录的话，那就不可能去关注它，我们就认为是没关注不了。

好，然后的话 return follow service，我们之前写过这个方法，还是follow，然后的话你把当前用户传进去 household，get user，然后就是get ID，然后 itd用户类型type user，另外的话再把有载ID传进去，判断一下当前用户对这个用户有没有关注过。

好回到刚才的这个地方，我们就可以调这个方法来做进一步的处理，那么这个方法我就这样直接往map里放， map点put，

pass follow做。



这样一个标志。

然后的话调这个方法还是follow的，然后把优点get ID存进去，我就会得到这样的一个结果。

好，这会 map里的数据就补充全了，补充全以后，最终我把集合要传给模板 model，点艾特，然后 users user list，好，最终我就返回模板处理模板就可以了。

那就是sat，然后followings网页的名字叫不是叫followings叫。

follow me。

点html就是它好，那么除此以外还有一个还有一个请求是要查询的，当某个用户的粉丝，那么其实这个方法的逻辑和这个是很像的，我就copy加以修改，首先改一下路径上肯定是要变的，那就是follow us，粉丝follow us，然后的话也需要传输某用户也是 get请求，然后这个方法名得改一改， get follow us。

其他的不动，然后也是要解析路径中的uzid也是要分页，也是要用model往页面传数据，在一开始我们要查询用户，判断它是否存在，如果存在我就把它产品模板不用变，其次对分页的处理，然后我也要每页显示5条数据，然后当然路径得改了， follow us。平常uzid然后 rose行数是find应该是follow。

好，follow count的条件需要就变一下。首先传的是类型。Entity tap user，然后其次传的是uzid这个和刚才正好是反的要注意，你要记得我们 Key是怎么拼的。好，然后就要查询了这个就是find follow us，follow service，点find follow us，所以说你看我们这个方法的定义，这俩单词比较接近的，所以说你在写的时候一定要搞清楚，不要弄混淆了，不然的话就很头疼，一定要搞清楚。

然后传递ID，传递俩条件，飞行条件没问题，然后要补数据，如果非空我就遍历，每次得到一个map，然后从map中得到 user，然后我判断一下当前用户对它是否已关注，这个逻辑还是一样的，最终把 user 传给模板，然后返回这个模板的名字，这个叫follow。

Follower二点html就完了，好了，那么完成以后接下来我就开始要处理这个模板，首先我要打开这个页面叫profile点html打开这个页面，因为我们是点这儿访问的列表页面的对吧？

所以说你要在这个地方把这个链接加上去，所以我们先处理页面的链接，我就搜一下 profile，点cmi好这个人关注了多少个人，这里不是有一个链接吗？对吧？

链过去把链接得改一下，th冒号，hren好，我改一下这个路径，那么我们当前的路径路径中带一个变量，所以我写个双出现拼这个变量，然后是 follow with那么下级就是用户ID，你看用户不就这样取悠着点什么对吧？所以很容易悠着点，ID就可以了。好这个路径就处理好了。然后再第二个关注者就是粉丝，这也有一个路径也需要处理。好，这里边它也有一个变量，所以说也要选双竖线follows。

然后也是刚才的优点ID，好现在我就把链接配好了，配好以后那两个页面你要处理 follow vee，还有 follower，不过这里面其实处理方式雷同，我们重点处理第一个，第二个就很简单了，搜一下 follow me，html，当然我们是首先第一次处理这个页面，你需要把他们的基本的内容配置好好，然后这个地方要改 hider要处理好，然后再看后面这得处理好。

接下来就处理中间的核心内容。因为中间是个列表的话，它都是一个数据的示例，我可以把其他的li删掉，保留一个以此为模板进行遍历就可以了。

好，删掉多余的保留一个就行，我们从前往后去处理它，首先我们在一个选项的地方，这里有两个可点的页标签，就是这能链接能链过去互相能跳，这俩链接得处理好。好，第一个是链接链，到 follow上去我把它处理一下。

好，那么它的路径就是follow with，然后的话 uzid当然因为你有变量是吧？变量悠着点，ID咱们 control不是传过来一个uzid好再有 a里边你看它是某某某关注的人，这个地方你得显示用户的user name，它得是一个变量的显示对吧？在这里我写th冒号，txt等于然后的话uz等于uz内。

好，那么类似的第二个是练到 follow上去，好，那么它里面的内容这个人名也是需要一样的处理。

好了，那么这个人就处理完了，处理完以后这个地方还有一个叫返回个人主页，其实就是返回到 profile，这个链接也得改，因为这是个写死的精彩的链接。改一下，那么我要练到的是user，然后的话 profile哪个user呢？User点ID这样。

好，那么关于选项的这个地方我就处理完了，处理完以后接下来就开始处理这个列表了，主要就是循环对 li进行一个遍历，每次遍历我们从集合中得到的是一个map，所以变量名我就叫map。Users。好，然后的话每次得到 Map我要显示的一些数据，首先你看这个地方当然这是用户头像先把它改了。好，那么map中首先有一个数据就是user对吧？Map的user然后点头像黑的URL就可以了。另外的话头像外面有个超链接能链到用户主页上去，这个链接我也处理一下。

好这个地方我要写成路径艾特，然后访问主页是要带ID，所以说写上双出线 User。

然后说。

fan不是发下级，是优质ID，是优质的ID，所以是map点优质点ID链过去。行了，然后再往下你看这个地方这地方写死成人名字了，你得写成变量对吧？那就是MAC点user点user name。

好，然后再往下关注于什么时候，这是一个关注时间，我把它换个行，不然的话有点写不下一屏，关注时间改一下。好，那么关注的时间我们需要进行格式化，diss点format，那么这个时间也是卖不离去卖不掉。Follow time对吧？然后给一个时间的格式。好，然后再往下就是分页，分页的话我们可以复印首页的分页，这地方就很简单，一句话搞定好。

那么除此以外还别忘了一点，就是关注它这个地方的处理，它和主页上的关注它的处理方式其实是类似的。

那么主页上关注他，我们之前在处理的时候是哪去了？在这我们之前在处理的时候，一个是这个地方你要判断这个状态，显示是关注它，还是已关注对吧？

另外的话一个是显示关注他和已关注，另外的话你要判断如果说我访问的空间是我自己的，你还不能显示，再一个按钮颜色的变化是通过样式来切换的，也是根据你是否已关注切换样式的。

最后我们关注的时候需要获取一个uzid那么 Uid我把它摆到了前面，用一个黑的来做处理，我们现在的处理方式和它是一样的。首先我回到刚才的页面，我把前面加上一个黑的，然后 Ittid value value应该谁？是map的user，是你当时正在看的，User的ID这块要处理一下。好，然后 Button它上面的 class我先处理 class，th冒号plus等于这里边需要拼一个变量，前后加上一个出现，不要忘记。

然后中间我们主要是拼的是这个地方巴丁音符这个地方，那么就是map点 has followed。

好，那么如果是followed的状态，我就显示另外一个样式，否则显示这个样式。那么另外的样式叫什么？我就拷一下，记不住，或者其实你可以把这句话整个贴过来也行这样的。

好，那么除了样式以外，别忘了还有其他的 content要处理，那么你要判断一下它在满足条件时才显示，否则不显示。那么这条件我这样写，登录log in user，如果说log in user不等于闹，你只有登录了我才显示关注你才能关注别人对吧？如果你没登录关注的权利是没有的，并且log in优点ID它不等于卖点优点，ID就登录用户，他不等于你当天看的用户如果等于的话，那就是一个人自己关注自己没有意义对吧？这是判断。

另外这个字也是需要根据状态去改的，Text等于那就是map点has followed。

好。

那么如果是已关注你就显示已关注，否则你要显示关注他，我copy一下。好了，到这儿我终于把 Follow这个页面给它搞定了。Follow s其实和它一样，我就快速的把另外一个也写一下，follow第二天的这里我能copy我就copy省点事儿，这个地方他俩用的样式也是一样的，拷贝还得展开，不然的话会拷的内容会多，这一展开后发现错了。

好high的，然后再往下我先把 li多余的删掉，再往下尾部不用处理，下面这两个样式也需要处理，直接copy得了，这两个GS不是样式。

好，最后处理正文的内容，首先是这个链接和它是一样的，但是你不要整个拷，因为它俩样式有所区别，链接地址是一样的，好，这里我也折个行 user怎么取？一样。

好，那么另外一个路径也是copy，好，那么内容的话，也是人名 copy。好了，copy完以后返回个人主页这个一样，好像基本都一样，但是你最好一个捋一遍，不然的话他万一有这个逻辑不一样的地方，容易容易搞混，所以说这个地方还是慎重一点。

好，接下来列表了，我先把分页拷过来。

好，然后的话其他的地方首先电力一样，好，然后再往里面看遍，历以后里边我们主要是显示用户头像，然后显示你要查看的每一个列表中的人的名字，关注时间，还有关注的状态确实都一样，我拷便利拷过来了，然后路径一样的，profile路径一样的，好，然后是user the head URL书写方式一样，然后人名一样的处理方式，关注时间换个行太长。

好，那么关注的按钮这个真是一模一样，直接我直接就把它都拷过来算了。好了，到这儿我终于把这个页面逻辑写完了，写完以后咱们得试一下，那么我把项目重启一下。好，启动以后咱们试一下。你注意我上课之前我对这个数据做了一些处理，就是我造了一些数据让他能够至少有两页，不然的话数据太少没法看对吧？

我提前照好了就跟大家说一下。

然后当前是以 AAA的身份默认已经登录了，那么这样我就看AAA的空间，就他看他自己的空间个人主页，你看我照的数据是当前的AAA关注了8个人，然后有另外7个人关注了他，我点进去看，你看AAA关注的人有这么多，因为当前的登录身份就是AA，他已经关注了这些人，所以状态是下一页没有问题，上一页没问题。

当然了比如说我随便点一个，比如说我不想关注这个人了，点一下刚才那是张飞点完以后，那么页面刷新了，那个人就看不到了，因为他就不存在到关注列表里看不到。Ok上一页下一页没有问题，然后但是你从这也可以切换，或者你返回个人主页，我看能不能返回。可以。

看这个关注者就是谁关注了AAA这个人有这么多7个人，其中 Ccaa也对他进行了关注，我再看上1页，然后我随便找一个人也关注他一下，比如说我就关注 ggg关注他没有问题对吧？好，然后这个地方能不能切换点一下，这是可以切换的，然后返回个人主页，你看他多关注了一个人，他关注的还是8还是7。好了，那么这次课我们就把关注列表和粉丝列表就给它完成了。好，这次课我们就演示到这里，咱们下次课再见。

## 4.7 优化登陆模块

34-优化登陆模块.mp4

这。

次课我们使用reads来优化登录模块，因为登录模块它有一些功能，我们原来实现的方式效率偏低，那么使用reduce优化以后，能够很大程度的提高效率，满足性能的需要。我们主要是优化三个方面，这里我列出来了，第一个我们是要使用reds存储验证码，因为什么验证码它是一个比较频繁访问与刷新的这么一个功能，大家可能很多人会反复点验证码，因为有的时候看不清对吧？

所以它对性能要求比较高。

那么第二个验证码它不需要永久保存，他就临时存一下，通常存的时间很短，可能存个几十秒，甚至是一分钟就够了，然后过一会儿就失效。那么目前我们是把它存到了30里，其实之前我们在存的时候我也说过，存到3s里面只是一个临时的方案，因为我们将来要是做分布式部署的话，那么CS有共享的问题之前我也已经讲过，对吧？

而我们把这个数据转存到readiness里，一个是它性能高，第二个 Readiness可以对 key设置一个有效时间，超过这个时间它自动就失效很方便。

再一个 release它也是一个数据库，所以说我们把这个数据存到了reads里，那么分布式部署的时候，所有的应用服务器都从reds里读这个数据，就避免了 section共享的问题，就绕过 section。

好，这是我们用reds解决的第一个问题。然后第二个我们可以用 Readiness存储登录凭证，原来我们的登录凭证是存到了 my circle里，但是大家也发现了，那么我们每次请求我们都要查询一下凭证，我们在拦截器每次都查对吧？

所以说这个登录凭证它的访问频率是非常的高的，那么如果不进行优化的话，每次都从这里取，那么性能会有很大的影响，所以说我们这节课用 readiness把登录凭证给它存下来。

然后第三个是用readiness缓存用户的信息，为什么？因为我们每次请求时要根据凭证查用户，所以也是每次都查，每次都从my circle里查，效率也低，我们也可以把用户缓存到rds里，但是我们这个登录凭证如果说用ready存，那么my circle其实就不用存了。我们 Log in take的表，你的数据其实就可以作废了，那张表就可以不用了。

但是优的数据我们不能作废，买SQL的表还是要保留，该存还是要存，只不过我们是把 user缓存到reds里，然后过一会儿过一段时间它就失效自动过期，其实也符合我们实际的业务，因为你想用户访问这个网站，他不可能一直都访问着对吧？

他可能访问用了十几分钟20分钟他可能就下去了，对吧？或者是顶多一两个小时就下去了，因此过一会儿它这个数据给它清掉，他可能正好赶上他下去了，就比较合适好了。

总之我们这次课就是用rise把登录模块这几个相对来说我们之前实现的性能偏低的功能给它做一个提升，这是我们这节课的一个主要的目的。好接下来我就开始写代码，我已经打开了开发工具，首先我们先来定义 radius的 key，因为我们要往release里存数据，我就打开 release，keep you too。

好，放大一点。

我们首先解决第一个问题，就是存验证码的问题，我首先需要定义一个前缀，

验证码叫 captain好。我就用这个单词，然后我这里需要提供一个方法，然后来拼这个验证码的 k好，那么方法名我叫get看不上。

k。

那么我们获取验证码的时候，获取登登录验证码的时候，应该是验证码和某一个用户是相关的，对吧？

不同的用户他验证码应该是不一样的，我们要识别出来这个验证码是属于哪个用户，有同学可能会讲我就加一个uzid这个不对，你想想，当用户验证码的作用是干什么？是辅助用户去登录的，当他打开登录页面看到验证码的时候，这个时候他没登录，他还没登录，我们还不知道他是哪个user，还不知道他的user ID对吧？

所以这个时候你还传不了user ID，这个怎么办？我们还需要识别这个用户是谁，可以这样，我们可以在用户访问登录页面的时候给他发一个凭证，这个凭证是一个随机生成的字符串，发给他，他存到cookie里，我们以这个字符串来标识用户临时标一下就可以了，然后很快让怎么说过期就行了。

所以这里我就传一个字符串，我叫owner，就是验证码的拥有者，其实就是用户的一个临时的凭证。

好，这里我就 Return，然后刚才的前缀加上然后斯贝利分隔符加上owner，好，那么 k我就这样处理好了，那么处理好以后，我们需要改哪个地方代码，因为我们这个是重构，不是开发新的功能，你想我们验证码是在哪用的呢？是在登录的功能里用的，是在log in control里用的。对吧？所以我打开 love in controller。

好，然后往下找一下，首先方法get卡片就是获取验证码的方法，首先我要把这个方法进行一个重构，原来我们是把验证码存到了 session里，现在我就不用它了，我把 section它注掉，然后你看我们这个方法首先是生成验证码，逻辑不变，然后存入筛选，我需要加以修改，我把这句话注掉，取而代之我要把它存到这个readiness，而存到reads里，首先我们需要构造 key，而 key又需要一个验证码的归属者，所以我们首先来解决这个问题，

就是验证码的归属。

刚才我说了我临时给客户端颁发一个凭证，这个是一个随机生成的字符串就可以了。

好，我就随便卡片。

然后等于肯定那些优秀点儿，January uuid就行了，那么这个凭证需要发给客户端，客户端要用cookie保存，所以说这里我就创建一个cookie，好，那么 cookie中存的就是它了，key。然后值就是这个字符串。

然后我再设置一下酷k它的生存时间其实很短，就可以sat max我把它设置为60秒，因为验证码很快就失效，60秒一分钟足够了，然后 cookie点set pass，它的有效路径我就设置为整个项目下都有效contact pass，那么生效时间路径都设置完以后，那么我们把它发送给客户端，那就是response点爱的，cooking添加到response里就行了。

那么验证码的归属有了以后，接下来我就可以去存这个验证码是吧？那就将验证码那么存入，然后我先拼 k那就ready是t YouTube点get看看，然后的话把 owner传进去，好，有了t以后我要往 readiness里存数据，我就直接我需要用到的 ready stop雷达，需要把它注入给当前的 BIM。

好，那么再回到刚才的地方，那么这里我就调radius.Template。然后的话，点ops for。

value。

我们其实存的就是一个验证码，就是一个字符串，所以说普通的数据结构就可以，然后点sat，然后的话值是什么？是验证码就是 Text。

好，那么这句话写完以后，你可以再另起一行，然后再调用他们类的给 k设计一个有效的的时间，或者也可以在这里直接设置，其实这个方法有承载的方法可以直接设置时间60，60是单位是什么？后面再设一个单位 Time。

unit点sex60秒，

所以超过60秒它就失效了。

好，那么当我们首次访问登录页面的时候，那么 get capture这个方法被调用，然后的话这个验证码就生成了，然后也存到了reds里，存进去以后什么时候用呢？但是我们在对登录进行具体的验证的时候要用，所以下一个方法log in也需要处理。之前我们是从 session中的把验证码取到的，现在的话我们得需要从risk里取，所以首先我还是把注掉。

好，那么注掉以后这句话就不对了，我也把它注掉，然后差产我需要单独来取，从readiness里取，这个好办，首先我先把变量的声明好，开不成默认为not，接下来我要取了。

那么我们要从我们要从 readiness取的时候要有 key，而这个key又需要 cover owner，而他的o又需要从 cookie里去，所以这个方法我们还得再加上一个注解，就是从cookie中取值的一个方案。

Cookie value。我们刚才往里存的时候 key叫咖啡欧对吧？我可以copy一下，好，然后声明变量用来接收这个值，这个key有了我们得判断一下，万一说客户端 key已经失效了对吧？Key已经不存在了，所以你判断一下它存不存在，所以如果是这样。you choose is black加我is not black。

Now。

如果说 p不为空，比方说这个数据没有失效，这个时候我才去取，否则就取不到，我们后面就直接再回到登录页面给个提示，验证码不正确就完了。

好，这里我就构造 key， read this key。

把。

这个owner传进去，我得到了key，然后我就可以从reads里去取这个值 for value，然后的话点get把key传进去。

当然我们得到的是一个奥巴加肯，这里你需要转型成词句就可以了，那么取到以后后面他该怎么判断该怎么用，这个和以前保持一致，这就不用动了。

好了，你看现在我就把这个代码重构好了，重构好以后咱们来进行一个测试，好我把这个项目启动一下，启动以后我来访问一下。

好，我先访问了首页，我要登录点这个登录页面，你看验证码生成是没问题的，生成的时候每次生成它会给客户端下发一个临时的凭证，然后现在我试一下，我看我能不能登进去，如果能登进去，比方说验证码肯定是能正常的取到。再刷一下，好填写完验证码以后登录 Ok，说明我们改完以后这个功能还能用。比方说我们改的是ok的，否则的话如果你取不到这个值，肯定就功能就失效了，因为我把之前的代码已经给注掉了。

我们第一项要优化的内容就很简单就优化完了，下来我们再进行第二项的优化。第二个就是我们要使用 readiness来存登录凭证，那么首先也是像刚才一样，我也是先定义 key，那么再回到 ridsqq然后的话我再加一个k这个前缀我叫 prefix ticket凭证对吧？登录凭证 q就叫ticket，然后提供一个返回这个key的方法，登录的凭证。

那方法明儿我就get tk的k。

那么因为你要获得的是登录凭证的详细的数据，你肯定要传进来，你登录成功的凭证对吧？

那就把txt那个字符串传进来，我就委托你 price退给他，然后加上贝类的分隔符，然后再加上这个凭证字符串就可以了。

好，完成以后，我们现在要使用reds去存凭证，代替之前的那个表，所以我可以把之前的 log in take the map给它废弃掉。就不用了，当然我也别把它删掉，我就把它设置为不推荐使用，就以后别用了。那么我就在接口上再加一个注解。

Deep bracket声明一下，组件不推荐使用大家就别用。

好，那么然后我们需要去重构一下之前我们用到这个病的哪个地方，就像我们在哪用的，其实就是在 user service里，我们是在登录成功以后，我们生成了一个凭证，然后做了一个保存，但我们登录以后存了凭证还要注意，那么我们登出的时候推出的时候还要把凭证删掉，那个地方也要处理。

再有我们在user service里之前还写过一个查询凭证的方法，那么其他地方都是调这个方法去查凭证的，查也需要重构，总之这三个地方要重构，好我就打开 user service，好我回到顶部，好现在他不推荐使用了，我干脆在这里把它注掉，不用了，当然不用以后肯定会报一些错，我们去把这些错误处理掉就好了。

首先这个地方就是登陆的地方 log in，那么登录成功以后要保存凭证把它注掉，然后取而代之的是把它存到readiness里，首先我得听到key，对吧？写错了，red is key，you too点get take the key，你把 txt 传进来。

那么take it我们可以从对象中获取，好，那么取到以后就直接存，我们存的话调用release time play的还没有注入，还得注入一下，好，回到刚才的位置，我接着写，我要存档。Read this Complete。然后点 ops包发什么？这里我还是发value。

好，有人说你这是一个对象，你怎么按照字符串来存其实也好办，那么我们可以把对象序列化为一个字符串，Jason字符串存进去不就完了，对吧？也没有必要说非得把它存到一个集合里，因为每个人有一个自己的贴给他，所以说我们每个人单独存一份就可以了。那for value存点set，然后把 Key传进来，然后再把对象log in take it传进来。

然后那么 rides就会把对象序列化为一个字符串，机身格式的字符串，然后保存这样就可以了。

好，这个地方存我们退出的时候你要把它删掉，但其实不是真的删掉，而是把他的状态改为一，这里我注定要重新改，但是首先还是得拼这个k k就是传进来的参数。

那么。

有了k以后，我们需要先把老给你推给他，先把它取出来，然后把这个对象的状态改完以后再存回去，要做这样一个处理，我就取 ready template，点ops for value，然后点get把key传进去，我们返回的是一个对象 log in take it。

当然这里你需要强转，因为它默认的话返回的类型是opx好得到这个对象以后，我要改它的状态，site，status把它改为一，就表示删除态，然后这个时候我再把它存回去，key存进去，然后值还是对象传进去可以了，我们就把新改变的值覆盖了原有的值，好。

最后那么这里还有一个方法是查询凭证，那么其他的地方要调查去查凭证，现在他也不对了，我们也需要重构，现在需要从 release里查，这就很简单了，首先拼 key，听完 t以后就直接取对吧，我就直接 return。

然后的话是ready stuff late s for value get ready scheme。

然后这里也得强转型一下。

好这三个地方都改完以后咱们就处理完了，处理完以后咱们还是测一下，我把这个服务重启一下，我重新编译，然后重启自动重启。好，已经重启了。我再回到浏览器，这样我先退出一下。这个退出报错了，因为是我直接点了，我不应该直接点，因为咱们现在实现方式已经变了，我们之前刚才是用这个数据库来存的，现在用reads来存，所以说它数据取不到了，自然就有问题，我们应该重新访问一下。

好访问首页，你看访问首页它这个登录状态就没了，因为这个凭证从数据库就取不到了，从ready这里取，Ready这里还没存对吧？你要重登录了。好我点登录，然后我用这个CC来登录，都PP9，好立即登录成功了，那就说明了我们改造也是ok的。然后你再随便访问一下其他的一些功能，看一下，比如访问这个帖子也是可以的，访问这个消息也是ok的。

我们发起任何一次访问，那么在任何一个请求里，拦截器它都会获取给你买它都会掉犯的方法，说明这个方法起作用了，或者我们可以在这里打个断点也可以，然后你随便刷一下，刷新一下，你看就执行到这了，然后 f8 f8走哪去了？算了，我这f9他走丢了，那个焦点没有问题，就说明这个方法被正常的调用了就ok了。

好，最后我们再来实现使用rise，缓存用户信息的功能比较简单，唯一需要注意的是我们这只是缓存暂存，你一定要注意这个数据最终就是过一会儿你要自动把它删掉。

那么我们刚才存登录凭证，这个不删，为什么？因为我们之前存到my circle里也没有删，而删除只是改状态，因为我们要保留用户的登录的记录，这样的话有些功能需要用到，比如说将来我们比如要出一个功能，看一下你最早是什么时候注册的，最早是什么时候登陆的，或者说你每一年你登陆了多少天把我们给一些奖励，这样的功能就需要访问这样的数据，所以这个数据你要永久保存，不能把它删了，但这个缓存数据因为优的表还在，你只是临时存一下提高效率而已。

好，那么缓存用户数据其实我们主要是也是重构user service，其实主要重构的是 Find ID方法。因为你想我们每次请求获取凭证，取到凭证以后要根据凭证查用户就调这个方法对吧？很多地方都调这个方法被调的非常的频繁。我们把每个user缓存到了 Reads里，你摘掉这个方法的时候效率就提高了。

好，所以还是重构 service。我们做缓存的话基本上就是做这么几件事，或者说分这么几步，第一步我们先尝试就是我们在查询 User的时候，我们先尝试从缓存中取值，有可能能取到你就用，有可能取不到说明缓存里的数据没有初始化进去，你就要做初始化，这两件事可以把饭的又在外地重构了，然后有些地方我们会改变用户的数据，你改了他的头像等等，这个时候你改了用户数据以后，你是不是得把缓存更新一下，对吧？

我们有两种做法，一种是把缓存更新一下，另外一种做法是直接把缓存给它删了，当我数据发生变化的时候，我们一般都是用删除的方式，当我数据发生变化时我把缓存一删掉，那下一次请求你又去访问这个用户，他就又重查了一遍就ok了，这样简单，如果你去更新数据的话，一个是麻烦，第二个的话我们更新数据有可能会有并发的的问题，两个人同时更新可能会有并发的的问题，所以说还是删除比较干脆。

好，因此要想把缓存的管理好，我们主要说做三件事，我把写一下，第一个第一件事我们是优先从缓存。

中取值，

当查询时先不去访问circle，而是优先从缓存中取值。然后第二步如果取不到，那么取不到时就初始化。

缓存数据取。

到那就直接返回。好，然后第三个，当数据变更时，当数据发生变化时，我们要清除缓存数据。

这是我们要做的三件事，我把它分成三个方法在别的地方去调就可以了。

好，这个方法其实主要是给 User service内部调用的，它是私有的就可以，先从缓存中取值取到的是一个u字，所以说这个方法我就要get，catch就从catch中去缓存，但你取的时候你要存就在ID阶段是哪个优点。

好，这个就比较简单了，首先我要去拼 k好像 k我还没有处理是吧？K还没拼补充一个k这个就是perfect，然后的话user。

好。

那么有了这个前缀以后，我再提供一个方法，就是用户 get user key。

那么你需要传入user，ID我得知道是哪个用户，然后我就。

use点加上。

space加上 user ID。

好这就可以了，再回到刚才的地方，这地方我就可以拼 k了。

数据把。

优质ID传进去，我得到了一个k然后的话我就尝试从 Rides里去取这个值。Ready这里我们怎么存，我们就直接存 user对象，它会序列化成json字符串，所以还是用 ops for value，我就这里就 return readiness campaign，然后的话点ops for y6，然后点get ready ski。当然它的反馈类型是object，你要把它转型成user，这是从缓存中取值。

那么如果取不到的时候，我们需要初始化缓存就是往缓存里存数据，当然这个数据是来源于my circle，我再写一个方法，这个方法我也返回user，当我初始化完以后把初始化的数据返回回来，叫一类图开始。也是要传输优质ID，你要告诉我你要缓存哪个数据。好，那么这个时候我就需要从my circle里把这个数据查到，那就user map。第二谁来办ID查，到以后那么我们再去得到 key user key好，那么有了t以后我要往 rise里存数据了，sad，然后 ready skip存的是我们查到的优点，然后注意给一个过期时间，因为是缓存，它过一会要失效，这个时间我可以给它长一点3600秒。

一个小时，这个时间其实不短，已经基本上够用了，然后的话最终我就return，我得到 user就完了。

好最后我们再处理第三步，数据变更时我们要把缓存的清除，这个很简单，清除我不需要返回值，只是把数据删掉而已。方法名我。

叫可丽。

儿开始清除缓存，那么参数是优质ID是哪个优质的缓存要指定？首先我需要拼 Key还是这句话，然后的话就直接把这个缓存删掉就ok了，这里也不用设置它过期怎么样就删掉就可以了。

很简单，不是防外流，直接操作key把这key删掉。

delete。



然后的话ready ski，当然了这三步我们需要在别的地方去调，我们从前往后过一遍，首先 find by ID 中你肯定是要调的。

那么就像我刚才所说的三步，但是原来这句话先不要了，这回我们查询用户应该是这样做，首先先从开始里去查，有点等于 get catch。你把ID传进去，那么有可能能查到有可能查不到，判断一下，如果user它是now表示开始了没有怎么办？你就初始化这个缓存，对吧？盈利的开始然后把ID传进去，这个时候我们会得到一个user对吧？我把它赋给 u 则变量，这个u则变量不就有值了。

好，所以在这儿到这了 u 的一定会有值，我把它委特回去给调用方这就可以了。这个查询方法就搞定了，好。

然后再往下看，后面的话主要是看一下哪个地方我们是修改的user，改的地方我们都需要把缓存清理掉，再找一下，好这是音色的，这是添加一个用户，这不是修改，这个不用动，然后 Active vision，你看激活的时候我们不是修改的状态对吧？

这就是对用户的一个修改，这个时候我就可以把优质给它把缓存清理掉在这个地方。Clear cat，然后你把优质ID传进来好就行了。

然后再往下看，主要是看user map，看它掉没掉，不对的方法好。在这你看 update header么，update header的时候也是一样，那么用户被修改，这个时候我也需要去清理一个缓存，好我需把这行代码注掉，然后重新来写，那么因为是要在这句话之后清理缓存，当然我在这之前清理也行，就先把缓存清理掉，然后再更改，反正也可以，我先清理。

这样还是有点别扭，因为这个方法我们访问my circle和访问readiness没有办法把它们放到一个事物之内，他们之间的事物是分开的，所以说万一说更新失败了，但我们提前把缓存清理掉了，这也不太好，这样我还是还是换一种方式，还是先把它更新掉，如果万一更新失败了，缓存我就不用去清了，这样比较稳妥一点，我先把它租掉，我就先去更新，更新的时候会返回一个更新的行数，当然我最终要返回 rose，这更新以后我再去清理。这样比较好一点好，然后再往后看后面就没有了。

总之就一共这么几个地方是修改了用户，然后的话我们都对缓存进行了清理，缓存的设置就设置好了，下面我们再测一下，看有没有问题，重新启动一下。

好，启动完以后我打开浏览器，然后访问首页没报错，其实首页上我们访问的每一个这个帖子，在访问的时候他都去查了一下，user能显示出来说明应该没问题，但是我为了保险起见我还是确认一下，因为隐含的规则我们表面上看不出来，这样我在隐匿的开始里加个断点，看看这个方法什么时候被调好，回到首页刷新一下，它没有被掉，为什么？

因为我刚才访问了一次首页，这个数据已经被初始化好了，所以第二次调的时候他没走这个方法，如果说你把某一个用户的数据给它改了，然后的话它触发了口令，下次它就会掉了，我给你试一下，比如说我把 u 字的我换个头像换个头像，你看程序那卡住了，我们看一下卡住断点之处，因为头像变了，所以说数据被清理了，清理完以后他又执行了一个新的请求，然后这个时候就触发了掉了重新初始化的方法，我就f9走下去，回到浏览器这个图形变了没有问题。

好，那么这次课我们就使用reads把这三个地方进行了一个重构，重构以后的话，这个性能是有提升的，当然我们现在的话还看不出性能提升多少，那么在后面阶段我会有一个小节专门给大家演示一下，我们用一个专门的测试工具去测某一个功能，看它性能怎么样，然后的话我们再去用release对它做优化，然后再测看怎么样到那个时候会比较明显。这几个功能我们先不测了，你先把它按照这种方式优化好，这确实是能提高效率的。好，这节课我们就演示到这里，咱们下节课再见。

## 第5章 Kafka，构建TB级异步消息系统

### 5.1 阻塞队列

这一张卡我们主要是学习卡夫卡，那么卡夫卡它是一个目前来说性能最好的消息队列服务器，那么它能够处理tb级别的海量数据处理的数据量是非常大的。所谓的消息队列其实就是为了发消息的，你像游客网这样的社交网站发消息，发送消息是一个非常常见的功能，比如说当用户给你点个赞的时候，那么系统会自动的给你发一个消息，告诉你谁评论了你，那么点赞的时候关注的时候也都会做类似的处理，当然也有其他的功能也会发消息，那么这里所谓的发消息并不是说像我们之前开发的私信不是这个用户与用户之间的私信，而是服务器自动的给某些人发送系统的消息，或者说系统的通知，我们私信列表里不是有一个私信，旁边还有一个通知，那个通知我们要用这一张来解决的问题。

那么像这些功能是社交网站的非常常见的功能，而且这些功能它的频率是很高的，很多地方都能够触发发通知，所以说我们需要保证性能的问题，那么我们采用性能最好的卡夫卡去发消息，那么就能够保证性能，所以说这张我们主要是来解决发送系统级的消息或者说通知的这样一个问题。

好，那么第一节我们先来学习阻塞队列，有人会说为什么要学主队列？不是说要学这个卡夫卡吗？这里我解释一下，因为什么卡夫卡也好，还是其他的信息对这个框架也好，它都是框架，他都是做了深度的封装。

那么如果不用这些框架，我们能不能解决类似的问题是可以的，那么我们可以采用注册队列解决类似的问题，也可以用它来实现这个消息系统。

所以但是因为主队列它比较简单，那么照比这个卡夫卡要简单很多了，所以我们很容易就能了解理解主队列，理解它以后，他对你这个理解消息系统对你理解卡夫卡它底层做了哪些事儿很有帮助，而且帮助是很直观的。

再一个对我们将来的面试也有帮助，因为面试的时候面试官可能会问你卡不卡相关的问题，也可能会连带着问一些稍微原始的实现方式的一些问题，看看你 Java 的基本功怎么样。

因为主队列其实是Java自带的API具体来说什么是阻塞队列？其实它就是一个接口叫blocking q阻塞队列，那么这是Java核心API中的一个接口，那么这个接口主要是用来解决县城通信的问题。当然了我们解决线程通信的问题其实不只是这一种手段，我们利用什么不Jack的类当中的胃疼的饭这样的方法也是可以进行线程通信的，只不过那个方法比 Blog跟q它还更原始更麻烦。

所以说我们通常如果说我们在gdk中直接选APA解决这个问题，我们还是更倾向于选 block、q注册队列，那么注册队列它怎么解决问题，其实它主要是依靠两个方法，这两个方法都是阻塞的，一个是put往队列里存数据，一个是take从队列中拿出一个数据，是存一个，take是拿出一个。

那么右边我这里画了一个图，就是解释了一下 block q它的大概的一个工作的方式，那么中间这些蓝色的小格子就是我画的一个假设它就是主队列，假设队列的空间一共有8个位置，它里面能存8个数据，有8个空间，然后那么它是解决线程之间的通信的问题，比如说线程一和线程二想要通信，那就可以这样形成一调用库的方法，把数据放到队列里，然后线程二调，用这个方法从队列中取一个值。

好，有人说这样做有什么好处吗？你注意这样处理问题的方式，其实它是满足一种模式，叫做生产者与消费者模式。那么在这个场景下，新城一往队列中存数据，或者说它是产生数据的线程，我们称其为生产者线程。二是从队列中拿数据使用数据，我们称之为消费者，我想很容易理解它很直观。

那么 Blocking q其实它就是在两个线程之间，或者说生产者与消费者之间构建了一个桥梁，构建了一个缓冲的空间，避免你两个线程之间直接打交道。

如果两个线程抛开了 block q他直接打交道可能会有一些问题，比如说有可能我生产者他生产数据的速度特别快，远远快于消费的速度，你可想而知了，如果说他直接把数据给他，他这边处理的速度慢，很快，他就达到了瓶颈，这边生产的话就得但是如果说它没有一个缓冲的话，它还在一直生产，还在占用的 CPU的资源，它没有被阻塞，这个时候系统的性能就会白白的浪费掉。

或者反过来说，如果说生产的速度很慢，消费的速度很快，很快它的数据就被消费光了，但是这边还抢着要数据，还在占用系统 CPU资源，还在获取数据，但是取不到，所以说也是做无用功，也是浪费系统资源。

如果说中间有这么一层block and q那就好了，那么我生产的数据放到队列里，我就可以继续生产，直到如果说我生产的速度快，直到把 q堆满了，当队列满了以后，那么破的方法就被阻塞了，或者说线程一就阻塞了，你知道阻塞的时候就是说他什么也不做，只是在这等着这个时候他也不会占用系统资源，这就是阻塞的好处。

或者反过来说，比如说我生产的速度慢，消费的速度很快，这个时候他很快把队列中的数据消费光了，这里没有数据了，这个时候线程二被阻塞，也就是说他在这等着从队列中取值，当你有值的时候，这个时候它也不去占用 CPU资源，对系统的性能也没有影响。

总之这个阻塞队列其实是在两个线程之间起到了一个缓冲的作用，能够避免 CPU资源的被浪费掉，能够从这个角度来提升系统的性能。

好总而言之，那么总之我们多个线程之间的通信用这种方式是非常切实可行的，你有数据就往队列里丢，然后你该做的事儿如果丢满了你就阻塞就完了。这边反正我就等着有数据我就去处理，没有数据我就等着，这样的方式非常的高效。

那么 block q它是一个接口，我们不能直接拿来用，我们需要使用它的实现类，然后 GDP当中带来很多实现类，有这么多，有 every blocking q底层是用速度实现的队列，有link blocking q用链表实现的队列还有其他的等等各种队列的形式，然后当然这里的每一种队列，那么它都有它的一些特点，或者说它有它的一些特定的规则，特定的用法。

那么我们这次课其实主要是告诉大家阻塞队列是什么，然后的话它基本上是怎么用，所以我这里随便拿一个简单的瑞普拉QQ给你演示一下。至于其他的，如果你想了解自己查一下 Java的一篇就可以了。

好了，接下来我给大家写一个小的程序，咱们来演示一下阻塞队列它是怎么用的，那么我打开我的开发工具，这只是一个演示的案例，不算是我们正式代码，所以说我把它写到 test包下。

然后这里我新建一个类叫blocking q test，主测队列的一个测试的案例，然后主测队列我就直接在main方法中写程序进行测试，我就不写这个结论里头了。

然后阻塞队列它仅是满足这个生产者与消费者模式的，所以说你这里得有生产者现成，还得有消费者现成，通常我们会额外的定义两个类，所以我在这里直接定义一个生产者和消费者的类，当然你在外面定义也可以，在这里面直接定义也行，但前提是不要写public，因为一个文件里只有一个类可以是public，好，我上面一个类叫生产者，producer，那么因为它是一个线程，所以说我要实现 one more接口，当然你要实现这个接口需要实现它的乱方法。

好，那么除此以外，就是当我实例化 producer的时候，线程的时候，我要求你调用方把队列传进来，把阻塞队列传进来，因为我们这个线程是要交给阻塞队列去管理，调用方式你要把队列传进来，所以这里我增加一个变量来接收一下你传入的组织对立 block q。

好，那么 block control这里你可以通过泛型声明，它里边存的是什么东西，比如说我就存整数，但是你存什么都行，这里我简单点存整数，然后我就要给变量取名叫q我希望你在实例化 producer类的时候，就把 q传进来给我，所以我再给它加一个有参的构造器， public producer，然后的话你把 blocking q传进来，好，然后我把它赋值给当前对象的 q变量。

好了，然后接下来我就写 run方法，这只是一个小的事例，没有什么业务逻辑我就随便写了，这样我先去开始一下，开始到异常以后我就简单的输出一下，好，那么在这里比如说

我要频繁的不断的往队列中存放一些数据，这样比如说我要循环个100次，就是说我是一个生产者，我要生产一共生产100个数据，每一个数据我要把它放到队列里，放的时候我们最好中间有一个间隔，因为你不管是生产什么的企业也好，还是组件也好，它不可能是中间没有任何间隔的，所以说我们给他有一点间隔，所有的实力这里我停顿20毫秒，好，那么每每20毫秒我要生产一个数据，就是一个数 i我要把 i交给队列来管理我就q这是一个put，这个方法就是一个阻塞可以阻塞的方法，当队列满的时候它就能阻塞我们一会执行的时候能看到这个效果，我就把 i存进去，i就是我生产的数据。

好，然后放到队列中之后我打印点东西，打印什么？我首先打印当前的线程。是谁？Straight character。点。Get name，我获取当前线程名，然后加上生产。

我希望它打印出当前谁在生产，然后生产完以后我再打印出q点sin它生产了一个数据以后，这个时候队列中有多少个数，我想直观的看一下，好，就这样了。

总之我生产者的产能有限，它只能最多生产100个数，每次把树放到队列里，然后我打印了这么一句话就完了。好，接下来我们再写一个消费者线程，然后我们再去调，同样的我也是在这里写一个类消费者，然后也是你要实现让他们结合好。

接下来处理方式也和刚才的生产者线程是类似的，就是说我们在实例化类的时候也需要传入 q这样我copy一下，当然这个类名你要改一下，其他的地方不用变，肯定三码没有问题，那么这个乱方法之内我写一个逻辑，就是说我生产者消费者我就一直不停的消费，只要你对这种有数据我就消费，我这样写也是先处理个异常。

有异常我就随便输出一下。

好，然后只要有数据我就一直消费，我就用 while循环死循环，然后每次循环我要获取数据，获取数据就是调用队列的，每次take就是从队列中弹出对手的值，然后我们消费的时候也有一定的时间间隔，也有个时间间隔，我就是热爱的点，实力的这也是模拟一个真实的业务，你想我们用户访问一个网站，无论是它产生一些数据也好，还是它使用一些数据也好，它肯定是有时间间隔的，不可能是无间隔的。

而且用户消费数据使用数据时间间隔往往还是不确定的，有时快有时慢，所以这个时间我不把它写死，而生产者服务器它处理这个数据的时候，我把它间隔写死成20毫秒，但这个我不写死。

我写一个稍微随机一点，那就new render点。

next internet。

然后1000，这句话的意思是在0~1000之间属于一个数，你可以可想而知，我们大概率的隧道的数应该比20要大，对吧？

因为这么大一个范围它属于到20更大的比二十大的数的几率是更多的，也就是说其实我消费者他的消费能力没有生产者那么快，好，然后间隔完以后我就q点take使用这个数据，至于这个数据是几我并不关心，反正使用了，使用以后我再打印一句话，还是要spread一点儿。

其他人失败的概念内容，我想看一下当前的消费者线程是谁，然后写上他消费了，然后再消费完以后，我再把 q的size打印出来。

好知道消费完以后队列中有多少数据，好了，生产者与消费者线程的处理的逻辑我就简单写完了，只是一个模拟，写完以后我回到命运方法当中我要去调，那么我在命运方法当中先实例化主色队列，因为生产者与消费者要共用一个主色队列，不到QQ。62位 lock in q然后它的底层是由数组来实现的，所以说你要传入一个数字，表示说默认的话这个数组的长度是多少，或者说队列的容量是多大，我写个10，这个队列中最多只能存10个数。

然后接下来我就实例化一个生产者，我需要实例化一个生产者的线程，线程体是 producer，当然你new producer的时候需要传入q好，那么然后我要启动这个现象，new spread以后要启动start，这就是有一个生产者现场在不断的生产数据，但是它最多生产100个，然后我们再创建消费者现场6选3。然后也需要传入 q然后启动这里我把消费者多搞几个，就好像一个工厂一样，我工厂只有一个，但是消费者有很多个。

好我再写两个，我整个程序当中就一个生产者不断的生产数据，三个消费者同时消费数据并发消费数据。好了，这个测试程序就写完了，写完以后咱们执行一下，看看看一下什么效果，它是一边生产一边消费，需要有一个过程，但过一会就停了，因为最终的话也就是

一共生产100个数据，过一会就停了，咱们稍微等一下，停完以后我们再来从头开始看好。

停了。其实我们主要是看头和尾就能看出一些问题来。总体来说刚才我们也分析了是生产者生产的速度快，它每隔20毫秒就生产一个数据，消费的速度慢，但它的速度是随机的，也有可能快，但是快的几率很小。

因此你看一开始生产的生产者的速度快的优势就体现出来了，那就生产数生产数据生产完队列是一、234一直到10，因为它速度快，所以很快生产者就生产了10个数据把队列给堆满了，这个时候他就不生产了。

然后这个时候消费者开始消费完以后变成了酒，变成酒以后，队列布满了生产者又可以生产，立刻填充为10，然后再消费9又10，再消费9，又10，以此类推。

无论如何生产者生产完以后数量是10，它是这个规律是保持不变的，它不可能超过1010是上限，总之在中途就是不断的交替了，反正你你刚消费一个我就生产一个，而且你看消费者他应该是交替的，一消费，线程二消费，线程三消费对吧？

他们是交替进行的。

好，然后最终中间过程都是这样的，然后我们看最后快结束的地方，最后那么你看最后是生产者，最后第一百次生产完数据以后，这个时候队列是满的10，然后他就不生产了，这个时候这三个消费者就开始消费，这个时候没有生产的行为，消费完以后9876543210直到0为止，这个时候程序还没有结束，消费者还在等着消费，但是因为生产者不生产了，停产了，所以说这个时候程序就一直阻塞在这不动了，这是这样一个过程。

所以说通过这个过程头和尾能看到 Put和take方法它的一个主色的效果，你明白这一点就可以了。

好了，那么这次课我们主要是介绍了主色队列，介绍了生产者与消费者模式，然后的话也给大家写了一个小的事例，把主色队列做了一个演示，那么课后的话如果你有时间，最好你去看一看其他的时间内他们的作用，因为当你去面试的时候，面试官可能不只是问到这个地方，不只是问到这种程度，他有可能问你其他的这些实现类，你自己去看一看会比较好。好了，那么这次课我们就演示到这里，咱们下次课再见。

## 5.2 Kafka入门

36-Kafka入门.mp4

这次课咱们来正式的学习卡夫卡，主要是带着大家入个门，然后让你了解一下什么是卡夫卡，然后卡夫卡有的一些什么样的特点，它有哪些术语需要你去掌握。另外我们把这个卡夫卡就是安装好，然后的话演示一下它的一些常用的命令基本的操作是怎么样。好首先我们先对这个卡夫卡做一个了解，做一个简介，那么这是官网上解出来的一句话，说卡夫卡它是一个分布式的流媒体平台，有人会问不是说它是消息队列吗？

怎么变成了流媒体平台，这也是它的一个发展历程，它早先只是一个消息队列，只是用来发消息的。

后来他不断的完善，不断的扩充自己的功能，慢慢的它的功能就不只是局限于消息队列了，也能做了一些校区队列之外的事儿，所以说那么现在它这个功能比较综合，因此说它是一个分布式的流媒体平台，是因为它功能比较综合，我们从精神上来说它都能干什么？

第一个它能开发这个消息系统，这是它消息队列的功能。

第二个它可以做日志的收集采集，能处理大批量的日志，而且通过日志它能够分析很多内容，尤其是比如说你想分析用户的行为，对用户的行为追踪做一些分析，它也是可以搞得定，最后他还能够做流式处理，就是说你的数据流，你不是普通的数据是数据流，那么交给他他也能够帮你进行处理。

总之它的功能比较综合，但是我们的项目中不会用这么多功能，我们只用到它的发消息的功能，这也是它核心的重中之重的内容。好，那么卡不卡就像我们说的是一个性能最好的消息队列，它是怎么做做到这一点的？然后的话它有哪些特点才使得它性能高？这里我们也简单的介绍一下。

首先卡夫卡它具有高吞吐量的特点，就是它处理数据的能力强，可以处理tb级的海量数据，这是它的能力，那么它为什么具备这样的能力？那么其实这个消息持久化这个特点能够揭示这一点，持久化大家都理解，就是把数据永久的保存到某一个介质里，类似于硬盘这样的永久存储的介质里，也就是说它会把消息存到硬盘上，而不是简单的存到内存里。

其实很独整息队列，他们是像我们上课所写的 Block, q他数据就存在了，对这里，然后我消费完以后，他就出兑了就完了，他没有永久保存，但是卡夫卡它是长久保存消息存到硬盘里去了，那么硬盘大家都知道它的硬盘的价格要比内存要低，而且硬盘的空间要比内存要大很多，这是它能够处理海量数据的一个前提，它把数据存到硬盘里，所以说才能处理这么大规模的数据。

那么有人可能会质疑说你存在硬盘里的话，读取硬盘读写硬盘的速度要远远低于读读取内存是吧？他说是不是性能就低了，怎么还能保持高性能，其实这是我们大家很多人的一个误解，就是说读写读取硬盘读写硬盘的效率高与低取决于你对硬盘的使用。

其实实际上我们对硬盘顺序读写，不是随机读写顺序读写是性能是很高的，性能甚至高于对内存的随机读取。

所以那卡夫卡就利用了这一点，它对硬盘的读写都是按照这个顺序的方式来进行的，保证了一个很高的性能，同时硬盘的空间就足够大，所以它能够保证它能处理海量的数据，保证高吞吐量。

然后第三个特点是它具有高可靠性，那是因为它是一个分布式的这个服务器，就是它可以做集群部署，你一台服务器挂了，还有另一台对吧？所以说它是靠分布式保证的高可靠性，有容错的这个能力。

然后第四个它是有高扩展性，就是说我这个集群如果说服务器不够用了，我想加一个服务器，它很方便简单的配一配就可以。

总之卡夫卡就有这么几个优点，因此它是目前最为流行性能最好的消息队列，所以我们就选择用它，然后你要想使用这个卡不卡，你需要理解卡不卡的一些术语，否则的话用的时候就会感觉很别扭。

首先第一个术语是broker，broker是什么意思？其实就是卡夫卡的服务器，那么卡夫卡集群当中的每一台服务器我们称其为一个博客，记住就好了。

第二个概念是组keep，即使组keep它并不是卡夫卡里的一个概念，它是一个独立的软件一个应用，那么它能够用来管理其他的集群，那卡夫卡因为也要做集群，所以说它可以用主people来管理它的集群，那么我们在卡夫卡的时候，我可以单独安一个主keep，或者我直接下载一个卡夫卡，它里边也可以内置主keep也是可以的，我们一会我们就直接下载一个卡夫卡，然后的话就使用它里面内置的主key本就方便一点。

然后再看术语叫topic。

主题说到这儿我先再多说一句，就是说消息队列实现的方式大致有两种，一种是点对点的方式，我们上次课写的 Block, q其实就是点对点的方式，就是生产者把数据放到了一个队列里，消费者就从队列里取值，消费者可能有多，如果a消费者取到了这个数据，这数据就出对了。

那么b消费者取的是另外的数据，那么是会产生冲突的，不会重复的。每个消费者每个数据只被一个消费者消费这种情况，这叫点对点。

那么还有一种实现方式叫发布订阅模式，就是说我把消息放到了某一个位置，然后可以有很多个消费者同时关注这个位置，订阅这个位置，然后读取消息，这个时候这个消息可以被多个消费者同时读到，或者先后读到，是这样的。

好，那么这个卡夫卡采用的是发布订阅模式是第二种方式，这种方式其实比较灵活，也比较方便，那么他把生产者把消息发布到的位置，那个空间就叫topic就叫主题，你就理解为是一个文件夹，用来存放消息的一个位置好。

然后 partition是分区的意思，是对主题这个位置的一个分区，你看右边这个图，那么总体来说，比如说我这么这个空间是一个主题，可以用来存放消息，那为了增强这个并发能力，为了增强这个服务器处理能力，他对 topic这个位置又做了一个分区，又把它分为了012，当然可以任意多个区域，这里画的图是三个区域，这个图是来源于官网，然后这个时候就可以产生多线程的方式，同时向这多个区域多个分区里写数据，这样并发能力就强了，所以一个主题又可以分为多个分区，然后每一个分区它是从前往后按照顺序往里追加数据的，追加写入数据的，可能是一条消息两条消息三条消息不断的会堆到分区里。

然后比如说你看后面这个图，这个是一个宏观，这是一个微观，针对某一个分区而言，那我生产者是在队尾添加数据追加数据，如果你消费者读数据的时候，这里有很多条消息，你就需要按照顺序按照索引去读。

比如说9~10，这两部分是合起来是一个消息，我就从9读到10就读到这个消息。所以说这个off site是什么意思？Off site其实就是这个消息在分区内存放的索引序列。

好，那么另外还有两个概念，还有一个概念叫replica，remake副本。什么叫副本？副本其实就是对数据做备份的，因为卡夫卡是一个分布式的这个消息引擎，那么为了让数据更可靠，他不是把数据只存一份，它是通过副本的形式对这个数据其实每一个分区有多个副本，存多份，万一某一个分某一个副本坏了对吧？我还有备份这个意思。

所以分区是为了提高容错率，冗余的多存几份数据，然后副本是分为leader、副本和 follow副本、主副本和随从副本。

那么主副本他的能力比较强，当你像从分区读数据的时候，或尝试从分区获取数据的时候，消费者那么主副本可以给你，这个数据可以给你做出响应，它能够处理你的请求，而从副本他只是从主副本备份数据，然后他不负责做响应。

那么万一说某一个时刻主副本挂掉了，这个时候集群就会从众多的重复本中选一个新的作为主副本作为leader，这是分布式的时候它的一个玩法。

好了，总之这些概念我们先普及一下，要然后我们用的时候好知道这什么是博客，什么是组批本，什么是topic等等，好不是那么陌生。

好了，那么了解了这些概念以后，接下来我们就去下载安装卡夫卡，然后来看一下它的一些基本的操作，那么卡夫卡是阿帕奇推出的一个软件，所以说非常的流行，那么它的官网是卡夫卡，点阿帕奇点org我已经提前把这个网站打开了，然后如果你想下载的话，这不有个登录的对吧？我点登录的。

然后这里目前最新的版本是2.3.0，然后这是带源码的，就是不带源码的。

然后不带源码有两个版本，一个是斯卡拉2.11，一个是二点12，它的底层是由斯卡拉来开发的，这个是斯卡拉的版本，然后他推荐的是2.12，他建议是我们就点这个，有人说它是tgj这样的一个格式，这不是Linux下的压缩包吗？能用吗？可以的。

卡夫卡它部分操作系统，反正它都是一个包，然后你下载完以后，解压缩它包里既有支持Linux的命令，又有windows的命令，它一个包就搞定了，挺省事的。

然后这里有很多链接你都可以点，然后它顶上给你推荐，根据我们的Ip它可以推荐你访问网址，这是一个镜像网址，国内的速度会比较快一点，你可以点这个。

好，我已经提前下载好了，所以说我就不点了，我下载的位置在这。

那么安装的话就很简单，就把它解压缩就可以了，我把它解压缩到某一个地方去解压缩，我的放到d盘work的下面，解压做完以后看d盘work在这儿，卡夫卡下划线2.12-2.3.0，然后那么安装完以后，这个卡夫卡你需要做一些配置，配什么？看一下配置主要是在 Conflict里。然后另外我们要启动卡夫卡，访问卡夫塔需要用它的里面带的命令都在变，一下默认的是Linux命令sh结尾，然后这里又有windows命令 bat结尾。

好，我退回去先配着 can figure，首先我要配一下主keep，就是管理集群的工具，这个是它的配置文件配一下配什么？其实主要是配 Data dir那么主keep在运行的时候会产生一些数据，这个数据要存在，你看它的默认的存放的路径是Linux系统的路径，我们没有这样的路径，这你得改，我改一下，我改成c c盘，然后work，然后data，比如说我要把这个数据放到目录下，然后主题本目录下我看一下work，data目录有了，但主keep没有关系，一会运行的时候它会自动创建就这样了，然后再配，另外一个server点process也是打开，这里其实也有很多配置项，我们不用都配，这个是必须要配的。

Log点dirs就是卡不卡？

它日志文件存放的位置，它默认的路径也是不符合我们 windows系统的需求，我改一下c盘，冒号，work data，然后data下放到这个目录下就行了，其他的就不用动了，好这样的话这个组keep我就已经这个卡夫卡我就已经装好了，那么装好以后咱们要试一下，当我们试的时候其实是要访问并一下 windows下面的这些命令我给你演示一下，但是命令你可以从官网的手册里能查到，我给你看一下在哪查退回去。

在官网首页上，这不是有一个到k门txt文档对吧？你点这个文档，然后点完以后，你可以看一点一点1介绍，然后1.3quick start， quick开始。这里边就是有一些快速的常见的命令的介绍，就在后面了。

好，那么因为比较多，我不给大家挨个细看了，我就是挑基本的主要的给大家演示一下，能够把它用起来，大家了解一下它到底能干什么就行了。

好，首先我要启动命令行 come on的，然后我要敲命令启动，得先启动主keep，因为卡不卡依赖于主keep，好，我需要先CD到CD到6，我刚才是不是写错了，我那个配文件配错了，我的意思是把这个数据放到d盘work定损项，然后我卡萨也是装的d版，但是我刚才写到了c版写错地方了，应该是d版改一下这是d版。

好，那么再回到命令行，首先我要去d盘，因为塔普卡装到装到了 d盘，然后我CD到卡普卡的根目录下，CD d盘，然后 work，然后是卡夫卡到这个目录下，然后我要执行b下的命令，我可以这样，我直接敲b目录，然后因为我是windows系统，所以我得进windows目录下，然后我要启动主keep，那里边有一个主keep杠server杠start点bat我要执行它，我就这样主keep刚sir刚start的bat然后我启动的时候我要指定我要用肯目录下主key products配置文件，我可以空格写上配置文件，你注意我当前路径是这个路径下，我再访问config，然后听下组keep up case。

好，总之整句话的意思是我要执行命令，然后的话用配置文件来执行来启动这个组keep回车。

好，主keep启动的比較快，启动完了启动完以后还得再启动一个命令行窗口，我们再去启动这个卡夫卡，还是要CD到对应的目录下，d盘work卡不卡CD过来以后启动卡不卡也是类似的，有一个叫卡不卡，刚server刚start的bat，我还是用刚才的方式b然后 windows卡夫卡server start bat然后也是我要用到刚才我们改的配置文件就 server叫practice，然后他也在啃录像，server点practice，好然后回车，你看这个卡夫卡也启动了，启动完以后这个时候你可以去看一下咱们 d盘work data，你看这个data下是不是有一个目录叫主k本，主k本上有这个数据的。

另外再看 data下有了一个卡卡洛克斯，这里边有数据了，我们在启动的时候它自动创建的数据，现在主p和卡夫卡都已经启动了，然后我要使用这个卡夫卡这样我再启动一个命令，行，林天凡他使用的方式确实有点啰嗦，好我再启动命令行，然后的话我要我要使用卡夫卡的命令工具，我要CD到对应的目录下先去d盘CD d盘work，然后的话卡不卡，然后的话并然后 windows我直接 CD到包含命令行工具的路径下，然后我这里要干嘛？

我首先要创建主题，因为卡不卡作为一个消息队列，它是采用的是发布订阅模式，你要把消息发布到某一个主题下，首先你要创建一个主题，再强调一遍，这个主题是什么概念？第一个它代表了一个位置，第二个它也代表了一种消息的类别，消息的分类。



比如说这是点赞的消息，我就一个主题，那是关注的消息，我又一个主题，所以说它有双重含义，我们首先需要创建这个主题，而创建主题我们需要利用windows目录下有个工具叫卡夫卡topics点bat利用那个工具这里怎么写我就直接写了卡夫卡 topics点bat我调这个工具，然后我要创建主题杠杠，create然后我要创建主题，我要在哪个服务器上创建主题，杠杠，bootstrap server，空格。

House冒号9092，你注意我们装的卡夫卡它默认的端口是9092，表示我要访问这个服务器，好，然后空格杠杠，reputation就是副本一就比方说你要创建多少个副本，我这只创建一个就得了，然后再来空格，party cense，一比如说分区你要几个分区，一个当然你也可以写多个，当然命令比较长，你也可以从手册上去找命令是有的。

好，然后杠一空格加杠TP就这里说我要创新，我要在这个服务器上创建主题那么一个副本一个分区。然后刚刚topic是主题的名字是什么？我叫test回车，它没有任何提示，这就表示创建好了。

为了确认一下到底好不好，我们可以再执行一下 Topics，命令，去查看一下所有的主题，它可以查看一下卡副卡杠topics，点dad，然后的话查看aaah杠least，然后杠，both drop，然后 server，然后 local house，9092其实和上面这个create差不多，也需要指定你要看的是哪个服务器，因为我们这里是没有搭建集群的，但实际生产环境中可以搭建集群，可能有多个服务器，你要看的是哪个服务器，所以你要指定这个服务器9092好这就可以了，回车那就列举出来的所有的主题只有一个就test。

好了，那么既然主题创建好以后，下面我就要往这个主题上发送消息，我们是以生产者模式的身份来发送，所以我们需要调用生产者点bat，就producer点bat命令才能发送，就这掉了。

卡副卡杠看守杠producer点bat调命令工具，然后杠 god list broker不是服务器的意思吗？其实它的意思和有点类似，就是服务器列表，你要往哪些服务器上去发这个消息，我们现在只有一个 Local house，9092往这上发消息，然后往这个服务器上你往哪个主题上发消息，杠杠topic test，我要往这个主题上发消息，回车以后很快它出了一个小三角，表示你可以写字了，你可以开始发了，我就写了。

Hello，hello，world现在又发了两条消息，每发一个回车发出两条消息发出去了，然后此时生产者在这阻塞着在等着你发的更多的消息，但我没发。好，接下来我们得再去通过消费者的命令去看一下我能不能得到这个消息。因为当前窗口被生产者占用了，所以我再启动一个窗口，这个窗口代表的是消费者，我要去读这个消息，当然了为了执行命令，首先我也需要CD到 D盘的目录下，d盘cdd d work卡不卡？

并windows。

好，我CD到目录下，然后我就像调生产者命令一样调消费者的命令是吧？卡夫卡看守看沙漠点办，然后刚刚 boss抓sir都不好死，9092意思是你要从哪个服务器去读数据？我要从9692读数据，然后读哪个主题？Topic，test，读 test，主题中的数据，然后站杠杠strong。比给你意思是我从头开始读所有的我就回车，稍等片刻之后，他很快就读到了你之前发过的消息。

Hello world。只要你发到这个位置，发到这个主题下，那么我就能从这个主题下依次读到这个数据，生产者与消费者他们之间就这样产生了通信。好，再来。你看这边我这放着这边再来再写，比如说 How how are you回车，你看这接到了我就这样写，好，很快就接到了，这两端就可以产生通信了。

当然我们这只是通过命令行的方式给你演示一下这个卡夫卡它到底是一个什么样的形式。

当然这个你了解一下对卡夫卡有一个直观的印象，也检验一下你的工具的安装的好不好好，不好用，然后后面我们更多的是采用java代码，尤其是采用spring去整合卡夫卡去访问卡夫卡，那么当然那个是更重要的内容。好，那么这次课我们就对卡夫卡做了一个了解，课后大家把它安装好，然后也把命令敲一敲，对它熟悉一下。好了，这次课我就演示到这里，咱们下次课再见。

## 5.3 Spring整合Kafka

37-Spring整合Kafka.mp4

这次课。

我们来学习如何使用spring整个卡夫卡，那么spring整个卡夫卡这个是比较方便的，尤其是在 spring布的框架的帮助之下，非常的容易，我们整合的时候第一步你得先引入依赖，我们需要去搜到包 spring卡夫卡。然后第二步我们需要对卡夫卡做一些必要的配置，这里主要是配卡夫卡的 server，因为你要让spring知道我要访问哪个服务器，另外还需要对消费者做出配置两方面。

最后就是写代码了，写代码的时候，因为消息队列它是满足生产者与消费者模式，所以说我们写代码也是抓住这两点。

那么对于生产者他发布消息，我们主要是利用 spring给我们提供了一个整合好的类叫卡夫卡template，那么这个类它带了一个方法叫散的，然后的话有两参数，一个是传入主题，第二个是传入你要发布的具体的数据消息就非常简单，对于消费者来说写代码的方式是这样，是通过一个注解来实现的，当然我们得把注解加到一个方法之上，这个注解是卡夫卡listener，就是一个卡夫卡的监听器，然后的话这里边你要写topics就主题，而且是复数多个主题，等于大括号里边写上一个或多个主题，表示说下面这个方法是监听这些主题，什么意思？

就是说当我这个服务启动以后，那么 spring就会利用这个方法，它就会监听你配的主题，一旦它发现这个主题上有消息，那么它就会调这个方法去处理这个消息。

然后它会自动的把这个消息包装成一个 can summer record然后传进来，你就通过 record得到消息做你自己想做的处理，总之非常简单。

好，下面我就给大家演示一下整个整合的过程，然后我们写一个小的例子来看一下，整合以后它的效果。首先我们先来引入依赖包，我已经提前的把每本日抛c的到com打开了，我们搜一下spring，然后的话是卡不卡？

好，这个就是 Spring卡，sport你看这个包名叫sprig卡就他了，然后点一下，然后我选一个比较新的版本把拷过来，打开这个idea，把它粘贴到泡沫点xml里，我把它粘贴到最后的位置，但是其实我还是把这个版本去掉，因为负泡沫里已经声明了，它默认的版本，而且这个版本是它经过测试是没有问题的，万一我们选一个比它新的版本，万一产生兼容性问题对吧？

不太好，所以说我们就会把它删掉，然后就利用负泡沫中所声明的版本，它是什么就是什么就好了，就这样就可以了。

那么引入了依赖以后，第二步就是要对卡夫卡做配置，主要是在 application practice里去配，这儿其实我们是针对这个类进行配置，它那个类叫卡夫卡practice。好，这个类我就不看了，然后我就直接的写我应该配什么。

当然它的配置项很多，但是我们比较有必要的是要配如下几项内容，第一个是spring卡夫卡bootstrap service，首先是要配 server，我们的 server只有一个local house，929092它好，这是第一项配置，除了 Siril以外，我们还要配的是消费者 concern，你这样写我 copy一下 spring点卡夫卡，点肯萨姆，然后我们要配的是消费者的组分组的ID。

好，这个值其实是在肯萨姆的一个配置文件里有，咱们可以去看一下，然后打开d盘，然后 work卡不卡？

看所有的配置文件都在这里，那么很显然这就是消费者的配置文件。打开看一下，咱们上次都没有看，你看这里面它也配了，这个服务器是它默认的，然后的话这就是组ID，group点ID其实我们刚才 Group ID就配它，然后它的默认值是叫test consumer group。

这里我改了一下，我改成了community，当然你改不改都行，这里我改了，如果你要改的话，你要注意改完以后你要重启服务才有效，我就把粘贴过来它了。

然后再来对于肯萨姆还有其他的一些配置，那么肯萨姆然后点有一项配置叫尼泊凹凸，k米特是否自动提交？

那么这个参数的意思是是否自动提交消费者的偏移量，我们从斯布林布特的官方手册最后的附录里能够看到它的一个解释。

那么咱们上次不是讲了消费者在读取消息的时候，它是按偏移量来读取的，那么就是说偏移量它读完以后，你要不要把它记录下来，要不要做提交，是这个意思，好，我们是要自动提交的，然后通常我们还会再来一个参数，就是沙漠点。奥to commit英特尔。好，那么它的意思是自动提交的一个频率，你多久自动提交一次？这里我写3000，就3000毫秒三秒。

好了，那么配置这几项就可以了。配置完以后下面我们就一个写一段测试代码，然后我们来看一下怎么去用这个卡夫卡，那么我在这个测试包下新建一个测试类，这个类名我叫卡夫卡太子。

好，那么在类的前面我需要引入那么几个注解，我随便打开一个测试类，然后 Copy一下，然后我在测试的方法写一个测试方法，放大一点，方法名我就叫txt卡不卡？

我们测试无非就是我要测试用生产者发一个消息，然后看一下消费者能不能自动的收到这个消息，把它消费掉。

当然我们消费主要是把这个消息打印出来，看一下能不能收到就可以了，这里我们不涉及业务层面的问题，通常我们在写的时候会把生产者与消费者的代码进行一个各自的封装，我还是为了省事，我就不新建这个类了，不新建文件了，我就在当前的文件里写两个类，分别封装生产者与消费者的代码，那先写一个生产者克拉斯。

卡夫卡。

producer。

好，那么这里我希望这个病由 spring容器来管理，我在这个病之前加上一个control。好，然后生产者刚才我们PPT上也说了，生产者发一个消息主要是依靠工具叫卡夫卡，他们累的那个工具是被spring整合了，它在容器里，所以我把它注入进来。

好，然后我再增加一个方法供外界调用，你调这个方法就是发消息，我送他一个方法，这个方法无需返回值方法名，我叫send message发消息。

那么发送消息的时候，首先你要传入消息的主题，另外你要指定那消息的内容。

好，那么你调这个方法的时候，我发消息就是我去调用他们类的去发就可以了，他不累的点散的，然后的话 topic，然后 content就完了，这是生产者这个病，那么接下来我再封装一个消费者病class，然后卡夫卡跟萨姆，好。那么消费者他不需要依赖这个template，因为它是被动的，它是自动的去处理这个消息，我们需要用到注解，卡夫卡为什么？

然后注解里主要是要写一个参数 topics，等于你要关注你要监听的那些个就是主题，比如说我一会要发的一个主题叫test，我要坚定的就是test主题，那么注解写完以后，一旦服务启动了，那么spring就会自动的去监听 test主题，或者说一他就是一个消费者的身份，他就是有一个县城阻塞在那，一直试图去读取 test主题下的消息，但是是阻塞的状态。

如果你没有消息就阻塞着，如果你有消息他立刻就读，就是这样的方式好。

然后如果说一旦有消息读到以后，他会交给他所修饰的方法去处理，所以这里我需要加一个方法，这个方法是处理掉一个消息，这个方法名我叫憨豆，处理一个消息message，然后他调这个方法的时候会把消息加以封装，封装成 consumer record我们从 record里就能读到原始的消息。

好，我尝试读一下。很简单 Recode点get错了recode点，买点这就可以了。

好，现在我就把生产者与消费者这两个病封装好了，下面我们就在测试方法里去调一下，我希望这样，当我运行这个测试方法的时候，我就发一个消息，但是发完消息以后，我不希望这个程序就立刻结束，因为你立刻结束，我们就看不到消费者消费这个消息的过程了，对吧？不能立刻结束怎么办？

我可以把当前的主线程实例问一下，阻塞在这，等一会儿不用太长，等个几秒就够，在等的过程中消费者它就会自动的收到这个消息，就会调这个方法，触发这个方法去处理这个消息，我们就能看到这个输出，这样就可以。

总之生产者发消息是我们主动去调去发的，我希望什么时候发就什么时候调。

消费者他处理这个消息是被动的，一旦你的队列中有消息，它就自动处理，但是可能会略有一点点延迟，因为你队列里边有好独整息对吧？他排队处理到你可能会有一点点延迟，但延迟不会很多。

好，我这里就写了，首先我多发几个消息，但是我发消息我需要依赖这个病对吧？我把它注入到当前的Test这个病当中，卡夫卡 producer，然后的话卡夫卡producer好我要发消息，那就 cup producer点 send message。这个主题我就叫test，这个主题其实我们之前已经创建好了，然后随便写为一号，好像再来一个还是一个主题再来一个消息说在吗？

完了这就发完了，发完以后我说了这个程序我不希望它结束，我希望它阻塞一会儿等一会儿消费者他消费的过程，我就写一下spread点，sleep sleep1000×10，1000毫秒×10就10秒钟对吧？当然这句话你需要take action这样就行了。好，那么写完以后我来执行一下 Right？好。你看已经输出了，你好，在吗？说明消费者已经成功的消费了，这个消息已经得到了，但是程序还没结束，稍微等一会等到10秒才结束。因为我主色了10秒，好就比方说成功了，所以说你一定要体会得到消息队列它的这种感觉，生产者发消息是我们主动去调的，而消费者处理消息是被动自动去调的，一定要把这个搞清楚。

好了，那么这次课我们是不用整合，卡夫卡就演示到这里，咱们下次课再见。

## 5.4 发送系统通知

38-发送系统通知.mp4

这次课咱们来实现发布系统通知的功能。那么具体来说是当有一个用户他发表了评论以后，那么我们系统要通知被评论的那个人，同理点赞的时候，关注的时候，我们都要通知了被点赞被关注的那个人，所以你可想而知发布通知的操作它是非常的频繁，因为我们用户群有很多这个点赞评论关注的行为是很频繁的，对吧？

系统的发布通知是一个非常频繁的行为，所以重也要考虑性能的问题，那么为了保证性能提高性能，我们就需要用到我们刚刚所学的卡夫卡消息队列，为什么需要用到消息队列解决问题呢？

这里我们要阐述一下，因为如果我们采用消息对立的方式解决问题，很显然评论点赞关注就是三类，不同的事儿我们可以定义三类，不同的主题，评论一种主题，然后点赞一个主题，关注一个主题。

一旦这件事发生的时候，我们就可以把这件事包装成一条消息，然后扔到队列里，扔到它相应的队列里，扔到队列里以后，那么我当前这个线程其实就是生产者消息的生产者，他就可以继续处理下一条消息，继续做别的事情，他就不用再管后续的业务了，而后续的业务是由消费者线程去处理的。

也就是说生产者和消费者，其实他们在处理的业务的时候，其实它是并发的，它是可以同时进行的，这种方式我们称之为异步。

好，我们再总结一下，我们在评论点赞关注以后，我们不是要发通知，其实就是要发一条消息，我把这个消息让他队列里就不用管了，我可以继续处理别的事情，然后由专门的消费者线程去处理它，这样的话我们称这种方式为并发的方式为e5的方式，同时进行的方式。

具体来说消费者现场要干什么？它主要是从队列中读到这个消息，然后的话最终往 Message表里存一条数据，因为你看我们之前做过私信，我们发一条消息无非就是往这个表里存一条相关的数据就可以了，对吧？

所以他就是做这样一件事。

好，那么怎么说，从这个技术角度来说，我们用的是卡夫卡消息队列来解决问题，然后不同的操作我们用不同的主题来解决，这是技术角度。那么从业务的角度，我们称解决问题的方式是这种事件驱动的方式，我们称评论是一个事件，点赞是一个事件，关注是一个事件，这个好理解三件事对吧？

我们在解决问题的时候，是以这个事件为。

这个目标。

以事件为主体来解决的，所以我们在开发的时候是基于事件对这个代码的逻辑再进一步的封装，在卡夫卡的框架的基础上，以事件为主体进行封装，而不是这个消息，这是我们通常的解决这类问题的一种习惯。

那么具体来说我们称评论点赞关注这个时候发布通知，这是触发这个事件，这是事件的触发时机。

我们具体要开发的时候需要第一个我们需要把这个事件对象定义出来，对事件发生的时候所需的数据进行封装，而不是说我们就发一条消息就拼一个字符串，那样的话太简陋了，我们拼一个事件对象，这个事件对象中包含了这条消息所需要的所有的数据，至于说你消费者想怎么拼，那就是你的事，这样的话会更具有扩展性一些，而不是拼死一个字符串。

好。我们封装了这样一个事件对象以后，我们因为是产生消息对的解决问题，所以说我们需要开发生产者开发消费者，生产什么？我们生产的是事件，消费的也是事件。当然你消费了这个事件的时候，最终你是要把这个事件转换成一条消息插入到数据库里，所以整个我们的开发是做了一个更高级的抽象，更高层的设计是面向事件驱动编程的。

好了，那么大致了解我们要解决问题的方式以后，接下来我们就来写这个程序，我打开我的开发工具，刚才也说了，首先我们需要定义一个对象对事件进行封装，要封装什么？事件触发的时候，相关的一切的信息。

好，其实这也就是一个实体封装数据的实体。

那么我在安贴包下新建一个类叫int实践，然后这里我提供一些属性，首先 Topic主题，不同类型其实就是事件的类型，我们再从卡夫卡的角度来说，管它叫主题，就是事件的类型好，然后这个事件是谁发出的，你要加以记录，我好能追溯到它的来源对吧？

因此我再加一个user，ID这是事件的触发的人，然后这个人做了什么操作，他是点了赞还是回复了还是关注了对吧？

那么我们要知道这个事件发生在哪个实体之上，因此我们还需要记录 NTT太实体类型以及 nttid实体ID另外实体的作者是谁？帖子也好，评论也好，它的作者是谁？我们顺便也记录一下，因为后面也会用到帖子作者，实体作者我叫安 itt user ID这样。

好，那么除此以外，那么因为我们这个事件对象它要具有通用性，那么有可能在某些其他的业务，我们现在只处理这三种事件，将来可能还会有更多的事件。

那么在处理其他的事件的时候，那么可能还有一些其他的特殊的数据需要记录，但我又没法去预判我到底还需要哪些字段，哪些属性怎么办？我再加一个map，我把其他的所有的额外的数据全都存到 Map里，这样的话让它具有一定的扩展性。我就叫data，数据然后我先给它识别化一下新卖的。

好了，那么有了这些属性以后，我们当前的业务很容易用这些字段封装，那么其他的业务也差不多，然后如果实在不够用map来封装基本上就ok了。

当然了。

接下来我得生成对应的 Set。

好，那么生成盖特赛的以后，这里我再稍微的改一下这些方法，为的是什么呢？为的是我们在构造这个对象的时候方便怎么改造，你看我这样做盖的方法有反馈值了，而sat没有反馈值，那么我把它加一个返回值返回就返回当前的类型，然后这里我有一天类似这样做有什么好处？

就是当我们在调用 set方法的时候，我set了 topic，肯定我还要set的其他的属性，我set完topic以后又返回了当前的对象，我又可以调用当前对象的其他的set方法，所以我们在写代码的时候就可以一曼特点赛的topic，再点赛的优质ID在点什么的，这样就方便，当然有的同学可能会质疑你为啥不写一个有参的构造器，然后一股脑把参数传进去，那样也可以，但问题是我们的属性比较多，如果说你加一个很多参数的构造器其实也很麻烦。

再一个有的时候可能某一个字段，比如字段有的时候不需要传对吧？你构造器里又声明了，有的时候也很麻烦，我这样的方式它既灵活，然后的话又方便，所以说我们通常可以这样做。

好 Site Top给我做这样的处理以后，其他的site也是类似的处理，然后这个也是返 site返回emit券类似 site itt type一样， set Entity ID悠哉ID。

好，那么最后对于卖法我要稍微再就特殊处理一下，那么 get data返回整个map中的数据不动了，那么这个site data。

我。

也是做类似的处理，但是其实我site data我想这样，我不想直接让外界传过来，整个一个麦我稍微改造一下我改造成什么，外界调用时只穿一个p以及一个value，然后这里边我这样做，我把 key value data点put key外流，原来它生成的是直接传整个 map，但是我现在改成了只传一对k外流，然后我把 k外流存到 map里，这样外界调用方便直接调对象点set data就传一个值，在set data又传一个值。

当然为了调这个site就是方便能连续调，我也让他返回去慢点，我听对c那么如果你有多个值要往 map里传，你就可以 event点set data传，一个再点set data传，另外一个我们这个方法可以无限的掉下去，这样很方便。

好了，那么事件对象我们就处理完了，当然如果你不用这样做，你不这样做，你用以前的方式做也是ok的，但是这里面我们稍微设计一下，为了以后方便，因为其实以后我们会处理很多的事件，到时候你就知道它的便捷之处了。

有了这个事件对象以后，接下来我们就开发这个事件的生产者与消费者，那么生产者与消费者要做什么？我们上节课通过测试类其实也演示了，只不过这里

我们所关注的或者。

说我们所处理的不是一个字符串，而是一个对象，就这样。

好，为了写这个代码这样我再建一个新的包，事件的生产者与消费者，我建一个新的包。对他家里封装这个包名我就叫一麦，然后先写生产者，生产者我叫 invent producer，然后 B需要交给容器管理，所以说加上complaint，我们上次课演示测试代码你也知道生产者他需要调用卡夫卡template去发一个消息对吧？

所以我需要把卡夫卡template给它注入进来。

另外生产者我们需要提供一个方法，事件的发送消息的方法，当然我们现在的处理的主体是事件，我就要处理事件就不要发布消息了，咱们从业务角度来阐述处理事件。当然了处理事件的本质就是发一个消息，那么这个方法拥有的方便外，外界调用不需要返回值，通常我们习惯于管这种方法叫发1万就触发一个事件。

当然你外界调用的调用这个事件触发的方法的时候，你肯定需要把事件对象传给我，谁调谁传，因为你调用方才知我是什么事件，它包含哪些数据对吧？

你传进来我们封装的对象就有用了，emat好，那么我们这里要做的就是发一个消息，把这个事件所包含的数据发出去，我就写个注释说将事件发布到指定的主题，那就卡夫卡template这样，那很显然首先它需要两个数据，一个是主题，主题的话咱们阴暗的点不是有吗对吧？然后其次我们要发一个字符串过去，那这个字符串应该包含事件对象中所有的数据怎么办？

最好的方式是我们把一曼转换成一个Jason杰森字符串，那么消费者得到这个字符串以后能把它还原为int，这样的话他就得到了一般的事件中所有的数据，然后做进一步的处理，至于说他怎么处理，那是根据业务情况而定，我们这里只是负责发出去就可以了，所以我这样写节省 object，ok，然后节省词句，然后把英曼传进来，这样的话我就发了一个消息，只不过这个消息的内容是一个计算字符串。

Ok了，那么有了生产者以后，我们需要再去开发消费者，我在1万的包下再创建一个新的类是事件的消费者跟summer，那么我在这儿也需要把它声明到这个容器里加上一个control的注解，然后这里边我一会在写代码的过程中，我需要记日志，因为你处理这个事件发布这个消息这个过程可能会有一些隐含的问题出现，我们一旦发现问题要记日志。

好，所以我先声明一个记录日志的组件，然后传入当前的类。

另外我们处理这个事件最终要干什么？咱们刚才不是也讲了，我处理了这个事件其实就是最终是要给某一个人发送一条消息，而发消息我们最终是要往 Max表里插入一条数据，这就是发消息在我们系统里要往max表里插入数据，你是不是得依赖于 max service，对吧？

所以我把 my sister service重新来依赖它。

好，接下来我们就要开始写处理事件就是消费事件的方法。

那么。

我们可以写一个方法，消费一个主题可以这样做，也可以就是一个方法消费多个主题，因为我们上次在想卡夫卡里森那然后 topics等于大括号，就是说一个方法可以消费多个主题，这是可以的，同理同时一个主题也可以被多个方法消费，其实他们是一个多对多的关系。

那么你看我们业务当前这个业务是评论时点赞时关注时发一个通知，而这个通知最终的形式是非常接近的，我们可以看一下精神一面看这个消息，最终通知的形式是非常接近的，就是说了某某点赞了你的帖子，某某关注了你，这个形式非常接近，所以因为他们很相似，这个逻辑，其实我们可以写一个方法，把这三个主题都处理掉，他们一类似是吧？

所以我这样写卡夫卡listen topics等于好等于主题，我们现在得定义一下主题常量，不然的话这里怎么引用对吧？我们定义常量，这样我找到长城接口好，现在我定义的是主题时间的主题，首先是评论好这个主题我把它定义成4种类型，然后的话是topic。Comment等于comment。同理再来主题。

点赞。

Topic Like，好，再来主题关注。好了，那么这三个主题我就定义好了，以后我再回到刚才的消费者这个类当中去，然后我实现接口，便于引用它里面的常量。我说了这个方法我要消费，这三三个主题我都要消费，我就这样写topic，comment。

Topic like topic followed。

好，然后我来定义方法，消费者这个方法probably why Handle我要处理的是多种主题，这里我咋写我写这样handle选一个代表 handle comment卖水，然后同时也把另外两个处理了，就是这样的。然后我们也知道这个方法需要加一个参数，用来接收相关的数据，那么这个参数就要啃萨玛瑞靠的，别看了。

那么在这个方法之内，为了让程序更严谨，因为这是一个公共的组件，尽可能的严谨。

所以首先我做判断，如果说return它等于not或者record来了，它等于浪，假如万一说这个值是空的，发了一个空的消息，我需要我就不进行下一步处理了，我就直接明确了，但是我要记个日志，那个点的消息的内容为空，所以说我没有处理。

那么如果消息的内容不为空，我们知道我们发的消息它里面的内容是一个杰森格式的字符串，我们可以把这个字符串恢复成相关的对象，恢复成emat我就节省 Object。

然后第二 pass，ok解析为对象，那么我们需要传入的是阶层格式的字符串，那就是recall点value点to string。另外你要指定具体的类型，就是这个字符串它对应的类型是什么？是emat叫class。

好，那么转了以后我们再做一个判断，emat如果等于二，如果你转换以后对象结果是空的，这个时候我还是认为不对，还要做一个判断说消息格式错误，如果说有值，但是你还还原不回来，那就是格式不对，这个时候还是好，那么如果说内容也没问题，格式也对了，我们就可以大胆的去发送站内通知，所谓的发送战略通知其实就是一个发战略性，是把这个就是构造一个message，数据插到

message表里。

好，我们再回顾一下 message表，这里面有ID，然后有消息发布人的ID，有消息接收者的ID，另外还有消息内容状态以及消息传递时间。然后之前是有一个叫 conversation，ID是会话ID是两个ID，拼在一起小的在前，大的在后。

但是你要注意，我们现在发的消息和之前的私信有所区别，私信是张三发给李四是两个用户之间发，而现在我发通知是我们的系统后台发给用户，后台的话，user其实真实不存在，这样我们可以假设后台也是一个优点，比如说假设 user ID是一我们招一个虚拟的用户，永远都是一向某人发消息，这个时候你康c是ID，你再去存两个ID拼在一起就没有意义了，因为肯定from ID是一固定的，对吧？

因此你看就像这样，这之前有以前的就是测试的数据，from，ID发通知的时候它一定是一所以你康ccid存这样的数据没有必要，这个时候我们这里存什么，我们这里就改存为主题，你是评论点赞存主题，因为我们要存一个主题对吧？通知对应的主题。

另外内容里头我们存的就不是一个具体的一句话，我们就存的是纯什么？纯，我们将来在页面上要拼出这句话所依赖的条件字符串，一个介词字符串，我们存一个json字符串，这个字符串中包含了我们要拼这句话的各种条件，我们到页面上得到这个对象就能拼这个条件这个意思。

好，你从我之前给的测试的数据中也能看到，大概的意思我们就按照这种方式来存，同时也是复用这张表，或者说这张表它存了两类数据，一类是人与人之间的私信，一类是系统给你发的通知，他们存的时候有所变通，注意。

好，那么说完了这个规则以后，回过头来我就开始要发送站内通知了，我要构造 Message，这里我写个注释，发送站内通知，主要就是构造一个message对象，然后首先 message site。

我们先site from ID，ID不用ID是自增长的对吧？

From ID刚才也说了from ID系统用户我们给他规定就是一第一个用户，你看我们的表里有这表里，因为这表里莫干山有条数据就是系统用户我们就要它当然了你这里可以直接写成一个一，或者最好把一一定义成常量，这样别人一读好明白这个一是啥意思？

为什么是一这样我在这里再加一个常量，这是系统用户的ID，因为我要这里要存的就是ID系统用户ID，那么就是个整数，然后的话我给它取名字叫sister，我们就在ID等于一表里已经有了，好，回过头来这里就应该写system就是ID。

好，这是消息的发布者。

那么接下来我们还得去构造或者说设置这个消息的接收者 site to ID我们这个消息是发给谁？你看一下 Invent。那么是当前这个事件的触发的，那个人比如说点赞，张三给李四点赞，这就是张三。张三给李四点赞，我们应该通知给李四，李四是谁呢？这个是李四发的帖子的作者，他对吧？

所以回过头来你这里要设计的设置的是until又在ID，那么我就写event，从event里取event，点get Entity有个ID，然后再往下要处理 Conversation ID site，conversation ID刚才也说了，这里我们存的不是真的是conversation，我们存的是主题，主题我们也可以从银滩那里获取掏给他，然后再来接下来我可以那个状态默认是0就是有效的，所以我就不用去设置了，默认这个对象里那个值就是0。

然后是site create time，当天时间就可以了。

除了这个以外，我们还要设置什么 content，这个内容应该是里边是包含的是一个对象，这对象包含的是页面上，我要拼这句话的数据，好拼这句话应该是或者说进去以后，那么这是评论的详情，那么这句话中应该包含什么？应该包含这件事是谁触发的，然后他是对哪个实体做了一个什么操作实体你要存下来。



另外你点击查看能练到哪去，因为他评论的是你的帖子，你要练到在这里没有具体做，你要列到帖子上去，而列到帖子上面我们知道其实就是列到帖子详情页面，列到这个页面我们需要知道帖子的ID是多少，我们才能练过来，因为这个路径上面是包含了帖子的ID，好，所以我们需要这样的一些数据回到这个代码里，我可以这样处理，我做到一个map，Map就是纯的是具体的内容，最终我把它转成阶层格式的字符串，存进去就可以了。

好迈克。等于6，好新map。

好首先 map点put我要存一个优质ID，就是纯这个事件是谁触发的？优质的ID，优质ID我可以从一般的里得到。好，那么有了uzid我就能知道这个事件是谁触发的，我就能知道这个人的名字以及相关的其他的内容就可以了。然后再来map点put anti tap实体的类型。

因为我要知道你是评论了一个帖子，你还是处理了哪个，你是对哪个尸体做出了一个处理，所以这个类型也要记录下来，这个类型我可以从事件对象中获取，因曼特太好，然后再来map点put NTT ID，具体实体的ID我们也需要记录起来，这个也是从这个事件对象中可以获取。好，那么注意大家这块不要乱，就是说我们这个事件对象是事件触发时封装了相关的数据，那么我们事件的消费者在消费这个事件的时候，我得到是原始的数据，但我最终需要把这个数据转化成一个message，这个message里包含了一些基础数据又包含内容，最终这个内容你是要拼出拼出这样的话，而拼出这样的话，我们需要知道的是用户的ID，还有我们操作的目标相关的信息，所以说我把这些内容都拼进来，所以说这里需要一个数据的转化，不要乱。

然后当然了，我 invent里可能还有一些其他的数据，可能还会有一些其他的数据，当触发事件的时候，如果你这个时间稍微特殊一点，可能还会有其他的数据，这些数据我也不方便存到message的其他字段里，你存到什么康复c这里啥都不太合适，我也一股脑存到 content里，总之不方便存的字段我们统统的存到content里。好，那就这样做。

我就先判断一下。

看一看那个事件对象里 map有没有值。一曼特点get data。

第二判断一下is empty加个感叹号。如果说它不为空，不为空，我就要把 In one的map里的所有数据拿出来，存到 map里，也就是存到 content的字段里，好，我就遍历一下慢点安全。

好，那么我便liemandt每次便利得到一个安全，然后的话in Mandt点get date安吹sat，然后遍历的是一个p86的集合，那么每次我能得到一个q86，每次我要把 q86怎么办？存到这个是内容，这样我把跟他们去写叫content，这样的话大家能够这个表里字段对应起来，这from ID to ID这是content这样好一点。

好，然后这里我就写了content点put的安全 get key entry点。Get value我就把 p value直接存到 map里，content里存了这些数据以及可能存在的一些额外的数据，就这样它存的数据比较多。

最后 counter我是构造了一个map对象，我还得把它存到放到 message里，不用message点site，content，我这里需要的是一个字符串，我需要把它转成一个Jason字符串，Jason object，然后点to Jason词句，然后 content。

好，那么现在 message我终于构造好了，构造好以后我就可以存了，我就掉message service，然后点ed message。

好到这儿我终于把这个方法写完了，这个方法它消费的是三个主题中的数据，而消费的逻辑都一样，就是发送一条消息，而消息的构造的方式也一样，就这样。

好，那么处理完以后，那么接下来我们需要调这个程序，现在还没有地方调，当然义卖这个我们不用去主动去调，它就是一个封装数据的实体，我们主要是要调这个生产者与消费者，对吧？你什么时候要触发事件，就去调一下生产者处理这个事件就可以了。

那么消费者我们知道他是被动触发的，只要你对这种有数据它就自动就执行了，对吧？这个不用我们主动去调，我们只要主动去调这个producer就行。

按照我之前所讲的业务，我们应该是在这三个地方去调的，评论的时候点赞的时候关注的时候是吧？所以那么我就打开对应的CTRL了，首先是评论，咱们一个来评论是comment controller，然后还有点赞like controller，还有是关注 Follow controller，那么这三个controller这里边我都需要做相应的处理。

一个来首先我先处理 comment controller。好，那么 comment control我需要处理事件，我需要注入producer，当然了另外两个CTRL也需要调，也需要注入，我干脆copy，然后再回头逐个处理。

那么你看这个方法是添加评论的方法，添加评论以后我就要通知了，以后通知，所以我在爱的comment之后做通知的行为，这里我要做的事情就是触发评论事件，那么触发评论事件首先你要构造事件对象，把事件相关的内容包含进来，好，你看我就谬1万，然后的话直接调这个对象点site topic的方法，因为是评论事件，所以说那个事件的主题应该是topic，我要引用常量，implements，community constant，那么另外两个也需要做类似的处理，我直接copy有了，不用了，好，那么这里应该写的是 topic，comment，然后因为我们这个方法直接委托了类似，所以我还可以继续调对象点，其他的site、方法、site又在ID，你看这样是不是方便？

好，这个事件是谁触发的？肯定是当前的登录者对吧？登录的人是谁？不就是他，所以 site UI d应该是它，然后 Site on TT type。那么这个on TT type，我们可以从当前的com的数据里得到他们的概念，安迪地带的然后再点sat，安迪的ID，安迪的ID我们依然可以从 comment里得到。

好，然后最后我们在拼，刚才我也说了，我们在拼那句话的时候，最终的话拼完那句话以后还有一个点击查看，而点击查看要练到帖子详情页面，对吧？

练到帖子详情页面，而练到帖子详情页面，我们是需要有一个帖子ID的，所以我这里还要传一个帖子ID，但不是每个事件都需要帖子ID，所以我在设置设计 EMAIL的对象的时候，没有一个明显的帖子ID的属性，我可以把帖子ID存到 map里，site data，那么 key我叫post ID，那么值对应的值帖子ID，咱们这个参数不是有吗？

好，这样，我消费这个事件的时候所需要的数据我都放到 EMAIL里，可能有些用不上，但我们宁多别少就好了，这里还还有一个小小的欠缺，实体的作者安推推优质ID，这里我没有拼，为什么没有拼呢？因为我评论的目标可能是帖子也可能是评论，对吧？我评论给你的可能是帖子也可能是评论。那么 Nttuzid我们就得查了，如果是帖子你要查帖子表，如果是评论你要查评论表，所以这个得有所区分的。

所以这里我做个判断，然后去查我这样，而且comment点get itt type，如果说这个类型是itt type post。如果我是给帖子做评论，indtuzid应该是存的是帖子的uzid在这里我需要查查帖子，查帖子里我可能还需要再注入帖子的service。

回到刚才的位置，这就查了。就是discuss post service然后的话点fine的白底。这个ID就是comment to get anti ID，评论的实体的ID，此时是帖子类型是这个好我得到了当前的帖子，discus我管它叫它盖的，这是我们评论的目标。

好，因曼特我再补充一个属性，site安推优质ID，Uzid就是他点get uzid应该是这样的，形式应该是这个数据。

好，再来，还有一个可能是我当前评论的是评论的是一个评论 comment的get empty tap等于comment等于的是ntat comment这个时候我这个to get就是一个评论了，我这也需要查他们to get等于 comment service点find。

应该是find comment by ID但是目前我 service里还没有，我还需要做一个补充，好，我首先看一下 find不是不知道哪儿是他们的map，没有这样的方法。我没有我做一个补充，好，就是根据ID查一个comment，我需要对它做一个实现。

我搜一下 comment，map摘写一个select，他的ID copy一下，然后 without是卡门类型，我查询的话可以这样copy一下，好。这个很简单，select这些字段 From common，然后再加一个条件就好了。Id等于ID。好 Map实现完以后，我再去找找到 service，然后在这里加一个方法，这方法public fund come by ID。很简单，直接就return。Come to my book。点c ID就行了。好，简单补充完以后，我再回到刚才的我程序进行不下去的位置，我再继续下去这个地方。

犯的他们的白d这个com的应该是什么？是com的点get安吉的ID，我评论的实体ID我的目标，那么这个目标里头包含了一个uzid我要把它存到 Email里，一曼特点site，Entity。

Use ID那就是它 get use ID这样总之这块稍微麻烦，在不同的类型，你要获取它的作者需要稍微的处理一下，到这儿我们一曼的数据就全了，那么全了以后我就可以发布这个消息了，我就调 producer in mind，producer然后去处理这个事件点fair in mind存进去。

那么你看我这个位置构造完一曼特以后，我只要调一下发一曼，那么我当前线程立刻向下执行，立刻去处理我这个页面的响应就完了，而后续说你这个消息的发布，那就是由这个消息队列去处理了，它可能略有一点点延迟个若干毫秒，顶多一两秒就发布到目标头上去了，所以这就是并行的，并发的我在处理后面业务的同时那个消息来处理了，或者说也叫异步的，我后续的业务在处理的同时，另外一个线程也在处理那件事，所以这样的方式就处理效率高。

而且一旦说现在有一个热帖评论的人非常多，我当前的线程只需要把一曼丢到堆这里我不管了，那么它的处理的能力就提高了，处理能力就提高了，现场也可以攒着很独整息，他慢慢处理好。这就是消息对立的好处，它起到一个缓冲的作用。

好，你了解了以后其他的什么点赞，这个follow其实就类似了，也是做类似的处理。好我们再去处理一下。下面处理点赞，好那点赞的话应该也是在这儿，我点赞完以后，然后我去触发这个点赞事件，我在这加一段代码。好，触发点赞事件，你注意我这个点赞其实是一个具有双重能力的一个方法，它一个是你点一下它是赞，再点一下是取消赞对吧？我们给用户发消息发通知，显然是你赞的时候发个通知他就行了，如果是取消他就没必要通知了，那就太恶心了，对吧？

所以这里我们得判断一下，如果说这个like status它等于的是一的话，假如说我当前的逻辑是点赞这个时候我才去触发这个事件，否则的话你不是点赞，我就不去触发这个事件了。好，那么触发这个事件无非就是还是构造 Email的，因为EMAIL的然后点site这个topic是topic like。好，然后点site优质ID，site。优质ID。那 c点的站还是当前用户对吧？那就是household get user get idea，好再来，然后这个实体类型site安推推 tap，安迪的tap这里不有吗？

好，然后再来site inttid参数里也有，我就直接写过来，好，然后再来 sat安推推又是ID，这里也有，好，然后注意最终我们拼了点的这是点赞的通知，你查看的时候也是说是某某人，其实这里这个页面应该是提示应该换了某某人，占了你的什么东西，然后的话点击查看，占了你的帖子，占了你的评论点击查看，一点击查看也是去到帖子详情页面，所以我們也需要一个条件就是帖子ID这里也需要帖子ID，好，所以这帖子ID我还是要site，data放到data里，还是叫poss的ID，但是我当前的方法里我是得不到帖子ID，当然你可以去判断，如果点赞的是可以点赞给谁我去查，这样麻烦我这样我重构一下这个方法。

我要求这个方法在家在多接受一个参数，就是post ID我要求你点赞的时候把它传进来，因为我們是在帖子下面应用去点赞的，那个时候很容易得到 ID传进来就好了。

所以这里我就传进来，当然一会我们需要改一下那个页面，改一下 GS，这样才行，好到这就可以了，咱们这个EMAIL就构造好了，观察完以后我们还是触发一下实践就invent producer，然后的话点发一卖把对象传进去。

Ok那么最后这个就是关注事件的处理，我们再打开 follow controller，那么关注的时候你要去发通知，取消关注不用。

好，那么同样的我也需要在关注之后做这样的事情， follow之后做这样的事情，这里我要触发关注实践还是构造 invent，点site，topic这个topic topic点topic follow，然后的话site user ID那是谁关注的？

肯定是当前的登录用户对吧？House头的get，user的get ID，然后 site itt type目前我们关注你看他也闯进来的 itt type和ittid所以说你直接就写这个参数就行了。

然后是site itt tid也是这个参数直接有了，然后我们现在其实只能关注人，所以其实安得太不一定是个优质ID，所以说我就这样写了一点 site。

On, ttuzid然后还是on ttid的，当然如果说将来我们可以关注更多的类型的实体，这个时候我们在这个地方还要再变通一下，当前我们这样处理就ok。然后关注和评论点赞不一样。

关注是这样的，就是说某某人关注了你，完了他关注了你完了，而不是你的帖子，你的评论他是关注了你这个人对吧？

所以说这里边我们通过这条消息，如果链接的话不是链到帖子详情上去，而是链到某一个人的主页上去，比如张三关注了你，你就链到张三主页上去，你看他关注了我，而这个评论点赞点赞的是帖子是评论你要练到帖子详情页面，所以他们需要post ID，这个不需要，所以这里我就不再带data pose ID了，这样就可以了，然后的话一曼特丢手点饭1万，好到这儿我终于就把这三三种事件触发完了，触发完以后别忘了咱们这个点赞刚才不是说要重构一下，加了一个post ID对吧？

为了方便，那么为了加 POS ID，我需要改一下帖子详情页面点赞的方法。

好我搜一下 xx点q点前面我们调的是 like方法，我们要找到这个like方法，然后依次的进行一个处理。

好，那就是给他再多传一个参数，最后你看这是两个三个，最后再来一个参数，这个参数是帖子ID，而帖子我们当前页面上就是post，所以 post点ID就可以了，我copy一下，然后我们再往下找，那么其他的两个方法其实也是一样，处理方式非常简单，就是在后面加这么一个条件就行，再找这有一个like也是加个条件，这有一个like加条件，好处理完了，当然你调的方法多了一个参数，你这个方法得改一改，这个方法我是定义在 GS上的 discourse点GS那么我就需要在这个方法的上面加上参数， post ID，然后我需要把这个参数传给后台，所以我再加上一个post ID问号，post ID好了，到这儿我就处理完了，当然我这只说处理了这个事件的触发以及消费，那么至于说将来在页面上怎么显示，我们现在还没做，我们这次课先不去处理显示，下次课再去搞定显示，这次课就是说当我做评论做点赞做关注的时候，我能把这个消息发成功，我上表里一看这数据有了，表示说这一步完成了就ok了，做到这一步就可以了。

好，写完以后咱们测一下，我启动一下这个服务，然后你这回我们在启动服务之前一定是要把组keep卡夫卡服务器启动的，因为我们当前的应用已经依赖于这样的服务器了，如果你不启动的话这会有问题。

然后这里关于卡夫卡的服务多说一句，组keep一般没什么问题。

卡夫卡应用，它 windows的客户端做得不好，就这个客户端经常会由于某些原因崩溃，如果你去看他的报错日，他经常会说某一条数据被锁死了或者怎么样，有了问题有了冲突他起不来，经常会出现这样的情况，你说这个东西还能用吗？

这个东西其实它windows客户端不稳定，但是它Linux没有问题，Linux很稳定的，而你要知道我们外部应用的服务器基本上都是Linux，基本上没有人用 windows服务器，所以说在上线的时候这个问题不会有，但是我们在练习的时候，因为我们大家用的常用的都是windows系统，所以我们就装了这么一个不稳定的客户端，希望你理解，那么一旦说你说我发现了启动这个事我启动不起来了，一启动它就自动就没了就关了，这个时候怎么办？

也很好解决，我告诉你你就打开这个就是你卡夫卡日志存放的位置，你直接把这个目录删了就行，你把这个目录删了，它里边什么数据都没有了，自然也不会什么锁死了，就解决了，这个问题很简单。

好了，我这个数据服务已经启动起来了，我们测一下。

好我先访问一下首页，当然我需要登录好登录，然后比如说我要对 AAA的帖子做出评论，好我要评论你就随便写一个别CC了，你好好这就评论了，评论完以后我去看一下我这个表里有没有对应的那么一个通知出现。我查询 Message，奥特曼ID d SC倒序，没查到，看一下有没有报错，估计是有一些问题。

报错了。

报错的时候我们看其实主要就一般看第一句话，看他大概报的什么错，他说在这里在 can some record里，肯萨姆也靠它弹出来这个消息，然后这儿有说明说在憨豆肯萨 comment message这个方法里报了错，然后报的是控制人异常的错误，然后就没有具体的说到底是哪个报的错，我们再往下看。

下面的话是底层的代码，我们不用去看底层代码，它一般比较严谨，不那么容易有漏洞，主要还是看一下和我们的程序有关的地方控制异常。你看这个类是我们以前写的 LP 的类对吧？他说这里报的错点一下，如果说这句话报控制人异常的话，那很显然是这个对象为空，调这个方法就会报错了，对对象有可能为空吗？

H 表示是和请求有关的对象还真有可能为空，那么之前我们没有生产者与消费者，因为我们当前的 LP 是拦截的是 service，我们所有对 service 的访问都是通过 CTRL 访问的，但现在我们有特殊情况了，刚才我们写了一个肯萨姆消费者，消费者里面我们也调了一个 controller，那是通过消费者去调的，不是 CTRL 去调的，消费者调的 service 不是通过 CTRL 去调的，所以这个时候在这次调研里它就没有 request 得不到，所以这块就出现了控制异常，这是我们当时写代码其实不是特别严谨。

好，那就做一个判断判断 request，这样如果我简单处理，如果 h 表示他等于那表示说这是一个特殊的调用，不是一个常规的页面的调用，这个时候我就不去记日志了，或者说你也可以做更其他的处理，就是说你在这种情况下你记日志就不记这个 IP 了，也可以这里我就干脆我就不记日志了，就退了算了。好总之逻辑你可以自行处理，这里我就做了一个简单的处理。

好，改完以后你得重新编译一下，它已经重启了，然后我们再来试一下，好。再评论一下还是你好，再看有没有报错，这个应该没报错，看一下表里的数据是否有变化。你看有了系统一给 111 发了一个通知，通知的主题评论，这里边包含了一些一些我们需要的数据 Uzid and the type 等等，这就行了可以了。

好再是除了评论以外还有赞，我已经点过赞了，这样我把它取消掉，然后我再给他点一次。好，然后再看 like 又是一条通知对吧？好然后这样我再评论点对 AAA 点评论，我找到一个 AAA 这个是 AA 的评论，我要给他点赞，好，然后再看又一个 like，你看刚才那个点赞的是这一类型是一对帖子这个类型是二，对 AA 的评论点的赞，好。然后关注已关注我取消掉，重新关注一下 AAA 然后看一下表里的数据，follow a to 111，对吧？

Tap 是三，那么到这儿就基本上证明我们对事件的处理是通的。当然中间出现了一个小问题，就是我们之前写代码的不严谨，所以说你在写程序的时候，尽可能量的要去多做判断它是否为空什么的，让代码更严谨，不然的话出现问题的话也比较恶心。好了，至于说后面根据这个内容，我们怎么去做展现，那是我们下次可能要继续探讨的话题。好，这节课我们就演示到这里，咱们下节课再见。

## 5.5 显示系统通知

39-显示系统通知.mp4

这节课。

我们来实现显示系统通知的功能，把上次课我们所发送的通知存到数据库里的那份数据显示到页面之上。那么这个功能我们在开发的时候可以把它分为三个小的需求，第一个是开发一个通知列表，就是我们点这个消息，然后不是能够看到这样一个界面，然后这里面有一个系统通知的这么一个功能，你点完以后能够看到通知的列表，那么这个通知的列表和私信列表不一样，私信的话我们显示的是不同的规划，那么通知我们显示的是只有三类通知，咱们整个应用当中目前只有三类通知，分别是评论类、点赞类和关注类。

我们只需要在这其实显示三类数据就是三条数据，然后每一类通知这里，目前这个地方不需要你显示所有的通知的内容，只需要你显示最后一条通知，最后一条评论最后的一条点赞的消息，最后一条关注消息就这样，好。

然后第二个小的需求是通知详情，那么针对某一类通知你点一下，能够打开一个详情页面，然后显示这一类所有的通知，当然基本的操作和私信其实是类似的。

最后还要补充一个就是未读消息，当然我们这个地方都有未读消息，但是这个地方也有未读消息，并且很显然这个地方的未读消息应该是所有的未读的私信加上所有的未读的通知把它加起来。

所以我这里未读消息指的是这块，好，我们就按照三个小的需求咱们一个来实现，我们首先来实现通知列表，那么我打开我的 Idea，那么要实现通知列表，首先我们还是从数据访问层开始写起，因为通知的话我们那个数据是存到了 message 表里，所以说这里我们需要。

在 message map 里增加一些方法，那么要想显示通知列表的话，我们这里需要增加三个方法，第一个你要查某一个主题下面他最新的一个通知，因为我们一共只有三类主题，咱们一个查查三次就搞定了，所以我们就第一个方法是查询这里写上，查询某个主题下最新的通知只需要一个最新的通知就可以，有了这个以后还要第二个，你还得查询某个主题所包含的通知的数量，这里我们可以看一下页面，因为什么？

那每一类通知后面有一条提示，这有几条规划，其实就是几个详情的数据，所以说我们需要查一个数量。好，也写上就是查询某个主题所包含的通知的数量。好，那么还有一个我们还要显示什么？还要显示未读的通知数量，对吧？你也得查到。好，所以第三个我们要查询未读的通知的数量。那么有了这三个查询方法，那么通知列表就可以实现了。我就把这三个方法给它定义一下。

第一个是查询某个主题下最新的通知，注意是一个，所以我要返回一个 Message，那么方法名我叫 select latest，最新的 notice，最新的通知，当然你需要给定参数，首先你是查询谁的通知，优质 ID 一定要传入，另外要传入主题是哪一个主题？这就行了。

再来第二个是查询数量，那么返回整数方法名我叫 select notice 抗体通知数量。那么很显然其实这个参数和刚才这个应该是一样的，也是某个人某主题通知数量，最后是查未读的数量。其实参数也是这两参数，只不过查的不是所有数量是未读的，需要多加一个状态而已。

那就是 select notice 安瑞未读 Count，还是这两参数，那么方法说明完以后，下面我们就去实现了这三个方法，我需要打开对应的 map 搜一下，那么我在后面增加这三个方法，先实现第一个 select，然后 ID 等于他 without tap 返回的是 message。

好，然后要查什么，我们可以从之前的 circle 里 call，因为都是同一张表。肯定是 from 这张表对吧？Select，然后这些字段 from message，然后我们需要这个数据加以限制，我们说了是查询最新的一条数据，怎么保证最新，我可以这样写。Why 这个 ID in 我对 ID 做一个限制，这里我可以这样写谁来？Max，ID strong message。当然你可以回顾一下之前我们查询最新的私信的方法，只不过那时候查多条，现在我只查一条大同小异，select max ID，然后将条件限制，status 状态不等于二，因为等于二表示私信通知是被删除的状态，所以把删除再排除掉。

然后 from ID 等于 1，为什么叫 from ID 等于 1 呢？因为之前我们发通知时你也知道通知一定是系统用户发给你的，所以服装卖地一定是一等。我们加这个条件的筛选是为了避免和普通的私信混淆，如果你不加的话，可能不小心会把私信查出来，加上以后就不会有奇遇了。

好，然后再按 to，ID 等于这个就得等于 uzid 了，uzid 就是接收者，当然还有一个条件是主题，那么把主题加上去，而主题我们是存到了康 c 三 ID 字段里头去，这个字段有点长，copy 一下 and come c 加 ID 等于参数 topic。

好，你看我 where 子句当中 in 当中，我们最终查到的是最大的 ID，最后一个 ID 最新的，然后一加以限制，我们就查到了最新的数据了就可以了。

好，接下来我们再查这个数量，你数据都会查数量就更容易了，我直接写一下，返回的是整数，好，其实查数量其实就是 circle，条件都一样，实际的是不等于 2。

From，ID 等于一。To，ID 等于 is ID com CID 等于 Top 一模一样，然后只不过这里我现在查的不是最大的 ID，而是 ID 的数量，所以其实把整个搜狗印拷过来，稍微改一下就可以了。

这个地方改成 com，这就行了。

好，那么还有一个方法，最后一个也是返回整数，然后那么这个方法其实和刚才差不多也是查数量，但是条件需要稍微的有一些改动，我先靠过来，然后首先也是抗台地，然后 status，注意我们现在查的是未读的数量，未读的时候这个状态是00表示未读，然后 from等于一不动to ID等于uzidok然后这样写没问题，但这样写的话，我们就只能查出某一个主题的未读消息数量。

那么你像这个地方这个数就查到了这个主题，比如说未读三，这个主题未读二没有问题，但你注意我们将来还要查整个通知的未读数量，当然你可以挨个查把它加起来，但是我也可以这样做，如果不传 topic进来，我允许这个方法不传topic，如果你不传主题，我就不拼这个条件，而不拼这个条件，我就是查当前用户所有主题的未读数量是不是也可以，这样就这个方法的适用的面就广了，对吧？

所以我们这样处理一下，这里我需要加一个判断，if test等于然后 topic如果不等于now，这个时候我再去拼这个条件，如果说这个topic就等于那表明你没有传这个topic，你想要的是所有主题都要数量，我就不拼这个条件就可以了。

好了，那么针对第一个需求，这三个方法我就都实现完了，实现完以后，下面我们来写业务层代码，我就在 service之下找到 message，service因为我们之前的私信的逻辑涉及到这里来，那么通知和私信是一个表，我就干脆想一起去。

好，我需要在后面加几个方法。

那么我们查询通知其实它这个业务也是比较简单的，中间也没有什么额外的处理，反正就是查我就直接把这三个方法声明，然后调一下map直接实现就完了。那么第一个方法我叫find，latest notice就是查询最新的这个通知我直接 Return my smile点。Select select。Let is notice，好，然后再实现第二个方法。Find notice com。还是这俩参数，然后 return message，map，select。No discount。好，最后再来一个find。Notice and read。Count。Return message，map点。Select这个好了，很方便，我们这个业务层就写完了，写完以后最后我们就处理表现层，然后写controller，然后处理请求，然后处理对应的模板就可以了。

当然这个control我还是利用之前的message搜一下message controller，当然我们之前已经引入了什么 house toda message service，user service这些常用的病，另外我们也写了一些方法，下面我们再增加一个新的方法用来显示通知列表，首先先声明这个方法的访问路径 pass，这个叫notice，然后 list通知列表，然后 MR的等于 request get这是一个查询，然后方法是公，有的返回string，方法名我叫get。

Notice。List。参数。就一个 model，为了向模板传数据。好，那么在这个方法之内，首先我要获取当前的用户，因为你要查的是当前用户他的给他的通知对吧？当前用户得到这个很容易，house偷的 get user，然后那么我们通知列表之上显示的是三种通知，每种通知显示一条数据，你就查三次把它都得到就完了。

当然我们每次得到的一条数据一个message，这个数据其实在展现上它并不完整，那么我们需要再把数量，所有的消息的数量以及未读的数量查到，然后组合进去。

另外我们查到的数据里可能有user，你可能也需要查到对应的user组合进去。

好这个我们这样的处理方式我们是轻车熟路，我就查了三类通知，首先我查询评论类的通知，然后我再查询点赞类的通知，先把注释写得清楚一点，最后查询关注类的通知好。那么这个就查 message等于 message service。Find latest notice。那么ID uzid就是当前用户的ID然后第二个条件是这个主题，这个主题是固定的，我们当前是查询评论类固定的，应该传进来一个常量，我看有没有实现常量接口，没有实现一下。好继续。这里我就写topic。

好，那么查到这套数据以后，我们需要给它补充一些数量优质什么的，数据需要做一个聚合，所以说因为是一条数据不是集合，所以我就直接创建一个map。用来封装聚合的数据，那么这个map我取名叫 message vo没有还行，然后做个判断，如果说我们刚才查到的数据 message他非空，那么我就去进一步查更多的数据，然后把它存到 MAC里。首先我要存的就是 message，那就message will点put，然后 message除此以外，这里还要注意一下，我们查到的 message，我们查到的 message，你看这个表，其他的字段都好说，from conversation的都好说它有具体的含义。

然后我们还有一个特殊的字段是内容字段，那么对于私信来说内容是一句话，直接可用，但是对于这个通知来说，这个内容是一个阶层字符串，我们需要把它还原为这个对象，然后我们好再去应用，不然的话你这样不好用，所以说我们在这儿需要先把这个字符串还原为对象，然后也放到map里，这样我们就方便去访问这些数据了。

好我就这样，而且我们得到这个数据的时候，我们得到 content的时候，你看它在序列化的时候，像引号它用的是特殊的转义字符 quot我们因为我们就要显示这个数据了，所以我们最好把它还原回去，我们怎么把一个转移字符还原回去也还好这样的。

我直接写了CC content，内容等于html YouTube还记得这个工具，我们不是可以利用它对一些html的标签字符转义对吧？他能转移也能反转，一样，我们可以这个html，然后 escape是转移，然后你可以安，反转就是转回来。

然后 message点get content，这样我们得到的 content里边就没有转移字符了，是正常显示的，然后我需要把它转化为正常的对象，而不是作为字符串。

我就Jason object plus object转为对象，把content的传入，然后我们之前转为字符串的时候是利用map来转的希map，所以这里指定的类型是希迈克，我们得到的就是迈克类型的对象，名字我叫data。

那么得到这个data以后，我要把 data中的存的数据转存到 message view里，因为macdvo是最终被模板所引用，要显示数据的来源的对象，就存咱们之前往 data里存了什么，主要存的是优质ID对吧？存了这个int的态度， ittid以及帖子ID这些我都处理一下。

Message view put首先是user，然后 user service点，find user body。当然我们页面上展现的时候肯定是希望要的是 user对象，而不是user ID，所以说这里我就做了转换，find by ID然后 ID我从data里获取 data点get user ID当然了你 get到的时候它不默认不是in，它是object，这是我们要转型。

好，然后是 an TT type，然后这个值从data里获取点get Entity，看看。

好再来 message view点put，那么接下来是ittid好，然后还有这个帖子ID，帖子ID我取名就叫post ID，然后 data叫get post ID。

好，那么除了 content的内容里的这些数据以外，我们还需要查到这一类这一类通知它的总的数量以及未读的数量，把这俩数量查到以后也放到 video里头，我就查那就先查数量，com= message service，fund，no discount。

那么我们需要传入的是uzid以及类型，那么我之前查这个地方不已经传过了是吧？

Copy一下，那么我就把烫的也放到屋里头去，message will点put，然后这个名字我就叫com，这是数量，然后再来还有一个未读的数量，我叫安瑞的等于 message servers点饭的notice read com，然后也是这俩参数 copy一下，然后也把它放到 video里头去，message video点put angry好了，那么到这儿这个数据我该查到的就都查到了，然后都放到了 macvo里，接下来我要做的把 message view然后传给这个模板，我只需要把它放到model里就可以了，然后给它取一个名字，和另两类的数据要有所区别，因为这是一个评论类的通知，我取名叫 notice，数据是my city will。

好另外两类其实处理的方式和这个是一样的，甚至数据都差不多，就是类型有所区别而已，所以另外两类这个代码我可以去copy一下，省点事儿。

当然了，因为刚才我们有这个变量已经叫了麦穗，所以这里你最好把它把这个类型去掉，我还是复用变量名，find还是 user，但是类型需要换一换，这个是点赞的对吧？

所以是topic like，然后我也需要聚合数据，也需要实例化一个vo这个时候你也把这个类型去掉，我还是复用同一个编上名就行了。

那么如果说不为空，我就put这个没问题，然后取这个内容把这个内容转换为 map没问题，然后我们从这个内容中读到一个数据，把它处理以后，存到mactvo里分别是user NTT tab，mttid和post ID那么处理的方式也是一样的。



好，最后这两个数量其实查的方式也一样，变量名你都可以一样，只不过类型要换成但是最后这个地方就变量名这个不能和刚才重复，因为你要这样传到模板上去的话就混淆了，这个地方改个名字，这个我叫like notice。

好，很快就改完了，我再稍微看一下过一下，看看有没有什么细节的问题。好，我暂时没有看出问题来，那么如果一会测试有什么数据的错误，到时候我们再调先这样。最后还有一个是关注类的，通知处理方式和刚才也是雷同 copy。

好，关注了这个类型得改改成叫follow好我们一起改 Follow都follow对吧？然后这个是存的是message，然后这个是解析的是content，user on，TT cap按ddid注意关注类的通知，它里边我们好像没有传post ID，为什么没有传？

因为关注类的通知只是提示某某关注了你，然后你在这里边呢你点击查看的时候，我们是链到了关注你的那个人的空间里头去，然后其他两位是某某评论了你点赞了你，评论你的帖子，点赞你的帖子，你点查看的话是链到了帖子上去，所以另外两个需要帖子ID，但这个不需要，所以我把 post ID可以去掉，当然你不去掉你这么放着也没事，反正那个数据空的啥都没有，你也用不上，页面上其实也问题不大。

然后 com安瑞没问题，这变量名改一下，follow service，好，最后还要注意咱们在这个页面上显示的时候，除了显示这三条数据，你想整个系统通知的未读数量也得显示，还有私信的数量，你在这也得显示，所以这两个未读数量你要把它查出来，好，我查一下，再补一下，就是查询未读消息数量分两个，一个是私信的，let every card message service，然后点find letter。I read count。

那么你需要传入user的ID，然后第二个条件是composition，idea不需要，因为我不是要某一个绘画的未读数量，是整所有的未读数量，所以说第二个条件是now好把它传到页面上去，model点艾特。那么第二名我就叫light angry的com，同理再查未读的通知的数量notice。安瑞的com。Message service，点儿饭的notice I read count。User点get ID，第二个条件也是now，因为我这里不关注某一个主题，所有主题的未读数量我都要查到，model点艾特还是变量名。

好，那么写到这儿我突然想起来了，你看咱们之前开发私信列表的时候，是不是没有带上通知的数量，按理说应该带上对吧？那个时候我们还没开发通知的功能，所以没法查。

但现在我们可以把补一下，你访问私信的时候把这个专利给带上，而这个通知的数量不就是这句话，是吧？我们回到先前的位置处理一下。好，司机列表你看我在这儿查过未读的私信的数量，let I read count再补充一个 notice，所以说我们在访问私信的时候也把说明带上。

好，那么再回到刚才的位置上，这就这个逻辑就完成了，完成以后我们就返回到模板，反馈到模板site，然后通知的模板叫notice点sm。好，那么很多的处理请求就完成了。下面我们来处理模板的话，首先我先这样，我先处理私信，私信把这个数量补充一下，先改私信的页面。

另外我们是在私在你看我们是点消息，打开这个页面，然后是在私信页面上点系统通知的对吧？这个链接也需要在私信页面上去配好，那么回到程序当中，我打开私信的页面来特点去店面搜一下，搜了以后我要找到能显示数量的地方，能配链接的地方在这了，在弹窗之前，你看咱们之前已经把私信配完了，包括路径，包括数量配完了，那么通知也需要类似的方式去做一个处理。

代码有点长，我最好也折个行，像私信一样，不然的话不好看。

好，首先我先配这个路径，这个路径刚才我取名叫notice list对吧？好，这是路径，然后的话下面是通知的数量，那么我们在配通知的数量的时候，也是像私信这样去处理，默认的我显示的是0，但是要注意，如果说这个数量是0的时候，你就不要显示0了，没必要显示个0我就会处理一下，或者把拷一下，我需要把它配到 span上面来，唯一的区别是变量不一样了，咱们变量刚才取名叫notice every county。

好这个地方就配完了，很简单，接下来我们就可以处理通知的模板，notice点前面，那么我就打开那个页面，notice点前面，这个页面是我们从来没有编辑过的，所以说我们需要从头做处理好。

首先声明这个模板，然后把静态的资源路径给它做一个处理，然后头部复用首页的中间的内容我先不管，我先处理后面的把Gs处理一下好了，接下来我们就可以处理主体部分的内容了，那么当然首先我们需要也是需要处理这个地方这两页签的连接，还有数量也需要处理，当然处理的方式其实和刚才私信的部位是类似的，只不过它激活的位置不一样，因为它页签激活哪个页签有点区别，我们就拷贝，也找到那个位置，然后再拷，然后你看它们的区别是什么？对于失信的话，他叶圈被激活，它上面带了一个active，对于通知它被激活这带个active，其实它俩一样，只不过Active地方不一样，我可以整个拷过来改一下就得了。

整个URL copy。

好粘贴过来，然后active需要切一下，因为我当天页面是通知的，x条应该加到a上看是不是对加到a上，这个地方也应该加到a上好了，那么这个链接数量一拷贝就完了，一模一样，然后我们处理下面的通知列表，通知列表里它没有分页，只有三类通知，而每一类就是显示一些相关的提示而已，也是比较简单的，我们做一个处理。

首先我们先处理，这里面有点乱，我给他写个注释，这个地方是评论类的通知，每一个li的是一类。第二个是点赞类的通知，然后是关注类一类处理，处理方式也雷同，所以我们重点把第一类搞定，那么后面两类就容易了。

首先你要知道，有可能你当前没有评论类的通知，没人评论，你有可能评论，什么都没有，你就别显示这个li了，所以先做个判断什么，如果说notice MAC点message里边的message，它不等于now它不为空的时候我才显示li如果它等于空li就不用显示了注意。

然后这里边这个三是未读的数量，这里我们需要处理text，等于那就是comment。Notice点angry。

然后这里也要注意，如果说未读数量是0，你就别显示了，我可以再另加一个if判断，其实也可以做用三元运算判断，你也可以这样notice点安瑞的如果不等于0成立，我就显示安瑞的把它call一下，如果不等于0，我就显示note三位，否则我就什么也不显示，我就显示一个空串空字符串，就什么都看不到，这样也可以。

好，然后再往下看，通知图标，这不是人的头像，固定的不用动，然后评论俩字不用动。然后这有个时间就通知的时间，但是你时间要格式化，然后一行写不下，我换个行格式化这意思。

Format。

那么这个数据就是notice message点，create time，然后给一个格式化的时间格式好，然后时间有了，再往下看，你看这个地方显示的是用户某某评论了你的什么，我换一个行，因为一会写上属性以后就长了换行用户这也换行用户，某某评论了你的什么多换几个行，某某我们需要改成变量对吧？

所以th能耗u text等于这里需要获取note map中的user，显示username，comment。

Notice第二user加又在内部ok了，当然那么用户外面有个链接，这个链接这个链接是你点这句话的时候，应该是打开分类的详情页面，详情页面我还没写，这里我先这样先写个井号，一会再把它替换掉，然后他评论了你的帖子，注意也可能不是帖子，也可能是你的回复，到底评论的是帖子还是回复，我们得看Andy的太动态的去改。

所以这里也需要做判断，评论你的判断那就是notice，然后点NTT看看这个类型对吧？如果它等于11就是帖子，所以你这里就显示帖子，否则就是回复这样就行了。

最后这地方还有一个共几条绘画总的数量就简单了。Notice不是notice。Company。Notice。Company好了，那么第一类通知我就这样处理完了。你第二类其实和他就很相似对吧？我们再处理第二类。第二类是点赞类的通知，首先你在A上也需要加以判断，如果like notice Jahr，message goodnow，我就显示mi，否则就不显示，然后未读的数量。Like notice点安瑞。不等于0，如果不等于0我就显示它，是吧？

然后否则我就什么也不显示。

好，图标不用动，这个字不用动，这个时间需要动一下，就这个时间和刚才差不多，有点长copy了，只不过它是从like notes里取的数据，其他的都一样，然后这个地方我也是给它换个行太长了，首先用户某某也是要显示变量，在这里我显示的是like notice。User。User name。上面这个链接也是先写成并，然后后面再改，然后这个地方也是要判断他到底是帖子还是这个回复。好，那就是like notice点安迪d看它是否等于一，如果是显示帖子，否则显示回复。

好，那么最后有几条规划，这个也是比较容易 Like notice.com好，完了还有最后一个关注类的通知也是类似的处理方式，就稍微啰嗦一点，一个处理 th冒号if刷了 notice，点message不等于not。

好，然后这个地方要显示的是未读的数量，那就是follow notice，点安瑞它不等于0，如果不等于0，我就显示艾瑞，否则我就显示一个空值，图标不用动，时间需要处理时间太长了还是copy，然后把这里面like notice换成follow notice，然后这个地方是某某某关注了你，就关注了你，没有什么帖子回复什么的，就关注了你好。

某某关注了你某某，我需要改一下。Th那就是 follow notice，然后点user user name好。当然前面这个路径我也把它改成井号，后面会替换这个数量也是和刚才类似，就follow notice.com，好了，到这里我就把通知列表这里边这些数据做了一些处理，处理完以后对不对？咱们最好测一下，我重启一下这个项目。

好，启动完以后我就打开浏览器访问一下，访问首页，我登录一下，用AA来登录，登录以后我查看这个消息，你看私信有一条未读消息我就不读了，然后通知能不能点呢？可以点。然后有三类通知，然后用户CC评论了你的帖子，用户CC点赞了你的回复，用户CC关注了你分别这个数量是这样的。

好，这是我之前所做的一个测试的数据，然后我们并没有查看过，所以说都是未读的，这里很多都是未读的，当然也有一些已读的数据，这个数据是我做了一些处理，有的是未读，有的是已读。

好，总之我们就先这样子系统通知我们就搞定了，然后待会儿我们在开发详情的时候，我们点进去再细看到底有哪些条数据，然后的话再把已读给它设置上就可以了。

好，下面我们就来开发通知详情，开发通知详情，首先我们也是先从数据访问层开始写，我需要在数据访问层增加一个方法，增加方法要查询某一个主题它所包含的所有通知。

当然我们这个是要支持分页的，因为通知可能会很多几十条几百条都有可能，所以我们要查询某个主题所包含的通知列表，这个注释那么返回是一个list，封装的是message，然后方法名我就要携带notice复数。

你查询的是哪个人的通知？优质ID是要传进来，另外是哪个主题topic要重新来，最后还要支持分页，所以再加上 Offset limit，好，那么这就可以了，然后有人说你要分页不还得知道数量吗？数量你别忘了，刚才我们不是写了一个查询某个主题所包含的通知数量对吧？这个就可以了，所以就不用再写一个查询数量的方法了。

好，下面我就来实现这个方法，打开 MAC xml ID等于它resolve tag。Message，那么查询的字段和刚才其实我们写过的是一样的，可以拷贝好查询这么多字段，那么加筛选条件why？

Standards不等于2，你删除状态的不要，然后 and from ID应该等于一，因为通知系统发出的 to，ID等于传入的参数都在ID，然后主题 conversation我copy了conversation等于它，然后因为我们查到多条数据，所以最好做一个排序，这里按照倒叙奥德拜，great time disc倒序，然后加上分式条件，好，那么数据访问层写完了，下面我们来写业务层也很简单，再加一个方法就行了，他类似的卖水 find notice优质。

Id。Topic。主题。分页儿条件，offset limit，然后直接return my sale map。谁来 notice？User ID。Topic of site limit。

好，那么很快业务层也写完了，我们接下来重点的处理这个视图层，那么视图层还要加一个方法，还是在 message control加的方法先声明了它的访问路径，还叫notice，然后这个是通知详情，所以下一集路线我叫detail，并且你要查的是哪一类通知，你要把类别topic传进来，我希望你通过路径把这个数据带进来，所以说我加一个再加一级topic。

好，那么请求的方式方法名我叫get notice detail。那么这个方法需要几个参数，首先你得把路径中topic得到对吧？我需要利用 pass variable，好，其次你要分页，所以需要声明配置参数，再者我们需要给模板传数据，所以需要加上model。

好了，接下来第一步我首先要获取当前的用户，因为你查的时候是查当前用户的详细信息，当前用户先得到，得到以后因为是分页查询，那么我对分页条件加以设置 site。Limit对。五我希望每页显示5条数据，因为我的数据比较少，然后配着点set，pass。Notice。Detail。

但是下集 topic你还得带上参数里，有了就是pass，然后配置的site rose行数，行数我们可以查，数据量可以查 message service点，find notice com把user ID传进去，然后把 topic存进去，下面我就可以开始查询了，查询某个主题下的通知，查到的是集合历史的，我叫notice，list等于message service的find。Notice。好，传入user的ID，再传入 topic，债权边界条件 get upset。Get limit。

当然我们查到的结果当中还需要聚合一些其他的数据，比如说优质实体这个数据你要把它搞处理一下，所以说我需要再声明一个集合，但这个集合里面存的是MAC用来存聚合的数据，这个变量名我叫notice vivo this实体化一个新的 release，然后做一个判断。那么如果我们查到的集合它是非空的，我们就去聚合，否则就没有必要，我需要遍历一个处理。

然后这是历史遍历，每次遍历的时候，为了存聚合数据，我需要先实例化一个map，然后存数据首先我把这个通知先存进去，map点put，notice好先把通知存进去，然后再存什么再存这个内容，通知的内容，那个内容不是一个阶层字符串，你需要把它还原为对象对吧？

然后便于页面的访问，我还需要获取 content，当然你需要html，you做一个反转移的处理。点html，an escape。Notice点get content。然后我再把 content转化为 mat。我们写上一个方法的时候已经处理过了，这里我就不啰嗦了。好，那么得到了内容这个以后，那么我需要把它存到 map点到首先从user这个得靠user service查一下，find ID，然后的话是data点，get user ID，当然你需要对它转型成int，另外 MAC点put Entity time，data get Entity tap map。

They are put Entity ID，get Entity ID。

然后有两类通知它是带 post ID的有一个不带，我管他带不带我都给他存上，你用就用，不用就算了，你不带的空置的不影响的有data点get post ID，你想如果说data里没有post ID，我得到的是not，所以你map中存在 post ID是not，而对于这一类通知就是关注那一类，他不会用这个纸，所以无所谓不会报错的。

最后还有你看我们详情页面，那么在详情页面上，每每个消息上有一个消息的发布人要显示这个人的姓名，但是这里其实显示的不对，因为这个消息其实都是通过这个系统用户发过来的，这里应该显示的是系统用户的名字，这个系统用户这个名字我们也查一下，表里是有的，我们查一下，好。

所以回到刚才的程序里，我们再补充一个通知的作者这个叫from user service。Find user by ID，这个就是notice，get from ID。好了，最后别忘了，我们还要把 map存到 list里头去，notice will list点，艾特，那么这个数据最终处理完以后，我们需要把它放到model里是吧？给模板使用，猫都得艾特，那么名字我叫notice。复数 notice。没有list。

好了，然后还要注意，你看我们之前开发私信详情页面的时候，我们还设置过已读，是吧？当我们访问这个页面的时候，这一页数据我看到了，我要把它状态设置为已读，当时我们处已经处理过了逻辑，现在我们也设置已读，它的逻辑和以前是一样的，所以这里我就直接用以前的方式来处理就可以了。

设置已读，首先我要获取需要设置已读的那些个ID list，integer爱丽丝等于之前我们分成一个方法叫get let Alice，然后传一个集合给他，就not this list，好，然后判断一下，如果说集合非空，里边有数据需要处理，我就调用message service的read message，读取这个消息，读取这组消息，这个方法是可以复用的，别管是什么消息都可以复用。

好，这就完了，最后我们返回到模板site note得听，现在我们就把请求处理完了，处理完以后接下来我们就来处理这个模板，当然这个模板我们是通过点通知列表，然后访问的页面，所以我们需要把通知列表上路径给它处理一下，刚才我们不是把它们都设置为井号了，对吧？

这里需要处理一下。

好，那么看一下这个路径，你看这是评论类的，这个路径我应该设置为什么 th冒号，大括号，然后 notice，然后 detail然后你还要传入，下一集传入的是主题，但是其实我们主题就三类，这个主题我们不用写变量，直接在页面上这不同的地方写死就可以了，这第一类是评论类，我就直接写一个单词，comment这样就行了，这不用通过变量取，因为这里是已经能够明确就是这个值。

同理另外两个链接是类似的，换一个单词而已，换一个主题而已对吧？这个是点赞类的，这个是关注类的。

好，现在我就有链接能链到页面上去了，最后处理详情页面了，那么详情页面我们首先把它打找到，然后它的路径它的名字叫notice detail。因为从来没有编辑过，所以说我们需要从头开始做一些处理。

好，然后这里处理，然后头部做一个处理，中间是通知列表，其实这里我们需要便利，然后逐个遍历梳理，所以说我们不用留这么多然后把其他的删了，留一个就够了。好，尾部不用处理，GS要处理好。

接下来我们就来处理主体中间的内容。

这个内容首先是这有个返回按钮，目前返回一个假的页面，我们需要做一个处理，然后这里边我们直接写动态的路径不方便写，所以这样我就第二个方法立刻等于第二个方法，然后我去在这个方法里写上我要去哪个页面，我在下面写一个方法，function back，这个方法也比较简单，就一句话，只要跳转到对应的路径就可以了。

就是location点x-f等于调整到context，pass加上notice，list。调到这就行，这就完了。

接下来我们在处理中间要显示的内容，主要是要对 lm做便利的处理，每次遍历我们会得到一个map，我们遍历的是集合，了解好，那每次遍历我们那一开头显示一个图标，图标之前我们是写死的一个图标，但是其实系统用户他也是用户，他有他自己的图标，而且这个图标没准也会换，所以说我们这里也是动态的去获取，那么系统用户发送通知的用户，我们可以卖点，不让优点对吧？

然后点 high的URL就行了。

然后再往下看，这个地方什么落地三班车嫌疑人，这个应该是发送消息的那个人的名字，我就这样写th冒号。You text Deng yu，map there from user username。好，再往下，下面的话要显示一个通知发布的时间，我在这把这个时间给它写上去，要格式化。这个时间应该是map点。Notice。Great。Time。好，时间有了。然后再往下看，下面就是评论的消息了。

就这个消息了，当然这个未必是评论，我们在 Notice页面上点评论，点关注不同类别的通知，最终都是到这个页面上，所以这个页面上是要写所有类别的通知都要在这里显示，但是不同的类别的通知他这个话术不一样，又有点相似怎么办？

我们写三份做判断，比如说如果你是评论我就显示这份，如果你是典当，你显示那份写三个4班就完了，好。

这样我在处理之前我把换个行，不然太长，先把这个格式整理好以后，再copy用户某某评论了你的帖子都换换行，然后这几点点击查看，大概分这几步就行了。

好我再copy一下，我先写完一个再copy，反正那个代码是可以复用的很多，好这个spa我先判断一下，如果说我希望这样，如果说当前是评论，我就显示 style，你怎么判断它是不是评论呢？

我们可以判断一下，这个就是参数，我们可以判断那个参数，你看咱们这个条件里不是有个topic，其实我们可以判断这个topic就通过这个topic它的值是comment like还是follow，我们就知道当前是你点的是哪个主题，对吧？

好，所以回到刚才的页面，我们对它做判断，我觉得topic注意我们这个表达式里其实还可以写调对应的类型的方法，topic是个字符串对吧？我们可以调它的e cos调，这个方法判断它是否等于comment。

如果等于comment，我就显示评论相关的一些提示，好，我们需要显示的是哪一个用户给你发的，哪一个用户给你发过来的，给你做的评论？用户谁给你评论的，那就th冒号，Yoo Tex等于Net点，user点。User name。谁给你做的评论？然后他评论的是你的帖子，还是回复这也需要做判断那就是map点安贴tat等于一就是帖子，否则就是回复。

好，最后还有一个点击查看，你这里你要把这个链接路径写好，我点的时候能亮去链到哪去，其实就是链到帖子上去。因为无论是帖子还是回复，我们都是通过帖子详情去看的，所以这个地方要练到帖子详情的页面。好，我就写一下这个路径帖子，相信页面这里面需要带一个帖子ID，所以说带个变量加双竖线拼一下，然后 discuss详情detail。

下一集就是帖子ID，那我就map post排第五。好，这是对评论的消息的拼接还有点赞，类似做处理copy了，那点赞的话，这个地方就应该是等于like，对吧？当然这里面显示的也是username，然后也是点赞了你的帖子，还是回复处理方式一样，然后点击查看的路径也是链到详情页面上去，处理方式一样再考虑。

最后是关注的话 Ecos follow，然后关注的话就是提示用户某某这个是一样的，然后后面没有这么多这么啰嗦，评论了什么就是关注了你完了，用户某某关注了，你当然也要点击查看的时候，你是链到用户上面去好，链到用户上去路径要改了，改成user profile它的主页，然后下一级是用户的ID MAC点user ID，那么这些个显示的消息处理完以后，最后别忘了还有个分页，这个分页我们可以复印，不用重新编辑了。

好了，到这个页面我就写完了，写完以后咱们再重新测试，我腾出来不久重新编译。好，编译好以后打开浏览器，然后先放首页刷新一下，然后刚才那是静态页面访问首页，然后访问消息，系统通知对吧？

然后你看这里头这个点评论用户CC评论了你的帖子，这有一些评论的数据，然后看下一页行不行，下一页末页没问题，然后前面的话是用户的头像，这是用户名、system、系统，然后这个时间好，那么这个评论的消息好像显示的不对，因为你看这里每一个消息都显示true force应该显示的有问题，所以这里面我们需要排查一下是不是有问题，我就回到刚才的代码里看一下。好，一看很容易就发现这个错误了，就是死办上这个条件应该是其实是做判断的，就是他要不要显示这应该是个艺术，而不是说把他的判断结果显示出来，不是显示too fast对吧？

这个地方把里边给覆盖了，显示0改成if，里边内容应该问题不大，主要是刚才这个地方写错了，所以它显示成too fast forth，一共三个，好再重新编译。

回到浏览器，我刷新一下，你看这回就对了，用户CC评论了你的帖子，用户游客评论你的帖子等等，都是这样，用户某某评论了你的什么，都是这样的，比如说我点击查看，能练到帖子详情页面上去就可以了。

当然这里还可以点返回，就回到了这个地方。回来以后，那么还有一条未读出来可能是我一没有点到那一页，点一下下一页下一页下一页。因为有好多页，我也不知道哪一页是我点完了回去好。

所以说你看都变成了这个已读的，刚才有很多未读的，现在都变成已读的，然后数据一共是30条数据，分为6页每页5条，没错，好，这是评论点赞。有5个被堵的站，一共有20条数据，进去看，——下一页，一共4页，每页5条20条我现在都点过了，应该是都看过了，返回没有？未读的了。关注看一下。关注数据比较少，一共就11条数据。三页返回。

好了，你看现在现在我们这个功能基本上是没有大问题了，然后刚才我是在评论里练了，详情页面赞里没有练一下提示还不对。点赞的时候你看这是评论，这是点赞这块应该是改个字点赞的，你的帖子或回复，这是关注了你刚才这个字没对好，重新编译，再看再重来。进去。点赞了你的什么？没有问题，然后点击查看，看到的还是一条帖子形成页面没问题，回去看关注，然后用户CC关注了，你点击查看看到是CC的主页，没问题。

现在最后还有一个小的地方要处理，这是我们一开始也说了这个消息未读消息的总的数量应该是他加上他对吧？我现在还没有做处理。

好，我希望处理一下这个消息，这个消息的未读数量的显示，很显然应该用拦截器处理，因为我们每个请求都有这个消息，这个链接都要显示这个数据对吧？我要统一处理，所以我需要再新写一个连接器，那么我就在CTRL intercept目录下新建一个连接器，取名叫message，intercept行不行？做Intercept，写完以后当然我这里需要加一个注解，把它交给容器来管理这个病。

然后我要查的是当前用户的他的未读消息数量，所以说我怎么获取当前用户需要注入host后的对吧？另外我要查未读消息数量，我得通过message service来查，所以说我要注入 message service，那么我什么时候去查，当然我们拦截器有三个地方可以去做处理，一个是请求之前就CTRL调之前，一个是调CTRL之后，模板之前，其实我们在调CTRL之后调模板之前就可以，因为马上就要调模板，就要用这个数据，这个数据要在模板上展现对吧？

所以我就重写 post。忘了。忘了实现接口，我说怎么没有那个方法实现这个handler in the second这样我就可以去实现它的 post handle。

这个方法在这个方法里，首先我要获取当前的用户，我要查他的未读消息数量，然后查之前我还得判断一下你这人登没登录，如果你没登录我就不查了，如果优点不等于你登录了，并且这个数据我要通过mode on the wheel携带，我得判断一下有没有这个值，如果它不为空可以携带，我就继续处理，然后查未读消息数量，一共是要查两个对吧？

一个是私信，一个是通知。Int light安瑞 count等于 message service的fund。Letter。I read count。有点点get ID。好，同理再来 Notice。I read can't message service there find notice I really can't。User要改ID，好，那么查到以后我们给页面传的数据要把它俩加起来就可以了。Model and view点AD object，这是总的维度，数量我叫奥安瑞，康的这个值就是把它俩加起来。

好，那么拦截器这个逻辑很容易，你写完以后需要进行配置，所以我还要打开拦截器的配置配置文件，我先把刚才的拦截器注入进来，然后效仿着其他的逻辑就加一分就好。

好，我加的是拦截器，那么这个拦截器对于所有的静态的资源它都不用拦截，然后对于所有的动态的请求，它都要拦截就行了。

好，那么处理完拦截器以后，最终我们是要在head里显示这个数据，而head我们所有的页面都是共用index，因为我打开 index找到显示消息的位置，这个链接是没问题的，数量我们之前没有处理，这里面需要处理这个就很容易了，显示一个值就行了，叫奥安瑞的抗体好。

当然了，如果它是0就不用显示，所以判断一下如果不等于0，它如果等于0我就显示空值。

好了，处理完以后咱们最终再测一下 CTRL f9，重新编译，打开浏览器，我试一下访问首页一条未读就他这个通知，这样我再给他来一个新的同志，我再登录一个账号 CC，然后比如说我对 AA的帖子再做一个回复，随便回复点啥，随便回帖回完以后我去看AAA有没有一个新的消息，重新登录 k0 g9两条消息。

好没错，现在我们通知的显示就完成了最终通知和私信，总的未读数量我们也显示的成功了。好了，那么这个显示系统通知的功能我们就开发完了，我们这次课就演示到这里，咱们下次课再见。

# 第6章 Elasticsearch，分布式搜索引擎

## 6.1 Elasticsearch入门

40-Elasticsearch入门.mp4

这一章我们来学习elastic search，xs它简称叫es但是目前性能最好，也是最流行的一个搜索引擎，你像大名鼎鼎的 Facebook get up 维基百科国内的百度等等，很多公司那么都使用 es进行搜索，那么我们的项目中也有搜索功能，我们就选择es这么一个专业的搜索引擎来解决我们项目中的搜索的需求。

好，那么第一小节我们来学习 es入门相关的内容，主要是你要了解这个工具，另外的话我们把它安装好以后，在学习它的一些常用的命令怎么去做基本的操作，那么你对它足够熟悉了，我们在此基础上再去演示这个代码怎么写。

好，首先我们看一下 EMS相关的介绍，这里有一些我提炼出来的一个重点的解释，说es是一个分布式的 restful风格的搜索引擎分布式大家都能理解，就是说我们是采用多台服务器进行集群式部署，这样能够提高整体的性能。

那么restful是什么？它是一种设计风格，这种风格它规定了前后端你应该按照什么样的标准进行交互，那么不同的类型的HTTP请求，你的请求的格式应该是什么样的？其实它是对请求的标准的一种描述。也就是说那么一s它采用这样 Res否这样风格的一个设计，就是我们通过普通的HTTP的协议就能够访问它，所以说很实用很方便。

然后第二个它支持对各种类型的数据的检索，它所支持的数据，包括你结构化的数据可以，非结构化的数据也可以，甚至是地理位置也可以，一些指标相关的数据也行，总之各种类型的数据我们都可以通过它来搜索，非常强大。

然后在搜索的时候，es它的搜索速度很快，可它可以提供实时的搜索服务，你要知道就是说搜索引擎它进行搜索它是需要分几步的，首先我们需要把我们的数据先提交到搜索引擎里，或者说我们把数据要在搜索引擎里再存一份，然后搜索引擎在存这个数据的时候，它会对数据进行分词，然后索引建立索引，然后通过索引再去搜索这样的方式提高效率。

所以说那么其实很多搜索引擎它都能做到搜索，但是有的引擎你在这个数据提交的环节它就比较慢，它不能做到实时，有的引擎是你提交以后，它这个分词索引速度稍微慢一点，不能做到实时。

那么对于 EMS搜索引擎，那么在这两个环节它都能够做到实时，所以说它搜索速度非常快，而且能够提供实时的搜索服务，因此才是目前最流行的一个搜索引擎。

然后第三个因为它是集群式部署，那么如果说你想在这个集群里加一个服务器非常方便，所以说它便于水平扩展，便于加服务器。

因此它每秒有了集群以后，它可以处理的数据量非常的大，它可以处理 Pb级的海量数据就总而言之 es是目前性能最好，然后的话最方便使用的一个搜索引擎。

那么我们想学会 es首先我们需要了解它的一些术语，那么我们在了解它的术语之前我还要再强调一下，就是说刚才也说了，其实就是我们要想使用es去搜索数据，那么前提是这个数据你要在es里再存一份，所以从这个角度来说，你也可以把 es当做一个特殊的数据库，也可以这么看。

我们买SQL也好，还是reds也好等等，这个数据你要想通过它搜索，你把它转存一份存到 es里。好，那么基于这个前提就有了索引、类型、文档、字段这样的几个术语，这是和它里面存储数据数据的形态有关的一些术语。那么这个术语我们和我们的数据库my circle可以做一个对比，这样的话大家好理解。你看我们my circle纯数据，首先我们得有database，得有库，对吧？我们的库不叫community对吧？然后其次库里得有表推广是吧？每个表的纯某一种类型的数据，然后第三个表里的数据是多行的，很多行。



第四个是每一行数据它有分为很多列很多个字段对吧？那么其实索引 Es中的索引类型、文档字段，它和 my circle当中的数据库表行列相对应，也就是说es中的索引与数据库中的 database，数据库相对应es中的类型和my circle当中的退保表相对应一个类型，就是一张表，一个索引是一个库，因为这个概念和我们关心数据库差别比较大，所以一开始可能很别扭，但是你这样我按照顺序摆摆出来，然后你按照这个顺序去记还好。

第三个的文档，那么一个文档相当于 my circle中一张表里的一行一个文档是——行数据或者说是一条数据，那么在es里这一条数据叫文档，然后注意这个文档的数据结构，我们通常采用的是节省节省结构，然后杰森中的每一个属性我们叫字段，字段其实对应的马赛克就是一列，好，所以它对应的是 database，这个对应的是黑宝，这个对应的是肉，这个对应的是color，是这样的一个关系。

好，但是在 es6.0的版本当中，这些关系逐渐要发生变化，主要是集中在前两个地方要发生变化，主要是他想废弃类型的概念，废弃类型以后就没有与表对应的术语了，谁与表对应？

索引，所以从6.0以后索引其实它对应的是一张表，一个索引对应一张表，然后还对应行，这个还挺厉害。

然后6.0它保留了这个类型，只不过类型都固定成一个固定的单词，而不是各个表明了固定成一个单词就完了，到7.0的时候就彻底废弃掉了。

那么所以因为我们用比较新的版本，肯定是6.0以上的，因此对应关系大家要搞清楚，现在最新的它对表它对应行，它对应列类型逐渐的废弃了解，但是我为什么要介绍，有人说你既然用新的你就别介绍了，这个还不行，因为我们在后续的操作当中还会看到残存的一些信息，如果不介绍的话，你会感觉这是个啥东西很别扭，所以来龙去脉还必须得知道。

好了，这是es关于存储数据相关的一些术语，大家要了解。

第二个是和集群有关的，那么因为我们是分布式部署，所以说这个是采用集群的方案进行部署，那么我们一台 es服务器或者是多台组合在一起，那么就是一个集群，然后集群当中的每一台服务器，我们称之为节点，当然在其他的服务器当中，其实这个概念是通用的是统一的，不只是es这么称呼，然后分片，那么你注意一个索引相当于一个，然后表里的数据可能会非常的多，那么分片指的是对索引进一步的划分，那么一个索引在存的时候可以拆分成多个分片进行存储，这样的话并发能力就提高了。

你把一个索引拆分成多个分片或者说多个空间的话去存，那就可以并且并发来存，就比较能力就比较强。

提高并发能力，再一个是副本，还有一个副本的概念，副本是对分片的备份，一个分片可以包含多个副本，我们有了备份以后，万一说某一个副本数据丢了，挂了，那么还有其他的备份，这样的话提高系统的可用性。

所以说再回顾一下集群式部署，分布式部署，那么可以提高整体的处理能力，提高整个的性能，然后分片，所以也是提高并发的能力处理能力，副本的事情要提高可用性。

以上是 es相关的一些特点或者是它的一些术语，我就介绍到这，大家做一个了解，当然你也可以从 es的官网去做更多的一个了解，那么es官网是这个网址好了，那么接下来我给大家演示一下 es我们怎么安装怎么去用，这个是es的网址我打开了，打开以后它会做一个判断，一看我这个是中国的IP所以它自动切换成中文了，这还挺方便，它还支持中文，但是文档它不支持中文，它只是一个大致的介绍才是支持中文的。

然后那么如果说你想去查看 es相关的文档，你可以点学习，文档就非常直观了。

点一下，然后它这里面有什么you last take stake。Electric站它的技术栈，它和electric search有什么关系？其实 elastic设置它是整个技术栈中的一个产品，它这个产品其实最早就是es这一个产品，后来的话慢慢的扩充，有了很多其他的组件，慢慢的形成了一个技术栈，越来越多东西，但我们学的就是 Es这也是整个 es公司的核心，我要看他的文档就点这当然有其他的版本，你可以点比如说点这是7.2的文档，

然后的话这是目录，你点目录还有一些介绍，然后的话你可以看往下翻下一节等等，这里我们就不在课上去看了。

当然了我不看这个文档，我要做的是下载，我再回到首页，然后你可以点这个产品，产品里面其实你点这也能找到下载的位置，你点这儿就更直接一点下载点一下，下载以后你看有这么多产品，其实都是围绕以last设置构建的整个这些产品加起来，less take stake它的一个站。

好，我要下载它点 download，然后目前最新的版本是7.2.0，但是这里我想说的是 es我们暂时还最好别用最新的版本，为什么？因为我们所使用的思政部的他的 Pom文件是继承于 pom父pom当中声明的 es版本是6.4.3，那么那个版本他做了一个测试是没有问题，和当前的思政部的完全是兼容的。

当然其实对于其他的工具来说，一般我们用一个更新的安装一个更新的版本的工具，其实也可以，比如说我spring布的声明这工具，我用的包是2.0，我们其实装2.2.2，33.0可能往往一般都问题不大，但 es它6和7之间版本变化比较大，还真的不行，你忘了刚才我说的就76和7这个过程当中，它是逐渐废弃掉了，类型变化比较大。所以 sport当中它他用的是6.4.3，我们用最新的版本真的就不匹配，我试过。

对于一s这个工具，大家最好用的是和死人不得内部一致的版本号，6.4.3，所以这里我重点强调一下，那么要下载6.4.3也好办，往下看一看你看它这有一个past历史，以前的版本我就点一下。好，然后他就倒叙的方式列举了以前很多个版本，找一下6.4.3，还有点冒，这6.4.3就是它了。

好，然后点进去，那么比如说我这是windows系统，我就下这个点 windows就可以，下载一个zip的压缩包就可以了。

好，那么我这个已经提前下载好了，我就不再课上下载了，然后我们来看一下，安装的话就很简单了，就把它解压缩就可以了。我就解压缩一下解压文件，解压到d盘work目录下。

好，剪完以后大家看一下 Less take search6.4.3，那么解压完以后我们需要稍微改一下它的配置，那么它的配置是在config目录下，然后是文件，elastic search点yaml我就打开它。

好，打开以后我们不用说每一项都配，它，这里有很多的注释，有很多的解释不用每项都配。

那么有必要配的是这么几项，第一个格拉斯特内集群的名字，因为我们后面编程的时候要指定集群的名字，所以这里最好把它配一下，它默认叫man，我改一下，我叫改成叫脑壳的，好然后再看，然后最好把这个配一下，pass点data，就是说我们 es我们要往里存的数据，这个数据存到哪？

你看它默认的路径是针对Linux系统的一个路径，你这里必须得改了。

好，比如说我要把它存到c盘，好的d盘 d盘，然后work目录下data目录下，然后新建一个文件夹叫electric，然后-6.4.3。好，那么然后再建一个目录叫data，我是把它放到这个目录下，当然你任意定义放哪都行。

再一个还有一个这个地方 pass点logs，就是说那么它在这个工具在运行的时候会产生一些日志，日志存到哪去，我也希望存到 d盘work目录下，所以说我copy一下，但是存的最后后缀就不叫data，叫locks。

好，这样就可以了，目前的话这个目录应该还没有看一下。Work data有，但这里 electric还没有，那么运行的时候它应该会自动创建，我们指定好就可以了。

好了，那么配置文件这就配好了，配好以后我们再来配置一下环境变量，因为我们一会会通过命令行的方式运行几个 es常用的命令，那么为了访问它的命令比较方便，我们最好配置一下它的环境命令 work you last take search6.4.3，那么它的命令是在b目录下，所以我把 b配到环境变量里，pass新建一条这点开编辑然后新建一条。

好，那么 es我就配好了，配好以后那么我们还要装一个东西，中文的分词插件，为什么？就是说因为搜索引擎和别的不一样，大家都用搜索引擎，你至少用百度对吧？比如说我搜互联网校招，那么他一定会对这个词进行分词，至少会分成互联网一个词，校招一个词，对吧？这就是分词，然后搜互联网相关的内容，搜校招相关的内容。那么如果某个内容和互联网和校招都有关，这个肯定是优先排到前面去，对吧？如果只和一个关键词有关，那就排到后面去，大概是这个意思，大家应该都能理解。

那么 Es它默认的话，它是支持英文的分词，英文分词很简单了，英文分词就是一个单词，一个单词中间有一个空格就分了，而中文不一样，中文的话我们一句话中间是没有空格的对吧？这个工具默认它没有带中文的分词的插件，我们需要单独装一个才能够支持中文分词。好，我们的牛河社区，咱们用户群体都是中国人，所以说用的都是中文，这个是必不可少的。而中文分字插件我们在哪找？Es官方没有，这是第三方提供的，那么我们在低的普上可以找到，我们找一下。

好，这里我已经提前打开了 get up.com，然后你可以在搜索框里搜一下 get up它的搜索引擎就是一s搜一下us take search，然后分词中文分词插件叫ak好，然后搜索回车，那么第一个使用的最多的你来take search and nice。

Ok，点进去以后这里有这个代码你可以下载，但是我们不是要看源码。然后后面一些有这个版本的说明，就是说 lk的版本和es版本的对应关系，其实它就是严格基本上是比较严格的对应关系。

总之6点差对6点差6.4.3，你最好就对6.4.3颜色对应，然后这里边还有关于 es的一些介绍示例你都可以看，插件的示例都可以看，我们不细看了，一会我给你操作下载的话是点这儿。他提供了一个下载的链接，还是在 Get up网站里点一下。

打开以后它这里列举了很多版本，我们要找的是6.4.3。下一页6点 4点3点进去点，这个就可以了，那么我这已经提前下载好了给你看一下它。

好，那么下载好以后我们需要安装，解压缩就可以，但是必须解压缩到一个固定的目录下，必须解压缩到我告诉你，反正我就直接解压解压文件，必须解压。

刚才我们安装的 elastic search的路径下，所以我打开d盘，然后的话elastic search6.4.3，然后你要放到普拉克因斯里边，然后还要在普拉克因斯下面建一个ak的文件夹，把它放到 lk的路径下要注意，好我选中plus新建一个文件夹，然后取名为ik然后选择ak点确定解压到这里，我看解没解压，说好 work plan。

Ok已经解答说好了，刚才那个过程太快了，都没看出来好，然后解压缩完以后，我们只要看看一个东西，让你对这个分词插件有一个更多一点的了解。

我们主要是看它的配置，这里有很多配置文件，有很多 dic结尾的文档，就是字典，他分词的时候他搜索的时候，那么他怎么知道你哪几个字组成的是一个词，他怎么知道有哪些词？

那么是靠这个来配的，这里面其实有很多内容，你随便看一个比如说面点DAC打开，你看这里面就各种词，他搜到有这么一个东西的，就知道这是一个中文词，这是一个中文词，就这样的。

好了，当然了他这里边内置了很多词典，但是我们的语言是有发展的，咱们每一年不断的会流行一些网络新词，词典里肯定是没有的，那么如果有网络新词出现，你也希望能够被搜索引擎做分词做识别，能够搜到这个词怎么办？我们需要在这里配在 ik and Liza这个里配。

好，我打开这个文件看一下，这里就是ik的一个扩展的配置项，然后你可以在这里配你自定义的词典，一个什么点DAC文件，就是说你可以在这个目录下创建自己的disc文件，然后在这里指定，指定以后你只需要把网络新词写到你的 disc文件里就行，怎么写？

就像他写法一样，就是一个词一行就可以了，非常简单。好，然后还有一个他还配一个什么？这里可以配扩展的停止词。词典它有两个词典，一个是我们搜索的词，一个是停止的词。好，我们看一下它这里面有没有什么停止的这个词，你看这个就是stop work，dic它是默认带的，你看一下这是英文的，再找一个这个再点开，什么叫停止词？

你看就是这样的了，以使这样的得是这样就是停止词，这样的话你配了这个词以后，那么他在分词的时候，他看到这个词就知道这是这个词条就已经结束了，到此就结束了，这种词就是停止词你也可以去配，但其实基本上一概不用配，因为网络新词它一般造不出这种东西来好了，总之就是说如果你想配置一些你就是一些新的词汇就在这里去配，主要是增加你自己的自定义的DAC文件就可以了，然后DAC文件你就参考它自带的去写就可以。

好，现在这个分词插件也装好了，还不着急去使用，我们再安装一个工具，这里我列举了这个工具的网址叫post Mark这个工具。那么postmark它能够模拟外部客户端，说白了它就能够模拟网页发送 HTTP 请求，为什么需要这么一个工具？因为其实我们直接通过命令行去访问 Es服务器，如果是查询某些东西还好，但是如果要往里存东西，这个命令太长了，记不住也很难写，特别麻烦。

那么因为 es的支持用HTTP的方式去访问，如果我们有一个现成的网页，能够有框往里填数据不就方便了，但是我们又没有网页，那么这个post Mark它就能替代网页，然后你可以通过框去构造一些数据，然后提交给 es服务器，这样比较方便。

当然其实不只是往es中存数据，即便是我们从es中搜索数据，如果说我们搜索的规则比较复杂，那个时候命令也非常难写，那个时候我们提交的搜索条件用它来提交也会比较方便。

总之为了提高 EMS我们入门的时候使用的体验，我们就用这个工具比较好，那么打开这个网址就可以下载这个工具，我把它下载一下，我已经提前打开了，这个页面get boost com，然后的话打开以后你可以点 get started，然后这里就download，那么我当前是windows系统，64位我就点它就完了，你根据实际情况去下载，我已经提前下载好了，看一下，安装一下。

好，稍微有一点点慢，大家别着急，咱慢慢来，安装完以后它会提示你注册一个账号，你就填邮箱，用户名密码，可以注册一个账号，然后那么我这已经注册好了一个账号，我就直接登录，就点刚才点31我登录，然后在登录的时候，你用这个账号或邮箱都能登录。

好登录了，登录以后，那么我们要访问 es服务器，其实就是利用这个页签，然后这里不是敲命令，而是在这框里输入一些数据，大概这样的操作，然后你看我登录以后，它会把我之前在别的电脑上也是利用这个账号所做的一些操作，他都已经给他下载过来，它同步过来了，这就是它的好处，当然我们不用看好。

那么现在你看我就把我这次课需要用到的这些工具都安装好了，再回顾一下第一个安装了以来CK设置服务器，对吧？第二个我们安装了这个分词的插件，第三个安装了一个模拟外部客户端的工具 postman。

好，那么这里头 es服务器和中文分词插件是必要的，toast man是不必要的，没有它我们也可以访问，通过命令行也可以，或者是我们通过其他的类似的工具也行，这个post不是唯一的，你要知道。

好，那么接下来我给大家演示一下我们怎么去使用 es首先我们演示通过命令行的方式怎么访问它，当然你在访问之前你得把它启动起来，它就好像一个数据库一样，你得启动起来。

好，我打开 EMS安装的目录，然后的话b打开b windows系统，我就是运行 bat当然如果你是Linux或者是 units，系统就是mic os那么你需要透明的行运行命令。

好，我用的batbat的话你通过命令行，访问也可以，直接双击也可以，这里我就省事，我就双击一下。双击以后稍等一会儿，它启首次启动的速度有点慢。

好，这已经起来了，你看当它提示绑定了9200的端口的时候就ok了，那么 es服务器它默认占用的是9200端口，好，启动完以后我把它先最小化一下，我打开面谈工具来访问一下，打开命令行工具作为客户端来访问一下这个服务器好。

那么因为刚才我们配了这个环境变量，所以说我可以直接使用 es的命令去访问它。

首先我们先执行一个命令去查看一下 es集群的健康状况，那么如果你去官网手册，他也是第一个也是建议你执行这样一个命令，看一下这个机器人是怎么样的。

好命令这样写，curl杠x杠get不是杠就是直接get，然后的话写上你要访问的服务器的地址，那么本机有个house的端口9200，好这是我要访问的服务器，然后get表示说我是一个获取 get请求获取，然后获取什么，后面你要接着写cat，然后的话健康状况 health，然后带一个参数v比方说你在显示数据的时候给我显示一个标题出来，好，我写完以后回车，我把它弄的大一点。

好你看它有两行，第一行是标题，提示你下面数据显示的什么内容，第二行是具体的数据，你比如说time是时间格，拉斯特是集群，status是状态，状态 green表示说很健康没有问题，如果是red就表示有问题，然后这里面有多少个节点，no的，还有多少个分片，没有等等，就这样。

因为我们现在里边还没存数据好了，这是我们查看集群的健康状况。

再来一个，比如说我想看一看这个集群中有什么节点，你可以这样写，curl杠x然后的话还是get获取，然后路口29 200-cat，然后的话我们查看的是节点 no，带个v比方说我也要标题，那么它会显示出其实只有一个节点，它IP是本地127.0.0.1，然后这是堆内存的占用量，然后 CPU占用量，内存占用量等等，还有节点的角色等等，好，总之你只能看到——一条数据，那么就是一个节点。

好，再来，那么刚才我们介绍 es术语的时候，我们介绍过索引，那么6.0以上的版本索引就相当于是一个表。比如说我想看一下我当前的 Es服务器里有多少个索引，怎么看？这样写。

杠x概念还是获取，然后后面跟的是服务器的内容， host9200，然后还是杠cat，那么查看索引这是index它的复数的形式，带上v表示我想看这个标题，事实上它只打印出一行标题，然后没有具体数据，比方说我现在没有索引，因为确实我们新装的里边什么数据都没有，对吧？

好，没有索引，我怎么去建索引呢？

这个也比较容易串联锁也是这样的，curl杠x然后我们采用的是put方式的请求， port表示说推送数据新增数据，然后后面写 local house，冒号9200这个服务器，然后斜线后面跟的是你要建的所有的名字，这里我随便写一个叫test，回车，然后它返回的结果是json格式的acknowledge two比如说成功了，超你好的所有名字就叫test，没有问题。

我想确认一下，我再查一下，再执行一下命令，查一下有多少个索引。

好，再执行你看有一个索引，但是索引的健康状况是yellow，它不是green，不是最健康的状态，为什么？因为我们在建索引的时候没有给他指定它分片副本，所以没有备份他没有备份，因此他认为你不是最健康的状态，有可能是有一定的风险，但是能不能用？可以用，只是一个警告而已相当于。

好，如果说索引我不想要了，我删掉那就curl杠x然后执行给力他类型的请求。那么 Http协议它里边包含了各种类型的请求，我们之前常用的是get post，其实还有put delete，那么它们都表示特定的含义，那delete就直观了。如果说你对 HTTP协议各种请求的类型还不太熟悉，那么你去翻阅一下，就像我给你提供的网址，我直接拿网址从里面可以找到相关的介绍，你去看一下好。

那么删除的是local host9200服务器上面的名为test的索引。你看返回的是数，ok删掉了，删掉以后我们再查一下，所以就没了。

好，那么以上是我们通过一些命令的方式来访问 Es服务器，那么如果说我们只是做这个查询，只是创建一个索引还好，当我们要进行复杂的搜索的时候，当我们要进行往s服务器里添加大量的数据的时候，这个就太麻烦了，就写不出来了。

所以接下来我再给大家演示一下，我们用这个post来代替一个网页，代替一个web，就是用它来代替web客户端去访问 es因为它是web客户端，它也执行的是HTTP协议，看会怎么样，然后我们在多讲几个es的操作，让你对 es有一个更直观的了解。

好，这样我把刚才我们执行的那几个命令挑几个咱们再用post man再演示一下，比如说现在我要用postman查一下 es里一共有多少个索引，我不是执行命令，我是直接做操作，首先我查询所以get请求这里选get默认就是get，然后这边你要访问的服务器地址，你要写上local house9200，然后你要做查询，后面加上刚看的，你要查询有多少个索引，index、复数加上v就是这样，总之你看比刚才命令稍微精简一点，然后点send或者回车，都可以触发访问点send，那么查到了只有一行数据标题，没有具体的内容，因为我们刚才不是把 test给删了，对吧？

好，下面我们再重建一下，索引重建的话是另外的请求类型，我再加个页签。然后这块选择 put。新建 put，然后的话local host9200是吧？那么所以名字还叫test好，然后回车就可以了。反馈结果还是 Jason 格式表示成功了。成功以后回到刚才的页签，我看这边能不能关掉，关不掉。就这样我再查一下，你看所以你就有了是吧？这是新建一个索引的方式，如果删除索引我再来个页签，我一般习惯的是一种类型的访问，我就来一个页签，这是得力的。

删除local past9200，然后 Test，这样就行了，然后回车成功再查一下，又没了。好，你看我们使用托斯曼一样可以执行这些命令，只不过它比我们刚才在命令行敲命令略微简洁一点。

好，我们再往下看，我们访问 es服务器最主要的是要往里存数据，然后好搜到这个数据对吧？不只是建索引，接收一下相当于建立一个表里就有数据，我们怎么往里面插入数据，给大家演示插入数据， put提交数据，然后访问的是6个house， 9200test索引，当然有人说你刚才不是把test给删了吗？

是我给删了，但你注意如果说我直接向es服务器执行一个命令，要求插入一条插入一个文档就插入一行数据，那么我在命令当中指定了不存在的索引，那么它会自动建索引，然后查数据，所以这个是一个隐含的，它有一个隐含的操作，只要我指定索引的名字它会自动建索引。

好test，然后再写杠dog，这个dog其实是文档的类型，那么因为6.0开始这个类型逐渐要废弃，但6.0没有彻底废弃，而是给它一个固定的名字，通常就固定为dog，而test晋升就降级为表明它就是固定就站位，不用管它，然后斜线，你要指定这个数据的ID，我第一条数据ID是一。

好再说一遍提交数据。用post用put请求，这是服务器 IP和端口，这个是索引是哪张表？ Dog占个位而已，不用管它，这是ID。然后你ID有了，你具体要存什么数据，这数据怎么提交？我们是在这儿在整个请求这是请求路径。我们是在请求的body里请求体里带上这一条数据请求体选 Body，然后数据有各种形式，我们选肉，然后这块选json，我们用json的形式提交数据。

所以可想而知，如果说你还是用命令行的话，我们想就添加用请求体的添加一个杰森格式的数据，那边内行就很难写，但是我们有波斯曼，我们点一点在这写就方便很多了。好，所以这里我写一个json格式的数据提交给 es服务器。好，比如说我要提交的数据包含两个字段，一个叫开头，开头就是标题，随便写hello。好，然后再来一个 content，内容，内容随便写，好，还有完了，当然你也可以加更多的字段， json的形式，这就写完了。

那么写完以后，下面我就要执行了，我就算了一下算了一下，我要看结果，好你看这个结果treated创建成功了， successful一成功了成功以后我们去查一下，我就查一下这条数据，怎么来查这条数据，查还是get是吧？

怎么查这条数据这样 localhost9200，那么test索引到站位，然后一比方说我要查这个索引之下，ID为一的数据， send你看我们最终查到了一条数据是属于test，索引类型是dog，可以忽略了 ID是一，然后版本是一就没更新过，然后它的内容包括两个字段， title和content这数据就有了，没有问题。

好，那么这是我们增加一个文档，增加一条数据，如果要删这个数据，如果要删，有人说如果要改数据改的话，其实还是用铺的，你把这个值一改再提交它，其实底层会自动的先删再新加，它是这样一个规则。

所以这个改我就不演示了，删怎么删，删肯定是得力的，我演示一下删，然后 Google host 9200test， dog占位删除ID为一的数据，然后 send你看返回说delete的删掉了，确认删掉了吗？我再查一下。Get一下放的force。找不到没有了，删掉了，好，那么到现在我们就知道了我们该怎么去建一个索引，我们怎么往索引中增加数据，修改数据，删除数据查询，数据增删改查，知道了，这只是说我们怎么把数据维护到 es服务器之内，对吧？

用正常改查建索引的操作。

但你要知道我们一s存在的价值是在于说它里面的数据能够被我们所搜索，搜索不是让我们查数据库 select from加一个条件加两个条件那么搜，而是说我们提供一句话一个句子，这个句子需要有人帮我们分词拆分成几个词条，然后再去库里去匹配，匹配的时候还不是固定的某一个字段，可能是我既想搜 title，又想搜content，又想搜某字段，我是一个全文的搜索全文的匹配，那么数据库这一点是做不到

的，它不能分词，它默认它也没法去支持全文的这种匹配的方式，对吧？

这数据库做不到，那么搜索引擎很容易就能实现，所以下面我给大家演示一下我们怎么去实现这种搜索，那么我们要实现搜索，肯定你得有点数据，对吧？你一条数据都没有怎么搜，我往里存几条数据存三条，意思一下，好我重新存数据刚才都删掉了，重新存put put然后还是往test里存，然后ID是一刚才删掉了，重来，ID是一。

好这个数据比如说所以它就两个字段，title content就可以。那么这个title我就要最后搞个中文的，咱们也好好看一下中文的插件好不好用？

互联网求职，然后content的内容是怎么想寻求一份运营的岗位，当然我们实际的项目环境里这个字比他多得多，我现在守桥意思一下得了，然后三个一条数据就好了，再来一条，第二个ID是二，这个抬头我叫互联网求职，这个叫互联网招聘好内容，再编一个内容，我就要招聘一名资深程序员好，然后send created成功了第二条数据再来第三条，第三条这个标题我叫实习生推荐内容是吧？

本人在一家互联网公司任职，然后可推荐实习开发岗位这么一句话上的created ok好。

我再确认一下，我刚刚提交的这三个数据真的存进去了吗？在这搜一下send。没问题有对吧？互联网求职，然后二互联网招聘没问题。三实习生推荐，你看三条数据都有了以后，接下来我要搜索了，这样我再建一个新的圈，搜索还是盖也是查，只不过它比刚才这个普通的长用ID长要复杂一点，你上哪去了？这好。

那么当然我们访问的还是local house，9200这个服务器，我们要搜的目标还是test索引，那么我可以加一个社区单词下划线社区这个单词，比如说我要从所以你搜索信息，目前我只是说搜索，但是我没有加任何条件，那就全搜出来了，Send。

好你看结果his就是命中了几条数据三条，然后的话你看这是第一条数据，互联网招聘，第二条数据互联网求职，第三条招聘，第三条就是实习生推荐，这三条数据都搜出来了，这是没有加条件设置搜索我加一个条件，比如说我想搜title，问号你加条件q等于固定的就是q等于，然后以开头为条件来搜索，就开头冒号，后面加上关键词，你想用什么关键词来搜？

比如说我想搜互联网，这个意思就是说我要搜开头当中包含国内互联网这个字的信息三个。好，你看命中两条，其中一个互联网招聘，一个是互联网求职没问题两条，另外一个没有互联网对吧？好，同样的我也可以去搜content，比如说我要搜content当中带有运营实习这个词的关键词的内容。好我send。

好，你看命中还是两条，第一条寻求一份运营的岗位，有人说不对，你这个条件是运营实习，他这里没有实习俩字，你注意这是搜索引擎，当你提交一个条件的时候，他会先对这个条件分词分成若干个词条，你最容易想的它至少要分成运营和实习两个词条，然后分别去匹配。

那么你看这句话是不是能够匹配上运营这个词条，当然是可以的，所以他就被收到了。

然后你看第二条数据，这里边没有运营，但有实习，所以它匹配的是另外一个词条，这是搜索引擎它内置的一个能力，它能够对你提供的关键词进行分词，我们写的是中文，就说明中文的分词插件起了作用，对你注意。

好，还有一种情况，比如说我不只是要搜title，我不只是要搜content，我要搜的是我既要搜title又要搜content，只要包含我这个词条就都会显出来，这个时候就不能简单这样写了，因为我们搜的逻辑有点复杂了，这是多个字段同时匹配，那么通过路径直接搞定了，这个时候怎么办呢？

把路径改成杠设计，那么具体的搜索条件，我们通过请求体去提交复杂的搜索条件，所以你看这就是post慢它的好处，如果是命令行，这个请求体就太复杂了，不好写。好我请求体我用json构造，所以选肉，然后这块选json好大括号，然后这里你要写写什么？Carry，就我要用什么样的形式来搜索，那么后面的话carry后面这个值是一个符合的值，还是一个json，还是个大括号？

换行一下，然后 k 冒号，y6 还是一个符合的对象，还是一个精神 k 要写成什么？Martin match，Martin 就是多个 match 就是匹配，就是多个字段匹配的意思，然后的话大括号里你还要继续写克，瑞克瑞写什么写的是你想匹配你的关键字，比如说我这个关键字就互联网，关键字互联网，好了，然后再来还要写 Feels 就是字段的意思，后面是方括号，这是一个数组，里边可以写多个字段，指定你到底要在哪些字段里去搜互联网。

那我既要搜抬头，又要搜 content，这个查询就构造完了，那么 es 它的搜索其实还有很多规则，远比这个复杂，那么我们了解到这基本上就能够理解他的大概的能力了，后面的那些更复杂的我们就不挨个去演示了，你可以从手册里去自己慢慢去看。

好 body 我构造完以后散了一下，然后看结果 his 命中三条数据，第一条标题中有互联网对吧？第二条标题中有互联网，第三条标题没有，但内容都有互联网，你看那么对于整个索引多个字段我们都能够匹配到，而且如果说你中文是有多个词组成的，它也能够进行分词，那么基本上就能符合我们的实际的业务的需要。

当然了我们现在这种形式只是说我们去安装完了 EMS，我们只是去测一下，看看到底好不好用。

然后在测的过程中，我们体会一下 es 它到底是有什么样的能力，到底是大概用什么样的形式能够去访问它，能够做什么，有一个直观的印象。

那么下次课我们再去用 spring 整合 es 然后通过 Java 代码的方式，然后去访 es 那个是我们日后经常要用到的访问的方式，但这个你最好也要做一个必要的了解。好了，那么这次课我们就把 esn 的入门的内容就介绍完了，我们就演示到这里，咱们下次课再见。

## 6.2 Spring整合Elasticsearch

41-Spring整合Elasticsearch.mp4

这。

次课我们来学习如何利用 spring 整合 elastic，那么整合的方式也并不麻烦，大致分为三个步骤，第一步你需要先把这个包导进来，那么这个包是斯文布的提供的叫 spring boots 大特带特。

Last 打完包以后，我们需要在 spring 布的配置文件里对 es 加以配置，那么有必要配的是两方面，一个是集群的名字，cluster name，一个是集群的节点 not 那么配完以后就可以调用 spring 给我们提供的关于 es 的 API 去访问 es 的服务器了，那么它给我们提供的 API 主要是两个方面是一个类叫 elastic search template，那么我们可以调 template 去访问 es 它里面提供了各种增删改查的方法。

然后第二个我们也可以等使用这个接口叫 analytic search repository。那么我们使用这个接口的时候，需要定义一个子接口来继承于它指定我们要访问什么样的数据，然后简单的声明以后，spring 会自动的实现这个接口，而这个接口当中又包含了各种仿 es 的常见的方法，所以说也很方便。

那么怎么说就这两个 API 的关系是其实接口的实现，其实它底层依赖于 template，但是我们使用接口的方式会更方便，所以我们一开始尽量去用这个接口，那么但是在某些情况下还不得不用 template，所以这两个 API 都会用到我课上，都会给大家做一个演示。

好，下面我们就来实现这几步，我们把配置好以后，我们写主要是写测试类，然后的话在测试类当中演示如何去访问 es 如何去对这个帖子进行搜索，我们把数据库当中的帖子存到 es 服务器里，然后去搜帖子。

好，那么我首先要打包，我先搜一下，我已经打开了 maven，reporter 到 com 这里你要输入 spring elastic search 搜一下。好，那么在这个结果中我们能在里面找到我们所依赖的包点进去，好，我选一个最新的版本，然后把它掏一下，粘贴到这个泡沫点 xml 里，稍等一下，它在下载。

好，那么因为这个包是 spring 布的提供的，那么在 spring 布的负泡沫当中，它对技术工具的版本做了一个声明，所以说我一般情况下像这样的包我习惯于把 word 去掉，然后就直接依赖于它附泡沫中所声明的版本，因为那个版本是他经过详详细的测试，没有问题的。



好，这个包打完以后，接下来我们需要做一些配置，我们打开 practice配置文件，那么在后面追加一项配置，es的配置我们主要是针对叫依赖txt类进行配置。好，然后刚才也说了主要是要配两项，一个是集群的名字，这样写我就直接写出来了，spring。

第二data，第二 elastic shard。Cluster。Name。

然后。

集群的名字，我们上次课在装 es的时候，我们不是改了它的配置文件里，我改了一下那个名字叫脑壳的还记得写上正确的名字，然后接下来我还要去配这个节点，因为一s它是集群式分布式部署有很多节点，当然我们这是单节点，只有一个，我们需要声明节点，spring data这copy一下，上面写错了，一来是txt，然后

cluster knows等于。

这里我只有一个节点，你要写的是这个节点的IP地址，冒号和端口，我们本机的IP就是127.0.0.1，当然也可以写localhost，127写错了，它和localhost的等价，冒号，注意我们上次的访问 es的服务器用的是9200端口，那么你注意 es它有两个端口，9200是HTTP访问的端口，它还有一个9300的端口也是默认启用的，那么9300是TCP端口，我们服务我们应用服务通常会用9300 TCP去访问它。

好，那么配完以后，接下来我们还要做一个小的事情，做什么事情我需要解决一个冲突，就是一来是txt，它的底层是基于native，然后我们之前所安装并使用的redis，它的底层也基于 net，那么这两者在使用native在启动native的时候有冲突，所以冲突主要是体现在 es它的底层的代码上，有个地方有点小问题，我们需要稍微做一个变通，具体哪有问题我给大家看一下，有一个类叫做night run time，90 run time。

好，那么我登录了一下SARS，下载一下源码，方便看好。

然后这一类当中有一个静态的类类当中有这样一段代码，你看set available processors就设置可用的处理器，然后你看这块他抛了一个异常，当 available process是不等于0的时候，它抛异常，然后有这样一个提示，如果说我们就现在配到这种程度，我们直接去启动的时候就会报这样的错误。

他认为 processor net的处理器已经被设置好了，被谁设置好了，其实 redis启动的时候就已经做过设置，他认为重复了，它就不启动了，就会有问题，那么报这个错那就得看这段代码是谁调的，这段代码其实这部分程序其实是由es Diao导致的错误，es哪调我们再看一下。

还有个类叫94。You tell us下载一些源码，这个类是es底层它封装的一个类，然后它这里边有一个方法在这个方法之内它就掉了。你看掉了90run time set，一旦这句话被调，基本上它里边就报错，因为它传的数据不等于0，因为 redis已经启动了 net，已经初始化了net，所以这个数不为0它就报错了。

我们希望它的别报错，因为你怎么说 red is它依赖于net，你依赖于net，这个都可以依赖，你不要因为人家依赖了，你就不能依赖，所以它底层的处理它没有考虑这种大家都依赖的情况，有点这块他处理有点狭隘了，好怎么解决这个问题？

他也留了一个口，你看这个地方有个开关，那么如果条件不满足，比如说它返回force，这就是return下面就不会被执行，就不会被报错，这报这也报错。但有个开关，这个开关如果force就不值钱。好，这个开关是什么？它会读取系统的一个属性，属性的key是它你只要把它值设置为force，就不会执行下面的程序，就不会执行这个检查了。

好了，所以我们需要做一件事，就是在我们的服务启动的时候，在比较早的时候把它设置为force，好，我在哪做？我在这里做 Application。你要知道这是我们整个应用的核心的入口的配置类，配置类是最先被加载的对吧？

那么我在这里写上这样一段程序叫post construct，大家应该还对注解有一点印象吧，就是说注解是用来管理这个类的生命周期的，主要是用来管理类的初始化的方法，那么它由注解所修饰的方法会在构造器调研完以后被执行，所以通常是初始化方法，我在 bin的初始化方法里去设置参数足够早，因为这个类是最先被加载的，这里我写个注释写清楚，是解决 net。

启动冲突的问题。

那么。

设置一个系统的属性，system点set12天，key刚才我call了把它设置为force就行。

注意这个force它需要的是一个字符串类型参数，我就字符串force它底层的会转，他用的时候会转型我们就不管了。

再一个解释一下我们这个处理方案是从哪找到的，从刚才这个You tubes里边找到的，所以说你可以去看94you twos，然后它里边有个方法，这个方法。

好，那么你课后可以自己去看一看它的这一个逻辑，这是我一开始使用es的时候也是这样，没有注意到这1.1启动就报错，后来的话跟踪代码跟踪到这，发现这个问题好，在他留这么一个开关了。

好，现在你看我已经把包也倒进来了，配置也配置好了，那么隐藏的坑咱们也做了一个解决。

那么接下来我就要开始写我们与需求相关的代码，我之前我也说了我们要做什么，我要把我们现在数据库里存的那些帖子想办法再存到es服务器里，然后我们去es服务器搜索帖子，那么有两种方式能够做这样的事情，一个是使用template，一个是使用repository。

那么我说了这种方案简单，我们优先用这种方案，当有些需求它不好解决的时候，我们再尝试用template，好，那么用reporter这个方案，它所有的代码实现类都是自动生成的，但是生成之前我们需要做一些配置，需要告诉他那我们就怎么说帖子这个表和es里要存的索引，它的索引不是和表相对，它俩之间是一个什么样的对应关系，然后这个表存到es里变成索引的时候，那么每个字段对应是什么样的类型，用什么方式去搜索，这些你都要去做配做配置。

那么这个配置它不需要我们写像麦迪斯那样的x面文件，我们通过注解就可以。

好注解写到哪上去，要写到实体类上去，因为我们是针对帖子的操作，所以我打开discourse post在这上通过注解进行一些配置，首先我需要在类上加一个注解，叫document，document里需要声明，index，name等于什么？需要声明，type等于什么？然后通常还要声明，shock等于什么？Shock等于什么？还要声明的。Replicas等于什么？

这俩是数字，不是字符串，就是说spring整合的Es它所提供的这项技术，它的底层在访问es服务器的时候，它会自动的将我们的实体数据和es服务器里边的索引进行映射，你实体映射到哪个索引上去，映射到哪个类型上去，映射的时候你需要创建几个分片，几个副本你在这里指定，那么我们将来调API的时候，如果它检测到没有索引，它会自动传你索引，然后没有分片没有副本，它会自动根据你的配置去创建，然后的话再往索引里插入数据，所以说我们需要做这样一个配置，这个配置就很好配，你首先给索引取个名字，实体类叫disc，post，那么索引的名字我希望这样取叫disc cast post全消息就可以了。Discourse post，然后tap，因为现在的es版本tap被弱化，将来会完全被废弃，所以我们当前的版本就把它写为一个固定的dog，就完了，就不用太管它，然后分片你随便这里我写6片分6个区域，副本你备份几份三份，当然你随便配，其实我们往往是根据这个服务器的处理能力来配，这里我就随便写几个，不要写太多就行了。

另外你这样配死人就知道了实体和索引有一个对应关系，那么实体当中有什么的属性？它和索引中的那些个字段是怎么样一个对应关系，所以说你在属性上也需要加以配置。

好，那么第一个是ID整个是组件，这个ID的话我们通常会加一个ID，主角写错了，加一个ID注解，那死不愿意看，这是ID，那么在建索引的时候它也会把这个数据存到ID的字段上去。

好第二个是user ID，优质ID是一个普通的字段，我们加一个feel，然后里边要声明它的类型type，然后是防废物的态度，接口的常量，最后的tap点怎么不自动弹出来了？Feel的tap点儿，inter整数。好，然后其实其他的都好说，重点是Title和content。你想一想我们搜帖子搜什么？我们主要就是搜这个标题和内容对吧？从这里边搜索我们想要搜的匹配的关键字，关于Title和content它的配置就尤为重要，当然它也是feel的，它也是普通的属性，但是我们声明的类型要等于feel the type点，text，文本，文本。

好，除此以外还要说明另外两个属性，一个叫analyzer等于什么？还有一个叫奢侈。来者等于什么？这是解析器怎么说？存储的时候的解析器，还有搜索的时候的解析器，什么意思？解释一下。举个例子，比如说我们要往 Es服务器里存的抬头，要存的抬头是这样一句话叫互联网校招。那么他存的存完以后，他需要给这句话建立索引，其实就是把这句话提炼出关键词，用这个关键词来关联这句话，将来我们通过这个关键词去搜的时候，因为有关联就能搜到这句话，就能搜到这个内容。

你想他在存的时候是不是应该尽可能的把这句话拆分出更多的关键词，创建出更多的词条，这样的话这句话能够就增加它搜索的范围是吧？

应该是这个意思，所以在保存的时候，应该是把这一句话把内容尽可能地拆分成多个词条与之匹与之匹配，增加搜索的范围，这个时候我们需要用一个范围非常大的一个分词器，那么我们所安装的中文的分词器，它里边有这样一个功能的分词器叫ak。

杠 max down word。

你看这个单词也很容易理解它的意思，最多的单词，他会把这句话拆分出最多的单词，然后与这句话建立一个索引，与这句话去匹配，增大它的搜索范围。

好，好比如说你这句话存起来了，将来我要去搜内容的时候，比如说我搜的时候，我恰好输入的搜索条件就是互联网校招，你想我搜的时候互联网销售他有必要说拆分出那么多词吗？当然有人说你 max word拆分这句话能拆分出多少个词，我给你举个例子，比如说他可以这样拆互联网一个词，互联一个词，联网一个词，网校一个词，校招一个词，大概五六个，其实这都是指互联网网校校招互联网对吧？

这都是可以成独立的词的，它给它拆分成五六个词，然后与这句话相匹配，但是我们搜索的时候有必要这么猜吗？你搜索的时候肯定是我们这句话反映出来的，我想搜的是和互联网有关的内容和校招有关的内容。如果这两项都匹配最好对吧？匹配一项也凑合，是这个意思。

所以说我们搜索的时候不应该采用这样的分词器，你没必要把它拆得那么细，而是我们用聪明的方式能够猜出你意图的方式去做拆分的稍微粗一点就可以了，拆分的符合你的预期一点就可以了，所以说这个时候我们采用另外一个分子器叫ak smart，一个聪明的分子器它会洞察你的意图，拆分出尽可能少的，但是满足你需要的这样的词汇。

好，总之我这里写的这两个字符串是什么？是两个分词器的名字。总之我们在存储数据的时候采用的是分词器，能拆分出更多的词来增加搜索范围。我们在搜索的时候采用的是聪明的分词器，那么拆分出较少的词汇，然后满足你的需求就可以了。

好，这是 Title，你title这么处理，再举个例子，刚才比如说 smart，我搜的时候，他可能只拆出互联网校招两个词就够了，别的他就不会拆这样的。

好，那么同理这个content也可以这样去处理，一模一样，它也是feel的，然后的话也是text也是存储是用分词器，然后搜索是如果有其他的可以被搜索的字段，我们一律这样写就可以了，都是这样写就可以了。

好了，然后再往后看，下面是 tap的话是个整数，整数其实和 uzid显然是一样的，对吧？好再来，然后下面是status，那也是这样的，对吧？好，然后下面是date稍微有点特别得单独写feel的，然后 tap等于 feel the type点。这里不有date，恰好是满足日期的这么一个需要。好，再来，下面这个是整数还是它？然后这个是小数double，你就feel the type的double，然后这样就行了。

好了，到现在我就把实体类就配好了，实体类，我们加上这些注解以后，它和es之间就建立了联系，它和es它整个对象和es中的索引有关联，每一个字段，每一个属性和es中索引的字段类型之间是这样一种转换关系。

就很清楚了，那么有了配置以后，spring底层才能够帮我们生成对应的时间内好很关键，那么配置好以后，下面我们就可以去定义刚才我们所说的 Reporter接口。

好，那么我在dao这里边定义，但是我不在这直接定义我再建一个子包，因为毕竟访问—s和访问关于数据库还是有区别的。

它的实现方式也不一样，所以我单独建个子包，把它归纳到子包里，子包我就叫it社区。

然后在这里边我创建一个接口，选择接口，因为我们这个接口仿的是帖子，所以我取名叫discus post reporter。

这样的接口一般都叫什么repository，那么这个接口因为是数据访问层的代码，—s可以被看成是一个特殊的数据库，所以我们需要在组件上加上 reporter这个注解，不是map，那个map是my business专有的注解，而reporter它是spring提供的，针对数据访问中的注解，然后我们这个接口不用声明任何方法，我们只需要继承于—s它就是只不过给我们提供的默认的接口就可以了，继承于us take search。然后的话repost继承给他就行。

我们继承的时候需要加上范型声明，好，你这个接口我们要处理的实体类是谁？是discourse post，另外还要声明实体类当中的组件是什么类型整形，这样就可以了。

那么负接口当中它已经事先定义好了，对es服务器访问的增删改查，各种方法我们这样声明完以后，加了注解以后，spring会自动给它做一个实现，我们直接去调就可以了。

好，那么到这我们就可以用了，那么我们用的话那要做一个测试，看看好不好用，我创建一个好，那么我在这儿新建一个测试类，这测试类我叫Eli stake设施test，那么它上面需要加上几个注解，我从其他的内容去copy，那么因为我们访问一来是社区，我们的数据来源其实是来源买circle，我们从my circle中把数据取到，然后再准存到 es里，而不是凭空捏造这个数据。

好，所以为了从数据库里取帖子，我把迪斯卡斯普斯的麦克注入进来，这样我还要充满c口里取这个数据discus boss的麦克这个变量，然后取得简短一点，对卡迪斯卡斯麦克。另外我们往 es服务器中存查数据，我们需要用到刚才的 report，所以也要把它注入进来，不可能 discounts poste repository，变量名儿我取得简短一点儿叫discus repository。

然后有些特殊情况，这个repost都是解决不了，我们还需要调用 template，所以我再把 template注入进来，elastic shirts，template，然后变量名不叫一来CK谈PK，简短一点。

好，那么在访问 es服务器之前，我想先打开 post曼，我们先看一下我那个服务器里它有没有帖子，相关的数据有没有索引？好，我打开 Post麦。

好，那么我要查一下9200，然后的话杠cat，然后这有一个现成的 index，我要看一下我现在这个服务器里有多少个索引，你看它目前只有一个索引test，就是上一个我所创建的目前并没有discourse post对吧？

那一会我们执行这个方法以后，看一看 disc pose的索引是不是被自动创建了，好知道此部分帮我们做了什么。好，我就回到刚才的程序里，我就开始写代码，首先我们要写的是什么呢？怎么去往—s服务器里加数据，添加数据，先把数据加进去。好，我写一个测试方法，别写错了。

叫test。音色插入数据，然后就调 Discard repository，那么调它的sale方法就可以插入数据，而这个数据这是插入一条，这个数据我从map里取到查到 disk us map点谁来把ID好，我得指定是用哪个ID呢？咱们看一下这个表，discard post ID。

我 order by iddesc倒叙一下，这数据还挺多的，随便找几个数据，比如说我就找这三条，241242243，我把这三条数据插进去，好，我这就写241，这是一条数据插进去，然后再来第二条，第三条这就行了，反正就这样就能插入一条数据。好我就执行一下，然后的话看一看这个数据能不能被插进去。当然我们在执行插入数据的，它底层会换成这个命令，在执行这样的命令的时候，它发现索引没有，它会自动创建，我们上次跟他用post曼去访 Es服务器，也发现了这样一个现象，对吧？它会自动的，所以你不用去特意创建首页。

好，我就运行一下。好，那么运行完以后，你看它解忧那张没有报错表示成功了。

然后的话。

当然了你运行的前提是你要注意一件事儿就是说你你来CK设置服务的启动，我这之前启动好了。

好，下面我们再看一下普斯特曼，我再执行一下命令，你看this time is post索引就存在就有了，然后比如说我想再详细查一下 discourse post，我新建一个页签就页签，就把改一下this pass，post search我要查所有的，然后包的设置为那不传包的数据了。

好，然后执行你看它命中三条数据，分别是242243241没问题，好，这是每次执行 CEO方法是插入一条数据，然后还有一个方法是能够插入多条数据，我们再演示一下，插入多条数据的方式，我叫test insert list。

好，那么还是调用 repository，然后点CF二 CF二里你要给它传入多点数据，传一个集合进来，我们也可以通过 map来查，discuss member，然后 select，discuss posts，然后我们查的时候，这里是需要根据优质ID来查，然后还分页了。

好这样我们就着方法多插入一些数据，把我们表里的大部分数据都插进去，相当于一个数据的初始化，这样一会我们好去怎么说一会我们好去做。

搜索得有这样的数据，我要插入哪些数据，它是根据user来插入，你看 uzid尤其是前面这个数据，101102103，然后的话。

12。

然后131132133134等等我们就插入，其实到这儿了，往下大部分数据就进去了，我就又在ID处理到134差不多了，其他的我就不挨个处理了。

好首先是101数据比较多，然后从0开始100结束，其实他没有100行，就这个人没有100行数据，所以说这个人的数据就都有了，101。然后再来就是102的103的，然后是111的112的，然后是131132133134。好了，我这样的话一执行，咱们表里大部分数据就都过去了，执行一下。好，执行完以后没报错，我们再通过 Post慢再看一下，我再搜一下。

好，那么搜完以后你看他一共命中了141条数据，然后这里他没有全列出来，其实他只列了第一页10条数据，但你看这个数据确实有了变化是吧？有了变化我们不用挨个细看，确实多了很多了，但是他没有列举全，他只列举了第一页10条数据就行了。

好，刚才我们所演示的是添加数据，可以添加一个可以添加多个，那么如果你想对数据修改也是可以的，再演示一下修改test update。

好，那么我想修改哪个数据，咱们一先找一条数据，比如说就找231数据，然后我们先看一下post吗里面231这个数据，这样我把 delete关掉。

兔子关掉先不用了，然后我要去看一下231的那条数据，local house9200，然后当前是discuss post。然后 DOC，然后是231执行，你看它这个数据是叫灌水，新人灌水，比如说我就要把这个内容稍微改一下，我们看一下它能不能变好。

那么首先我先把这怎么改，其实改我们还是调 seo方法，把之前的新增的数据再覆盖一遍，就是改这样我把231这个数据再重查一遍。

discourse post。

等于迪斯卡斯麦克点谁来？克白d231，我把它查出来它是什么新人灌水，然后我要改一下它的内容，post点、site，content改成叫我是新人，然后使劲灌水。

好稍微改了一下，改完以后我再通过discourse，repository、点、sale，C5一个去保存它，那么它就会把原来的数据覆盖掉，这就是修改还是C5方法。好，执行一下，看看执行完了没报错，我们再通过post曼查一下，数据变了，表示修改成功了，对吧？好，这是修改，删除的话也很简单，只要调 delete就行了。

再来一个方法。

test, delete,

那么如果是3亿条数据，就是discourse、repository、点，delete，白底它自带一个方法叫得力的白底，比如说我要三231执行一下看看，好执行完以后看boss的慢搜一下，数据没了被删了，然后也有方法能查删掉所有的数据，但这个一般我们用的少，因为这太危险了，一下删掉所有的数据风险比较高，对吧？

但是也介绍一下，也演示一下 discuss repository点，delete，就直接把所有里的数据就都给删了，好来执行一下，执行完以后我们再去post慢搜一下，设置一下，啥都没有。

好了，那么你看我们争三改啥就都演示完了，演示完以后接下来就应该演示搜索了，但是现在数据没了，没关系，我再执行一遍 insert，重新把这些数据插进去，然后这些数据就够我们去演示了。好，执行完这个数据就有了，咱们再确认一下，对吧？已经有了141条数据没问题。

好，下面我们再演示 es的最核心的功能就是搜索，这些只是说把数据存进去而已，搬运过去而已，最重要的是搜索的功能。

那么我再新加一个方法，方法名我叫test社区。利用什么搜索，利用 reporter进行搜索，就用接口进行搜索，那么搜索的时候，我们首先要构造搜索的条件，它有一个组件专门是构造搜索条件的，其实不只是条件，你搜完以后你要不要排序，你要不要分页？

并且搜索结果你要不要高亮显示，什么叫高亮显示？

比如说我要搜，我搜的时候我搜索条件是互联网，而你返回的一句话当中，你返回的这句话当中，比如说是这样一句话，这还是因特网，比如说你反馈是这句话，互联网求职暖春计划，那么你是不是应该把匹配的这个词给我变红是吧？得把它变红，那怎么变红？

其实不是说es能够把它变红，但是es能够把匹配到这个词前后加上一个标签，比如说这样的，他把互联网这三个字匹配到以后，它前后加一个标签，这个标签由你指定的，你想加什么标签都行，a b等等都行，比如说我加个em标签，互联网我写错了，有人说这就变红了吗？

可以的，因为你返回的结果当中这个关键词词条前后有这样一个特殊的标签，我们最终是要把文字放到网页上去显示，而我们网页上不是有css，样式文件里边我们可以写em它是红色的，它不就红了对吧？

所以说你来c测试还有能力把返回的结果当中匹配到的那些关键词给你点亮，做高亮显示，它实现的机制就是在这个词前后加上标签，那么你需要做的是文本显示到网页上的时候，你要给这个标签加一个样式就可以了，你注意好，所以说我们构造搜索条件要构造一系列的内容，一个是我们搜索这个关键词要存进去，要构造进去，然后排序的方式要构造进去，然后分页的方式也要构造进去，还有高亮显示怎么做，也要构造进去，把所有的搜索的要求构造进去以后，你调一下它的方法，他就给你返回你要的结果，这样的。

好，所以我们花的很多精力是构造查询的条件，那么我们利用的是这样一个接口。

叫社区科瑞。

这是spring给我们提供的一个组件。

设置carry等于那么它有一个实现类叫native native search carry，其实它的实现类是native search carry，但是它还给我们提供一个工具类叫native search cover builder，这个builder能够构建一个native search cover实现类，用它比较方便，所以我们用 build构造一个接口的实现类。

好，我new的 build怎么去构造也很简单，首先就调这个方法叫 with carry，然后这个方法还是返回对象build你在with carry。

好，可以无限的位置下去，好，这个位置块是构建搜索条件的，搜索条件我们另外一个对象去构造，整个搜索条件还需要另外一个对象去构造对象叫 carry builder。

然后我们比如说我想搜什么，假如说我输入了搜索条件叫互联网寒冬，我希望能够既从开头里搜，又从content里搜，就是我是一个多字段的搜索，是一个全文的搜索，这个时候得这样写，quiet builder点没有引入是吧？

稍等 carry写错了是carry builders，我说怎么没有 carry builders点点儿叫Martin max carry，你看Martin max carry我们上课写利用 post曼执行，搜索的时候不是用过 Martin max carry就是多个字段同时匹配对吧？

然后 carry你需要传入几个参数，一个是你的搜索的词条关键字，我是互联网寒冬，很显然他拆的时候会拆两次，一个是互联网，一个是寒冬。

好，然后我要搜的是哪些字段，一个是开头。

一个是content，就这样，好，搜索条件就这样，就构造好，好像是不是少了半边括号，我确认一下写错地方了。

好，当然你也可以写更多的字段，我们只需要写两个 title和content，然后写完以后你可以继续构造完搜索条件以后再去构造排序的条件，搜索完结果以后我按照什么来排序，那就位置 sort排序的条件我们用什么来构造是 sort builders，查询条件是块儿多的，这个是salt的，然后点儿feel。然后 feel的是out，我希望按照什么字段来排序呢？

我们看一下这个表，这样我希望按照tap status以及死杠来排序，因为我们最终按照这三者来排序的，type意味着它是否置顶，这个status意味着它是不是加精，死杠意味着这个帖子它的价值，我们优先按照这个置顶来排序，如果你置顶的话，它肯定是排到最前面对吧？其次是按加精，如果你没有置顶，但加精它比一般的帖子要排前面去，最后如果说你既没置顶也没加精，我们或者说你都置顶都加精了，我们再看 score，这样的是按照这三个字段依次来排序。

好，我们每个sat只能指定一个字段，首先我是优先按tap来排的，然后 tap。

点你。

是按tap是正序还是反序，我得order。

然后。

sort order点disc，我是倒叙，这样的话置顶的会被排到前面去，写法是固定的，把它记住，同理再来，我就再来一个copy一下，我刚才这个排序说错了，我是要按照tap10-，然后和时间来排，tap置顶肯定是优先把它排到前面去，然后这个状态其实状态加精会被核算成一个分数，它在分数里体现，所以其次按分数来排，你加精的话这个分是比较高的。

如果万一是一个帖子分分一样，我就再按这个时间来排，是这个意思，刚才我说错了。所以第二个是scar，好，scar的话，我也是按照倒叙把分高的排前面去。

好再来。第三个如果说你这两个帖子它分都一样，我就按照创建时间 create。

time来排，

然后也是倒序把最新的排到前面去。好，这是排序的规则，然后我们查这个数据不可能说一下，把所有数据都查到，万一匹配个几万条你都查出来也太乱了，对吧？

我们就查按分页来查询，虽然你还得构造分页的条件，那就是这点位置，然后分页 k宝，然后构造分页的条件，用配置request，点of，然后我们需要传两个条件，一个是你当前要显示你要查的是第几页，我写的0是第零页，其实就是第一页，它从0开始，注意第一个条件指的是第几页，第二个是这一页，你要显示最多几条数据，10条，然后再来点儿位置 high light，其实feels就可以了。

就是说你要指定哪些属性，哪些字段你要做高亮的显示，你要对哪些这个词进行高亮显示，你怎么高亮显示是反馈结果变红的处理方式。

Highlight feels其实因为你搜索的关键字可以拆分成多个词，你希望反馈结果里哪些个字段，哪些个词它是高亮的，这里可以指定多个字段，那么每一个字段我们是用这样的方式来指定 new还赖的 builder，我们利用 builder里面有一个叫field的对象，我们利用它来构造高亮显示的字段，首先是对抬头字段我需要做这样的处理。

抬头字段当中我匹配的那个词，我这高亮显示我不是说了前后加标签，标签你来指定 pre Tex前置标签，我希望是一m然后再点。

post。

后置标签，后面的标签，我希望是em结尾。好，这是一个。

再来一个，因为我们还有一个字段也可以匹配CTRL对吧？字段中如果有这个关键词被匹配到了，也是这样的，前置是em后置是em结尾one。好，那么当我把高亮显示字段也配好以后，最后别的就不用了，这些就够了，然后我们就调用 Build对象的一个点build的方法，点build的方法一执行，那么整个它就会返回 search carry对象的接口的时间内，那么就可以了。

好，我终于要把这个查询条件构造完了，虽然说第一次比较麻烦，以后就方便了，因为这个格式基本上就你可以复用，可以copy，在基础上改就快了好。有了以后我就查了，查的话就是分页查询怎么查呢？Kiss reporter。

点。

search你。

把这个search carry传进去，它就能帮你做分析查询，返回的是分页的数据，它的分页的数据利用一个配置对象进行封装，这个配置不是我们自己写的配置，是人家给我们提供的配置。

这个好，然后配置里面它直接存了，实体是多个实体 this cost，你可以把配置看成是一个集合，它里边封装了多个实体类是这样的，当前这一页的实体类都放到这里来。

好，那么通过这个配置对象我们得到很多数据，我把它显示出来给你看一下，首先配置点get total elements就是一共我们查到了多少条数据，一共有多少条数据匹配，好，再来 page点get total配置。按照我们当前的分页条件一共有多少页？

好，再来配置点get number。当前我们处在第几页再来配置点get Set，每一页最多显示几条数据。

好，最后再来最后就是我们要看配置里的数据了，因为它里面封装了多条数据，我们到循环遍历。

discus post post配置这个配置是可以直接遍历的，它应该是实现了某个接口看一下，它继承了你看它其实继承了艾特瑞宝接口，所以它是可以被便利的关掉回来。

所以我就便利了它便利的时候，每次遍历我把这个post打印出来，你注意这个post是我们从一s服务器里搜出来的，好了，我就来执行一下这个方法。

好，那么我们看一下这个结果，首先你看他返回说一共有111行数据是匹配的，然后一共是12页，对不对？对的，因为我们一共111条数据，每页显示10条，那可不就是12页，然后当前处于第零页，每页显示10条数据这都没问题，后面它是显示10行数据是10行吗？12这是第二行。3456789 10没问题，每一行都包含相关的关键字吗？你看互联网，互联网互联网这都开头有互联网可能就匹配。

这个的话开头里面它没有互联网，但是它的内容里有互联网，对吧？当然我们还有一个词是寒冬，它会拆分出寒冬这个词，是不是有一些帖子里有寒冬呢？

你看还真的寒冬，所以说有很多是这两次都匹配的，所以这两次都匹配，它会放到前面去，第一页你看都是尽可能都是两个词都匹配的，再往后可能有的内容只匹配一个词，es很聪明，他把两个字都匹配的优先排到前面去好了，但是对再看一下这个结果。



还差点事儿，但是有人说我刚才看寒冬也好，对吧？还是一个互联网也好，你不是说它会前后加一个em这种没加。这里我要说的这个问题就是怎么说 Reporter它的底层的默认实现类，他在查询结果的时候，他确实把es返回到带有标签的，高亮显示的内容已经返回了，但是他没有把这个内容合到这个结果里来。

就是es给我们反馈的结果里是包含一个原始的结果，就是你匹配到的结果。

还有是高单独它会把高亮显示的部分再给你，它是两份数据，你需要把这高亮数据整合到原始的数据里做一个替换才可以，但是它默认实现了底层没有做这样的事情，那么你跟踪源码的话能够看到这一点，具体来说源码是怎么回事，它的底层 reporter的底层，你跟你跟踪设置方法，它的底层是调用了us take，我先这样 us take time play的。

点carry for。

配置它底层是调了这个方法，然后一共是要三个参数这个方法，一个是search carry，然后还有一个是class，就是说你要查询的是哪个类型的数据，class要传进去。

第三个它要传这样一个类型的东西叫。

search。

without a map。当然我这是一个事例，我的助调我是想告诉你，一 repos它的底层是调了他们的方法去查询数据查到的数据，由这个map进行一个处理。

如果你要把这两两份数据组装在一起的话，你要用 map进行处理，但是他没处理，他们利用 map去处理。

这没办法。

所以有这么一个小问题。但我先解释一下它的底层获取到了高亮显示的值。

它。

的底层获取到了高亮显示的值，但是没有返回，他没有做进一步的处理好。

那么鉴于这种情况怎么办？你要么你去重写他的某些方法，那就很麻烦，要么我们干脆就不用设置方法了，我们就用report用 Template的方法，我们直接用这个方法就完了，对吧？我们直接利用这个方法就可以了，反正它底层也是调这个方法。

好，那么有人说你干脆就一开始就用，我想让你看一下还怎么说，这也是一种方案，只不过这种方案不完善，但它的这种方案我们能够通过它看到它底层的一个实现机制和缺陷，然后我们自己去解决它，这不挺好的是吧？

好，接下来我再写一个测试方法，然后我们利用time类的去做搜索，看一下效果怎么样。

test search by template。

那么我们利用template来查询它，也需要我们传入 search carry，这个search carry我们可以还是利用刚才的智能代码都不用动，一模一样，

好。

有了它以后，我要用他们做查询，怎么查其实也不难，当然它返回的也是，它返回结果也是配置也是它那配置等于analytic template。

然后第二发瑞所配置，这个方法需要我们传三个参，第一个设置carry，第二个是实体类型 discuss post点class。第三个是社区伪造的map，这是一个接口，我们可以自己实现这个接口，然后的话在接口里去处理这两部分结果的这么一个合并的问题。

我在这里做一个匿名的实现 new然后的话。

社区。

伪造的迈克尔好，他把方法的自动把我加上来了，总之这段代码执行完，就这段代码跨日方配置它会得到结果，它得到的结果它会自动的交给迈克去处理，迈克是得到这个参数，它可以利用这个参数去处理这个结果，就通过它能得到所有的结果，这样的，然后把结果做一个封装返回就可以了。好，我们得到配置以后后面怎么用，其实和刚才的用法是一样的，你智能代码也可以拷过来，这个去掉就怎么用是一样的，就中间的这段处理稍微它稍微麻烦了一点，好，这里怎么处理我们来写一下。

首先我们先通过response去获取变量名太长了，我把它改短点，首先我们通过瑞斯帕斯先取到他这次搜索命中的数据，那就是get配置得到了多个命中的数据，得到的是这样一个类型search。

His。

好，设置his它里面封装的是多条数据，我们得判断一下你到底有没有值。那就his第二get total his，这是返回命中的数据量，总数如果小于等于0，那就意味着你没查到数据，如果没查到数据我就直接结束了，Return now完了。

好，那么如果不小于等于0，如果是大于0的，就说明你命中的数据我就需要做处理，我怎么处理呢？

我最终要把这个数据怎么说封装到一个集合里，然后返回这里我声明一个集合，技术里装的是discourse，好，然后我需要遍历命中的数据，然后把数据找到自己做处理完以后放到集合里就可以了，我要遍历命中的数据，每次遍历我们会得到一个search hit。得到一个命中的数据。

好。

那么每得到一个命中的数据，我要把这个数据包装到实体类当中去返回。

好。

所以我实例化了，实体类我们就从命中的数据里，它命中的数据的形式，我们上着课通过post曼也看到了，它返回的不是Jason格式的数据，那么在这个对象里它是把Jason格式的数据增长为了map，所以我们从hit里能调用map得到每一个字段的值，我就取一下hit点get SARS as map。

就是返回map形式的数据，然后再点get ID，我要得到组件，然后把它测试确认一下，我先把它透视卷一下，得到这个字符串，我先得到字符串ID，当然了我们POS的存的是整数，你存的时候再转一下就好了，点site ID，然后的话inter value of ID这就可以了。

那么其他的任何类型的数据你都可以这样先to string一下，把它转成字符串，然后存的时候需要什么类型再转一下，这样就比较方便了。

好，我就不废话了，我下面把所有的都这样处理一下。User ID。get south as MAC。然后get然后user ID透视卷，因为他get得到的是一个object，透视镜一下，然后再转post sad，优质ID好，这是优质ID，然后开头这是原始的开头，并不是抬头高亮显示我们需要单独获取，用单独的方式获取，有人说你直接获取高亮显示的抬头不就完了吗？这还不行，因为什么？有可能当前返回的title里，它没有匹配的關鍵字，那个关键字在content里，这title没有匹配，content的匹配了，有这种可能对吧？

所以我们这样处理，我先把原始的title渠道设置到实体类当中去，原始的高端的渠道设到Pos里头去，然后我们再去看高亮显示的字段，有我们就覆盖，没有就算了，这样是比较科学的，好然后再来get抬头吐死病，然后post sat开头，这个就不用转了。

同理，空乘。

好，除了抗震的以外，还有斯特尔斯，然后是create time，我们需要利用一个date，然后我们需要这是一个这个字符串，注意这个字符串是它其实是浪类型的一个字符串，它不是年月日的形式，它是一个浪类型的，我们可以把它转成浪，然后的话再设置给date。

就是说—s在纯日期类型的时候，它是把它转成了一个浪整数类型的字符串来存的，注意。

好，然后还有评论的数量。好到这儿我们这些字段就都设置完了，接下来就处理高亮显示了，处理高亮显示。

的结果，那就是说我们看一下它有没有返回高亮显示的内容，如果有的话是抬头还是content还是两者都有，我们把这个title和content相应给它覆盖成高亮显示的内容就可以了。

好，怎么获得高亮显示的内容也很简单，hit。点get，highlight feels。然后点get。

抬头我。

想获取与抬头有关的高亮显示的内容，我们得到的是什么呢？得到的是一个还来的feel，得到的是这样一个类型，这是一个字段，它是对一个字段的封装，我取名叫抬头feel的。

好，然后。

判断一下他到底是真有价值还是说假有价值，如果它不等于空，假如说它确实是存在的没有问题，这个时候我就要抬头给它覆盖掉，我要从这个字段里把高亮显示的值取到抬头，feel的点get fragments。

Fragment返回的是一个数组，有人说为什么是个数组？你想一下，我们抬头中匹配的词条有可能是多个，你比如说我们搜的是互联网寒冬，有可能Title里既包含互联网又包含寒冬，甚至有可能包含了它两个互联网这个词，所以说对吧，所以匹配的可能是多个。

好，他如果匹配多段的话，我们这里设置这个title，我们要第几段，我们就要第一段就可以了，咱们不用每一段都都给它设置上去，只要第一段就可以了，所以我就get然后下标0取第一段，然后to spin。

好，这样就行了。

同理，那么content也是做一样的处理，我把这个拷贝一下，只不过这个叫。

content。

这个叫compt feild。

费用的判断，

如果它不为空，那么我们就site content。然后从counting的field里取第零个值，然后to，string这样就可以。

好，到这儿我们实体类这些数据就处理好了，处理好以后别忘了把它放到集合里，类似的要艾特poss。

好。

这还没完，最终我们需要返回一个结果，当然这个结果里面应该包含集合中的数据，返回的它不是直接list，它返回的结果里你看我们使用的时候既包含什么，一共有多少条数据，有多少页，当前页是几什么包含的数据比较复杂，是个聚合的数据，不只是一个集合，但是它包含这个集合，它返回的类型是什么？

是这个类型。

所以我们需要构造这个类型的，它是一个接口，我们要构造它的一个实现类，好了，那就构造我就在后面return这个类型，它的时效链是类型。

Apple我这里写了return。

你有Angel。

然后当然这个时间内我们需要传好多个参数，这个参数的顺序你可以去看repost的底层的源码去做一个参考，这里我就直接写了第一个传集合，第二个传方法的参数配置，apple把传进去，返回结果里。

好，然后再来我再换一行，下面的是his点get total his就一共多少条数据要传进去。然后是response点get要返回这个对象，具体说他到底要这个干什么我也不太清楚，反正他是这么写的，我也照着写了，具体没有细看。

然后是get scar ID，然后是get max4杠，总之它底层需要这些东西，我就把它把这些东西给他就完了，其他的就不用了。好了，终于我自己把这个数据处理好了，处理完以后咱们来试一下，好，我来执行一下这个方法。

好，你看这个结果还是111条数据，12页当前第零页每页10条没问题，然后后面是10条数据，然后每一条数据你看互联网前后是不是有这个标签的是吧？内容韩东强是不是有个标签的对吧？当然有人会发现说你这个内容不是完整的内容，完整的内容是很长的，你这怎么变短了？

因为就是说你的内容其实是一篇小的文章，可能有1000个字，而我与关键字匹配的可能有几部分，一部分两部分甚至多段，那么它返回的只会返回匹配的那一段前后有关的内容，它不会匹配所有的内容，你想你去百度搜东西不也是吗？你搜到的结果是一小段一个局部对吧？

你不是一整个一篇文章没必要，所以这一点 Bs也帮我们做了处理，它返回的结果里是匹配到的有关的那一段内容，而不是整个完整的数据，这是非常合理的。好了，那么这次课我们使用spring整合—s怎么去用？这些就演示完了，这些就够了。最主要的是最后的搜索，你要把它搞明白，尤其是如何构造搜索条件，如何去处理高亮的显示，至于说正常改数据就比较简单。好，那么这次课我们就演示到这里，咱们下次课再见。

## 6.3 开发社区搜索功能

### 42-开发社区搜索功能.mp4

这节课我们来开发搜索功能，那么在开发搜索功能的时候，我们正好就要用上我们刚刚所学的 e list 的社区，那么在开发这个功能的时候，我们大概要开发三个方面的内容，第一方面写服务层的逻辑，我们需要提供搜索相关的服务，或者说搜索相关的业务。

具体来说是这样的，当我们添加一个帖子的时候，发布一个帖子的时候，我们应该把新的帖子存到es服务器里，所以我们的搜索服务的组件里应该提供这样一个存储的。

方法。

再一个那么如果说我们删帖子，我们也应该从—s服务器把这个帖子删掉。当然了我们现在删帖的功能还没有实现，但是我们在开发搜索服务的时候，先把从es服务器删帖的方法先准备好，然后以后可以直接调用。

然后第三个就是重点就是我们要在组件里提供一个搜索的服务搜帖子好，这是第一方面，那么这个业务层我们处理完以后，接下来我们要处理表现层，首先我们需要解决这些帖子，什么时候增什么时候删，这样一个问题，我们采用事件的方式去处理。

发布帖子的时候，我们采用异步的方式把这个帖子提交到es服务器，然后还有增加评论的时候，那么增加评论以后，帖子的评论数量就会发生变化，这个时候我们也将帖子一步的提交到—s服务器，相当于这是一个修改了帖子。

当然了我们采用异步的方式，主要是为了提高性能。你这样当我们发了一个帖子以后，我只要把这个事件丢到消息队列里，我们就可以继续处理下一个类似的请求，对吧？就不用等待，所以说异步的它可以并行的去处理一些事情，这样比较好。

既然是异步的话，我们把在增发布帖子时，在增加评论时触发了这样的一个事件，我们需要在消费者组件里加一个方法来消费这个事件，这是我们需要基于事件来做的一些处理。

好，那么当我们把这个数据同步到了es服务器以后，剩下的就是查询了查询的时候，我们要想显示出搜索结果，我们需要在CTRL那里处理搜索请求，然后在对应的 Html里显示这个结果，这个就容易了。

好，总之我们整个功能的关键其实集中在前两个部分提供搜索的业务，然后对于事件要做出相应的处理就好了。

好，下面我们就开始写代码来实现搜索的功能。

首先我想先解决一个小的遗留的问题，当然目前这个问题还没有表现出来，但是如果说我们任由它摆在我们这个功能开发完以后就会表现出来，所以我提前把它先解决一下，这是之前没有注意到的一个小小的环节，主要是 map 的配置文件，discus, post map, 然后这里边我们不是写过增加帖子的方法，对吧？

这个方法你看我少了一个东西，少了一个 p 图片，那么如果你不声明 t party, my business 就不知道你这个表你增加的时候哪个是主键，然后就会有一个问题，他增加完以后它不会把生成的主键存到你实体类当中去，所以后面如果你实体类想要用到组件，没有，而我们后面处理这个事件恰恰是要用到 ID，所以说这里需要稍微处理一下，但我需要写上的 k 等于 ID 解决小的问题。

好了。

下面我们开始正式的去开发我们刚才所说的这些内容，首先我们要处理业务层，那么我在 service 目录下，我新建一个新建一个 service 组件，因为那是搜索，而且是基于伊莱斯 txt 的搜索，所以组件我取名叫伊莱斯特设计。

然后。

service, 那么我需要给它加上一个 service 注解，另外把我需要的用到的 b 注入进来，那么因为我们这是要往 es 服务器里存，这个帖子相关的数据我就可以用到上我们所写的 disk as post repos 的人。

咱们。

不是。

写完了吗？好。

我们就利用组件，然后除此以外，我们在搜索的时候，我们肯定是要将结果当中带上高亮显示的那些个字段的。上节课我们学 es 也知道，我们要想高亮显示我们还得用到 e list template 对吧？所以我把 Template 住进来，estate, search, template,

我把这个变量给它起的简短一点，也稍微短一点，不然太长。

好了，那么有了这两个病以后，我就可以去实现我刚才所说的那几个方法。首先我需要提供一个方法，能够向 es 服务器里提交新产生的帖子，那么这个方法公有的不需要返回值，方法名我叫 save, 然后 discuss post。好，那么当你 save 的时候，很显然你得把帖子的数据把实体传进来，怎么去增加？我们上节课在测试类里演示了非常简单是吧？就 discard repository 点 c 只有一个，好这就可以了，非常 easy, 然后还要提供删除，修改不用，因为你再 save 一次，就是修改，删除需要提供一下，然后将来好用删除。

我叫 delete this。

cast post 需要你传入 ID，那么我还是调 discourse, reporter, 然后点 delete by ID, 这就可以了。

好，最后我们需要提供一个搜索方法，public, 然后我们上节课学过的就是演示过搜索，那么我们搜索以后它能够给我返回一个配置类型的数据，然后里边封装了多条，这个帖子我就直接返回这个值配置。

但。

这个配置不是我们的配置，是 spring 他给我们提供了一个类型配置，然后里边封装的是帖子的实体，然后这个方法我就要 search this cast pose, 搜索帖子。

好，那么你搜索帖子的时候，你肯定是要传入你搜索的一个关键字，我叫 key word。另外因为我们搜索是支持分页的，你要传输分页条件，那么你要传入当前，你要显示第几页？

Carry的表示的是当前是第几页的意思，我们之前写的分页都是传一个offset，它是offset是左眼不是第几页 discourse reporter或者 time内的在分析查询的时候，它需要这样一个参数，所以说我们就变通一下。传的是carry的，然后还有每页显示多少条数据 Limit。

好，那么这个方法中的查询的逻辑，我们没有必要再从头开始写一遍，咱们直接打开我们上次课的测试方法，把代码拷过来，在此基础上稍微改一改就可以了，因为这个代码太长了，然后写一遍也不白写，那么那个测试类叫做。

elastic shirt tests。

好，那么就是最后一个方法里的逻辑，当然 system out输出不需要，我只需要把之前的拷过来，拷过来以后还得改一改，咱们从头往后理一理，首先是构造查询的对象，查询条件，这里边就不能把这个条件写死成寒冬了，互联网寒冬了，你应该是写这个参数，key word对吧？

然后我们查询我们搜索的字段依然是开头和counter，因为还是这个帖子数据，那帖子我们搜的就是title content，然后排序依然是优先按照tap排，如果tap一样再按照分数排，如果分数也一样，就按照时间排都是倒序不动。

另外我们要分页，这个分页条件不能写死，第一个条件是当前页carry，注意 carry从零开始，我们传的时候要注意这一点。

第二个是每页显示多少条数据，就是limit变成替换就好了。

最后整个高亮显示之前我也说了，前后加上一个特殊的标签，但这个标签由你来定，那么我们加em就可以，我们在搜索页面上已经其实不是搜索页面了，我们是整个个logo点css里边对 em标签做了定义，em标签显示为红色，所以说你加em它就会显示为红色就可以了。

好，这个条件就这样就行。

然后我们看查询的逻辑，那么返回的最终的数据，最终的数据，其实我们可以直接这样，我可以直接return，我可以直接return，然后瑞特 carry for配置。然后这个是传的条件，分别是e list，take search，然后的话类型以及伪造的map不动。然后再往下看，在这个方法的实现的过程中，我们首先得获取命中的数据对它做了判断，这都没问题。

然后我实例化的一个集合在集合之内我实例化的一个实体，然后根据命中的数据去构造实体，把它放到集合里，最终返回这里，因为我们访的还是帖子，所以说这些字段名字都不用动，整个逻辑就这样就可以了，高亮显示对于title和content也是一模一样的，处理方式也不用动，那么最终return的值也是这样去构造的，没有问题就这样了。

好就这样了，那么这个方法我们经过上次课的代码的改动以后很快就实现完了。好，那么当前我们业务层这个代码就已经搞定了，下面就是要处理。表现层刚才我们也说了，我们主要是要用事件的方式，用一部的方式去向es服务器当中同步数据，这样效率高。

我也说了我们是在两个地方去同步数据，目前来说发布帖子时还有增加评论时，删除帖子目前还没做，先不管，我就在这两个点去触发一个发帖事件就可以了，把这个帖子把这个事件加到队列里就可以了。

那么我们找到了这两个位置，首先发布帖子，我们可以找 Discourse很抽了。好。发个帖子不就是这个方法对吧？在这个方法的后面，当我把帖子发布成功以后，这个时候我应该触发发帖事件，然后把新发布的帖子存到s服务器里，对吧？应该是做这样的操作，那么我就实例先创建一个事件对象，You want。

等于6in mind。

然后点sat topic。

这个发帖事件我们还没有定义常量，我还得定义一下，好，我在这儿加一个主题发帖，topic tablets。Ok然后回过头来，所以我这个主题就是topic，public之后我们要设置 user，ID是谁触发的事件，那应该谁呢？当前的登录用户 user点get ID。好，然后还得设置安TT态度，实体类型。那么你这个事件是关于哪个实体的？

我们是关于帖子的安推 Tag post，还有在site int ID，实体的ID，实体ID就是帖子ID，post点 get ID，为什么刚才我要修正小的问题，因为你看郑家湾帖子，这里我不是要用帖子来ID对吧？如果不修正的话，这就会有一个问题，所以说修正一下之后这块就没问题了。当然了其实还有安推优的ID，但是我们发布帖子的事件里其实用不上这个数据，我就不去获取它了。

因为毕竟获取的还是有点麻烦，一还得做一步多一步，我不去取了用不上就算了。然后我要触发这个事件，我就调调 In one to producer，我需要把它注入进来之前还没有注入。

好，那么注入以后，在这个位置in mand producer，然后 fare in慢把你们的对象传入，这个事件就被触发了，完成以后还有一个地方也需要处理。我们发布评论的时候，当我们评论帖子的时候，那么评论帖子以后，我们会修改帖子的评论数量，那帖子就变了对吧？相当于改了帖子这个时候你还要触发一次这个事件，然后的话把帖子的 Es里的数据给它覆盖掉，其实是一个修改的行为。

好，那么发布评论我们去找comment controller第一个方法爱的comment就是当然我们就一个方法，我们在最后触发完评论事件以后，我们再去触发发帖事件，但这里你需要判断一下，因为我们这个评论可以评论给帖子，也可以评论给回复对吧？你只有评论给帖子的时候，我们才做这样的事情。

注意判断一下，如果说 comment点get on TT type，它是等于NTT type post，是帖子的话，我们再去触发发帖事件，好，因为这个变量前面被已经有了，所以我复用一下我就不重新声明了，event等于new event，然后 topic依然是public，然后 uzid看一下uziduzid其实就是一个host，就当前的登录用户，当然你从 comment里取也可以都行，这里我就comment要get有字ID，然后实体类型是post，没问题，然后是帖子ID，那帖子ID呢它参数这些数据设计完以后设置完以后，我们触发这个事件 very much，现在这个事件已经发布了触发了，接下来我们需要做的事情就是去消费这个事件，我需要找到这个事件的消费者，invent can summer，然后我在这里增加一个新的主题，主题的处理的方式和之前有区别，所以说我们就不要继续用这个方法了，我们再新写一个方法，这个方法主要的目的是消费发帖时间写清楚，然后加上注解卡夫卡雷森的topics。等于就是发帖的主题 topic，public这样就可以了。

他只消费这一个事件一个主题，那么方法的声明为public没有返回值，然后憨豆 public method他消费的是发帖的这个消息，然后的话参数是 ok，summa，we calm，好，当然了，我们首先也需要做出类似的判断，这个消息的是不是为空格式有没有问题，也是做出同样的判断，我call一下。

那么如果说我们得到的消息非空格式也没问题，我就要处理这个事件处理非常容易，我们只要怎么说，我们只要从这个事件的消息里得到帖子ID，查到对应的这个帖子，然后的话把它存到es服务器里就可以了。

这很容易，那么首先我要怎么说，我要先查询一下这个帖子，那查询帖子我们需要调用，我们从数据库里查，我们需要调 discourse的service，我看一下我当前有没有注入，还没有补充一下。

好。

然后我们查到这个数据以后是要存到es服务器里对吧？所以说我们需要访问 es你来take service service，所以也把它注入进来。好，那么有了这两个service以后，我们继续在这儿，首先我要查询出这个帖子 discuss，post，调用discuss post service，然后饭的办ID方法。

帖子ID我们从 EMAIL的世界里可以得到很容易 EMAIL点get on，tvid其他的就不用了，其实就用这一个值就可以了。

那么有了这个帖子数据以后，我就要往这个服务器里去存了，那就依赖txt service，然后点CF把pose的传入，好这样就行了。

那么这个事件消费咱们也处理完了，最后就是从展现，当我发一个帖子，这个帖子能够同步到es服务器里，我就能搜到它下面，我们就是做展现。

因为这是一个新的独立的功能，所以说我给它新建一个CTRL，我在 CTRL的目录下新建一个类，这个类我就叫社区，很丑了，好，那么加上很多的注解，那么我们要做查询的话，肯定是要用到e list, service先把它注入进来。

好，另外我们还要注入另外的两个service，一个是your service，一个是like service，为啥？因为我们收到了帖子以后，我们还要展现出这个帖子的作者，还要展现出帖子点赞的数量在页面上。

所以说你想我们是不是得用到这两个service去查相关的数据，就需要我就把它注入进来，一个是user service，再来再来一个是like service，这些service都注入以后，下面我就开始实现了搜索的逻辑，我先声明它的访问路径，pass等于search。

好，然后因为是查询数据，所以请求方式我设置为get，那么方法说明一下，返回实现方法名就叫设计好，那么这个方法就是需要传入一些参数，首先你要传入的是就是你搜索的关键字key word要接收一下。

好，另外你还要分页传输分页的条件，分页条件我用我们自己封装的配置对象来接收，最后我们查到的数据需要反馈给模板，我需要用的model。

然后这里我说一下 Key word，因为我们当前不是pose的请求，是get那么get请求 key word，你就不能用请求体来传，那get请求是要么你用路径后面带问号出来，你要么是用路径中的某一级来传，我用带问号的方式来传，这样的方式，我这个路径是叫设置对吧？

然后后面问号

带上 t word等于什么？大概是这个意思，用这种方式传 T word那么算法说明完以后，下面我就来实现了，实现就比较容易，首先我就上来就搜索帖子，那搜索帖子的话，我就调in last exercise service，然后他的设置方法设置的时候我需要传入 key word，另外要传入当前是第几页配置点get carry这就是当前页。

但是我封装的配置当前也是从一开始，是从它自然的一个顺序开始，但是我们这个测试方法里要求是从0开始，所以说你要减一做一个处理，最后每页显示多少条数据是配置的get limit这样。

好，我们搜索的结果也是一个配置对象和类名冲突了，我需要是配置，因为冲突了会所以它自动的会带上包名，以免产生歧义有点长，但是也没关系。

然后这个对象中它的风格的类型是实体类 discourse post，我们得到的就是一个搜索的结果，我叫 search without。

好，这个帖子我就搜索完了，搜索完以后别忘了，我这个帖子里得到的只是一个discuss post的实体，它里面包含了优质ID，我们需要我们还需要更多的user的消息，所以我们需要查 user。

另外帖子有多少点赞数量，我们也需要做查询，所以我们需要把这个数据再做一个处理，再聚合一下，我聚合数据我声明一个最终的聚合的结果是一个集合，然后的话里边分成一个map，那么这个变量名我叫discuss post，这是我最终要返回的结果，实例化一下，然后做一个判断，如果说 result。

它非空，

这个时候我们才去做进一步的处理，我需要便利，它便利的话，每次我会得到一个铁子 discuss post，好我遍历的是伪造的，好每次遍历我要实例化一个map封装聚合的数据，然后我先把这个帖子存进去写清楚，map点put post。

好把帖子存进去，其次我要存帖子的作者那就map点put user这个user我需要查 user service，点find user by ID这个u在ID我们从帖子里可以得到 post要get user ID，然后我们还需要得到点赞的数量，map点put，数量我叫like count也是查 like service，然后 find Entity like god那么实体类型是我没有实现接口，实体类型是帖子的类型，然后实体ID就是帖子ID，post点get ID这样就行了。



好，那么我得到了聚合数据map以后，我需要把它装到集合里，它好这个数据就聚合完了，聚合完以后，那么别忘了我们需要把得到的最终数据传给这个模板，model的AD，然后 discuss posts，把聚合后的数据传给页面。

另外当我们返回到这个页面的时候，我希望把刚才我所输入的关键字也把它就是设置进去，也带上这个关键字好，知道我刚才搜索的是啥，所以为了我们在页面上取关键字显示默认值方便，所以说我再把这个关键字也传给这个模板，t word。

好。

最后别忘了，我们还要设置一下分页的信息，好，能够实现分页我就配置点set pass。

那么这个路径是设置后面会带一个条件，t word别忘了，然后等于提问，然后还要设置一下一共有多少条数据，它好计算的总的页数 sat这个多少条数据，我们可以从这个数据里取，我们上课不是演示了吗？它里面封装了一共有多少条数据对吧？可以从里边取，我判断一下。如果说他万一是闹，我就认为是0条数据，如果他不是闹，我就从这里取get total。Alex。然后盖的偷偷爱你们家，他返回的是浪，但是我们赛特肉丝这里要求的是in，我们得强转一下。

好，那么设置完以后，最后那个模板在site目录下名为设置的sml好这样就行了，那么现在我就把Kantola处理完了，处理完以后，最后就是写这个html，html首先我们需要处理的是搜索框，我们看一下，我打开今天页面看一下先其实每个页面都有搜索框，所以我们可以先在index首页上去对搜索框搜索做处理，那么其他页面都复用head对吧？那么就都可以复用这个代码了。

所以我打开index点html我在这里处理搜索好，我找一下搜索的框的位置在这，然后首先我声明一下请求的方式，master等于注意是盖，因为我刚才说明的是get，然后提交的路径等于这个简单就是search，这就可以了。

然后我们这里是一个文本框里的值是要提交给后台的，所以我们需要给它加一个name，取个名字name。

Key word。

另外当我在某一个页面上输入了这个关键词以后，当我一点搜索又回到了这样一个页面，就打开了一个搜索的结果页面。

那么在搜索结果页面上，我要显示刚才我输入的条件，所以我要在这儿显示一个刚刚输入的key word，我就用。

value。

给他一个默认值，然后。

key word。

当然你首次访这个页面没有key word的时候，它显示为空不影响，但是你输入key word以后，那么这个就会有有一个值就可以了。

最后这是一个搜索按钮，type等于submit你一点，触发了提交数据这样就行了。

好，这个是我们搜索的条件的提交，最后还得打开search的html我们好显示搜索的结果，我再打开这个search点html，这个页面我们第一次处理，所以我需要从头开始对它做一个设置，先声明弹力，好，那么样式文件这里需要说明它的路径，然后黑的我们需要复用，好，那么复用完以后，中间的内容我们一会再说，这个内容先别一会再说了，这个内容其实就是一个帖子列表，对吧？

搜索结果列表还有多个li我们只要保留一个就可以了，我先把它删掉，不然碍事，好尾部不用处理，折叠起来，GS要处理一下。

好了，接下来我们就处理中间的核心区域的内容，帖子列表，很显然我们需要遍历刚才我们所搜索到的结果在li上加上th each，每次遍历我们会得到一个map，所以这个变量怎么样？我就叫map，我们遍历的集合叫做discus后4次遍历，每次遍历你看首先我们显示的是帖子的作者的头像，这个地方头像我们需要处理一下。

好，我从 map当中取 user，然后取优字的头像拍的you are，这就行了。再往后，那么这个地方这是标题，我们需要做动态的处理。好，那标题就是map，点post，取到 post，然后点开头就可以了。另外这个标题它应该有链接，能够链到帖子上去，这个链接我也处理一下。

好，那么这个链接链到应该是帖子详情页面上应该是路径中包含了一个帖子ID，所以包含一个动态的值，所以我要拼一下加上双出线。这个帖子详情页面的路径是diss pass，然后的话是detail，然后是帖子的ID，你就map点post点ID这样就行了。好，再往后这个地方显示的是帖子的内容，我就th You text map点。Post点，content，内容。

好，然后再往下看，这个地方是显示的是帖子的作者的名字，这也得改，那就是map点、user点，user name，作者的名字，然后发布于后面还得有一个帖子的发布时间太长了，我换个行。那么发布时间 Th 冒号text等于。那得格式化一下是吧？这意思？

点format。

然后是map点post点create time。然后给一个时间的格式。好，然后在这儿我们要显示两个数量，一个是赞的数量，一个是回复的数量。为了写这个代码方便，我在这个数量前后加上一个标签吧，这个是点赞的数量 Th text等于map点。

like。

com，然后是帖子的回复的数量，帖子实体当中直接就有这个数据。

Th模号text等于是map点post，comment，count。好，然后最后别忘了还有个分页是吧？分页的区域我们可以复用首页的分页的逻辑， Ph冒号，replace等于首页。然后名字叫配置内选。

好，那么到这儿那个页面就处理完了，现在所有的代码都写完了，写完以后咱们可以最终测试一下，那么启动一下服务，当然你在启动服务之前，你要确保你的卡夫卡还有 es是要去启动的，因为我们要访问这些服务器要用到它们对吧？我这里提前已经启动好了。

好，那么我的服务已经启动完了，启动完以后咱们测一下，我先访问一下首页，访问首页以后我先不去发帖，我先去搜索，看一下能不能搜索到目前现有的数据，我们目前的话一s服务器里是有数据的，因为我们上次做测试的时候，已经把大部分数据都搬进去了，对吧？

我们可以测一下，比如说我要搜的是互联网。

寒冬。

这样一个关键词 key word，然后点搜索搜到了一些结果，然后每一页显示的是10条数据，后面的回复数量通常大部分都没有回复，然后你看这个结果这个是标题有互联网，然后内容有寒冬有互联网，这是这一段匹配的文字，他给你列出来了，他不是列出整个所有的文字是第一段匹配内容，然后你看第二个标题没有匹配的词，但是结果里有就都有，以此类推就这样。

然后我们看下一页，当然我照的数据数据是挺多，但是其实数据大部分都一样，就这样好。一共有多少页？末页一共有12页，咱们上次测试的时候也是这样，对吧？好，我再来一个，比如说我搜的不是互联网寒冬，我搜因特网。

好像我记得这个数据里也有一些是因特网的词好搜索因特网。寒冬。寒冬是吧？带有因特网的可能比较少，带有寒冬的比较多一点，也有一些带因特网比较少，但你看到了我们这个数据是根据我的搜索的关键词会发生相应的变化，而且是匹配的，说明我搜索没有问题可以了。

但是我发一个新的帖子，我能不能立刻收到，我们也测一下，我需要登录一下。

还有问题。

信息能够然后123好我登录了，然后我需要发一个新的帖子，发一个什么就发一个跪求offer，然后就随便写说谁能给我一个offer，我给他张晓丽随便瞎写了，好写完以后发布，发布以后我能够看到我所发布的帖子，这里能看到，没有问题点个赞，然后回复一个 Hello。自己回复自己的，然后好让他有个数量，然后做了一个操作以后，咱们经过了几秒钟应该可以搜到了，因为我们只有这一条一个实践扔到队列里，他立马应该就处理了，很快就能够把数据同步到 es服务器里，我们试一下，我这回搜什么搜offer，还有显示的怎么这么恶心，搜到了是吧？

跪求offer，但头像显得太大了，别的倒是ok的就太大。我们之前的旧数据里也有什么offer，但被搜到了，因为它时间比较早，赞一恢复一，为什么这么大？我们看一下页面头像，其他的标签其实这里边写上了，在这里用内嵌样式写成了宽和高都是50，但这个没写的话，所以他有的时候就就会有问题，这里写一下style，你可以参考别的地方的盈利者，就头像他都写了，就这个地方忘了，style等于。

y然后50像素，

宽50高50像素，完了就写这么一句话，样式说明宽高可以了。

好完了以后可能f9我重新编译，然后回到这个界面我再重搜一遍搜，这回就好了。

50 50好，你看我们刚刚发的数据立刻就搜到，这是一个能够实时的搜索到我最新的结果，这样一个工具非常的好用，当然也是十几页数据，因为很多都带 offer好了，那么这次课我们就利用 e list，结合着之前我们所学的消息队列，然后把搜索功能实现了这个功能，它包含了我们之前刚刚学过的两部分新的内容，那么大家做的时候一定要就多去把这两个内容多加练习，把它练熟了。好，这节课我们就演示到这里，咱们下节课再见。

## 第7章 项目进阶，构建安全高效的企业服务

### 7.1 Spring Security

43-Spring Security.mp4

到目前为止。

我们已经把牛客社区的主要的功能都开发完了。那么这一章我们主要是对牛客社区的项目进行一些补充，进行一些完善。那么我们完善的内容主要是集中在两个方面，一个是我们要提高系统的安全性，再一个是我们要进一步的提高我们系统的性能，主要是集中在这两个方面，这两个方面也是你去面试找工作的时候面试官非常关注的。

好，那么我们第一节就是要解决安全性的问题，那么我要学的是spring surprising，那么security就是安全的意思， spring security就是spring为安全提供的这么一个模块。

好，我们先来了解一下 space quality，那么它是一个专注于 Java应用程序，提供身份认证和授权的框架。然后它的强大之处在于它可以轻松地扩展，以满足自定义的需求，就是说怎么说这个框架本身功能很强大，其次它又很方便与扩展又很灵活，所以说。

它。

能够满足我们应用的方方面面的安全性的需要。

甚至来说如果说你用spring cloud去做微服务开发的话，那么surprise在微服务当中也能够发挥很强大的作用，总之它能够提供一个非常全面和完善的这么一个解决方案。那么它的特点有这么几点，第一个它能够对用户身份的认证和授权，提供全面的可扩展的支持。

你注意我们对整个系统提供安全性的保障，其实主要的一方面就是管理系统的用户的权限，而管理用户权限其实主要就是集中在两方面，不管是什么框架，都是从这两方面入手，一个就是认证，一个就是授权。

那么所谓的认证就是判断用户你有没有登录，如果你没有登录，你根本就不可能让你去访问，像私信发帖这样的功能，这是第一个层面。

第二个层面是授权，那是认证以后要进一步判断你有没有访问功能的权限。你比如说我们将来要做一个功能，给帖子置顶加精，甚至是删除，那不是说你每个用户登录以后都能做这样的事情，你需要有管理员或者是版主这样的高级别的权限，你才能做这样的事情。

第二个环节授权是判断用户有没有访问当前功能的权限。好，那么 Spring security从这两方面它都能够提供很好的支持，除此以外它还能够防止各种攻击，比如说固定攻击，点击劫持，什么csrf等等，所以说能够保证我们系统避免被攻击。

第三个它能够和外部开发相关的技术很方便的集成，比如说server类的API其实就是Java ee或者是spring mvc当然其他的外部开发技术其实也可以集成，当然了我们的课程用的是 spring mvc，那么因为它们都是 spring全家桶范围内的技术，所以说它们之间可以无缝的进行衔接。

然后 spring security它的手册你可以从官网上能够查阅的到，然后这里我就不在课上给大家去查阅了。关于 spring security，那么我们在初学之初需要大致的了解一下它的底层是靠什么样的一个思路去管理系统的权限，你要大致做一个了解，然后我们才好展开了后面的学习。当然了你在这个基础上你说我大致也了解了，然后后面的案例我也搞明白了，我会用了。那么你在这个基础上你再去阅读一些技术资料，然后把它搞得更细，甚至你看一些源码，跟踪一些源码，把它搞得更细，那就更好了。

好，那么我在课上给大家演示一下，讲解一下 spring security它底层的一个大概的一个机制。这我需要画一个图给你做一个首先我先不说色块的，我们先说 surprise mvc，大家都知道spring mvc它的核心是什么，是不是叫做dispatcher，salade，这个是spring mvc的核心组件，那么所有的请求都是要提交给他进行统一的处理，然后他会把请求分发给一个的控制器，然后具体用某一个控制器处理这个请求。

我们一个应用当中只有一个SUV类的，但是却可以有多个控制器。我们先理一下我们当前CMV当中的这些个核心组件他们之间的一个关系，那么很显然 dispatch和 Ctrl之间是一个一对多的关系，这里我花三个表示多个就这样一个意思，把请求可以分发给给他。

那么在请求分发给CTRL的过程当中，那么还有一个组件能够起到作用，就是拦截器intercept，那么拦截器可以拦截访问CTRL的请求 intercept，好，然后拦截器也可以有多个也可以自定义，所以 despite和 intercept和control它们之间的关系就比较清楚了。

那么 dispatch分发请求在分发的过程中可以被拦截器拦截到最终是把请求分发给CTRL处理。

好，那么整个这些组件构成了spring mvc的一个底层的核心，我把它框起来，这些内容是spring mvc的内容，我来标注一下。好，这是spring mvc有关的内容，然后你要注意 Controller和 intercept这完全是使用mvc这个框架，它所封装的它所提出的一种理念，这是super mvc的特有的组件，但是 dispatch servlet你注意它是组件是满足 Java ee的规范，Jarvi的规范中有关于serve的要求。

那么dispatcher servlet是一个具体的 servlet接口的实现类，就是说Java ee它提供了提出了规范是一个serverlet接口，那么dispatch solve的满足规范，它实现了接口，所以它其实一个它是死人mvc的核心，再一个它是满足Java ee的规范的。

我。

把它变成蓝色，就表示它具有一个一定的特殊性，然后 Jarvi的规范不只是 caelate，还有一个组件也是 Jarvi的标准，或者说也是一个比较核心的组件叫filter过滤器。

那么过滤器它和dispatch之间的关系，就好比是拦截器和CTRL之间的关系拦截器，可以拦截对comptroller的访问，那么philter可以拦截对servlet的方法，但是他 philter和spring mvc其实没有直接关系，她是Jarvi的标准，那么同就跟这个一样 filter也可以有多个每一个filter可以解决一个问题，一方面的的问题。

所以实际上我们在开发的时候，无论你是用surprise还是用查IE，那么你其实都可以利用 filter对dispatch进行对 service进行拦截。

好了。

那么。

注意蓝色的部分是Java意义的规范，我们先强调一下把它标注一下。这个我把它标成蓝色，通过颜色你要知道不同的组件它归属于哪一个规范，然后 spring security它在底层要统一的利用统一的机制去处理系统的权限，它是怎么做的？

其实它是利用了 filter，它是利用了 Java e 的规范 filter 组件，而且它的底层有很多个 filter，具体来说其实应该。

大概是11个filter，比如说有1个filter专门用来验证账号密码对不对？是做这个认证的操作的。

然后有一个菲斯特只专门做退出操作的，有一个filter是专门把你认证的，结果你的权限加以存储的，然后还有的filter是做一些其他的事情，总之每一个filter做一件事，它的底层都是通过 Filter对整个请求加以拦截，然后实现权限管理。

好了，所以说 surprise它对。权限的控制，它这个是比较靠前的，如果说你没有权限或者你没有登录，你压根都访问不了dispatcher servlet，更别说CTRL了。

所以说它。

判断的时机特别靠前，比拦截器比肯德基都靠前，大家要了解这一点。

然后后面当然有同学说我没有具体去接触过 filter，我不知道怎么写，后面我们会有这个代码给你演示说我们怎么写一个filter，我们怎么自定义一个filter，我们解决我们自己特定的问题，这也是可以的。我们会有演示。再一个如果你想深入的去研究 security，那么你就得多去研究它的这些个标准，有人说我有必要去研究吗？其实我觉得还是蛮有必要的，因为什么？Spring security可以这么说，它是spring所有模块当中最复杂的一个模块。

当然这个复杂不是体现在应用上，我们在使用的时候其实很简单的并不复杂，但是我们在学习的时候，它的底层的机制，这11个filter你是很难一下子把它都吸收的，概念一下子太多了，你记不住，而且它们之间的调用关系对吧？

也是错综复杂，你一下可能不是那么容易能够把它吃得透，所以如果说你有精力把SQL研究透了，其实但你把整个spring其实也就很容易就能搞透，为什么？

因为square的是管理整个系统的，它和其他的组件之间是有一些关系的，他是管理其他的组件的，相当于你把它搞明白了，其他的组件也能够加深你对他们的理解。所以说怎么说你课后自己再多去研究一下色块的，我觉得是一件很好的事情，对你就业的话应该是有帮助的。好，这是它大概的原理。

有同学说如果说我想深入研究怎么办？你就是买书的话，还没有说专门一本书专门去写SQL的，有的话也是一些英文的中文的几乎没有，然后的话你去看 Spring的官网，当然它有解释，但是官网的解释一个是它是英文的，如果你英文水平不好的话，看着也费劲。

第二个官网上只是就着这个技术进行一个必要的解释，他没有一个系统的案例去穿插的讲解，说这个东西到底应该是怎么用，然后有些源码是怎么回事，他它是点到为止的，所以说如果我想研究的更透一点，对源码有一个更必要的了解，对它应用有一个加深的了解，我从哪去看这个内容我给大家推荐一个网站，这个网站叫spring for all其他的话其实倒无所谓，主要是 spring secrete，因为它比较难，所以说你去找相关的技术文章，这个质量是参差不齐的，我去找过很多这样的文章，最后我发现就这个网站 spring色块的模块，它提供了一些文章，质量是非常高的，所以你可以去阅读这些文章，有入门级的，然后的话还有源码级的，你把这些好好读一读，然后自己。去一跟源码，不用每一个都跟那几个核心的 feel你好好看一看，那么整个死人死快的，你对它的理解就到位了。

这么一个难的模块你能把它吃透了，那么其他的就好办了，面试的时候你也就有了一定的心理的优势好了。

那么 spring security它的大致的原理以及你参考的一些文章，我就介绍到这儿。下面主要是给大家演示示例，给你看一下这个技术我们怎么去应用。然后这个事例我这样，我并不是直接在我们之前的牛客社区项目上直接去引入 Security，为啥你直接一引入，他立马就会产生效果，他立刻就会对整个系统进行权限的控制，他立刻就会接管你这个登录和退出，然后的话你搞不好搞乱了。

这个连我们。

正常的功能都访问不了了，因为被他拦截了。

所以说这个比较怎么说。

他管理的范围比较大，涉及的面比较广，咱们还是要慎重。再一个你看我们现在的这个项目，它的代码越来越多了，代码实际上已经很多了，然后我们这个代码本来就多，我们再引入一个比较难的东西，难于学习的东西引入进来以后，哪些是我们新学的内容，新的代码，哪些是重点，它淹没在了这些代码当中去，你你抓不住重点了，就更乱套了。

所以为了你更容易学习技术怎么办？那么我给你准备了一个demo级别的项目，当然这个项目也是对我们游客社区的一个简化到不能再简的程度，只保留了几个必要的功能，而且是简单的一个模拟实现，我让这个代码尽可能量的少，然后这个代码现在还没有加入 Secret，然后我给你看一下这个小demo是怎么回事，有什么功能，然后我在简单的代码基础上给你演示secret，你一目了然就知道哪一块是我们新学的内容，哪块是重点，这样的话便于你去理解 secret。

然后我们学完这个色块的基本的应用以后，把它消化了以后，我们再将它引入到我们项目中去，解决我们项目中的问题，而解决的时候，我们可能会有一些变通，有些地方我们可能就省略了，有些地方甚至我们就把它忽略掉了，也有这样的可能。

好了，首先给大家看一下我这个简化以后的demo，当然你现在还没有这个demo，你别着急，这次课课后的资料里就会传 demo，那么你在课后练习的时候，你就引入我 demo，在这个基础上去开发色块的案例就可以了。

这个demo非常简单，你看首先这个数据访问层我只保留了一个user map，里边就一个方法select by name，然后有对应的 use map，xml文件，这就是我们之前项目的一个简化，对吧？没什么好说的，一看就懂就是dao层数据访问层，然后业务层我只保留了一个user service，里边就一个方法放在user白内容。

当然我为什么要保留这样一个方法，因为它跟这个登录有关系，你登录的时候不是要调这样的方法对吧？

而我们权限管理要管理的一个方面就是认证判断你登没登陆要用到这个东西，这是you do service，另外表现层我就保留了一个controller，我是把所有请求都放到这一个controller里，你看有什么访问首页，index访问，帖子 discuss访问，私信lighter，还有一个访问管理员的专属的功能to me。

当然还有一个我们打开登录页面的功能 log in配置，就这么几个方法。

然后我们看一下这些方法对应的页面，你看这个是首页，这首页我也精简了，就保留了一个标题，然后几个链接能链到其他的网页上去就完了。然后除了首页以外还有帖子页面，我就一个标题，你知道它是帖子页面就可以了，都不需要显示内容。

然后私信也是一句话，然后登录当然登录有个表单了，但是这个登录我们这个表单提交到哪去，这个还没实现，我们是用security来接管，请求用security来实现这个认证，所以我还没有写。

然后还有一个ADMIN，管理员专属页面也是啥都没有，反正就是一个标志完了。好，这就是我整个项目的全部非常的精简，一看就能看得懂。然后当然还有错误页面，404 500我们后面也会有用，好就这样了。

接下来我这样我给大家跑一下，目前的配好了能够直接运行起来，好我启动一下服务，启动以后，那么我来访问一下首页也是community index，首页就这么几个链接，能够练到帖子详情页面，回去能够练到信息列表，回去能够练到登录页面填数据，然后能够一退出就退到了登录页面。

除此以外还有一个隐含的路径，这里面没有列出来，叫阿德妹能够访问管理员专属页面，就这样。

然后现在这个系统是有残缺的，你看我没有登录，我居然能访问帖子详情页面，我居然能够这个访帖的详情页当然可以没登录也可以访问，但是我居然能访问信息列表，这也是不应该的，对吧？你得登录以后才能访问。

再一个我还可以访问，好的命这个是不应该的，即便是我用普通用户登录了，我也不应该访问我的命，我必须得用管理员身份登录才能访问到它对吧？应该是这样的一个逻辑，目前我们整个权限都没有做管理，登录的认证也没有去处理。

那么下面我就用security来解决这方面的问题，给你演示一下，那么我们使用SQL的第一步，肯定是要倒包的对吧？我去搜一下这个包，我打已经打开了，这个没问题，抛弃的人，我搜包spring。Boot。Secret就是然后点进去选一个版本，然后 Copy。好，那么我把它粘贴到后面的xm二当中去，然后这样我把这个版本去掉，因为负泡沫中已经声明了版本，它声明的是哪一个我就用哪一个。这个得稍等，他需要下载一下，稍等。

好看底下滚动条走完了，说明下载完了好。

那么下载完以后，就是说我们只要做这一步，我们只要导入 surprise包，它立刻就会产生一个效果，它立刻就会对我们整个系统做一个安全的保障，做一个权限的控制好。

我们可以试一下，我把这个项目重启一下，启动完以后我再打开首页，然后刷新一下。

你看一刷新 secret，一看你没登录，他立刻就给你一个登录页面，这登录页面是它自带的，当然我们也可以用自己自己的登录页面去替换它，但它默认我们还没有写代码，所以他就是用他自己的有人说用账号密码是什么，他默认在内存里给你生成了一个账号密码，你看这个日志里就有这么一句话说生成了一个安全的密码，这么长的一个字符串，账号就叫user，我们可以利用这个账号密码登录一下，看看 user 密码套进来登录，然后你就可以访问这些功能了，他这个账号权限很高，什么都能访问，就这样。

那就是说我们只要一引入secret包，立刻权限控制就生效。

当然了，这只是说明他管理的面比较宽，但是我们不能这样用，我们不可能说用他给我们生成了一个随机的账号密码，随机的密码，这也太复杂了，这样一个也不可能所有人都用这么同一个密码，这个只是说只是说它的一个入门级别的一个事例，我们怎么去把它的登录页面换成我们自己的，我们怎么去利用我们自己数据库里的账号密码去做登录，然后去做认证的，接下来就给你做一个更详尽的演示，我先把这个服务关了。

好，那么我们开发的时候，首先我们是开发业务层，我们想做认证，我们想做授权，那么我们需要在业务层做一些处理，或者说业务层或者以及数据访问层，首先我要处理的是 user 实体类，为什么要处理它就是说我们在做授权的时候，当前的user具备哪些权限，你怎么体现？

其实。

每每应用它是一个权限的体现不一样，稍微复杂一点的可能是会有一个单独的这么一个角色表以及权限表。

User关联了角色，再关联到权限，他们之间是一个多对多的关系，一个用户有多个角色，一个角色有多个权限，经过一系列的关联用户就有多个权限可以这样去设计。

当然你学数据库的时候可能也学过这方面的知识，然后我们应用权限设计比简单一点，一般互联网应用的权限可能就没有那么复杂，我们的权限非常简单，主要体现在 user 的type字段上。Tab字段是啥意思？

咱们再回顾一下，我们看一下优则表，建表的脚本，tab你看零代表普通用户就具备普通用户权限，这就是权限的意思，一代表的是超级管理员，二代表的是版主。我们tap这个字段就表示说当前用户具备哪些权限了，当然我们这个比较简单，就一个用户只能有一类权限，比较简单。好，但是我们将来在用security做授权的时候，那么我们要的不是这个态度，我们要的是一个字符串，这个字符串能够明确表达你的权限的含义，所以我们需要这个定义出来这样的字符串。

好，那怎么去做？我们通常是这样做的，当然这个做法不是说唯一的，我们通常是这样的，我通常是让User实体类实现一个接口叫做user details，这是secret一个标准的方式，实现user对tos接口，那么这个接口中规定了一些方法，我们需要去进行实现，我在后面去写。

好，那么第一个是这个方法，account，net x zero Excel的这个方法的意思是什么？就是说如果你返回处，返回处的意思是表示说账号没有过期，你返回false就账号过期了，这个意思这里我们要返回数，账号不过期就是我们认证的账号默认不做过期的处理好。

然后第二个还要处理Is the count not。那么这个我们通常也返回输入的意思指的是账号为锁定。好，然后还有一个方法叫做is credit sales。Excel。那么这个方法我们通常也返回处的，它的意思是凭证未过期，凭证就是我们认证的一个结果，你登录成功的一个结果。

好，然后还有一个是is enable，通常也是返回出，反馈数它的意思是账号可用，如果你返回force你就不可用。总之我们在实现接口的时候，基本上这4个方法都把它设置为数就可以了。好，当然了可能有一些极端的情况，你需要再把它设置为force，我们这里都没有提供赛的方法，不需要我们就都是处然后还有一个更为关键的方法需要去实现叫get authorities Authority就是权限的意思。

那就是说这个用户他所具备的权限你要返回回去，然后因为一个用户可能有多个权限，所以返回的是一个集合，然后grant的office是一个一个父类，那么我们集合中装可以装父类的，这是一个接口去装它一个实现类就行了。当然我们我们这个权限比较简单，我们是一个用户只有一个权限，所以说即便是一个集合，我们这里也是只存一个数据。

我就这样我就先声明一个集合 granted Australia release最终我就返回集合，集合里我只装一个权限，每个用户只有一个权限，我们这里ID然后我要实现接口，new做一个默认的实现grant authority实现它，要求它里面带的一个方法叫get authority。

那么每一个grant authority里边，通过这个方法封装一个权限，多个grand or city就是代表多个权限，我们这里只加一个唯一的权限，这个权限刚才我们也看表了，我们是通过tap去判断的，你不同的tap应该返回不同的权限的，就权限的名字，所以我们需要对态的进行判断，我这样判断，因为它是一个整数，我switch态度好，然后case如果是——代表的是超级管理员，当然了我们一起有其实有三个，一个是零普通，一个是管理员，一个是2版主。

我们现在先不考虑那么多，我们现在只管两个，假设我们这个系统里只有两个，一个是0普通的，一个是管理员TC，管理员我就委托一个字符串，咱们。表示管理员的意思，好，然后default，否则我就return普通用户，user好就完了，这个return now不需要，因为我这里已经返回了这个数据，现在这个接口就实现了，那么如果说我现在得到了user对象，我调用这个方法就能知道user对象它是奥特曼还是普通的用户，我就知道他有什么权限，我就可以给他去做授权相关的操作了，所以这是一个基本的准备工作。

准备好了，准备好以后，然后下一步我们是还要做一件事，也是一个准备，我们需要在这个user service里让user service实现一个接口，通常是这么做，如果你不让他实现，让别的组件实现也未尝不可。

但通常都是这样的优点，实现优点对TS service，那么这个接口只有一个方法，需要我们实现添加，它需要我们实现的方法，这个方法叫漏的user by user name，它的含义其实和这个一样，就这个接口要求我们提供一个方法是根据用户名查用户这个方法的实现，其实我们完全可以调find use by name对吧？

所以说我这里就直接return历史点，find user by name，然后当然它这个参数s应该就是user name，我稍微改一下，直观一点就可以了。



好，那么这个接口是 Siri 的底层的一个接口，那么 Siri 在做认证的时候，在检查你是否登录，检查你的登录的情况的时候，它需要用到这个接口，因为你想我们自己去实现登录的时候，你不也是要用这样的方法吗？Find my name 对吧？我根据你传入的 your name 去查一下，如果查到了数据，这说明内部对了，我再去判断是吧，就是这个逻辑，因此色块的底层它能够自动帮我们判断账号密码对不对？

它需要依赖于接口，所以说它需要我们提供这样一个接口的实现，我们就这样就实现了。

之所以在 youth service 上实现这个接口，是因为因为这 service 里有这么一个方法，只不过名字不一样对吧？我们稍微调一下就 ok 了，这样方便最方便好了。那么这两个就是 User 和 user service 这两个接口都实现完以后，那么我们基本的准备工作就已经准备好了，准备好以后，下面我们就需要利用色块条对我们整个系统进行认证以及授权。

怎么做色块的，它并不需要我们去什么帖子私信那些个组件上去挨个处理，其实它底层也是基于它底下是基于 filter 的去拦截，能拦截很多个大量的请求。

所以说。

我们只需要写在一个类当中，就能够解决所有的需求。

这个也就是 surprise 快的配置类，在这里去做相关的一个配置相关的处理。

下面我就写来写色块的配置类，那么我在肯包下新建一个类，我取名叫 secret，当然因为它是一个配置类，所以我需要加上注解，configuration。

另外 SQL 的配置类有一个要求，继承于 web security can figure adapt，我们继承于父类，然后重写他的一些方法，就能够解决一些问题，这样就可以。那么在这个类当中，首先我要注入的是 user service 组件，为啥？因为组件实现了 user detail service 接口，而这个是这块底层要依赖的，所以我把组件注入进来。

好，然后父类我们在配的时候通常要重写它的三个方法，这三个方法都叫肯德基参数不一样，然后我先重写，它的参数是 web security，Web security 重写的时候，我们这里这个方法比较简单，一般都写一句话，on the web ignoring，然后第二 and matches，然后里边写上什么，我们通常这样写。

那么这个 ignore 是忽略的意思，然后这里是路径，风格的路径，你看我忽略的是什么？路径下的资源是属于 SASS 下，星星表示所有的资源，我们知道 resource 下我们放的是什么，它不是 Java，它是一些静态资源，其实主要的是 static，还有主要是 static，主要是图片、GS、文件 css 那么这些资源我们在我们其实也可以再直接写一个 HTTP 的请求去访问他，对于这样的请求我们需要这块的对它进行拦截吗？

需要用 filter。对。那些请求做拦截，当然不需要，因为静态资源对吧，没有什么可保密的，随便访问随时访问，所以把它忽略掉。这个意思是为了提高性能，这里写的注释就是忽略静态资源的访问，那么这个方法主要是干这个事，其他的基本上就不用动了。好，这是第一个我们需要重写的地方。好。第二个还是要重写另外的一个肯飞的方法，然后这个方法是带这个参数叫奥森 tk manager，builder 比较长，我们重写这个方法。

好，我直接就重写这个方法的逻辑了，这个方法其实是做什么？这个方法内部。

主要是做认证，咱们权限管理主两两个方面，一个是认证，一个是授权，这个方法主要是对认证进行一个处理。

那么我在写这个方法之前，介绍一下这个方法所涉及的一些核心的组件，首先有一个组件叫做奥森 tk manager，有这么一个组件叫奥森 tk 是 manager，这个组件它是一个接口，它是认证的核心接口，是认证的核心接口，我们这个参数叫奥森 tk 是 manager，builder，这个 builder 是一个工具类用来构造接口的实例的。

好，那么这里写一条，builder 它是用于构建 manager 接口，构建 LCD 跟 manager 对象的工具。好，这里为什么要写？因为一会你就知道了，它的底层用到的组件比较多，很容易就混淆了，所以写清楚注释便于你课后的理解。

好，另外因为它是一个结果，它肯定有实现类，那么它比较常见的或者说默认的时间内是谁，你得知道叫做 provider。Manager这个类它是奥森tk是manager，接口的默认实现类默认会用它的，就是说标点默认构造的是这样一个类型的数据。

好，那么我们在这个方法内部，我们怎么去做这个认证？其实只要进行配置就可以默认不用我们自己去写认证的逻辑，它会帮我们实现，但我们需要提供一些配置。

好，那么我们需要这样写，这叫内置的认证规则，就是我们调查默认的认证规则，这样去做。Os调接口调标点，然后点 user detail service，然后把user service传输进来，为什么要这样做？因为它底层帮你去做认证，它需要得跳这个接口，他才能够根据账号查出优质这个数据才能判断你登录的对不对，对吧？它底层需要依赖这个接口才能做查询。

另外我们通常还会这样写点pass word，encoder，然后里面实例传一个传一个pass word，include的实例就是对密码进行编码的一个组件。

对密码接口有好多个默认实现类，比如说我用 New叫Pb kd，然后 to pass word，include它里边需要传一个，我随便写12345这个就是一个加密工具，这个是盐，那么它会把传入的密码带上盐之后进行一个加密，你看我们去登录的时候不也是这样做吗？我们需要对密码进行加密吗？对吧？所以说只要提供了 detail service，提供了加密工具，那么它底层就能够自动帮我们去做一个登录的认证。

当然了默认的规则跟我们当前的优则表，我们当前的数据的形态不匹配，主要是我们的加密的组件，不是我们之前用的不是对吧？而且我们是每条user数据，每一条user它都有一个自己的sound，往往是不一样的，是吧？不是说一个固定的sound，所以这个逻辑虽然说能够解决问题，但是和我们目前的系统的现状不匹配，所以说我是给你演示一下可以这样做。

如果你是开发一个新的系统，基于这样去设计是可以的，但是我们现在系统已经开发到这种程度了，和它不匹配，我就不用这种方式了。

如果说我们系统这个数据的形态和它不匹配，我们有没有什么别的办法能够处理当然是可以的，我们可以自定义这个认证规则，就不用他默认的规则了，只不过稍微麻烦一点，而且我们自定义认证规则的话，我们还需要了解更多的组件，我先把自定义认证规则的这个代码的基本结构先给你写一下，然后我来解释组件的含义，20点also take action，然后的话 provider这里边我们通常这样写，new authentication provider好了，那么这基本上就是这样一个结构，我们需要给标的对象传入一个什么？

传入一个奥森tkc provider这么一个组件，在组件这是一个接口，在接口里实现一些逻辑，那么才能够自定义我们的认证规则。

好，你看这里又涉及一个新的组件叫奥森德克森特为的这是个什么东西？解释一下，就是我们刚才所说的所谓的manager它不是认证的核心的实现类，Provide manager它和奥森也可以跟provide有关系，provide manager他持有一组 authentication，provide而且每一个奥森txt，负责一种认证好，现在概念多了起来，逐渐多了起来，而且你会发现这些组件它的名字稍微都比较激进，比较相似，很容易就搞混淆，所以说这一课估计你课后还得再反复再看个一两遍，把这个概念把它理顺了，可能就好了。

是这样的，provider其实它里头有一个集合，它在集合里装了多个奥森特克森普瑞德，然后就是说是这样，这个普瑞的manager他不直接自己去做认证，它里边包含了一组组件，然后让组件去做认证，而每一个这样的组件负责一种认证，为什么是这样的呢？

因为你知道我们登录的话，可不是说只有账号密码一种形式，我们通过账号密码可以登录，可能我们通过第三方微信 QQ也可以登录，甚至我们少个指纹也可以登录，我们刷个脸也可以登录，是不是？我们在这种登录的形态是非常的多样化的，扫个码也可以登录，有多多样化的，不只是一种方式。

那么他这样做的目的是为了能够把所有的这种登录的模式都能够兼容了。

因此我们再回顾一下所谓的manager，他并不去自己亲自去做认证，而是它包含了这么多organization provider，每一个organization provider负责一种认证，你负责账号密码认证，他负责微信认证，他负责什么认证，是这样的一个逻辑。好，这在设计模式上叫做委托模式，这是满足了一种委托模式。

好，解释一下，也就是说所谓的manager，它是将认证委托给了organization to read。好，那么我们当前的游客社区的外网应用，我们采用什么认证？我们只有一种认证方式，账号密码认证，所以我们需要传入一个账号密码认证的这么一个 provide。

所以我们这里实例化的一个奥森dkc provider，那么在这里我们实现了一个账号密码认证的模式，这个意思好了，这里我们还需要实现这个接口，目前还没实现，那么实现这个接口之前，我们还需要做一些必要的解释，那么首先你看这个方法它有一个参数叫authentication，你还得理解它authentication是个什么意思，奥森txt它用于封装认证信息的接口，它是用于封装认证信息的接口，什么是认证信息账号？

密码。

这就是认证信息，所以这个对象是用来封装账号密码的，对吧？我们需要从这个对象中得到账号密码，我才能进行认证，它是用来封装认证信息的接口，你注意不同的实现类代表不同类型的认证信息，这是一个接口。

你不同的实现类可以代表不同的认证类型的认证信息，比如说实现类代表的是账号密码认证，实际上代表的是QQ认证，微信认证等等，我们这里用的是账号密码认证，它里面封装的是账号密码的数据这样的。

好，然后还要注意这个地方。

那么你看这个方法，这里边这个叫奥森tk的方法，就是做认证的逻辑的，这里边我们写认证的逻辑，这个是个什么 Support，这个是个啥解释一下。这个方法指的是就是说你要返回我当前的接口，它支持的是哪种认证，它支持的是哪种认证类型，它体现的是我当前能够支持的认证类型。好，我们通常怎么写？我们通常这样写。Return Return。然后的话这样写user name，然后它是word also take action。Token点class，点e cos，然后的话 a class类型。

好，就是说如果说当前我们认证的时候类型认证结果是这样一个类型，那么我们就支持我们或者说我们当前的接口是对这种类型进行认证的，这个是啥东西还得解释一下这个东西。

它是它就是authentication接口的一个常用的时间内，它就是接口的一个时间内，它代表的是账号密码认证，所以我这里写克拉斯一口是他意思是我当前的接口，支持的是账号密码的验证的模式，而不是微信不是二维码，其他的内容是这个意思。

然后整个方法我再解释一下，整个方法所体现的是什么呢？它反映的是当前的奥森txt，当前的接口，支持哪种类型的认证，归根结底它支持的是这种类型的认证，支持的是账号密码的认证，好了，你看这个方法特别麻烦。关于一个小小的认证，他搞出了这么多个概念，这么多个组件，里边体现了很多种这个模式，为什么这么做？因为我们在使用的时候要想扩展性更好，更灵活，那么它底层就要做更多的设计，才能支持这种灵活性，所以它的底层就搞了这么多组这么多概念。

因此我们在初学这个词块的时候，主要是往往容易被这些概念给弄晕了。那么当然你的学习可以有几种境界，第一个你可以说我不去纠结这些概念的理解，我先把这个程序先写出来，先跑通了，大概了解了是这么做的，这是第一个层面。

第二个层面我再去细抠一抠，这个概念代表什么含义，把它理顺了，大概的理解透了。第三个层面我再去阅读一些手册文章，或者我自己跟一些代码把它搞得更透一点，那就更好了，就这样。

好，我们这个方法还没有实现完，继续怎么去做这个认证，这个就简单了，我们只要从这个对象里得到账号密码判断就可以了。有的service我们也有，说白了我们把之前我们log in中的逻辑搬到这里来就差不多了，好我就写了，首先我要请求的内容那就是错了。Authentication，然后点get name。这就是u的name，好，再来还得获取密码是吧？Pass word等于 also education点。Get返回的就是密码。不过它返回的默认不是死菌类型，我们需要转型一下。

好账号密码我就得到了，得到以后就判断，首先我去查一下用户存不存在，我就调 user service，点find user by name传入user name。然后当然了我们也可以我们详细的判断流程得判断一下，这个又在存不存在对吧？判断一下 Uzi有没有激活这里，我简化，因为我们重点不在于这段逻辑，我们重点在于整个secret，它的整个的流程大体的结构，这些细节我写的不那么细了，因为本身这就是一个demo。

好，我就写个判断为user，如果user等于not，那就意味着账号不对是吧？当我们判断登录不成功的时候怎么办？我们抛异常。后面有专门的philter能捕获到这个异常，对异常进行处理，所以整个色块的底层都是feal驱动的，我们只要抛异常就有人处理，我们需要抛的异常也有这个规定，因为是账号不对，所以我们需要抛的是user name，然后 not found exception。账号不存在这个提示。

好，那么接下来如果账号存在，那我就得判断密码，而判断密码之前密码得加密，因为我们库里存的都是加密后的密码，对吧？我就pass word就等于我们调工具类，工具类我也有看没有这些优秀点。

Md5把原始密码传进来，还要加上盐，那就从优点里获取 Get salt，原始密码加密之后做一个加密，MD5加密，得到了加密之后的密码，我们判断一下这个加密之后的密码，数据库里的密码是否一致就可以了。如果说user的get它是word，它 equals用户传入的password不等号，如果不等于的话表示密码不对对吧？那也得抛异常。

此肉牛就叫bad，就要credit so exception就credit soon的它表的是表示的是一种也是一种凭证，也是一种令牌，其实它代表的大概是密码的意思，但是它为什么不叫密码？还是那句话，因为你登录的时候未必是账号密码登录对吧？你可能是刷脸登录可能是指纹登录，所以你登录的时候密钥它未必是密码，所以说它用这样一个单词来代替，笼统的代替。

好，这里写密码无证据。好了，那么如果说密码也正确，最后return我要返回的是认证的结果，而这个认证结果我们需要用我们需要反馈的是什么呢？我们需要返回的是这个接口的实例，而我们当前支持的是这种类型的实例对吧？所以你得返回这么一个类型。Return new，然后优质内部pass word奥森的k神头肯 Talk里第一个参数我们要传入的是我们认证结果的主信息，通常它是奥布加克的类型，但我们通常传user。

第二个是传认证的一个凭证，我们通常是传密码或者是能够代替密码的那个东西，我们这里就是密码。

好，然后第三个我们要传的是当前用户的权限，你user的get authorities就能得到，这个不是我们已经实现了这个方法，对吧？

好，这里我解释一下这三个参数，它的你看这个它的代码，它第一个参数叫这么个名字object，第二层是叫这么个名字，好解释它的含义，刚才我虽然说了，但是怕你忘第一个 principle，他的意思是认证的主要信息，一般我们就存 user就完了。

然后第二个 credit service，通常我们这是证书的意思，证书的意思。那么在账号密码模式下，通常我们就传的是密码好。

然后第三个参数authorities它代表的含义是权限就没有歧义，悠着点get authorities。好，这样就可以。那么当。

登录的时候，我们一点登录按钮，那么色块就能够捕获到请求，就能够拦截到这个请求，当他发现这是一个登录的请求，那么他就会调用这个接口这个方法去做登录认证，然后登录结认证结果会存到这个对象里，后面某些地方会用到，就是授权时会用到，是这样的。

好，那么关于认证的逻辑我们就处理完了，刚才这个方法是关于认证的逻辑，除了认证以外，别忘了还有授权的逻辑，整个权限管理核心的就是认证以及授权，那么还是要重写一个can figure方法，那么这个方法参数叫HTTP这个时候我要给你看一下super中的代码的逻辑，它里面是有实际的逻辑的，好你看一下，当然它的逻辑比较长，大概看一下你看hgdp点，also all the right request。

点 any request 是任何的请求，任意的请求，然后 all c austerity, Austin kitty。什么意思？大概的意思就是说任何的请求都要启动认证，所以说你看我们只要以引入了 security，那么 Can figure 就这个配就默认配置文件它就生效了，然后这个方法就生效了，生效的时候它因为这段逻辑就生效了，就任何请求都要被权限控制，所以为什么我们引入了 SQL 以后，一访 index 一访任何一面他都逼着我们去登录，就是因为这个逻辑。

所以说如果你不想这样，你不想上来，就所有的请求都被都被判断怎么办？你把它覆盖掉。所以这里头你看我把这个 super 一去掉，那么父类的逻辑被我当前的配置文件给覆盖了，那么默认不是所有请求都要被处理掉，那么就不会出现这样的情况。

我们说白了我们就可以避开它默认的登录页面，我们通过这种方式避开它默认登录页面，我们的目的是找我们自己的登录页面对吧？我们怎么去找我们自己登录页面，也需要在这里进行配置，我们需要做登录相关的配置，你得告诉 secret，那么我的登录页面是谁，那么我的登录表单是谁，或者我登录表单是我唯一提交的时候是哪个请求，好，能够请求它拦截到你登录的请求以后才能够调接口，他才能够获取账号密码，才能够调接口，然后调认证的逻辑是吧？

所以说这里你得告诉他一些关键的信息，这个怎么配呢？固定的模式是 hgdp 点 form log in 登录表单相关的配置，然后的话首先配通常我们会配 log in 配置，你要告诉他登录页面是谁好，我们登录页面是谁呢？看一下 home CTRL 是这个请求 log in page。

好，那么我告诉他我的登录页面是 log in page，然后还得告诉他我提交表单时处理登录的请求的路径是谁？他好去拦截这个路径，帮我们做认证，但是我们现在路径还没有，我这里提前配好 log in PRO session ui 就是登录的处理的路径我叫老兵，那么一会儿我们表单上就必须配这个路径，它才能处理掉。

好，那么你配了这个路径以后，当我们一访问一发出这个请求的时候，它就拦截了，它就掉这个逻辑了。调这个逻辑以后，它最终在认证的过程中可能会成功，也可能会失败，而失败的表现是抛异常。我们怎么去处理？成功或失败有两种方式，一种方式是可以 success for word URL 我成功时跳转到某一个路径，这叫转发到某一个路径。

然后的话然后点 Failure 发布的 ui 失败时候跳转到某个页面某个路径可以这么简单的处理，但是我们往往不是说我们跳过去就可以了，我们可能在成功时或失败时可能还要带一些数据，还要做一些特殊的处理，所以这样的方式比较局限，那么还有一种替代方案更灵活，比如说我可以这样写，success handle 成功的处理器，那里边需要传一个接口，我们在接口里面自定义我们成功时处理的逻辑，这个接口是 new 奥森特 kitchen success handler 我们需要定义这个结果，类似的失败的时候也是这样 failure handler，然后的话这里边也是你有实例化一个失败者的接口，你有奥森特 k 选飞了，handler 就是这样。

那么这个逻辑怎么定义？当然我们可能在实际的真实的项目里会稍微复杂一点，当前的 demo 我做的稍微简略一点，如果成功的话，我就跳转到首页，我就跳转到首页，我就可以这样写 Is bus，它参数名都搞得太长了，我就叫 request，我就叫 response，简短一点，这个也是一样。

好，那么成功的时候认证成功的时候，我就 response 点 send redirect，重定向到首页，我就 request 点 get contact us。

获取项目名，然后的话加上首页的路径，index 这就行了，那么失败的时候怎么办？我们失败的时候要回到登录页面对吧？并且给一个错误提示，回到登录页面给个错误提示。那么回到登录页面能不能用重定向的方式注意是不可以，因为如果我重定向到登录页面，那么重定向那么会让客户端发一个新的请求。既然客户端发了一个新的请求，请求变了，我怎么去向下一个请求传参不方便传？可能你只能用跨请求的组件 qk section 的去传，有点麻烦。

我们还有一种方式就是我可以把参数绑定到 request 里，然后把请求转发到登录页面上去，转发和城乡不一样转发是保持在一个请求之内，我们是可以通过 request 去绑定参数传参的。

当然转发到底是怎么回事，转发和成像到底有什么区别，我一会再详细来解释，我先写着我先往瑞快上绑定一个错误消息，sat开始的消息叫艾瑞，然后这个消息就是exception，是因为你失败的时候他传一个参数就是这个异常，你看失败的时候他把异常传进来了，其实就是我们之前抛的异常，这是异常的负接口。

我就直接把这个异常然后的消息发到那个页面上去，然后我要转发到登录页面，怎么转发呢？用request request点get request, dispatcher。我要转化的页面的路径是log in配置，然后点发word转发，把我当前的request response转过去，这样就可以了。

好了，那么到这儿我想解释一下这个转发到底是怎么回事。之前我们没有具体的解释过，而且转发合同对象其实很像，他们都是将让请求做了一个跳转，但是有一些微妙的差别，所以我们对比着来讲画个图。

好，首先我们先来回顾一下重庆相，因为我们相对熟悉一些，回顾一下重庆相，换个颜色我画的方块假设代表它是浏览器，然后我再画一个方块就代表服务器，那么重定向是这样的，就是浏览器有访问服务器，访问的是它里面的某一个组件，假设这是a组件，好，假设我服务器里有一个a组件，当前我浏览器访问就是a组件，那么有的时候访问a组件以后，A组件其实没有什么结果可以反馈给你，为什么没有结果可以反馈给你？

比如说我访问的是一个删除的功能，对吧？数据都删掉了，给你反馈什么结果呢？这个时候可能需要另外一个组件给浏览器显示一个显示一些内容，但组件和它没有直接关系，比如说这是删除，这是查询，删除以后我想回到查询页面，而删除和查询有直接关系吗？没有，它是两个独立的功能是吧？这种情况下我就适合用重定向两个独立组件，两个独立组件，我还想让它跳转，我就适合重定向，而不适合直接这样去调。

如果直接这样去调，a和b本来没有关系，就产生了关系，它俩就产生了耦合，这种方式是不好的。

而重立项可以降低额度，重定向之前我们也说了，他的意思是这样的，服务器给浏览器一个响应，告诉你我没什么可以给你的，它的表现是我访问了一个302状态码，当然了除了302状态码以外，还包含了一个路径，说我建议你去访问它，那浏览器一看好吧，我就访问b然后b会给他一个结果，展现b的访问是浏览器自己去访问的和a其实没有关系，a只是给了一个建议而已，a和b之间没有直接发生关系，没有耦合度就这种方式是一种低耦合的方式去跳转，然后但这种方式也有一些小的问题说不是说问题，有一些地方不好处理，比如说我a还想给b带个数据，但通常不会的，假如说a想给b带个数据就不好带了，因为这是一个新的请求，request都被重置了，对吧？

你通过request没法去带数据，这是一个request，没法通过开始带数据，两个请求又想共用一份数据，你只能用cookie或session这样的，这是重庆项它的这种感觉。

我们再看这个转发好，还是浏览器和服务器之间的这么一个故事，还是比如说服务器有两个主键a和b好，那么比如说浏览器访问了a然后有一种场景是a它只能处理请求的一半，剩下一半需要由b来处理，需要a和b共同合作才能完成整个请求，这个时候a就可以把请求转发给b让b继续处理，然后由b给浏览器做一个最终的响应。

那么整个过程是一个请求，因为是一个请求之内，所以说a可以把数据存到request里，b能够取出来，他请求没有重置，他俩用的是一个request，a转发给b的时候，要把request response给b你继续处理，就是这个意思。然后还有一个差别是什么？

就重立项的时候，因为浏览器访问b的时候，他知道我要访问b他本来访问a的时候地址栏写的是a当他访问b的时候，他知道他要访问b地址栏就变成b了，而这个他访问a的时候，这里是a访问a然后由b给个结果，但浏览器不知道有b的存在，他还以为是a访问的结果，他不知道有b的存在，所以他的地址还是a的地址，这也是一个表现的情况，现状就总之如果说我服务端有两个组件，我想去跳转的话，你分析一下它俩之间是协作完成一个请求，有耦合，还是说是两个独立的功能？

没有耦合，如果没有耦合你就重新下。如果有耦合，如果是要共同完成一个请求，你就转发是这样的。有人说能不能举个转发的例子，其实我们之前没有用过，没有用过转发，这里我举个例子，比如说假设a的路径是log in，然后B的路径是log in page。你看A是log in，比方说a是什么？A是提交登录的表单，那么提交登录的表单有可能是我们在服务端一认证失败了，失败的时候我们就需要回到登录页面，那么而登录页面log in配置是另外的一个请求，那么它俩之间就可以产生依赖，因为我需要依赖于他，我自己怎么处理一半，另一半可以由他来处理，这个时候我们就可以不转发，就是我们刚才的场景，当然我们之前没这么做，我们也可以怎么着，我们也可以在里直接return一个登录页面模板，written login的html也可以，你转发给他也可以。

这有什么区别呢？就是说如果我在a里直接return登录页面登录模板相对于转发的话，转发的话是复用了这段程序，如果说log in配置请求方法当中还有别的逻辑，它能被复用，它这种方式其实优于直接return一个模板，总之两种方式都行，这只是一种方案，而我当前不得不用这个转发了，因为我当前是处于这么一个handler方法之内，它又不是CTRL的，我也没有办法return一个模板的路径，它也不能识别对吧？

所以我这里我只能转发了，迫不得已就这样好了，你看我现在把登录相关的配置也配好了，你登录配了，退出也得配，你登录时是我们认证的，你退出时要把认证结果给删了，然后做一些处理好，所以接下来退出相关配置把它处理好。退出的配置是HTTP点log out，然后点log out，URL，你推出的路径是什么？我们就是老高，你看这个controller我这里还没写，我页面上可能写了一个假的路径，跳过去了，我要求退出的路径是报告的，一会再处理，然后退出的时候有可能会怎么说，就是退出的时候他最终是会成功的，成功以后我们需要做什么处理，需要给一个指示，我们可以log out，you are log out。

然后 Success URL你提供一个成功以后跳转的路径，或者是也可以这样log out success hand给一个处理器，可以更灵活的处理，我们用处理器的方式更灵活一点，那就new log out success。Handler在这里他给我们提供了request，response以及认证结果这些对象都有，你想说怎么样的处理都行，我这里就跳转到首页，也就强制跳转到首页，这个可以重新下，你退出和首页没有什么直接关系。

它俩不是协作的关系，我觉得成像response点散的redirect，或者如果你不太清楚哪些用重新向哪些用转发的话，你就优先用重定向，如果重定向你发现我不好传参了，有些问题解决不了，你得考虑很准的，你可以这样去考虑问题。Request点get contact pass加上首页。Index可以。

那现在我是对登录退出做了配置，这是与认证有关的，除了认证以外还有授权，我们还要对授权进行配置，主要是配说你哪个权限，你拥有哪个权限能访问，哪个路径，是权限与路径的对应关系，固定的格式这样写。Http点，Authorize request，然后点and matches。

写上路径，比如说 lighter私信的路径，然后点has any of the city。换个行不？换行就这样了，意思是这个路径有哪些权限才可以访问呢？Has any or有任何权限就可以访问它，你要配权限，我这样配user。革命什么意思？只要你拥有user或ADMIN，任何一个权限都可以访问到，这是合理的。你只要登录了，有了一个任何权限都可以访问它，私信好，然后再来and mattress。

我们的系统中还有一个路径是ADMIN管理员专属的页面。我写has authority。这里我就写好的，没有。意思是你有管理员权限才能访问这个页面，你只是有点防不了，它底层会做控制的，会做拦截的。最后再来个暗的，暗了之后还能再拼点别的东西。And exception handling。

那么有可能有这种情况，如果说你压根就没登陆，一看你根本权什么权限都不具备，或者说你登陆了你是优点，但是你要访问它的命，这个时候你权限不匹配了，它会报错，我们怎么处理这种权限不匹配，或者说你没有权限的错误？

我这样写 Access单位的配置我要指定一下，当你没有权限的时候，我访问哪个页面，我就调到这个页面带你的页面这个路径，而这个路径我现在还没写，我需要补充路径，不然的话这就不对了。

这个路径我也是把它写到 control这里，目前没有，我再加一个写个注释，就是说拒绝访问时的提示页面，或者说你没有权限你的提示页面，我copy一下，那么访问路径带你的拒绝，然后这个是get就可以了。Get就行。

然后方法名我叫get denim配置。返回的页面是谁 era? 404。当你没有权限的时候，我给你返回一个404的错误页面，比方说不存在，你没有权限访问，对于你来说这就不存在合情合理是吧？好了，现在我终于把 security就配置完了，那么配置完以后别忘了我们认证的时候，你还需要提交表单，表单我们还没配对吧？

所以说我们现在配置完了，路径也搞好了，但是表单还没配，我需要处理表单 login，点html，首先我要声明表单提交的方式，post没问题，然后表单提交的路径，写一下th action，我们手快的让它配置的时候，要求路径应该是老in，才能够被所谓的拦截到，所以你要按照规则来配。

然后你要提交表单的话，这个数据你要取名字对吧？Name等于user name，然后 password，name等于 password，因为我们在security做认证的时候，这里边它需要获取数据，那么你需要取个名字，他可能被它拦截到，而这个名字需要按照这样的标准来取。

好，这个验证码我们先不管，一会再说。

另外那么如果说我一点登录，他这一认证发现不对不对，以后还回到这个页面，你还要给一些错误提示，我们把这个提示到段落里，th冒号，txt等于艾瑞尔，错误信息就是艾瑞尔对吧？

另外的话如果你错过的时候，我希望回到页面的时候账号密码有一个刚才的那个值不好，基于这个值改，那就th y6等于那就是参数，我们请求参数里边的有的内容，这个就是请求参数里边的 password给他一个默认值，等回来的时候，好，最后登录按钮是300米的一点，就登陆，这个没有问题。

登录处理完以后再处理一下这个退出，因为退出我们看一下首页，首页之前是就写了一下，退出的时候就直接到了登录页面，我们需要重写，为什么要重写？

就是说色块的要求你退出的话，必须采用post请求，他这种请求就是get，所以不符合要求，我们重写一个，因为是POS的请求，所以我需要有个表单，然后 master等于post，这是他的要求，没办法，只能这样，action等于这个路径咱们在配置文件里写的是log out，按照规则写成log out，然后退出我就加个超链接就行，别的数据都不需要然后退出，然后它的href等于因为我退出是要提交表单，所以你不方便直接写一个链接，我要提交表单怎么办？

我可以用GS来实现。

那么如果你这个链接这里写的不是链接，而是GS，你要这样声明java secret冒号，我这样写 document forms意思是我获取当前页面上所有的form，而我只有一个form，那就是 d0的 form，0得到的数有方明的forme，然后对 forme Diao，沙漠的方法就提交了，这就是一段简单的GS就可以了。

好，这个退出咱们也配完了，最后为了让好看一点，我想在首页上再加一个欢迎信息，如果你登录成功以后，这里显示欢迎你某某某，注意我不是在每一个页面上都显示，我只在首页上显示，既然你要在首页上显示一个信息，我需要在首页访问首页的时候在这里处理一下，我需要获取到你登录的信息，然后发给首页，我写个注释，我们认证成功以后，我们怎么去获取认证的数据，这里解释一下，就认证成功后，那么这个认证结果会通过 security contest，holder，screen，rule，security context这个对象当中，这是底层有一个filter专门干这个事儿，当你认证成功以后，他拿到了认证，结果他就调 hold把这个结果存到对象里，所以他认证成功以后，他存到这个对象，你要取结果就从这个对象里取，我取一下 security，so clarity、context、holder点get contact。

就得到了这么一个对象，然后这个对象里就存的是认证的结果，你就代表get authentication认证结果，而奥森的kitchen我们之前往里存了什么，你看我们认证的时候，这里边我们不是往里存的user，而我们现在页面上要显示的不就是user相关的内容对不对？这个user是不是我们是按照以 print Principle来存的，对吧？所以你可以在这儿就是点get principle。

我们得到的其实就是 user这里，我这样我先当作object来处理。然后判断一下，判断一下OB机instance of user确认一下，如果它确实是user，那就说明你登录成功了，如果没有登录成功，可能会给一个匿名的给一个别的结果，或者有别的情况存在，我怕如果说你直接就把它当成优质来处理，可能会有一些问题，所以最好判断一下，如果是优点我们就model点ed这里还没有model是吧？



说明一点model我就model点ed edge build，然后的话log in user登录的user， user把 obj实际上就是user传到页面上去，那么这回登录页面上我就可以显示刚才的登录结果了，对不对？首页上就可以显示这个登录结果了，在这里我补充一段内容，欢迎信息加个段落，我在段落里显示这个信息 Th做个判断，因为只有你登陆了，我才显示你不登陆我就不显示了。

我就log in user，如果它不等于now我才显示这个信息，这个信息我显示成什么？显示成欢迎你，然后的话十八里显示的是某某后面带一个感叹号，然后18我就需要我这边放用户名了。Th text等于老b有点内容，这样就行了。

好了，那么到这儿我们终于应用了 Sql，目前的话我们应用它认做了认证，主要是体现在这个方法之内，也用它做了授权，主要是体现在这儿。当然这两段代码其实也是和认证有关的，是这样的。

好了，那么处理完以后咱们试一下看看行不行，那么我重启一下服务。好，那么启动以后，咱们来访问一下，打开浏览器，我再访问首页，刷新首页可以访问，然后我访问帖子详情，没登录，可以访问我，访问私信，访问不了。强制我登录。因为我没有权限没登录对吧？没有认证过。

好我访问ADMIN不行，强制我登录，我们刚才的配置就生效了，你没有认证就必须登录好吧？我就登录AAA登陆，咱们这个表里不是有AA的用户对吧？Aa登录先不用验证码登录，欢迎你AA表示登录成功了，认证结果已经写入到了康泰斯里，好，登录成功以后再访问详情没问题对吧？

再访私信列表没问题，普通用户就可以访问信息列表，访问ADMIN他就到了404页面说你访问的页面不存在，这是404页面给的提示，你看授权权限的管理也就ok了是吧？我再回到首页。

好，你看目前的话，我们 security表现的状况就是ok的，当然你也可以退出以后你再访问私信又不行了，是吧？这就是色块的一个作用。从认证到授权都能够做全方位的管理。好到这还差一点，我们在登录的时候可能还有验证码的，对吧？我们可能甚至还有这个记住我这两部分怎么去处理，这个也给大家演示一下。

那么首先我先说一下验证码，验证码我们不能够直接在这个地方去处理，这个地方对象里只封装了账号密码，没有验证码，没有记住我相关的信息。就是说 Security它的职责分工非常的明确，你这个方法只是对账号密码进行认证的，就只有账号密码存进来，和别的无关。然后如果说我要验证这个验证码怎么办？

那你要知道验证码应该是在账号密码之前验证，如果验证码都不对，账号密码就不用验证了，应该是这样的。我们怎么办？我们这样做，我们可以增加一个filter，在账号密码认证之账号密码认证的filter之前加一个filter，专门解决这个事儿，这需要制定一个field来解决好。我给你演示一下，那就在这儿加一个配置，增加 filter，然后处理验证码，这样的话hgdp点l的 filter，你看爱的filter，爱的filter比方是在谁之前加，在谁之后加，你可以指定我是要在谁之前加对吧？

比方第一个参数你的filter实力，我们可以在这匿名实现一个future，就是new future实现接口，然后注意还有第二个参数，第二个参数你要指定你要在哪个filter，哪个filter以哪个photo为目标，在它之前加处理账号密码的 Filter，底层的filter叫u.name，然后也就 u the name password also take action。Twitter。

当然你要指定它的类型，剩下的我们就需要在这里写逻辑就可以了。

Filter匿名的filter就会在它之前被执行，在它之前拦截请求好了我就接着写，首先你看 filter的参数类型， server later request，而我们之前用的是HTTP server request，什么关系是这样的，这个是副接口，这个是子接口，当然这个方法调的时候肯定是传他们的一个实现类，我们通常用类型的方法多一点，我们这里可以转型一下，因为传的实现类，那么虽然说你用的是一个副接口来接收，但是我其实可以把它转型为这个接口，因为它传的是他们的视线内。

好转型成类型，好用 HTTP request等于servility bequest。当然你向下转型的强zhuan response我转一下方便一点。好，那么转完型以后我要做什么？注意我不是每个请求都要处理的，你一定是登录请求，我才要处理验证码的逻辑对吧？你不是登录请求我就不用管，我要判断一下，如果说request点get subvert passed，然后点echoes路径它等于log in。

如果说我们当前的访问路径是log in的话，说明你是登录的业务这个时候就有验证码存进来对吧？我就要得到这个验证码，假设这个验证码的名字叫WiFi扣的，我取一下 WiFi扣的。等于request点get very cold。好，我得到了这个验证码，得到以后我要判断验证码对不对？当然我们按理说应该是再提供一个方法，专门生成验证码，然后验证码存到 rise或筛选里头去。

我们这次课重点不在这，验证码生成我不写了，我就假设验证码固定就是1234，我就判断他等不等1234，然后简单处理，那么如果说外人说扣的，他等于闹，他是空的，或者是 wherever code不等于1234不等于前面加个不等号对吧？

这个时候我都认为验证码不对，验证码不对，怎么办？我需要给一个错误提示 Request叫sat actual，然后说验证码错误，好，然后我需要回到登录页面，我需要转发，转发的话，咱们可以这不是写过吗？拷一下。

好request，get this picture，然后 log in配置forward。转发过去以后return后面的代码不用执行了，因为到此为止。好，否则如果说没有return就说明什么？说明要么是不是登录请求，要么就是您找对了，后面还要写一句话，Feel turn，这有一个参数叫菲尔德欠，你要写一句话，菲尔德欠点future，然后的话 request response，这句话是什么意思？这句话的意思是让请求继续向下走到下一个filter，如果下面没有philter，他就走到Savitt，让请求放行，让请求继续向下执行。

如果说你这里没有写这句话，请求到此终止，不会向下走了，所以通常我们这句话一定是要写的，或至少是有些情况下是要写的，有些情况下不走，如果你验证码不对，请求别往下执行，如果验证码对请求继续执行这句话有特殊的含义，因为你有多个filter对吧？

好了，这是我们对验证码的一个那么如果你想处理一下记住我也可以，我在这里再把这个记住，我也给大家配一下，记住我HTTP直接配 Remember me，组建专门配记住我，然后首先是一般会配token，reporter你要告诉他，那我记住我你用什么来记，你是把用户的信息记到内存里，你还是把它记到数据库里，你要给定一个存储用户数据的方案，然后它默认带了一个类，我就用默认的new in memory什么talking repulse the apple这句话很显然他要把用户记到内存里，它默认一个实现。

如果你想存到reds里存到数据库里，你就自己实现一个接口，那个接口就叫应该就叫talk reporter自己实现一个结果自己去做就可以了，这里我就不去处理了。整个流程你知道就可以了。

然后再来偷看，然后 validate seconds过期的时间，你希望记这个数据什么时候过期给个时间，我3600乘以24这是秒，second是秒，1小时3600，就24即24小时。

然后再来 user detail service，user service，你还得告诉这个记住我组件 user detail service，为什么？

因为你这次登录他记住你了，当你把电脑关了，当你下次再访问这个网站的时候，它从内存里能得到根据你的凭证能得到你的用户信息，能得到你用户的比如说ID，得到ID以后不是得到用户名，得到用户名以后他得给你查出你用户的完整信息对吧？他得帮你查出完整信息好，让你就是帮你自动的过认证的那一关，他怎么查就用这个来查你，得告诉这个东西。

好了，到这儿验证码和记住我咱们就这就配完了，就这么配完以后登录页面你还得做一些处理。

好，我再回到登录页面，首先处理验证码，验证码我说了，我认为它默认就是121234，我觉得写死了1234，就是1234不动了，然后你得给它加一个name，对吧？Name等于 WiFi扣的和刚才我接收的k是一致的。

除此以外还要记住我再加一个段落，这里面写一个记住我得是说勾选的是吧，那得是tap等于checkbox，然后 name是固定的，一定要叫 remember me，然后给一个提示记录好，这样的话就可以了。

我们再次做一个测试重启。

好，那么我打开浏览器，首页访问首页，目前是没登录的状态，我登录一下我先不记录我这回我还是用AAA登录，如果写一个错误的验证码登录你看生效了验证码错误，然后我写一个1234，没有记住我登录欢迎你没问题对吧？

好，但是我没退出，我关掉浏览器，再访问 local house index，这是没有登陆的状态，为啥没登陆？因为我没点记录过，他没有去记对吧？好。再来登陆 Aa, 1234，记住我登录欢迎你。关掉再访问 local house index。欢迎你AA为什么会这样？因为他记住了，当你再次打开网页的时候，你网页里通过库里有一个 Talking有个凭证传给了服务器，服务器根据凭证找到了user name，然后根据user service detail查到了你这个用户，然后这个消息就看到了，他记住你就帮你绕过了认证的过程，或者说他帮你自动登录了，就是这样的。

好了，那么这次课我们花了一个比较长的篇幅，就是把 security它的概念，它的大概的实现原理以及使用的方式给大家做了演示。

那么课后的话你一定要不要看它是一个demo，你一定要好好自己去试一试，因为确实还是挺麻烦的，你第一次学概念太多了，你不试一下的话不容易记住，自己亲自试一试，试完以后你对他有了感觉，那么下次课我们再利用 secret，我们把它应用到我们的游客社区的项目，去解决我们项目中的问题，提高我们项目的安全性。好，这节课我们就演示到这里，咱们下节课再见。

## 7.2 权限控制

---

44-权限控制.mp4

这次课我们来做权限控制，就是把上次课我们所学的 security这项技术运用到游客社区这个项目中来管理我们项目中现有的这些请求的权限访问的问题。那么我们在做开发的时候，大概需要从这4个方面入手。

第一个方面我们需要把之前我们所写的登录检查功能给它废弃掉，因为我们之前写过一个拦截器用来进行登陆的检查，当然了，这是一个比较这个基础比较简单的权限的实现，那么我们用色块去实现的话要代替它，然后的话会比它更安全，功能更强大，所以把之前原有的简陋的检查给它去掉。

然后第二个是我们要做用色块的做权限控制，主要是集中在两个方面，第一个方面是认证，第二个授权。

那么关于授权我们现在这个项目开发到这个阶段有很多请求，这里边有一部分请求需要进行权限的控制，有一部分不需要，因为有一部分请求是你不登陆谁都能访问的这样的功能，就不需要我们就把需要控制的那些请求找出来，然后把它配置好，而对于认证，那么因为我们当前已经把登录退出功能已经开发完了，如果说我们按照上节课我们学的这块的案例，那么去弄的话，我们基本上得把。

登录。

还有退出相关的逻辑重新再来一遍。

我觉得没有必要，我还想用我们系统原有的认证方案怎么办？我们想办法绕过色块的认证流程，我们采用原来的方案，当然怎么绕过其实也比较简单，那么我们在做的时候，大家注意留意就好了，很容易。

然后最后一个大家还要关注一点，就是关于 csrf配置 csrf它是一种比较常见的攻击手段，那么security它默认就启用了，防止csrf攻击。但是我们在开发的时候需要有一些细节需要去处理，这样的话才能够彻底的防止这种攻击，所以说我也会给大家讲解关于csrf方面的内容以及做相关的配置。

好了，那么这么几个方面我们一步一步来做，我就打开这个idea开始写我的程序。

首先因为我们这个项目中还没有引入secret，所以我们需要先打包，打包我们可以从我们上次课的案例中把那段代码贴过来，我就不去搜了，我用都得派的提前已经把这个项目打开了，然后的话已经打开了泡沫的xml我把这段代码copy一下，然后粘贴到我们现有的项目中来，然后很快它就加载完了，因为这个包我们本地已经有了，好这是第一步。

第二步这算是第零步，就是我们做一个前置的工作。第一步我们正式要做的，首先是把登陆检查拦截器给它配置给它废弃掉，那么我们没有必要把这个类删掉，我们只要把它的配置让它不生效就可以了。

而拦截器的配置是在 Web mvc can figure类当中进行配置的，我们打开看一下，它 log in required intercept，我们之前利用它就是拦截了一些请求，这些请求必须登录以后才能访问，现在我不需要了，我们需要我们要用色块来替代它，我赶快把它注掉，但是这也得注掉，不然就报错了，这样就行了，那么拦截器就已经被废弃了，分析以后，那么接下来我们就开始做下一步操作，做授权有关的配置。

那么我们就需要写 secretary的配置类，在写配置类之前，我想先在常量接口里先增加几个常量，因为什么？我们在配的时候需要指定说你是权限，我让你访问这个请求对吧？我们系统里支持哪些权限或者用户有哪些身份，我们最好先把它在常量接口中加以定义，一会好引用。

好，那么我就打开常量接口，在后面加几个常量，首先是加一个用户的权限，就普通用户，那么我加 host writing user，普通用户就等于user，然后其他的我就copy我们的应用当中其实主要就是用户有三种类型，就有三种权限，一种是普通用户，一种是管理员，一种是版主。

那么我们看优质表，看见表的语句当中有这个描述，大家可以看 Tap类型的描述。

好，再来一个，第二个是管理员，管理员变量名叫奥斯瑞特，然后20，这个是2.0，好，再来整体第三个版主，然后是奥斯瑞特毛德瑞特好了，那么这常量我就设置好了以后，接下来我们就对色块的进行配置，主要是做授权有关的配置。

那么我就在肯配合包下，再新建一个色块的配置类。

那么首先在类上我写上看，微信另外类需要继承一个父类，叫做web security configure adapt，另外我让它实现常量接口，这样我好引用它里边的那些个常量好了，上次课我们做了演示，就是这个类我们通常是重写它里面的三个方法。

那么第一个我们再写一遍，第一个第三个都是肯菲格，第一个的话是带web squared参数的 QQ的方法，那么在这个方法里我们通常做一件事儿，就是忽略掉对静态资源的拦截，静态资源直接访问不用拦截，那就是web ignoring，然后 aren't measures。然后的话resource星星所有resource之下的平台资源都可以直接访问。然后第二个是重写带有什么奥斯瑞特什么的这个方法，然后在这里我们去做认证。

但是刚才我也说了，我们现在的项目已经写好了登录退出方面的这些认证的内容，最后我们要绕过这个环节，所以我不重写了，不加了，但将来怎么去绕，后面再做补充，我们先直接进入第三个环节，就重写带有HTTP security参数的方跟这个方法，那么在这个方法里我们主要是进行授权，那么你需要梳理一下，主要是看了我们项目中的所有的啃臭了，因为我们所有的动态资源的访问路径都是在啃臭那里进行定义的，那么你可以把你的每一个CTRL都看一遍，每个方法的路径都记录到某一个能记录下来，记录到比如说某一个记事本里，然后一个过一遍，哪些是不用登录就可以访问的 pass掉，哪些是需要登录以后普通用户能访问的，我们需要集中起来去管理，但是我们现在还没有管理员和版主特有的访问的路径，都是普通用户能访问的。

那么将来会有管理员，还有版主特殊的功能将来再说。

好，这里我提前已经过了一遍，已经整理好了，我就直接写了。然后你们在写这段代码的时候，这个地方可别让我抄，因为咱们每个人在开发的时候命名习惯不一样，我们写的路径可能是不一样的，你照着抄的话可能就起不到相关的作用，所以你要注意。

好，那么授权的话我就这样写，Http，然后点also rise request，然后点ant matters先声明一个路径，这上面一堆路径，这一堆路径我垂直写，这样我们一行一个路径阅读起来比较方便，然后在先写个引号占位，然后再点这些路径 has any authority。

对于这些路径，只要你拥有以下任意的一个权限，你就可以访问，这里面写上权限，然后等等再往后写。

好，我这里就先写路径了，首先就是首页你可以直接访问，然后关于优则里边看头像，看主页也可以直接访问，但是 user里边有一些不能直接访问，比如说 user、site，用户的设置你不登录是是不可以进行设置的。

好第二个还有user upload，如果你不登录你是不可以上传头像，好。

然后还有 diss Cass at就是发送发帖子，那么如果你不登录不可以发帖子，还有comment at后面我再写个星星，然后被他这么写的，因为comment at下面它还有一级是一个帖子ID的一个参数，所以还有个下级我就杠星星代替了。

那意思是你添加评论你也需要有权限默认不登录，没有权限不行。好，这是4个再来，然后是私信，私信我们所有的功能，都是以let开头杠星星表示私信所有的功能，那都是要有权限的。还有和私信的类似的是notice通知，

那么也是你。

至少要登录你才能接受通知，如果你是一个匿名用户，你根本就没有通知。好再来，然后的话是like点赞，也需要有权限才能点赞，然后是follow以及安抚了。

好了，我整理了一下我当前的所开发的各个功能，一共是这么多路径，那么其中这些路径是你需要登录以后才能访问的，而登录以后你是什么权限什么权限都行，也就是说你是普通用户，你还是管理员，你还是版主都可以访问这些功能，所以这里面我需要写上三个权限 Authority，user，普通用户，authority。

管理员admit authority， moderate。版主这三个任意一个都可以。

好了，当然我们现在就这些路径，这些路径需要有任意的权限都可以访问，其他的没有特殊的了，我们暂时先不用去配别的路径了，将来的话我们有管理员专属的路径，有版主专属的路径的时候，我们再去补充，好，那么再往下可以再接着写什么，除了这些以外， any request，除了这些请求以外，其他的任何的请求我就 permit on，就都允许统统都允许，这样就可以了，这是对我们现有的路径的权限的配置，除了这个以外，我们还需要配一下，那么如果你没有权限的时候，如果 SQL在底层捕获到了你没有权限的那些种种异常的时候，我们应该怎么去处理？

好，所以还得再去配一下权限不够时的处理。那就是HTTP，然后点exception handling，然后点上次课我们是配的是 access单位的配置，当你权限不够的时候，我们给了一个错误提示的页面简单的处理，但是我们现在是一个正式的项目，这种处理方案就有点简陋了，我们上次跟小demo这样处理是完全没有问题，为什么？因为我们当前的项目。

怎么说，有多种页面，有多种请求，有的请求是普通请求，有的请求是异步请求。

所以你对于普通的请求，他期望你服务器给他返回的是一个html，我们。给他指定一个页面，我们跳转过去就可以了。但是如果是异步请求浏览器期待你返回的是什么，他期待你返回的是一个Jason，这个时候你给他返回一个html就不对了是吧？所以说我们需要有甄别的去对待，因此直接返回一个页面的这种情况就有点简陋了，我们可以换一种方式，就是用这个 excess denied handler。

当我没有权限的时候，我们用一个handle一个主键，特殊的去处理主键中的逻辑，你可以任意的定义。

用另外除了这个之外，我们还可以再把配置一下，就奥森tkcn angry point，这个是配什么呢？这是配就是说你没有登录时我怎么处理，而刚才的 access带 handler，它是配的时候就是说你登陆了，但是你权限不足时的一个处理，我们把配好以后处理的方案就更为详细的。

好，那么奥森tk选安追point就是你这个需要认证时，我们需要进到这个点里，发现你没登录，你需要认证，我们需要进到组件当中，它需要我们传入一个组件来解决这方面的问题，我们需要实例化一个接口做一个匿名的实现。

那就是new奥森tk审，然后的话安全判断这个好，那么另外这个是没有登录时的处理，另外再来点 access Denny handler，这里边是我权限不足是怎么处理，我就还是匿名实现一个接口new access。

xs dynad handler就这样，那么这个接口当中分别有一个方法，这个方法就是当发生这件事时，我们需要做哪些处理，当然我这里先把注释写一下，这里面是没有登录时我们要做这样的处理，这个是。

权限不足使得处理。

然后你看这个参数像没有登录的时候，他给了你一个request，给了你一个resource，然后还给了你一个异常，这个异常就是你没登录它封装的一个错误提示，在这里我们需要怎么处理考虑请求的方式，如果你当前是一个普通的请求，我直接就跳转重定向到登录页面强制你去登录就可以了。

那么如果你是一个异步请求，这时候我们就不能给你返回网页，要给你返回json，我就去给你拼一个阶层字符串给你一个提示就完了。然后浏览器上给一个提示，或者是浏览器上，我们通过前端的手段再强制你去登录就可以了。

好，如何去判断当前请求是同步还是异步，我记得好像以前我们在哪个地方在处理异常的时候好像做过类似的事情，这里我们再回顾一下，主要是看这个请求的消息头的某一个值，某一个值是request的get header。

那么 header的key叫x-requested-with，然后 value好。

我们。

通过 key可以取到一个值，这个值是个字符串，词句取名叫x-requested-with。

好，然后我们对这个字符串进行判断，如果说我们得到的这个字符串它的值是xml，然后 HTTP request，如果是这个值，那么就意味着它是异步请求，如果不是这个值就是同步请求，因为异步请求他期待你返回的是xml异步请求object，那个里边的x代表的是xml，但是现在被机身基本上给代替了，但原来原有的意思指的是xml，所以说如果他期望这是一个返回xml的请求的话，那就是一个异步请求。

好，这个时候我要给浏览器输出响应，结成字符串，这里我手动去处理 unique response response，点 site，content，看声明我要返回的数据的类型，application。

play表示的是普通的字符串，但是我们要确保这个字符串确实是阶层格式，那么前台才能够进行解析。另外的话再声明一下，点site，字符集好支持中文好。然后我们就通过response获得 Response字符流，然后向前台输出内容。

好我就 write杰森字符串，我们可以通过工具去拼 community，YouTube，然后点get杰森斯郡，那么当你没有权限的时候，服务器拒绝你访问的时候，而不是因为服务器报错，是因为你没有权限的时候，这个时候我们通常返回的是403的状态码，然后的话这里给个提示说。

你还没有登录，好。

这是异步时的处理，那么如果不是异步，我就直接重定向，我就用 response1.3的 redirect，然后的话request点Guide。

Context pass。点Guide。对context parse。然后的话加上我确认一下我们访问登录页面是不是烙印好像不在这儿在这儿。是的，就是说我们打开登录页面和我们提交登录请求路径都叫log in，我们默认的请求方式是get，所以说访问的是 Log in，没有问题。好了，没有登录时就处理好了，然后权限不足时的处理，其实跟他类似，也是要分开。

你 x的是返回json还是返回html分开处理，我考虑一下，当然这个参数名改一下，不然的话太长，这个方法也是给了你传递来了 request的response，还有一个异常里封装了相关的内容。

好，这里我们也是做一个判断，然后取header中的参数x-requested-with，然后判断它是不是等于这个值，如果是返回json的，这个时候我的提示要进行一个修改，改成说你没有访问此功能的权限。

给个明确的提示，否则的话就要重定向，那么这个时候我们重定向到没有权限的界面，而不是登录，因为如果说他走到这儿了，说明他已经登录了，但是他权限不足，你没有必要让他再重新登录一遍。

这个时候权限不足的页面，我给他再来一个路径叫denied代理。

当然目前我们的项目中还没有处理这个路径，我把它处理一下，这个简单我们可以copy，我们上次课那个小案例中不是写过吗？在home control到最后这个就可以对吧？把它copy过来，我们现在也是这样，如果你没有权限，我就还是跳到404的错误页面，就当没有。好，回过头来我把这个代码也是。

拷到后面看。

抽了里这就行了。

好了，那么我们继续配 Secretary，有人说这还要配什么？不是已经配的差不多了，还有一个细节小的细节需要处理一下，那么这个secret它框架它管的比较多，然后关于退出他也管，而且默认情况下它就会自动拦截名为log out的退出的路径，他认为log out的路径就是退出，它自动拦截，然后做了处理。

因为我们知道雪块它的底层都是通过filter去拦截去做权限的管理。那么 Filtered的代码的执行在dispart zirlite之前，肯定是在comptroller之前，所以如果说它提前拦截到了log out，那么帮我们做了退出的处理以后，到此结束，他就不会让程序往下走，所以后面我们自己写的log out就不会被执行。

现在我的想法是我想执行我自己的logo，我不想走他的那段逻辑，因为我自己都写好了，这个时候怎么办？这里我先写个注释，把刚才的这个话再稍微总结再说一下，省得忘了。

就是夸它的底层默认会拦截拉不到的请求，然后进行退出的处理，那么我们只要想办法把它这个逻辑给覆盖掉就可以了，我们覆盖它的默认的逻辑，那么这样的话才能执行我们。

自己的推出的代码。

如果你不做这件事，我们推出代码就不值钱，就很奇怪，怎么把它覆盖掉？

其实很简单，Http就是我们怎么去处理登录相关的内容，是通过form log in，它返回一个与登录表单有关的对象，然后去对表单做处理，对吧？退出的话是HTTP点 log out，返回一个和推出有关的页面，这里边是对它进行配置好。

我得到这个对象以后，我点log out URL，我把它拦截的请求给它改了，你看一下 log out，下载一下源码，这是改 URL你看它原来的 URL是几是多少 log out，是吧？

所以它的底层就会拦截这个路径，我把它改成一个别的，他就拦截别的，但他就拦截的路径，其实我们不处理就可以绕过这件事。这里我随便写一个，比如说我叫security老高的，好，我们程序中有这个路径吗？根本就没有，我们这只是一个善意的欺骗，然后的话好让程序执行到我们的方法里去而已。

好了，那么现在我们基本上就把色块的k给它配好了，配好以后，当然我们配的主要是授权，而认证就没处理，认证它会走log in，control了我们自己的认证。

但是你注意有一个问题，我们上次课写的认证的逻辑，你看我们在色块的框架下写的认证的逻辑是怎么样的，就是这样的对吧？

最终认证完以后，我们会把认证的信息封装到一个token里，那么这个token会被so quiet的一个filter获取到，然后 Filter会把 token存到secret contacts里，那么后面他进行授权的时候，他进行判断你有没有权限的时候，它都是从secret contact那个对象里得 token，然后看 token来判断你的权限，这是10块的它的底层的逻辑。

你想我们现在走自己的认证，我们有这个东西吗？如果你没有这个东西，security它是没法帮你去做授权的，因为他不知道你的权限到底是什么，所以我们是绕过了他真正的逻辑，但是那个结论我们还得想办法存到是quality context里面怎么存，因为我们已经绕过了它的认证的逻辑，所以说我们就没有必要像上次课那样，我们优质实现一个接口，用这service实现一个接口，没必要那么复杂，但是我們也需要做一些必要的处理，怎么处理？

我先打开 User service，我想在这里再加一段逻辑，加一段什么逻辑？

就是你想我们最终要把用户的权限存到色块的抗txt里，首先你用户你这个用户我查到以后他有什么权限，我们还得做一个就相当于做一个适配，根据获得用户的权限方法你至少得提供出来，我们得用是吧？这个方法我就直接在 Service里加一个，当你需要的时候你就调一下就可以了。好，我加一下这个方法，别的都不需要了， public然后用户的权限 collection extends granted，然后的话of reading。

好，然后报上名我就要get奥斯瑞斯好了，然后这个方法我希望你调用的时候给我存入邮政ID。就是说你希望获得哪个用户的权限，你把一个u在ID传给我就好了。那么这个方法的实现也比较简单。首先的话我根据优质先查询一下优质ID，然后通过根据优质ID查优质，然后通过user的type去判断它到底是什么权限。

查一下user。

等于当前的service里就有这个方法，find by ID Uzid传入，好，我得到这个u字以后，要对它的type加以判断，而判断的结果要存在一个集合里，所以我这里还是先实例化一个集合，最终我返回的是集合。

好集合中的数据我添加一下，我们的系统里每个用户只有一个权限，所以说叫爱的一次。然后的话艾特的是实力，我就new granted。

好。

我把它里面的方法给它实现了，这个方法的实现我们主要是根据判断这个态度，然后的话去返回一个就是权限的字符串，好我就因为tap是整数，我就switch，然后的话 user点get tap。

好判断，那么如果说它是一的话，我就return常亮，这常亮估计是没实现接口，实现一下还实现了那就省事了。Return authority优点。如果是不对，如果一是好的妹子管理员，然后case二，如果是二Return版主moderate好，然后的话，否则默认我就。

委托 user。

就是普通的用户，当然因为我这里默认就返回了值，所以这个return now就不再需要了。

好了，那么 user service我把查询某个用户他的权限就做好了，做好了准备。

做好准备以后关键是什么？关键是我们什么时候去就是获得用户的权限，并且把用户的权限的结论Token存到content，其实也不难想象，大家想一想，我们之前。也是做过显示用户的登录信息这样的事情，那个时候我们就判断了用户的。

凭证ticket。

根据一个凭证的判断他是否登录，我们做过这样的事情对吧？

然后登录以后怎么样？

所以说我们登录成功以后会怎么样？会产生一个结论，产生一个tk的让用户去存，然后用户每次访问我们服务器，我们服务器会去验证 txt，看你过没过期，看你对不对，最终有一个结论看你登录成没成功，所以说是在那个地方在拦截器里，我们就知道你登录是否有效，如果你登录有效的話，那就意味着你是一个有权限的用户，这个时候我再去查你的权限，再去存到对象里就可以了，后续就能够用它来授权了，这个地方就可以了。

所以我需要打开我们当初实现逻辑的 intercept，叫做love叫做 login。Take it, intercept。那么你看在这个地方，在这我们查询了凭证，然后判断了凭证是否有效，那么如果if内部的程序执行，就比方说有效以后，我们把user存到了 House toda里，后面我们好能够得到u的相关内容，然后我们之前不也用出色块的康txt，厚德厚德主要是厚德什么，主要是保存什么？

保存的是色块的康txt，就是保存认证的结论，所以其实它的功能和 householder是类似的，只不过house透的存的是普通就是用户的信息，而色块的康txt存的是认证的结果是类似的，所以我们在这儿去存就可以了，我这写一下在这我要干嘛呢？构建用户认证的结果，然后并存入 secretary contact。

当然我们通过什么存通过的是他。好的，康txt，那么我们存的目的是以便于10块钱进行授权好。因为我们没有认证，所以说越过了那一步，我们需要把这个结果自己存到这个地方来，后面授权才能进行下去。



好了，接下来我就创建一下认证的结果，认证的结果是存到一个接口的，叫奥森贴可以选，那么它这个是接口它的实现类，如果是用户密码认证的方式的话，那就是user name, password, authentication, 然后 Token里边我们通常是纯三个数据，第一个存的是user，用户存的是用户，第二个我们通常存的是密码。

Get password你可以参考我们上次课所写的事例，它是怎么构造的。

好，然后第三个存的就是它的权限，就是user service。我们得查了get authorities然后把 ID传进来，ID就是user的get ID，好，他这报错我们看一下，他大概的意思是说其实这个类型不匹配，估计是我引错了，因为你看我放在上面它提示我什么呢？他提示我他需要的是这个类型他们开的什么，但是我们其实这他们跟他没什么关系，这是security对吧？

估计是不是引错了，因为很有可能这个单词很多包下都有透明的，看一下是不是这样，你看确实引错包了，当时没细看，重新引入一下，好，这回就对了。那么我们构造完认证的结果以后不用再删了，没必要购置完以后，我需要把它存到色块的context里。

刚才也说了，所有的context我们是通过所有的context后者去处理的，我就直接访问 Security contact order，然后的话点site，context这里面需要传入一个手快康txt，我就实例化一个 new。

squire contact impl因为它是一个接口，那么这个是它的实现类，然后你拗的时候要把凭证传进去，存进去就可以了。

好，那么这样就行了。我们在每次请求的一开始就做判断，如果说你登陆了，那么我就得到了用户，并且我就得到了用户的权限，存了后边的话授权就可以进行了。

然后的话你别忘了，最后到我们请求结束的时候，你看我们 house偷的都已经清理了，我们保存权限的逻辑其实也可以做一个清理，所以我们把它清理掉。Security context后的点，它有个可列方法是把这个数据清掉。那么这个地方处理完以后，咱们还要注意另外一个地方，就是我们整个程序中不是还有一个退出的功能吗？你退出的时候是不是认证的结果也应该清理一下，对吧？

所以下面我再打找到退出的功能，把它也清理一下，那么它在 Log in很多了之内，log out。当然我们退出的时候先调了 Log的方法，那么接下来我们再去清理康泰CK二，contact偷的肯定可以了。

好，现在我们 security就配置好了，最后还差一个小的环节需要进行一个配置，就是 csrf也需要配置好，然后最终我们再进行一个统一的测试。那么关于CS2f我先说一下什么是CS2f它的基本的原理是什么？他这种攻击的模式原理是什么？那么我们预防的话又是用什么样的方式来预防，什么原理来预防？

我需要画一个图，首先画一个方块，代表你的浏览器，这是你的浏览器，那么你的浏览器这是服务器，这是我们的服务器，比如说你的浏览器之前已经访问过服务器，他已经登录过了，他登录过以后是不是服务器已经给他发了一个 ticket对吧？然后它已经存到了cookie里，所以此时浏览器已经保存了 Txt在库给你存的。

好，那么比如说这个时候浏览器又向服务器发出了一个请求，这个请求比如说是一个是一个盖的请求，但这个盖的请求是打开一个带有表单的页面，那么接下来用户就是要提交表单的数据了，但是他得先打开这个表单页面，他发这么一个请求，这个请求我标一下，这是一个让他放吧，就这个请求是希望你返回一个带有表单带有form的页面，那服务器就给你返回这个页面，这个没有什么问题，你访问它就给你返回，返回以后这个时候比如说按理说用户应该是填这个表单，然后的话提交，但比如说这个时候他没有直接去填，没有直接去提交，而是做了另外一件事，做了什么事，他这个时候把网页停在这，他去访问了另外一个网站，这个网站就随便写，比如说某网站这个网站。比如说它是一个不正经的网站，不好的网站，然后里边是有病毒的，那么用户访问了这么一个不正经的网站，这个网站通过病毒窃取了用户的cookie，cookie的话，其实存在客户端是不安全的，它是能够被别人给窃取到的，那么你想他窃取了你的cookie，而 cookie又是你 cookie中的ticket，又是你身份的标识，所以这个时候这个网站就可以就是模仿你的身份，伪造你的身份去访问这个服务器了，而这个时候你的表单还打开着，那么他也要提交给谁，所以这个时候他就模拟你的身份，然后向服务器帮你做一个提交。

当然提交表单通常都是post请求，他帮你做了一个提交。

好，那么如果这次请求提交的是一个转账的相关的数据要求转账的话，你想想你的账户是不是就很不安全了，对吧？这个时候你的数据你的钱，你的什么经验，你的道具，你的什么东西都有可能被人窃取，这是一个很危险的一件事。

总之这就是2f所谓的CS，2f攻击某网站，他盗取你cookie当中的凭证，然后模仿模拟你的身份去访问这个服务器，并且利用表单向服务器提交数据，来达到一个不可告人的秘密来获利，这样的这种攻击都是发生在提交表单的时候。

好怎么解决方式也比较方便，说一下secret的解决方案。你不是窃取了txt吗？

Sql是这样的，他这样你访问服务器服务器给你返回带有表单的页面的时候，那么色块儿会顺带着在这个表单里生成一个隐藏的数据，隐藏的一个凭证，tk的不是tk的是to k。

当你引入了secret以后，security默认会启用，防止csrf攻击，这个时候你再去访问服务器，要求获取一个带有表单的网页，这块儿也会在这个表单里给你带一个隐藏的数据，这个数据里会包含一个token一个凭证，所以你的表达里就有一个token，而这个token每次请求都是不一样，是随机生成的，那么某网站它能窃取到你的cookie，但是他没有办法猜到你的token，所以当他提交post请求提交数据的时候，服务器一看take的对，但没有偷看意味着什么？

这就是伪造的身份，就是有问题的。而你提交的话，因为你的表达里有这个偷看，一提交给服务器，服务器一看头跟对头，跟对，身份没有问题。

好，这种这就是色块的它底层的一个解决攻击的一个方式，大家了解一下。好，那么这样现在我可以把这个项目启动一下，咱们看一下这个token，我们现在它已经能够帮我们生成tokens了，我们看一下直观一点，我启动一下这个服务。

好，启动以后打开浏览器，然后访问首页，访问以后。

这样比如说我打开某一个帖子详情页面，我打开有玄学帖，打开以后你看这个玄学帖它里边不是有这不有表单吗？对吧？我说我们现在在我们引入色块的前提下，我们再去访问一个带有表单的页面，这个表单里会自动生成偷看，有吗？怎么看？我们可以右键查看网页，源代码看一看，这个是服务器给我们返回的html对吧？看里面有没有Token，它在比较靠后的位置，在福特之前这就是一个form，你看form的后面紧跟着form是不是就生成了一个隐藏的黑的，它的name叫csrf它的value是一个随机的字符串，所以这个csrf就是CRT帮我们生成的，防止csrf攻击的一个凭证，这样的。

好，但是还有一些情况他没有办法处理，就是异步的时候，因为e步的时候可能你连表单都没有，是我们从页面上硬提取的数据，然后单独写了个请求发给了服务器，你可能连表单都没有，他怎么知道给你生成CS2f凭证呢？对吧？

所以如果是异步请求，这个就需要我们自己处理了，我们再把异步请求处理方案给大家演示一下。我就演示这个发帖，因为发帖的时候它就是一个异步的是吧？好怎么处理呢？首先我先找到发帖他所在的页面，那就是index点Excel，我们是在这个页面上点发帖的对吧？

Index点x前面，既然你可能没有表单，有可能没有表单，我Csrf生成在哪，我换个地方强制它要求它生成就可以了，我可以在这通常是在还得在这，通过麦特标签要求服务器把csrf凭证生成在header里，然后我们发请求的时候从这里取，先存在这，这里写个注释就是访问该页面时，然后在此处生成csrf。

令牌。

好。

怎么写呢？Matter，然后的话它有一个name，搞错了。name等于-CS2f然后当然这个要生成的话，生成值我要生成到content的属性里，那么我们需要通过变量来解决，因为每次都不一样，所以说得用timeless的语法，然后这里面写什么csrf点投给。

就是说当我们引入secret以后，那么每次请求都会有一个c请求当中都会有一个 csrf对象，我们可以通过它获取一个头盔在这里显示，然后还要再生成一个东西，异步请求的时候，我们发请求的时候，因为没有表单，我们怎么去把这个数据传给这个服务器，可不是通过请求体，它是通过请求的消息头传，消息头是一个key一个value，而这个token是个value，它的key叫什么名字，所以我们还得获得 key，所以这个 name等于还是杠csrf但是没完还杠黑点，这是请求消息头的 k th content等于是csrf点hi的。

内容。

好了，总之当我们访这个页面的时候，那么 secret它就会在这两个matter里生成csrf的key和value，在matter的count那里生成。那么我们发一步请求时，把这个标签里的值取到就行了，这样就行了。好，我们发布帖子的时候是在哪儿发的一部请求？是在当前页面所对应的js里找一下index点GS特别是 Public方法对吧？然后我们可不是在这里直接带上这个数据不在这里去传，而是在请求的头里传。

我们如何去改变请求头，在我们发请求之前，在这写发送

ajax请求之前，我们要将csrf令牌设置到请求的消息头中，提前做好一个设置，那么发的时候就会带上这个消息头了。

好，首先我要把刚才的那两个值取到先取 token到了括号，我们用几块选择几块选择器，那么我们要获取的是matter，元素我就写matter，然后元素的属性，name、属性，固定的name等于大csrf这句话选题就这么写，意思是我要获取迈特元素 name等于CS2f的迈特元素，我要取这个元素的，不是取整个元素，是要取它的属性值。

Attrh的缩写取属性值取什么属性？Content属性值，这是头k好，再选 name， header，然后 header matter name等于csrf杠 header。

然后。

也是取它的 content，好，然后我们再写一句话，dollar document，然后点 send。

这句话的意思是在发送请求之前，我对整个请求做一个设置，对请求的参数做一个设置，这里面需要给一个匿名函数，函数需要写上三个参数，exh二还有 options，其中 xhr就是发送异步请求的核心对象，我们需要通过它来设置一个header，那就xhr点site request，header好，设置请求头头的名字是header，key是header值是头肯，这样就行了，你这样的话你发请求以后发请求的时候，那个请求你就会携带这个数据，这样我先把这句话注掉，我发完请求得到响应以后不刷新页面，这样的话我能看到本次请求的情况，我看看这里有没有这个东西，然后没有问题我再把它再取消注释。

好了，那么做完处理以后我们重新编译一下，他没重启，有的时候他感知的不那么敏感，我就干脆强制他重启一下。

好了，那么重启以后咱们重启以后我们试一下，我还是访问首页，访问首页首页以后右键查看源代码，先看一下，你看这回我们首页上是不是有这两个麦特名字，对应的名字内容，这是随机的，字符串这个是t对吧？没有问题。

好，然后接下来我要发布书，发布帖子了随便写一个啥，我这样发布之前我先右键检查一下，然后点 net work，然后我们发布的时候请求在net word里就能看到，我就可以去观察它有没有传这个数据，然后发布成功以后它没有刷新页面，我直接看你看爱的请求点看，主要是看请求的消息头，这是响应的 request。

你看这里边都是k86k86，我们设置的不就这个 csrf token，而这个值是取自麦特标签里，对吧？这就没问题了。

好了，所以说正是因为有了这样的一个限制，那么就可以避免 csrf攻击，当然了好使以后我得把恢复一下还是那刷新页面，但是我们现在已经启用了CS RS攻击，我们每一个异步请求你都得这么处理。

如果你不处理服务器得不到 token，这个时候他就会认为是被攻击了，他就不让你访问了，所以你哪个都要处理一下，但是我们这个程序中还有挺多地方是异步请求的，那么挨个处理的话比较麻烦，所以说为了图省事我怎么办？

反正我把这事讲清楚了，我把它注掉了，我就不往下接着演示了，我好让我这个程序能够执行下去，有时候你注掉以后你这都没有凭证了，你怎么去怎么去让他通过授权这个认证检查好办，我们只要在手快的肯费格里禁用掉，CS 2f就可以了，禁用他就不走这个逻辑了就ok了。

但如果你想走，你挨个处理就好了，你可以课后自己处理，我把它注掉了。

另外 index.html我也把它注点，其他的页面我就不演示了，就这样了，然后回到 config，下面我要让他不启用 csrf攻击的检查，怎么办？

我可以在这儿在授权的时候，在这儿再加一句话，不是没谈好以后再加个，and然后点CS二f点等于c宝，那么它这回就不会像页面生成凭证，也不会去做相关的检查了，就这样。

好，那么现在我整个都配完了，配完以后我们做一个完整的测试，再重新启动一下。

好，启动以后我打开浏览器先访问手机刷一下，然后退出，现在你看我是没登录，没登录，你看我能不能访问首页，可以注册登录页面可不可以访问没问题对吧？

这些不控制的请求是可以访问的，比如说我要硬敲一个路径，light list你要知道这个得是登录以后你拥有任何一个身份的时候才能访问，但是我现在就硬访问行不行？

回车不行强令你登陆对不对？好再来个notice，list也不行再来。

User setting还不行，你看我们刚才所配置的这些个路径，我随便敲了几个，我想没登陆的时候强制访问，不可以说明这段代码是有效的，我们授权是成功的对吧？

没有问题，好我就登录，我用CC登录，然后登录登录以后我再去访问刚才的功能，想消息通知点报错了，说明可能是以前我们写的哪些逻辑，可能有漏洞，或者可能是我们做的权限的控制，有一些代码产生新的问题，我们得去看一下，出了问题以后我们首先看一下客户端有没有报错，很显然他报错了，这服务端报没报，报错了，我们看一下这个错误信息，往前找从头开始看，那么看错误信息的时候，你不用说整段去阅读，你像什么艾特time leave等等这些代码，就是说报说time leave里报的时候基本上是不可能的，或者说极难遇到的，就是说它底层框架报的错，一般的问题都是和我们有关系，我们主要是看第一句话，它有它的提示，每一段的第一句话的提示，他说你看他说这个模板里边有问题，解析模板的时候有问题，什么问题不知道，我们再往下看。

还有别的段落，你看这一段说cosplay什么原因表达式？12表达式，就这个表达式有问题，表达式有问题。好，我们再往下看这表达式有没有问题是为什么有问题？

我们再往下看，下面还说这个表达式有问题，你再往下看，有没有更明确的提示到这有你说他说property或者是属性message找不到，在map里找不到message，好，然后你看我们是follow notice点message，如果说假设my follow notice，它根本就不存在，或者follow notice它就有问题，我们当然就找不到mic。所以说我们得去检查一下follow notice这里写的是不是有问题，我们看一下。好，我们就打开什么呢？打开通知我们打开Message control了去看一下，然后我们访问的是not历史的，我们看一下。

好，就是这个方法。

Follow notice它是一个message will，那么你看我们之前写的确实有一点有不太严谨的地方，你比如说我现实六化的希迈克，但一旦说我们恰好就没有这一类通知，有没有可能有这种情况，确实这个人谁都没有关注，对吧？

那么通知的数据就没有，所以这句话不会执行，不执行以后我们就往包里装了一个空的希迈克，所以你尝试从空的希麦克里取麦克取不到对吧？

就这样的一个问题，这个问题怎么解决其实也好办，比如说我这样，但是这个问题其实不是他这一段代码的问题，我们这三段代码都是这样处理的，都有问题，我们都重新捋一遍怎么处理，把这句话放进来。如果说 Message不为空的话，我再去实例化 map，最终我再去把 map存到model里，如果为空的话就算了，对吧？这样的话就能解决这个问题了。

好，所以第二段也得这么处理挪下去，但这个得实例化一个map，然后 model挪进来，这句话也是这样的，model挪进来，那么 controlled代码改完以后，我们还得最好看一下 html，看看html上有没有什么问题。

那么我找一下notice点html，从上往下看，我们最开始对数据的判断是在这儿，我判断了如果max不等于我就进行下一步，这里应该是这样写。

Notice不等于not，如果这个notice等于not，就意味着没有数据不等于那我就做下面的显示，否则就不显示同理这个也做一个处理，这儿也做一个处理好了，这三个地方处理完以后，我们再试一下，我再强制重启。好，启动以后再试一下。刚才我是登录了是吧？

我先访问首页，登录中还有，然后访问消息，私信可以通知也能看到，而当前这个人确实没有关注的通知，所以说它是一个报问报错，我们之前可能是开发这个系统通知的时候，那段代码写的不够严谨，所以这里面出现了一些问题，稍微改了以后就ok了。

好，然后比如说我现在登录以后，我能不能访问账号设置可以的，能不能改个头像？没问题。

对，就发个发帖子没问题，点进去，比如给帖子点个赞，回个帖，可以给回帖点个赞，给回帖再回个帖，再给回帖点个赞，没问题，然后比如退出也ok，好，就是没有登录的时候，我们那些功能不能访问我，登录了以后就都可以访问，然后顺便还改了个bug就ok了。

但是现在我们用户的身份还比较单一，我们都是普通用户，他做的操作，后面我们再开发一些管理员，还有版主他们特有的功能的时候，我们就能看到更丰富的一个授权的状况，以后再说了，我们先进行到这里，好。那么这次课我们就演示到这里，咱们下次课再见。

## 7.3 置顶、加精、删除

45-置顶、加精、删除.mp4

这次课我们来实现置顶加精和删除的功能，那么虽然说这是三个功能，但是他们在实现的时候非常的相似，所以我们可以一并把它们都开发完，那么我们在开发的时候主要是从三个方面来写这个代码。

第一个是功能层面的实现，也就是说当我们点击置顶的时候，我们要修改帖子的类型，然后把它设置为置顶的类型。

当我们点加精或删除的时候，我们需要修改帖子的状态，把它修改为对应的状态，当然这三个操作是针对帖子而言的，那么基本功能开发完以后当然很简单，主要就是修改，开发完以后，我们需要对这三个功能做权限管理，因为这三个功能显然不是普通的功能，一般的用户是无权访问的。

我的要求是这样，就是说版主这样的拥有版主权限的用户才可以进行置顶和加精的操作。

那么只有管理员才能进行删除的操作。

因为我们之前上次课已经把 security融入到我们这个项目中来了，它已经能够起作用了，所以说要想管理一个新的功能的权限，我们只需要在色块的KFC里进行一个配置，几行代码就可以搞得定，所以也比较容易最后那么这三个按钮我们在页面上显示也需要加以处理，你普通用户操作不了，你就连看也不让他看见对吧？

那么你最好是做这样的设置，版主能够看到置顶，将精管理员才能看到删除，那么其他的人都看不见。

那么这个按钮的显示我们因为是要通过权限去确认它显示与否，怎么去在页面上做这个事，那么这个 time leave可以给我们提供支持，因为模板上的事情由来解决。

Time leave它是支持 secret，它里边有一些标签能够支持 security，不过这是他们内部后来在某一个版本升级出来的功能原来是没有的，所以说你要想用这个功能，我们需要单独导这个包。

再一个我们从他们官网的手册里，他默认没有这方面的详细的描述，没有这方面详细的文档，我们在这个地方社区里能够找到一个小节，有一个大概的介绍很笼统，然后他这里建议你去 get up上去找他这个项目，那里边的话有稍微详细一点的介绍。

所以说如果你想去看这方面的文档的话，我们去get up去看好了，那么大概介绍到这，首先我们第一件事，因为我们要引入一个新的技术，所以说我先给大家看一下这个内容，找一找手册的位置，再一个我们把包引进来，然后我们就按照这三个方面去做我们这次课的开发，首先我先打开给他，然后去搜一下 timely色块的相关内容，我就直接搜 time leave，然后的话是quite收到以后就是这个东西 time leave，exercise，然后 spring security点进去，它这里边有好几个版本，34，那么组件模块的版本，它和色块的版本是对应的，如果你的色块的版本是5，你就用这个5，你是4就用4，我们其实用的是5，我们的 spring boot，它底层的pom声明的版本是5，所以我们用这个当然了，所以我们一会去搜包的时候搜的是第五个版本，然后后面他有一些关于这项技术的一些介绍，你可以去看，后面有一些小的实例，有一些代码可以去看。

好了，那么我们就不要在课堂上详细看了，我们要做的事情是把5这个版本的包找到，然后引入到我们项目中来，那么我就打开没本repost到com，然后搜一下我搜time leave，然后是不远 security。好，那么就是它了点进去，然后选择一个版本拷一下，然后把它粘贴到我们项目中来。

好这个版本我可以把它去掉，因为我们的副炮已经声明了，版本他用什么，我们默认就用什么，没有必要追求一个最新的版本。

好了，这个包打完了打完以后，那么接下来我们就按照这三个环节去开发，当然我们在最后一个环节才能够利用上 timeless这块的相关内容用的时候，我们再看它到底是怎么用的。

首先我们先做基本的功能开发，先实现置顶加精删除的基本操作，那么在实现的时候也是我们先来开发数据访问层，提供几个修改方法，然后开发业务层也是提供这么几个方法，然后 CTRL处理相关的请求，然后页面我们把展现的部分处理好，按照这么一个流程来，这没什么好说的，我们就直接来写这个代码。

好，首先我们先来开发数据访问层，因为是对帖子的操作，那么无论是置顶加精还是删除，你最终是要修改帖子，所以说那么我把我先打开迪斯卡斯pose的map去增加几个修改的方法，那么这三个操作一个是修改类型，帖子类型，一个是修改状态，所以说我需要加两个方法，update态度，然后你需要传入ID，根据ID来改，类型类型也是整数态度。

好再来 update status，根据ID改这个状态，那么声明完以后，我们需要打开对应的配置文件，然后它把circle写好搜一下。

对。

discuss postmarital，xml，那么我在最后加两个update语句，第一个是改这个类型 tap，然后是 update call一下不对的表site，tap等于参数 tap，然后 when ID等于参数ID，这就完了，非常容易再来一个，第二个可以照 copy。

另外是update status，然后这个是site，status等于参数，status，条件也是yid等于ID好这就行了，完成以后下面我们就在 service里提供这样的修改方法，然后好让表现图去调，然后再打开 discuss post service。

在这里我再加点方法，先是update tap，参数是ID，然后的话另外一个tat，直接return discard。Post迈克点update。Type把参数传进去就可以了。

好，另外再来一个update，这点事。

ID。

实在的直接委屈，discuss postmental的update，staters。好，那么业务层写完以后，下面我们开始写表现层，首先处理 controller，那么因为是帖子相关的逻辑，所以说我找到 discuss post control了，在这面我再加不是一个，其实一共要三个置顶加精删除三个请求，我们一个来先处理置顶的请求。

然后先声明一下。

路径，置顶我叫Top，然后 method等于request，request。

Method点，post。

因为你要想置顶，你需要提交数据进来，所以说这是post，然后请求我们用的是异步，请求就是页面上。

点完这个按钮以后不整体刷新就可以了。

然后所以说我需要加上 response包点注解，那么这个方法返回10句方法没有叫site，Top设置置顶，那么页面上需要传入的参数是什么？其实主要就是帖子的ID，你传入帖子ID，我根据ID把类型改成置顶的类型就可以了，就需要传输这么一个值。

那么实现也非常的简单，就是调 Service，迪斯卡斯pose的服务，然后的话点不对的 tag，把ID传入置顶的类型是多少，咱们可以从表里看一下。

这个得搞准确了，我们见表的语句里头写得很清楚，解释点一下。Ddl语句零普通的帖子默认的一就是置顶，所以把它改成一就可以了。你别忘了这里，我们虽然说做的操作比较简单，但毕竟帖子发生了变化，帖子发生变化我们需要做什么事呢？我们需要把最新的帖子的数据进行一个同步，你要把它同步到 Certainly，对吧？因为什么？我们好搜的时候能搜到最新的帖子，最新的数据，所以我们需要再触发一下帖子发布事件，就像我们发布帖子的时候这样再触发一次这个就发帖事件，我可以copy一下，好好。

好，然后粘过来，看一下哪些地方需要改，首先new event，然后 Topic，public没有问题，优质ID这个肯定是当前的用户的，我就house偷的 get ID类型 post没有问题，然后 ittid就是帖子ID就完了，然后触发一下可以了。

好，那么发布完以后，最终我要返回一个数据给一个提示成功就可以。

那么return community又退化，get精神给个0。

成功的状态就完了，这就置顶。

那么加精和删除其实和类似下面，所以我直接copy，然后在这基础上改，这个是加精的路径，我叫做 wonderful，然后也是pose的请求，那么这个方法名叫site wonderful。

好。

刚才刚才我写错了是update time对吧？我这怎么写上update status？置顶是update time。好，然后加精的话是update，status倒没有问题。Update status加精的话，这个状态应该是几？再看一下，状态默认是零正常，一是加精，二是拉黑切入删除，所以说这个也是一没有问题。

好，这个逻辑就是这样的，然后也是你加进以后帖子数据发生变化，你也要触发发布事件，那么相当于把数据重新的同步到 Es里去，这段话和刚才是一模一样不用动，然后最终我们返回一个状态就可以了，家丁很快就实现完了，最后还有个删除也copy好删除，那么路径我叫德丽婷，也是POS的请求，那么这个方法我叫set，delete就设置删除的状态，传输参数也是ID，那么删除态可不是一了，是二刚才我们看了拉黑的状态，然后那么帖子删除以后怎么办？

我们就不是触发发帖事件了，你不应该说把这个帖子在更新到 es里，因为es里存一个删除状态的帖子没有意义，我们应该是从es里把帖子给删掉，这样搜就搜不到，它应该是这样的。所以我们应该触发一个删帖事件，但是我们现在可能还没有删帖事件，我看一下有没有这个类型，你看没有，我们需要再加一个删帖的主题，写一下删帖，delete。

好，再回到刚才的 control里，这个地方改一下改成三体，然后也是当前用户，也是帖子的ID触发这个事件没有问题，然后你别忘了，那么这个删帖事件因为是我们新加的事件之前没处理过，所以说我们在事件的消费者里还要把这个事件也去消费一下，所以我们找到那个事件的消费者in mind12码。

其实删帖时间和发帖时间的实现逻辑非常就差不多，我们可以copy消费，删帖事件 topic。改一下别忘了。Delete，然后憨豆 believe to message。好，然后前面两段先判断这个参数是否为空，得到英曼的判断，英曼的是否为空，这个是可以一样的。

然后后面这个地方就不是safe，而是delete，所以说这句话不要了，我就直接 elastic search service，delete，discourse，post service里得力的方法，我们之前就已经写好了，这里需要传入帖子ID，我们可以从e MAC对象里获得 in one的点get int ID，这就是好，这就完了，那么现在我们就肯抽了处理请求的层面已经完成了，最后是要处理这个模板。

那么我们置顶加精删除的操作是在帖子详情页面上去发起的三个按钮在那个页面上，所以我们需要打开帖子详情页面，discus detail点steam。

好，那么就在这个位置，这三个按钮，首先你看它上面没有直接定义事件，其次上面也没有加什么ID什么的，说很有可能这三个按钮就是这个事件压根现在就没有做任何的处理，我们确认一下，我们看一下当前 html它所依赖的GS里有没有对这个按钮的处理，如果没有的话，我们需要重新来写。

好，我看一下，那显然是迪斯卡斯点GS了，搜一下迪斯卡斯点GS你看这里只有点赞，其他什么都没有，所以我们需要自己把这三个按钮事件定义好，把整个请求的过程处理好。这个代码太多了，我关掉一下。好。首先为了定义事件，引用这个按钮，方便我给它加一个ID，当然你也可以直接这样讲on clean，或者我们直接通过纯GS的方式，在页面加载完以后，然后动态的给它绑定时间也可以，我采用后者。

Id这个ID就是置顶，我叫Top置顶的按钮，Id等于加精的按钮，我叫wonderful body，然后删除按钮。

搞错了。

等于。

delete。

body。

好，那么不光是要 ID加ID是为了我们一会儿定义事件方便，那么我还要做一件事，我在这三个按钮之前加一个隐藏框里存一下帖子ID，因为你想我们无论是点置顶加精还是删除，我们都要向服务器发一个异步请求，请求传的唯一的数据就是帖子ID，我从哪得到这个帖子ID对吧？我干脆就在这写一个黑的，然后后的话里边放上帖子ID就完了是吧？

好写一个太太黑的，然后的话给他一个ID，我方便通过GS获取它，这里存的是帖子ID，所以我叫post ID，然后它得有一个值。

这个值我们可以通过就是帖子详情的这次请求里我们是传过来了这个帖子的数据的，所以我们很容易得到帖子详情请求里我们传过来的帖子数据应该是什么，你看这都是POS的对吧？

所以这个地方就是post点ID就可以了。

好了，下面我就去 js文件里，根据 ID给这三个按钮分别定义事件，然后去发异步请求，通常我们是在我们要写一个逻辑出来，页面加载完以后就说明html标签都已经加载完了，这个时候我用GS得到这个标签给它绑定动态绑定一个时间，一般这么做。

写这句话，到了括号里边写一个匿名函数function，这句话的意思和GS里的window点 on，low的等于function是一样的，就是页面表示页面加载事件的意思，这个函数是这页面加载完以后会调用的，页面加载完以后我要获取安全版权事件，我先获取 Top button。绑定一个单击事件可立刻。然后单击事件绑定一个函数，函数名写上去叫site。对，叫site。Top这个函数一会要写，不然就会报错。好，再来一个，然后不是点赞，是加精 wonderful button。Kelly可立刻单击事件，单击时调用set。



Wonderful。

好，再来删除得力发展，单机的时候sat得力。好，我是声明了说我点3千9的时候分别调这三个方法，但是这三个方法我还没写，于是我就写是吧？一个写先写在这里，好，这个方法非常简单，就直接发出一个一步的post，请求访问服务器传餐就可以了，那就到了了点post。

第一个参数第一个是路径康txt pass加上。

Discus。

然后是Top置顶的路径是Top对吧？然后第二个是我们要传的参数参数的k就是ID它的值看一下值这个东西。好，我们根据ID很容易就能得到这个框，从而得到它的值。我就到了括号 Post ID，然后得到这个框它的值点vi方法。

可以了，然后处理响应过来的这个结果是一个普通的字符串，但是满足阶层格式，所以说我们可以把它解析成js对象，到了点。

plus Jason这。

一段传进去，我们就得到了一个对象，然后对它作出判断，如果被它的扣的等于0，假如说成功了没有问题，否则就是失败了。

好，那么失败了怎么办？失败了给个提示，其他的就不用做， alert。Data，点message。好，那么如果成功了，你要注意我们要改一下按钮的可用性，因为如果我点过了置顶就不用再点第二次，那么我点过置顶以后，我就要把置顶按钮设置为disabled，不可用这样你就不能再点第二次，这样比较合理。

好，所以这个时候我要把置顶按钮设置为disa保函，怎么设置先选中它，是Top8天选中这个按钮点at1t2。那么主要是改这个按钮的属性AD，第二是改属性，改哪个属性呢？

Dc保的属性，然后把属性设置为DC保的就可以了， DC保的等于DC保的他就不可用了，这是置顶的逻辑，那么置顶你会了，其他另外两个方法其实非常的相似，我再拷贝一下再来一个加精方法名叫sat wonderful copy一下，然后提交的路径要改成wonderful，然后参数也是这么传，返回的结果也是这样去解析，如果是正确的时候，我们需要把加精按钮设置为不可用，所以说这个是加氢按钮，而嘉靖二者是他不可用，否则错误给个提示这就没问题了。

好再来最后是删除 site得力，然后路径是得力的，参数依然是这样的，处理的结果依然是这样处理，然后你注意删除完以后，我就不把删除按钮设置为这个不可用了，如果这个帖子删除以后，那么说明这个帖子已经没有什么存在的意义了，也不用去看它了，这个时候怎么办？我就直接跳转到首页就完了，怎么跳转？我们通过GS是这样写，location，点href等于 contact加上index。

好，这就可以了。

好了，那么现在我就把置顶加精删除，它的基本的功能已经实现了，实现以后咱们先测一下，如果没有问题，我们再对权限进行一个管理，那么我就启动一下服务，好启动以后打开浏览器，访问一下首页，然后我得先登录一下，我先用一个普通用户登录，因为现在没做权限管理，其实你用谁登录都可以，我们先不纠结这个事，一会再去做权限的处理。

登录以后比如说我要把做处理，置顶点一下，置顶以后它变灰了，但数据对不对？我们就要查一下表，看一看这个帖子是谁呢？叫QQ的帖子，我倒叙一下，因为他的时间比较新，倒叙，

最后一个。

QQ你看态度变成了一没有问题是吧？好再来我再加精一下，也变灰了，说明页面的逻辑没有问题，我们再看一下这个表。

状态变成一没问题，好再来我给它删了，以后它没有刷新，而是跳到首页，然后你看没了以后我们看一下这个状态。变成二是吧？可以的。好，但是现在其实有一个小的这个小小问题，我们看一下什么问题，我再来比如我点它置顶，置顶完以后，比如说我刷新一下当前页面，你看这个状态就不对了，为什么不对？是我点置顶以后，那么根据反馈值我对这个按钮做了处理。

但是你注意如果说我们刷新这个页面，或者说我通过列表进入到这个页面的时候，你是不是应该在页面初始化的时候也把这个状态改一下，如果当前已经是置顶了，你就把它设置为disabled，对吧？

所以这块得处理一下，不然的话就会出现刚才那样的尴尬。好。我先把它处理跟权限没有关系，只是可用不可用的一个——一个小的插曲。好首先置顶我这样写th冒号主要是要设置它的 Disabled的属性对吧？那就th冒号，disabled的等于我要判断一下post点PAD当前的类型它是不是一，如果说post点tap已经是一了，我就这个返回处意味着它就不可用，因为它已经是置顶的状态了，已经是置顶的类型了是吧？就这样处理。

同理加精也是类似的，th冒号对c波动这个得看是得到啥。Status等于一等于一意味着你已经是加氢的状态了，这个时候你就不可用，不能再次加氢了对吧？那么再来对删除也做一个处理。

Disabled等于然后boost点status等于2，那么如果你已经是删除态了，我们就让它不可用不能再点了，那么做完处理以后可能还不久重新编译，因为改动比较小，而且改的是模板，他有的时候感知不到，我强制重启，重启完以后重启完以后打开浏览器我再访问首页，然后你看这个帖子我点进来，置顶就是不可用点不了对吧？我点一下加精刷新一下还是对的，首页重新进来没有问题，好了，现在就是说我们先不考虑权限的情况下，功能的操作是ok的，接下来就是处理权限的问题，这个就比较容易了。

那么首先我要配一下权限，我需要找到 So quiet can figure，so quiet can think，那么配置权限的方式和之前这段代码是类似的，只不过我们再加一点就可以了是吧？我就在点at卖特斯，然后再点has any hospital，那么这里边我先设置置顶和加精，置顶的路径是discus Top，然后加精。

Discourse wonderful。

那么对于这两个路径，谁能访问是版主，而版主是这个moderate，你具有这个权限就可以访问这两个路径，好。

同理再来 And matters，然后点还是安利是瑞典。再来除了那两个路径之外，不是还有一个删除吗？

那就是disk us delete，

然后谁有权限去删除，好的妹。

好，这样的话就是这两个功能只有版主能用这个功能，只有管理员能用。

这样。

我们权限就丰富起来了，如果说我们的系统中占有更多的权限，那就好办了，比如说有运营的人员，有这个产品的人员等等，有测试人员等等，他们能做的事可能都不一样，其实都是类似的配置方式。

这是我们从这个功能上来说做的一个配置，这样配完以后，我们先不管能不能看到那个按钮，我们先试一下。

那么再重新重启一下，好，启动以后打开浏览器刷一下，然后我先退出一下，推出以后首页，然后比如说我随便再找一个我点置顶，没登录点置顶，你看因为我们是一个异步请求，它没有重定项，但是它给我一个提示说你还没有登录加精删除是吧？没问题，好我登录我用一个普通用户登录，这用户应该是没有权限的对吧？

好再来置顶。没有权限加精删除对吧？没有权限然后再来退出，然后扣的21，闹扣的21，它是他看一下我们表里原有的数据，测试数据不用倒序，它稍微比较靠前。

111213他们的眼影一样密码一样，其实都是123456，这三个是管理员。

2122232425这三密码也是123，这5个人密码也是123456，然后的话他们是版主，好，我就用21登录，版主登录123456，登录以后跪求offer，我能置顶吗？

可以加精可以删除，没有权限好再来退出，我用 now colder11这是个管理员，123456，为什么说他是管理员，我们可以看这，你看这5个人是二，那3个人是一是吧？

好继续，然后我用 Code11管理员登录，三KH二跪求阿尔法，你看我置顶将近已经点不了了，我先把删除点了，可以点对吧？

然后我再换一个，看我点置顶，没有权限加精没有权限，所以你看无论是我没有登录，还是我登录的时候用的是普通用户，或者是有一些特殊权限的用户挨个去测，他都能够准确的判断出我能不能访问这个功能。

好，所以说我们从服务端权限安全的角度已经到位了，但是页面上还没到位，就是说既然说你不让我点置顶和加精，你干脆别让我看见对吧？我们直接对于不能访问的那些用户，我就干脆让他看不见就完了，怎么处理？我们一开始所说的，我们需要用到他们对于此不认可的一个支持，需要用到它的一个特殊的标记，那么在模板上获得当前用户的权限，从而对按钮是否显示作出判断。

那么组件我们刚才已经包已经导进来了，那么我们在页面上怎么用？

首先因为它是一个新的组件，我们需要在模板上在顶端顶部也写一个类似于这样一句话，就说明一个 name，space命名空间，那么组建的面空间其实就有点不一样，怎么写还不好记，我们去搜一下。

在get up里头，这里有这其实就是一个手册，简单的手册，我往后找他，在最后的位置告诉你说你在这个html里边你需要写这个秘密空间是这样写的，所以你没有必要手写直接 Copy，然后把它粘贴到这里来。

然后你看我们这样声明表示说我 time leave的语法它是以th为前缀，这块是SEC比方说命名空间里的语法是 SEC为前缀，表示secret安全的意思。

好，具体来说我们在按钮上在这个模板里头，在标签上怎么用，我们再找到那三个按钮。

首先我先处理置顶，置顶我是希望只有只有这个版主才能够看到这个按钮，你就这样 SEC以SEC开头冒号，然后 authorize表示这个权限等于里边。

has any。

奥斯瑞铁然后写上一个权限，我们的权限就是moderate。

说白了这句话和 can figure里逻辑是一样的，can figure的逻辑不就是你有什么has any also reading，然后里边写一个字符串，对吧？当然我们在页面上如果你有多个值也是逗号，写多个值就可以了，has any else ready？这里就一个就是版主有权限。好同理加精也是类似加精的话也是版主可以操作，所以一样的。然后删除类似删除的权限是管理员在那里，这就可以了，所以我们通过这种方式也能够起到像服务端配置这样的效果，但是它影响的是元素是否显示，只有你有这个权限它才能显示。

没有就不显示，好，我们试一下，我再重启一下。好，启动以后打开浏览器，然后刷新一下，这样我先退出登录，没有任何身份，我去看任何一个帖子，一个案子都没有，对吧？我都没登录登一下。Cc就是普通用户ayj8什么也看不见，因为他是普通用户。

好再来，然后q211这是管理员8，guq还是只能看删除对吧？那俩看不到能不能点可以点，帖子删掉了，好再来，然后扣的21这是版主再找一个可以看到，但删除看不到能不能点可以的。

好了，那么这样的话我们置顶加精和删除这三个功能我们就完整的开发完了。

总之它的功能实现非常简单，重点其实我是想给大家演示的是权限管理相关的内容，那么权限管理一定要注意，其实它包括两个层面，一个是在服务端真正能够过滤掉，就是没有权限的那些用户不让你访问这个功能。

第二个我们在页面上要友好一点，既然你不应该访问，你最好也别看见眼不见心不烦，这两个层面都达到了我们你对 Spring security，那么它的运用和理解就比较到位了。好了，这次课我们就演示到这里，咱们下次课再见。

## 7.4 Redis高级数据类型

46-Redis高级数据类型.mp4

这节课我们来学习redis的高级数据类型，主要是学两个类型，一个叫help log翻译过来就是超级日志，另外的一个叫beta map，翻译过来是位图。那么这两种类型它高级在哪？它都适合用来对网站的运营的数据进行统计，而且在统计的时候它非常的节约内存，效率比较高，这是它的一个特点。

那么我们这次课主要的目的是为了了解这两种类型，了解他们的优势和一些原则以及适用的范围，那么下次课我们再去运用它，解决我们项目中的问题，来统计我们这个网站当中的一些有用的数据。

好，我们先来看一下 Help log这种数据结构它采用的是一种基数算法，适合用来完成独立总数的统计，什么叫独立总数？

应该有的同学也听过，就是说一个网站在运营的时候经常要统计他的UV就是独立访客，那么一个人一天之内访问这个网站，多次这算一个访客这算一个UV就是说我们对于这一类统计，我们在统计的时候是要对多次请求进行一个去重的，那么害怕 log特别擅长做这样的事情，擅长在哪？

他在统计的时候，数据占用的空间非常小，那么无论统计多少个数据，它只占12k的空间，你要知道我们统计一个网站的数据，比如说统计用户有关的数据，网站的用户可能是非常多的，他可能有几十万上百万甚至上千万甚至更多的用户，那么如果你用其他的类型，比如说你用集合 site它也能去重对吧？

但是你把这些用户都存到集合里，占用的空间都大了去了，但还是log，无论数据数据量有多少，它最终都只占12k的空间，它这么神奇，其实是有它的代价就是说它这种基数算法它是一种不精确的统计算法，它说白了就是一种估算，但是这种估算的准确率非常的高，或者说它的误差非常的小，它的标准误差官方公布的大概是0.81%。

好了，所以说对于那些你不要求100%这个准确的精确的数据，同时数据量非常大，那么你希望占据空间小的这样的数据，我们用 happy love来统计是非常合适的。

好，这是第一个我们要掌握的数据类型。第二个是bit map，贝特曼其实它不是一个独立的数据结构，你从那个risk官网上去看，你选命令不是分类，都没有单独的这一类，它实际上就是字符串，但是它是一种特殊格式的字符串，它这种字符串我们是要按位存取数据，那么每一位我们只能存0或一二进制的数据，你可以把它看成是一个只有01构成的一个字符串，说白了我们也可以把它看成是一个bad数组。

有人说它只能存0和1的话，好像似乎它存的内容很有限制，什么时候适合用它，那么我们适合用它来存储大量的连续的数据的布尔值。

这里我举个例子，比如说我要统计用户的签到，对于每一个用户来说，每天都要统计今天你到还是没到，明天你到还是没到，那一年365天，天天都得统计，然后要统计若干年，那么你看每天只记录到或不到，我们01就够了，0代表没到—代表到对不对？

然后如果说我们用其他的类型去存的话，这个可能你存的是一个布尔的处放式，你还是存一个一个其他的能表达这个含义的单词什么的，那么这个空间都是比较大的，但我们用 beta map来存就会占有很小的空间，而且很方便。

我们这样转存0101，比如说第一个第一位的数字01，代表的是今年的第一天你到没到，第二位代表的是你第二天到没到？

第三百六十五位表示你第三百六十五天到没到，明白吧？

这样的数据是连续的，那么所以我们在存这个字符串的时候，每一位代表的是连续的值的索引，而每一位的0和1代表的是真和假到还是没到，所以如果说一个用户我们存他一年签到的值你都可以算出来一年就365天，那就是365个365位的值365位，那么每8位是一个字节， $1 \div 8$ 年的数据量也就是40多个字节，所以说它存数据空间也是很小的，但是它相对于 help log来说，

它能够统计精确的值，你这一年010 101，我想统计所有你到的有多少个数量，它是能准确统计出来的，是这样的。

好了，总之这两种类型的他们的机制不一样，适用的场景也不一样，我们这次课主要是解决他们的语法的问题，然后的话我们下次课再去应用，接下来我们就写一点程序，然后来体会一下这两个类型我们怎么去用。

好，我打开开发工具，然后因为这是一个 reads有关的问题，现在我们只需要了解语法，所以我们只要写一个测试类来测一下就可以了。而之前我们写过ready的测试类，我就找到那个类，接着在里面写代码，Ready, stats。

好，我找到后面去在后面追加一些内容，首先我们先来演示 help log怎么用，我说一下我想做的事情，我想这样做。

我要统计。

20万个重复数据的独立整数，就是我一共有20万个数据，那么这些数据当中有一些数据是重复的，你统计一下去重以后有多少个数就相当于统一独立访客感觉差不多。好，那么我就写一个测试方法，谢谢。

方法名我叫哈尔泰斯坦，hyper log log。

好。

那么因为我们要往 Rise里存一些数据，统计数据你得先存进去，你把数据先存进去以后我们才能去统计，当然这个数据我们是造的自己造，但首先我们要存这个数据得有一个key，首先定义 key，随便 read this key等于test冒号，hyper log hll缩写，然后01这是有关hll的我们的第一个k好，然后我要造20万个数据，我就循环是吧？

我先放循环就可以这样放！然后一回车它能生成一段代码，就是！我完了从一开始！小于等于个十百千万10万，就是我从一循环到10万，然后我把每个！存到还存到还不拉的结构里面去，怎么做？

Red is template点ops for你看有这么一个类型 ops for happy love，然后的话添加数据艾特那么指定key再指定值a那么这行代码执行完以后，那就相当于我会往这个数据里一共存10万个数据是不重复的，但是我说了我一共有20万个数据要重复怎么办？

我们再循环一遍，我还是循环10万次，然后每次我要产生一个随机的值，随机的范围就是1~10万之间，这样的话就有重复数据了对吧？好，我要一个随机值等于。

mass点render。

然后这是小数乘以8,000万×10万，然后再加个一。当然了你还是小数，你得把它转成整数，这样才行。

好，那么 master random的是0~1之间的半闭半开区间乘以10万0~10万之间的半开区间加一就是一到100,001之间的半闭半开区间再取整，那就是1~10万之间的闭区间，这个范围之内的一个随机的数字数字，好，然后我把这个数字存到这个结构里，存的是 r这样我们一共向 hip log里存了20万行数据，其中有重复的，但哪个重复了不知道随机的，这个数据我们自己知道的，我们很清楚，如果说去重以后应该是多少？

显然应该是10万，因为它数据范围都是做了在1~10万之间，所以说就是10万。

我们看一下我们利用 help log它的API统计的结果是多少和10万之间的误差有多大？好，我直接就是输出统计结果，这样我先写一句话，统计的话就是red is Complete点 Happy love，然后的话统计这里边的去重以后的数据的数量 size，然后把 t传进来，我就得到了一个值，这值应该是浪吧，是塞子，好，我把这个值打印出来看一下。

写代码写完了以后我把它执行一下，这个单词我写错了， ready？

改一下，好执行一下，然后它这个过程有点慢，为什么？因为我们造了20万个数据他得跑，对吧？所以说他得需要一小——会时间，但这个还行，不是那么慢，还算比较快。然后你看最终的结果99553，精确值应该是10万，而我现在得到的是99,553，那么误差是很小的，这个结果是可以被接受的，是ok的。

好了，那么再来这个还不拉，除了我们这样存数据这样去统计以外，它还有一个很有用的方法就是能够对数据进行合并，比如说我统计了一号的UV我统计了二号的UV以此类推，我统计了这一月每一天的UV那么我们在去查询的时候，有的时候我们可能有这样的需求，我要的是这一周这一周7天的UV你应该把这一周7天的每一天的数据合在一起进行一个合并。

所谓的合并就是说本来是你这个人在这一天访问的，多次访问算一个UV如果是按7天来统计的话，就是你这个人在这7天之内访问多次算一个UV是这样，它能够自动进行一个合并，它有个API能够很方便进行合并。

下面我给大家演示一下，当然我不造那么多数据，我就造三组数据，那么就是将三组数据合并，然后在合并后的重复数据中统计独立整数。

这句话说的这么别扭，就是在统计在统计合并后的重复数据的独立总数，因为本来每一组数据都有可能重复，那多组数据合在一起就更重复了，对吧？我们在重复的基础上统计一个独立总数这个意思。好，我再写一个测试方法，这个方法名我叫test hyper log log。

有你。

好，那么首先我得有三组数据，这三组数据我就不搞这么多了，每一组数据我就搞一个1万条。好，但是现在我还是定义一个key新的key。

第二个t test hll然后02，然后我往这个数据里存1万条数据，l等于1。然后小于等于1万。好，那么我把每一个l放到这个数据里，l read this template ops for hyper log around add write is key，然后的话存的是 l好，第一组数据就有了，同理我们再来第二组，第三组数据一共就三组，我可以copy。

这是第二组数据，我把 k303然后这个数据我就从不从一开始了，我从50015001开始，然后一直循环到15,000。反正一共也是1万条数据，然后存到三里不要搞错 t好，这样你看第一组数据和第二组数据他们是不是有交叉有重复数据是的。从5001~1万，这部分数据是重复的对吧？就我特意照一些重复数据，好，看看他统计的到底精不精准，如果没有重复数据就没意思了是吧？

好了，再来一组叫44，那么 l我。

从。10,001然后。

一直到循环到2万。

然后也是为了让它和第二组数据有一定的交叉，然后存的时候这个是 ready4，然后存的是 l没有问题。

好了，那么这三组数据其实我们口算也能统计出来一共一共有多少条数据，一共有3万条数据，那么去冲以后有多少条应该是2万条，对吧？因为这个数据的范围是做了在1~2万之间的，它没超出这个范围，不管怎么重复，而且一定是连续的，然后不管怎么去重，一共是2万个。

好，我们给它合并一下，那么ride is happy，log类型它对数据合并是把这三种数据合并以后再存回到 release里，存的类型也是help log。

所以我们一合并会产生一个新的数据，我们需要定义一个新的key，我叫优点k unit，然后等于的是test paper log，然后的话unit然后我要进行合并了，radius template ops for help love，然后点它有一个克林方法的第一个 key写的就是优点的key，就是你合并完以后数据要存放的位置，后面你可以写多个key，你要合并的那些key，我们分别就是reds二，对吧？

3。

还有4。

当然你这里传的是数组也可以都行。好了，那么这句话执行完以后，那么 unt对应的数据里就有相关的指标，我们看一下这个结果还是 kind of sad。

然后 write this template for ps for have a log。然后 size我要看的是u0k这组数据我把 size打印出来看一下，这个写完了非常容易执行一下，这回数据量比刚才小了一些，所以说应该是略快一点，我们看一下这个结果已经出来了。

一共是19833，那么准确的值我们自己算应该是2万，但是它给我们估算的是19833也是非常的接近，误差是非常小的。

好了，那么通过这就是这两个方法，我们基本上了解了还不log，然后它的一个使用情况，它的用法，然后数据量大与小，其实也可以看，但是可能看的就不是那么准确，大概也可以看ready? Cic烂11。然后的话 Release里有这样一个命令，它叫做in for memory。就是看内存的一个使用情况。

然后这里边它有一些内存的使用量，那么你看比如说第一个使用的总内存，比如说你在执行操作之前看一眼内存，执行之后看一眼内存，然后两者一减，然后一换算大概能算出来它十几k但是这是总的内存，因为你创建数据它k也是占空间的，它可能是比实际12k大一点，但是它的值只占12k是这样的。

我们就不在课上看，你可以自己课后自己看一看。好，那么以上是关于 Happy love第一个类型。

下面我们再来演示一下第二个 bit，MAC写不出是说我要做什么，我想做的事情是我要统计一组数据，这里我就先不说我统计的是签到了，因为一会我还要做一个运算，签到的业务和运算好像不太搭边，或者说签到的逻辑用不上这种运算，所以我不说它是签到反正就是我要统计一组逻辑，一组数据的布尔值，一会我对它做运算，只是停留在一个语法层面。

好，我要统计一组数据或者值。

写个测试方法，test，bit map。

当然了统计的值，首先我们得有数据得先存数据进去，然后才能统计，所以说还是要说明一个key。Ready，skip test bit map01，这是我们要统计的第一份数据，首先我要往里存，这个数据我要记录一些值。

好，刚才我也说了 beta map它不是一个独立的数据结构，它其实就是string，只是对一个string的特殊的操作而已，所以我们其实还是ready time late点ops for value还是访问死菌，然后我要往死菌里存的是按位存的数据怎么办？

调这个方法site wait，然后声明 t因为是按位存，你要指定我现在是第几位索引是多少，以及它这个值是多少，比如说我索引是一，当然索引从零开始第一个位置，那么它是0还是1这个值，但是这里边他让我们传的是布尔的true force，说白了我们g0和1其实就是用它来代表true force，所以说它这里面是传的是一个逻辑一个，而不是0和1，这也合理。

好，再来两个数据，比如说这是第四个位置处，然后比如说这是第七个位置处，有人说那里没有存force吗？Force不用存，你只有设置它默认的话，连续的人默认每个位就是false0，那么你存了这个数以后它才是一，而我们最终要统计的往往就是处的个数为真的个数，就以签到的业务来说，你哪到哪天到了，我们很少说统计一共你全天哪天不到，一般都是统计哪天到，当然你不到可以通过这个减法去算出来，但用的比较少。

好，这是记录进去的，记录进去以后我们可以查每一位他这个值是多少，我们再来查查单个某一位的结果。好，我就直接输出这个结果了。

Release template ops方案6，然后取某一位的值get beta存是site取是get，然后 ready ski，比如说我取第零位，你注意第零位我可没有设置为处，它默认是false，第零位比如说我取第一位，我再取第二位，第二位，好，一会我们看一下这个结果，然后当然这种这不是统计，这只是查某一位的布尔值情况，我们要统计说一共有多少个数怎么办？

再。

演示一下统计的方法是不在这个对象之内，我们从对象访问不了，我们需要通过radio底层的连接才能够访问那个方法，我们怎么去获取radio底层的连接呢？可以这样做。

Ready是太累的。

点sq就是说这个方法是我们要执行一个rights命令，那么执行的时候它需要我们传入一个回调的接口，我们在这里做一个匿名的实现 new。

Red is called back，它自动的帮我把实现的方法加进来了，然后它里边有一个方法接口里边有个方法叫 doing ready，然后这个方法传会有一个参数是readus连接，这写的简单一点connection，就是说当我们Diao sq的方法的时候，这个方法底层的会掉，我们提供的 call back，然后它在调的时候它会自动调这个方法，它会把 reds连接传进来，这是red底层的一个逻辑，我们这里可以去做一些处理，并且返回一些值，反馈结果最终会返回给 sq的方法，我们可以通过这个地方接受一下最终的结果。

那么接受以后我可以最后把 OB机打印出来看。

好。

剩下的就是说我在这里怎么去利用品牌安全统计这个数值了，非常简单。有了connection以后，我们只要调它的 Beta count方法，然后的话传入 red SK但是因为按位统计，它需要我们传的是bad数组，好吧，red ski不是死菌吗？

我们点get best就得到了 better速度传进去就可以了，它会帮我们统计这种back里边的一一的个数，这代码就写完了，我们试一下看这个结果怎么样执行。

好，我们看一下最终的结果，首先是这个地方我们按位取布尔值第零位false第一位数第二位false没问题，因为我们存的时候其他的默认都是false，但是只有147是处，所以这里面一处往其他处放没问题，那么统计最终的数的个数三个也没错。

Ok没问题。

好了，这是我们往里存值，以及我们如何去统计这个值。那么如果你有多组数据，这之间还可以进行一个布尔运算，因为我们纯的值是数放，那么它能做什么运算？它能做与或非的运算，比如说处或force结果还是出真或假结果还是真，是这样的。

好，那么这个也有用，我们下次课

会用到这样的逻辑。

好，我们这次跟他先演示一下怎么去做这样的事情，几个注释统计，三组数据的布尔值一共有三组数据，然后我并对这三组数据做二运算，就是我们做二运算的情况用的多。

当然如果你是做这个案的，其实和二一样，就操作时候换一下而已。好，我再写一个测试方法，叫test bat，因为是运算运算符，Operation。然后这三组数据我得先造一下，首先 steam red is t二等于text bm然后02，然后我要往这里存一些值 release template，点ops for value，点set fit，write this key。二比如d0的位置，我要存的是true。



好，再来俩，然后第一个位置是处，第二个位置是处012都是处我这样存的，然后再来一组都去copy了。k就是33把都换成三，然后这个第二组数据我是234为数，01和其他的方式，你会发现这两组数据是不是有交叉的情况，对吧？就是二都是处他们有交叉的情况，好再来一组4。好，这个我设置是4，56是数，其他的位置是force。你这么看这3组数据，比如说你都把它看成是一个每一组数据都是有6位00六七0~六七位的这么一个值，那么第一组数据就前三个是处，第二组数据是中间三个是处，第三组数据是后三个是处。然后我们对他说奥运算会得到一个什么结果，主要是看这样的一个事情。

那么运算也是和刚才 help log类似，它运算以后会产生一个新的结果，需要存到一个新的地方。

所以我们需要再声明一个 key，我就叫red，red is keep等于test beyond。然后的话or因为我做的是奥运算直接一点。好，那么运算我们也需要用到 connection，因为预算更高级了，你像刚才我们统计一个数量都得用 connection合并，也需要用到connection，所以说我们需要写刚才那个代码结构，

object obg。

等于write this template点，s cute。

然后 new。

red。

is called back。

好，那么在内存的方法里，我们需要做奥运算用的connection，就connection参数名改短点，connection点bit op bit operation就是要做运算了，然后你要做什么运算，你首先要把运算符指定我们通过一个就是一个接口 bit operation第二二是red is string command。

第二bit operations，第二先证明运算符，然后你要指定我预算结果存到哪个key里，应该是存到 release key里。那就ready ski there get back。因为位运算它需要你存的是这个二进制的数组 Bat的数组，然后你要对哪几组数据做运算，你再把那几组数据的p传进来，那不就是234对吧？

然后就是write this key二，当然也得改成也得置换成办法。Read this case三，get backs，where is k4？Get backs。好，这样就行了，那么他就把234这三组数据之间做一个奥运算，你要知道二元算的出结果就是出，对吧？好了，执行完以后执行完以后我们会得到一个 red SK里就成了一个新的值。

我们要统计这里边数据的情况，就是里边真为真的数据的结果，我就connection点bit count。K你要写的是ready ski要get best好。

返回这个结果，

我最终把这个结果打印出来是吧？Ob界。好了，当然了，这个时候你也可以把 ready key对应的每一个值弹出来，它最多也就是0~0~六七七个值，因为这些数据合并在一起最多就7个值，我依次给打印出来 c是out，ready，stop late。If you ask for value，然后 get bit where it is key连。然后打印出来以后我们看的就更直观一点。

好，那么写完以后我再执行一下这个方法。

好，那么执行完以后我们看一下统计结果是7，然后你看我们打印的0~6，这7个值都是数统计的没有问题，为什么是这样的？因为 Sk2他012去处，虽然说其他的两组数据可能012 false，但是和它一二运算以后变成了处。同理它中间三个值是处，它后面三个值是处，它们互相一做运算都是处了，就是这样的。

好了，以上这两种rise高级数据类型，我们在使用的时候往里存数据以及统计数据的一些API的演示，当然我们更多的在意的是统计的一篇，或者说我们能够利用它统计什么，那么现在语法知道了，我们下次课再去给大家演示，我们到底能够解决项目中的哪些实际的问题。好，这节课我们就进行到这里，咱们下次课再见。

## 7.5 网站数据统计

47-网站数据统计.mp4

这节课我们来实现网站数据统计的功能。那么要统计的数据有两项，一个是UV一个是dau首先你要了解这什么是UV什么是dau我们在统计这两项数据的时候分别要注意什么，那么我们适合能用什么样的数据结构的去统计这个数据，我们先要对这些内容做一个了解，下面我就给大家介绍一下这两个概念。

首先说优威它是 UNIX visit这个单词的一个缩写，那么翻译过来就是独立访客，就是说那一个一个人他访问这个网站一天访问多次，他算一个访客，但是我们如何去识别这个人呢？我们并不是通过这个人的ID又在ID去识别的，我们是通过他的IP IP地址来统计。

当然我们是去重以后进行统计，也就是说为什么要用这个IP来统计，因为我们希望把匿名的用户，那些游客没有登录的用户也统计进来，就是说UV我们关注的是访问量，而不是关注你到底注没注册，你到底登没登陆，不管是谁，只要你访问就可以，就算这个意思。

所以我们每次访问都要进行统计，因为你不能确定这次访问他到底是不是一个新的用户对吧？所以每次访问都要统计，反正每次访问我都把用户的IP存到一个数据结构里，然后他最终去重以后就得到一个结果。

好，那么像这种场景其实特别适合于我们上节课所讲的还不唠不唠进行统计。因为 help log最大的特点是存储的空间小，而我们这个网站的访问量可能会很大，每天可能有几十万甚至上百万甚至更多的量，那么你需要有一个好的存储空间小的结构去存比较方便，所以这个功能我们使用 help log进行记录进行统计。

然后第二个统计的数据是dau它是这样一组单词的缩写，daily active user，翻译过来叫日活跃用户，什么叫日活跃用户？

其实不同的网站它的定义不一样，有的网站认为说你只要今天访问过我的网站一次你就是活跃用户，有的可能认为是你今天不但访问你还要特定你要访问，比如说我的主页或者你要访问我的某一个区域，你才算活跃用户，也可能有的网站认为你需要访问我多少次才算活跃用户，每个网站的界定是不一样的。

那么我的要求是什么？就是说只要你今天访问一次，你就算是一个活跃用户要求比较低，有人可能会想你这么定义的话，是不是跟独立访客UV一样的注意是有区别的，就是我们统计活跃用户，我们通常是通过user ID排重统计，也就是说统计只是统计登录以后的用户，对于那些游客没有登录的用户是不关注的，所以说看起来好像一样，但其实又有细微的差别。

Dna我们在统计的时候，我们更关注的是用户他的有效性，我们更关注用户的它的一个核心的程度。

所以对于这样的一个数据，我们在统计的时候要求它是一个比较精确的结果，不能是不精确的。

所以那么我们也是需要每次访问，我们都要去进行统计，那么因为我们要一个精确的值，所以我们用Beta map比较好，因为我们用它的统计存储的空间也比较小，性能也比较好，再一个它能统计出精确的结果，而happy love达不到这个要求了。

有人可能会想你统计你用 beta map来存这个数据你怎么存呢？其实这个很好存，因为什么？我们的网站的u在ID是什么？是整数对吧？我们要记录的是什么？我们要记录的是优质ID，他今天访没访问过访问过，就是一没访问过就是01就表示活跃，0就表示不活跃，我们可以以ID用ID整数作为beta map的索引来存，比如说ID为101的用户，那么我就在beta map第101的位置存一个一ID为202的用户，就在202的位置存一个一，很好理解吧？

用这样的方式可以存，这样的话我们每天可能是有几十万个活跃用户访问了，那么我们只需要用一个bit map，然后有几十万位去存就可以了，所以说它也是比较节约空间的，这么多用户，我只需要用这么多位用一个数据就能存，也是比较节约空间的。

好了，那么大致了解了这两个概念的区别以后，下面我们就可以来写这个功能，总体来说比较简单，因为其实我们也看到了，我们但凡是和 radius 用 radius 来实现的某些功能，一般代码都不会很多的，因为 rise 它存储的存取数据的方式都是比较简单的，所以那么我们在实现每一层的代码的时候，我就是这两个功能一起实现。

比如说我写 service 层代码这两个功能我就一起把它写完，因为一共也代码也不多，好，我们开始写代码，我打开我的开发工具，因为我们需要用的是 radius，所以首先我需要定义 radius 的 tea，我先打开 red skuto 好，那么我就需要增加两个前缀，第一个前缀我叫 UV 是用来统计 UV 相关数据的前缀，我这个前缀就叫 UV 好，我 copy 一个再来一个 dau 日活跃用户 dau 好。然后我需要定义这个方法，然后来返回对应的 key，首先我要返回的是一个单日 UV 的 key，就是我们统计 UV，我们记住这个 UV 是以天为单位去记，每天都记，然后我们实际去查数据的时候，查看数据的时候，可能我想看某一天的 UV 这是能支持的，很显然有的时候我们可能会看那一周的 UV 如果你看一周 6 个月怎么办？

你把这一周 7 天数据把它做 1 个合并，我们 happy love 不是可以进行合并就很方便，所以我们按照单日的方式来记做这么一个 key，怕被克斯代理均衡 key 方方面面的 get UV t 因为你要获取的是单日的 UV 那是哪一天你是不是把日记传进来？

我只要求你传一个日期的字符串就好了，我就不要求你传一个 date，而且这个只包含年月日就可以，我们只是按天统，不用时分秒我这里就 return，然后刚才 plus 杠 UV，然后加上一个分割符 split，然后再加上 key 就做好以后，刚才也说了，如果你想统计一周一个区间之内的 UV 的话，我们可以进行合并。

我们上次课也讲了，如果我对这个数据进行合并，它会产生一个新的数据也存到 radio 对吧？

你产生新的数据不也得需要有一个 key，所以我们再来一个区间有 v 从哪天到哪天之间的一个 UV，这个我就要还叫 get uvk 但是参数不一样，因为你是一个区间，所以说肯定是有两个日期，一个是开始，一个是结束，我叫石军。对，开始的日期再来一个。And 对结束日期。好，我就 return。Practice just beat it。然后加上搞错了，然后加上 start date，再加上一个 spill it，然后加上案例。

好。

这是关于 UV 的两个 key，同理，那么我们统计活跃用户也是需要类似的两个 key，一个是某一天的这一天的活跃用户，一个是区间范围之内的几天的也是类似的。好，那么这是单日活跃用户，我叫盖 dauk 我需要传入一个 date 为天堂，practice deo 加上 split 加上好再来。

然后是区间活跃，用户 get 还叫 dauk 需要两个参数，一个是起始的日期，一个是结束的日期准确，study 加上 speed 加上 and it 好了。

t 我就定义完了。

定义完以后，那么我们开发的话，首先开发数据访问层，因为我们用的是 rise，所以说我们数据访问层不用单独写一个组件，我们直接上来就写 service 就可以了，去调 ready 他们练就行了，所以我要写一个 service，这是一个新的功能。

我创建一个新的 service，我在包下新建一个类，这个类我叫 data service，数据统计我就简短一点数据了，然后加一个注解，service。

好，那么因为这个例子当中我们重点是要用 release，所以我把 release time late 先注入进来，另外我们在统计的时候要频繁的和日期打交道，每一个方法都得和日期是紧密相关的，那么日期的话我们又需要对它进行格式化，而我们需要的格式化成年月日的形式，不用时分秒以日为单位。

所以我我在前面提前先室内化一个适应部队的方案，后整，这样的话我在整个类当中随时就可以用它，我到处都要用的。6CM 的 format 给他一个格式，简单点 yy yy mm dd 就是年月日就完了。

好了，那么接下来我们就来写有关统计相关的方法，那么要统计这个数据其实两个方面，第一个方面是我要把这个数据记录下来，在每次请求当中我要截获请求，把相关数据记录到我 Release 里。

然后第二个当我想看的时候，我能够提供一个查询的方法能够访问到，一个是记，一个是查。

好我们一共是统计两个数据，我们就分开来做，我们先来处理 UV 的统计。第一个方法我要写的是将指定的 Ip 计入有为，因为是统计的是 IP，好那就很简单，说明一个方法不需要反馈值记录，不叫 record。

Uv。

记录的是 iip 所以你把 IP 传进来给我，记录就简单了，首先我就得到 k 然后往里一存就完了。10 卷 ride is key，等于 ride is key。

Youtube 点 get UV 好，这里你需要传入一个日期的字符串，传入一个日期的字符串。我们每次统计的时候肯定是统计当前时间的，所以我需要把当前时间格式化，然后转成字符串传给他，很方便就是 date、format、点、format。

然后的话 new。

date 就行了，今天的好我得到了 key 以后，我要往 readiness 里记录。我 run this template。

点 ops for happy love，然后 l 的一个值一个 t 然后这个值是 IP，好，记录这个数据的方法就完成了，然后还要需要提供一个查询的方法，能够让人很方便地查到这个数据，所以再来一个方法统计指定日期范围内的。

UV。

反馈浪，因为这个数据可能会比较大，然后统计就计算比较 accurate，然后 UV 因为是一个时间范围，从哪天到哪天，所以我需要你传入两个日期，你期望看的那两个日期，一个我叫 star 开始日期，再来一个我叫按的结束日期，好，那么这个方法之内首先我先对两参数做判断，那么任何一个参数为空，我不允许必须两个参数都传我才给你统计。

好。

如果说 start 等于 not 或者。

是 and 等于 not，我将抛异常 through new。给一个提示给一个参数。

不能为空。

好，那么如果参数没问题，我们就要怎么处理我们统计范围内的 Uv 的话，其实就是把这个范围内每一天的 k 做一个合并，合并的话我们需要得到先得到一组 k 我得先收集到这一组 k 做成一个速度才行，对吧？

然后好调合并的方法。所以我需要今天从开始日期遍历到结束日期，然后把每一天的 key 得到，然后变成把它放到集合里，我需要整理 Key，那么我要整理该日期范围内的 key。

首先我先说明一个集合准备存整理到的 t 试卷，我叫 t case。

new a released，

然后我整理的时候我是从这儿便利到这儿对吧？那么从那日期怎么便利，它可不像 int 对吧？然后每次循环以后，我这个日期是不是得加一天对吧？所以说为了能够对这个日期做运算，我们需要用到 kind，等于 kind 点儿，get insist，先室内画一个 Kindle，但是 Kindle 里面的日期还没有，我需要进行一个设置。

凯伦的点 site 看，我先把 start 设置给他，我让 calendar 里面包含的日期是开始日期，然后我就利用 calendar 做便利，用 while 循环 while 凯伦的点 get time，就是日期点 after。然后我前面加一个感叹号，好，我们看一下是什么逻辑。凯伦的点 get time 就是时间这个时间如果晚于 and date，加个感叹号就不晚于 and date。

也就是说这个时间是小于等于 and 的时候我就循环，它一旦大于 and 我就结束循环，就这个意思，也比较容易理解。

然后当然了，我在循环内部我首先要做一个处理完以后开展得加一天，让他无限的趋近于案子，最终等于案子结束这个循环对吧？必然死循环。我这个里边是开了的，别忘了开了点点加一天，单位是开了的点 Date，我指定的是加多少天一天这样。

好，在加之前我需要整理的是key，我需要得到一个key，这个key很方便，就是 cutu ready cutu然后点 get UV key你把看了那笔所包含的日期传进去， data format点， format方面的 data就是开了你的点 get，这就是我们所得到的一个t我要做的是把 t 加到集合里去， t list at t好了，这就整理好了，当我们循环之后 key list整理的就是所有的这个时间范围之内的key我需要对它进行合并，所以下一步是合并这些数据合并完的数据， key我要进行一个声明， red is key等于 ready skill to点get UV key。

好，这个时候我需要传入两个日期开始到结束，当然我要传的是字符串，所以data format，点format， start再来df点format，这是合并以后数据的key，我接下来就要进行合并了，那就是readiness template。

Ops for help log合并有你，然后第一个参数传的是合并以后的key，然后后面的是其他你要合并的那些数据的key，我们就是集合了key list，当然你需要把它转成数组，他要的最终是数组那种集合转成数组， to和瑞。

好这就行了，这句话执行完以后，这个数据你就存了，合并以后的值，我们只需要做一件事，返回统计的结果就行了，我就return radius template和ps for have log，然后点统计数量site ready skip。

好，那么统计uvr这个方法我就完成了。那么除了ewl以外，我还要统计活跃用户，其实也是类似这样两个方法，只不过我们的数据类型不一样而已，我再接着写。

接下来我要将指定用户进入到 dau数据子区加指定用户记录 dau public不需要反馈值， record什么 dau然后我们统计dau的时候，一般是按照优质的ID来统计的，所以传入就是ID这个也简单，首先还是先得到一个key t这里你也要传入的是当前的时间的字符串对吧？ Df点format，然后new data有了p以后直接把它存到 radius template ops for value。

然后调的是site，背的方法，传入key。

然后刚才我一开始也说了，我们存的时候怎么存，我们是以邮政ID整数作为索引，然后存的是布尔值有没有活跃，是否活跃，我所以这个参数是uid然后数你调这个方法说明肯定就是这个用户已经访问了，他一定是活跃的，对于其他的我没有调的，那就默认就是false，就这样就可以了。

好，同理我们还也要提供一个能够统计能够查询的方法，那就是统计指定日期范围内的DNA开了一个浪开口 dau那么这个参数也是开始结束两个日期。

好，那么整个逻辑和刚才其实统计类似，一开始也是判断一下参数怎么样，如果参数没有问题，我们也是当然我们背题map我们不是合并，我们是要把日期范围内的每一天的do做一个之间互相之间做一个运算，做一个奥运算。

如果说我要统计今天的活跃用户，你只要用户今天访问过网站一次，你就是活跃的。如果我以一周为单位，7天为单位统计这个用户是否活跃，那就是说在7天之内，你任何一天你只要访问过一次，就算是活跃，这是我们从产品角度去对活跃用户的一个定义。

所以你看我们这样的逻辑应该是做什么预算？奥对吧？任何一天你活跃那就是活跃，所以用奥运算好，基本上你要做奥运算，我们最后那一步也是需要你传的参也是也是一组 K所以我们首先第二步也是要整理 k整理完k以后再做运算，所以整个的过程和类似，尤其前面这两个类似都把它拷一下。

好，第一步判断参数为空，我就更改就这样。第二步是也是整理该日期范围内的t也是类似的，但是我整理 T的时候，我们 beta map的运算，它需要的t的类型是bad速度，所以这个集合点我存的不是死病，我是bad速度这样来处理好。

然后也是就开了盖，因此先16万一个开了，然后先把它设置为起始日期，然后便利的方式一样，如果开了你的 after and去反，不是这样的话，那么我就每次遍历我会得到一个key，这个得重新写了，好每次遍历得到一个t那就是red is t YouTube点get daudauk然后你要传入凯伦的所包含的时间，当然需要格式化一下放慢一点，然后是calendar点， get time。

好，我得到 key以后，最后我要的是这个我先这样先得到一个strain student。

好，我需要把 key 转成 bet 数组放到集合里也很好办对吧？Key list 点 at，然后 keep 叫 get best，然后 k 整理完以后，我就把这个日期加一完了。好，最后要做的事情就是进行一个奥运算好，那么进行奥运算，我们不能够直接使用思俊结构去调研这个方法，我们需要用 red 底层的连接去调用，我需要这样写，我就直接 return。

Red is Complete，然后 sq 然后传入一个匿名的，ready 是 call back 的实现 ready come back。

当然它这个方法默认返回的是 object，但是我们这个方法期望的返回的是 long 我强转一下，但是前提是我要保证我在这里最终实现的时候确实是返回浪，好想转一下，剩下的逻辑我们就是在 doing readiness 这个方法之内去实现了，把这个参数先明天再简短一点，那么我们做奥运算之后所产生的数据也需要存到一个新的数据里，也需要一个新的 key，在这里我先把新的 key 做一个声明。

Read is key 等于 read is key。Youtube 点 get dau key。两个时间开始 start 结束，and 好，key 有了以后我要做合并了，那就是 connection。

点儿 bit 不危险，第一个参数传的是你要做什么样的运算，奥运算好，然后第二个参数传的是 k 就是运算完以后你要存到哪去 k red is key，当然他需要的是 bet，所以我点 get bets，然后后面的参数你要对哪些 key 做运算，那就是集合里的数据，key list 我需要转成速度 to are we？

然后需要单独处理一下，我需要声明它要转成什么样的类型，它最终转成的是 new bad，一个二维的 bad 数组。好，那么这句话执行完以后，我们就可以进行统计了，当然我们统计的是 K 的值，我就直接把统计的结果返回 connection，然后点被 count red ski。对，get back。

好了，那么 service 我就完成了，一共写了 4 个方法，完成以后，接下来我们就可以写表现层的逻辑，那么表现层的逻辑一分为二，第一个就是我们什么时候去记录这个值。

第二个我们去查看两两个方面，记录这个值我一开始也说了，我们需要每次请求都得记，因为每次请求都有可能是一个新的访问对吧？既然是每次请求都要记，你说我们在哪里写这个代码，很显然我们是在拦截器里写比较合适，对吧？因为这是一个非常通用的处理方式，那么拦截器很方便能够拦截那么一个请求对吧？于是我要写一个拦截器，我在 intercept 包下新建一个拦截器，专门解决这个问题，那么拦截器我给它取名了叫 data center。

好了，拦截器前面我加上一个注解，另外它需要实现一个接口 hand in the second。

好。

那么在这个类当中我需要记录的是记录的是 UV 或者是活跃用户，我需要调 data service，所以我把它注入进来。另外对于活跃用户，因为我们记录的是优质 ID，所以我还需要得到当前登录的用户的 ID，我再把 house 偷的注入进来。好了，我们在什么时候统计？当然了我们在 Free handle 在请求支出统计就可以了，所以我重写 Free handle。

当然我只是做一个统计，最终统计完以后请求继续向下执行对吧？

你这里不要返回放，你要返回数，让请求能够继续下去，我只是加个塞统计点数据而已，那么统计首先我统计 UV，然后再统计这个 d 给 u 都在这一个软件系统统计了，那么 UV 首先统计 UV 你要统计的是 IP IP 怎么得到通过 request，对象就能直接得到非常简单。

String IP 等于 request 点 get remote，这就是 IP，当然如果说我们在访问的时候，我写的是 look house，他这个 IP 有点特别，他写的好像是 0.0.0.1，那么如果你是正常的一个 IP 地址，比如说 127.0.0.1 它就没问题，所以 house 有点特殊，但是它是不是个错误。

好了，然后我就记录到 data service 里面去，data service 然后点 recovery，把 IP 传入就可以了。

好，然后接下来统计 dau 我需要得到当前的登录用户是吧？那就 user 等于 house 偷的 get used，那么有可能用户没有登录，这意味着可能是 now 对吧？所以要判断一下，如果 user 是 now 没登录怎么办？我就我就不统计，如果它不等于 not，我才统计这样的一个逻辑。

Data service点，record dau然后是user点get ID。好，你看我们记录数据就非常简单，统计完了，记录完以后拦截器目前还没有生效，你想让它生效还得再做一个配置，我们需要找到 web，mvc肯定吧？在这里面我们再把拦截器注入进来配一下，这个叫data interceptor。

好，那么在后面我也是把它注册一下，data intercept，当然对于静态资源我们不需要拦截，我们只拦截就是对CTRL的动态资源的访问。好了这就配完了，记录数据的层面我们就完成了，接下来展现数据查询数据，那么展现数据我需要写一个control的来处理这样一个请求，那么我在control的报价再新建一个类叫data control的。

好，那么写上注解很臭的，然后显然是要调data，service把它注入进来。

那么这个control里我们需要写三个方法，第一个方法是我要访问统计页面，打开统计页面的样子，大概是这样的，里边有两打开页面以后有两个表单，然后在这两个表单上你想统计网站UV你就填这个日期，然后一点一点按钮，就统计了活跃用户也是类似填日期点统计就可以了。

那么如果你点开始统计，我就要统计网站UV我需要一个方法，你点统计用户要提供一个方法，所以一共是三个方法，打开网页的方法，统计它的方法，统计它的方法。

好，我就写一下，首先我先实现打开统计页面的方法简单了，先说明他的访问路径，路径我叫data，就data，然后的话请求的方式，master等于这里面我写个大括号，两个 Request，然后 request，点 post，为什么要这样写？

我一会写另外两个方法的时候再进行一个解释。

反正我先这样写着，总之这个方法既可以处理盖的请求，也可以处理POS的请求，然后方法需要返回的是模板路径，方法名我叫get data配置为这样，模板它在site二的面目录下名为data点开卖，所以我这样写site。

好的妹 data好了，写完以后这就可以了，我们一访问这个路径就能打开这个模板就可以了，但是模板一会我再处理，我先把另外两个方法也写了，也非常简单，我再写一个方法处理统计网站UV的请求。

好，那么瑞克斯卖品 pass等于data，UV master等于request点 post，因为你也看到了我们页面上要想统计的话得出来，要想统计的话他得选两个值对吧？然后点按钮，那就相当于提交一个表单，所以说要闯进来一些表单里的数据，我们用的是post。好这个方法名我叫get UV那么传的参数就是两个日期，一个是开始日期，一个是结束日期是大的。但你注意这个类型可不是普通的基本类型，它是一个日期。

那么我们页面上其实我们最终传过传的时候传的是一个日期的字符串，日期的字符串怎么转成这个data，它默认是不好转的，因为我们服务器也不知道你传的这不是什么格式对吧？

所以你需要告诉服务器我传的字符串这个日期是个什么格式，那服务器就知道了，既然是这个格式，我就能帮你转成对应的 data，它底层就可以做转换，我们怎么去告诉服务器这个日期格式，用一个注解叫date，time format。

然后对time format里边我们就写一个参数叫pet格式等于我希望日期的格式是这样的，yy杠mm杠DD页面上我们传的格式是这样的，那么同理也是一样的，太长了，换一个行就是这样的。

当然我最后我的结果要传给这个模板，我需要还用的model，model也声明一下。

好，那么方法内部的逻辑非常简单，统计的结果是个浪，我就等于data service点，calculate UV打死大的程序，把按的程序完了，那么统计结果我需要传给模板model，ed h build，然后的话我叫UV without就统计的UV的结果就UV另外你还要注意，那么我们统计完以后去哪个医院其实还是去统计页面，当我们就是一点这个按钮以后，就一点这个按钮它访问的服务器，服务器做了统计，又回到这个页面，回到这个页面的时候是不是起始年就开始年月日结束年月日还保留比较好对吧？

怎么保留？你看我这个页面访问了服务器，服务器再返回到这个页面的时候，你再把这两年再返回回来，它就能做一个默认显示对吧？所以这个时候我在model里再把这个参数再传回给模板好做一个默认显示。Model点a的h的

UV start对的。

然后就是start。

好，再来 Model点at attribute UV。

and date and。

好为什么这个参数名我取名都以UV开头，因为我们这一个页面上能统计两组数据，而且他们的数据结构数据形式还很像start and start，and如果你就叫start和and的话，这两个地方就容易混淆了，所以说我们最好把这两组数据做一个区别，加个前缀好区别就这个意思。

好，那么接下来我就等回到这个模板就行了，那模板的话其实你还可以这么写，这样是可以的。

那么还有一种办法，我也可以用这个forward转发，注意如果我不写forward，我这样写，我是返回一个模板，这个模板的返回给谁，返回的是返回给dispatch serve late，因为这个方法是他第二个 disc就是为了得到这个模板，他就去让这个模板去做后续的处理，这是一种方式。

For word，其实是什么意思是就是说声明我当前这个方法只能把整个请求处理一半，我还需要另外一个方法继续处理请求。

注意另外的一个方法也是一个和他平级的能够处理请求的方法，而不是模板，我发布的到谁我可以发布的到 data就是这个方法，仿路径其实就是它了，我处理一半，然后交给你接着处理，交给他处理，他里边干了什么，里边不就是转转到了这个模板。

当然了，我们在这里其实你直接写这句话也可以，但我这样写主要是告诉你，其实我们发过来也是可以用的，之前虽然说过一遍，但是我怕有同学还是不知道该什么地方用，这地方就可以用。那么假设如果这里还有其他的逻辑的话，我这么写这段逻辑能够被复用，如果我只是规范这么一个东西，这个逻辑就不能被复用，所以说这样写是有一定的好处的。

好，我这个就用forward了。

那么接下来我就可以解释了，为什么说我这个地方要支持两个请求方式，因为当我调这个方法转发到转发给他的时候，你注意这是虽然说程序从这儿跳到这儿，请求从这儿跳到这儿，但是这还是一个请求，转发是在一个请求之内完成的，因为他接收的是一个POS的请求，他把这个请求转给他还是post，请求在处理的过程中请求的类型是不能变的，所以就要求他需要支持POS请求，这样的话才能够被转发，否则就不行，要注意。

最后我们再统计一下活跃用户，那么统计的方式和很像，我就copy一下，那么这个路径我叫dau Post，然后方法名我叫get dau也是两个参数，一个是start，一个是然后 model这个就不变了，然后我们统计的值得重新写了，dau等于data service点，calculate dau然后把 start传进去，把and传进去，我就得到了统计结果我需要把它传给页面的时候，这个不叫dau result值是dau好，然后当然了为了页面上能够再显示继续显示这两个开始日期和结束日期，我把它也传回去，改个名字。

Deal，start it，deal and ok，然后转发到统计页面没有问题。

好，CTRL的这三个方法我就写完了，写完以后最后就是处理这个模板了，模板叫data点html我搜一下，这个页面是第一次处理，我需要从头开始处理一下。

好，然后这个地方路径要处理，这个地方header要处理，好，内容一会再处理，然后尾部不用管，GS路径要处理一下，最后处理中间的内容的部分。

首先你要统计你需要配这个表单对吧？我先来配统计网站UV的表单，那么首先请求方式，method等于post。另外要声明仿邮件，action等于get Uv然后那表单中我有两个框，这两个框你要给它取个名字对吧？



我把这个数据提交过去，name等于和统计UV的方法的参数匹配，start name等于然后当统计有结果以后，再回到这个模板的时候，我需要这两个框能显示默认值，所以我要写th冒号y6等于当然这是日期，所以我需要格式化一下。对此点放慢点，然后是UV是这样，格式比较简单，就是年月日。好，那时分秒的就不需要了。那么按的是同理的，我把这个拷贝一下。

当然按的是UV and对格式是一样的格式完了好再看，当然了你统计结果需要在这个地方显示，在一个十八里显示，所以在span这个位置我需要写上th冒号。

txt等于。

统计结果 UV without按钮，好，按钮得改按钮，它是8条，我需要把它改成3个问题好了，统计优雅就是这样，统计活跃用户不是类似的吗？

放然后是nice的等于post th action等于data dau这是关于 form的处理，然后这两个框也是一样的处理方式。

首先 name还叫start，这个还叫im参数名一样，然后 value topic一下什么事？This the format deo study，再来dau and date这是默认值。当然这里也需要改成submit，然后结果需要显示结果，就是dau伪造好了，这个表单我们很快也就处理完了，因为它数据比较少，处理完以后我们就来我们先测一下。

其实还差了一个环节，就是权限的环节，这只是功能的一个初步的实现，但是还有一个问题就是怎么说，这种功能不可能是每个人都能够访问到的，对吧？

它一定是网站的管理者才能访问，所以我规定只有管理员能访问他，那这个好办，我们之前有色块的配置，我们只要把它配一下就ok了。

那么关于全新的配置，我们打开 CK，然后找到这，你看之前这部分请求是管理员专有的，我在这里再加一个路径不就可以了么？当然这不是一个路径，其实是三个路径，data，UV data dau我可以这样写data星星，反正凡是以data开头的你都不能访问就完了。好，那么完成以后，下面我们就来做一个测试，启动一下。

好，那么启动完以后，我打开浏览器，然后先访问一下首页，然后先不登录，但有人说我通过哪去点路径，其实我们在这个hide里不去配这个路径，因为这个路径默认普通用户，甚至连版主都看不了，所以就没有必要配了，管理员想访问怎么办？

他自己敲这个路径，稍微麻烦点自己敲。好，我敲一下，比如说我是一个游客，我就想访问这个data，你看访问不了。对你没登陆没有权限对吧？好我登陆了登录了，我家访问了，对他访问不了，不存在没有拒绝是吧？Ok，所以说这个权限首先是生效的，要想访问怎么办？你必须得有用管理员的身份。我比如再来一个21。

版主看看他能不能访问，然后data也不行是吧？没办法，我只能用管理员了，然后偷点11。

好，然后管理员登录以后他直接敲路径，管理员知道回车就可以访问了，我就填数据统计对吧？今天我是第一次起这个服务，其实就刚才操作了那么几下，就那么几个请求，我们看统计的对不对？首先我统计UV点这儿今天依然是今天然后我开始统计一个因为什么？

虽然说我换了几个用户，但是UV是按IP来统计的，都是一台电脑都是一个IP，就一个当然这个时间你可以调，比如说我调成26号到28号统计什么都没有，比如说我调22号到28号，统计一个，因为今天包含在这个范围之内，是吧？就这样了。

好，这是统计优惠，我在统计活跃用户，比如说今天还是今天统计三个，这是按uzid来统计的，你看刚才我是不是用普通用户，我用这个还有版主管理员，三三个用户分别登录我对吧？

所以说是三，而游客那个是不算的，对吧？所以是三，当然我再换个时间，比如说换成明天到28号，没有了是吧？然后换一下换成22号到28号，三没问题。当然因为我现在数据量小，所以说虽然小有好处也好算，那么数据量多的时候你再看一看，你自己去看好了，那么关于网站数据统计，我们就讲解完了。好，这次课我们就演示到这里，咱们下次课再见。

## 7.6 任务执行和调度

48-任务执行和调度.mp4

这次课我们来学习任务执行和调度，就是说有些功能它并不是我们通过浏览器，然后主动的访问服务器去处理的，有些功能它是服务器定时启动定时运行的，它自己才能运行。比如说每隔一个小时我要算一下帖子的分数，比如说每隔半个小时我要清理一下，我们这个服务器上临时存了一些文件等等，那么像这样的需求，我们都需要任务调度的逐渐的去解决，很显然任务调度的组件它应该是基于独者程的对吧？

他俩的自动运行肯定是要启动一个线程，这个线程它独立的运行。

那么我们在程序中但凡是要用到独者程，那么一定是通过线程池去使用的，因为我们创建一个线程是有开销的，并且开销比较大，那么使用线程池去管理线程，能够让线程复用，能够提高处理能力，或者说能够节约一些资源。

所以关于多线程这个话题面试的时候也愿意问，那么问的时候一定会聊到线程池这个层面，所以说这是一个面试的时候比较愿意聊的话题，大家要多关注多去思考这方面的问题。

当然我们这次课要解决定时任务，其实我们最终用 class 工具就能搞得定，但是我想借着这个机会把线程池有关的内容咱们尽可能的全面的介绍给你，让你对现场职能有一个整体的了解。

首先就是jdk它就自带了行程尺，然后常用的有两个，一个叫SQL的service，这是一个普通的线程池，它能创建普通的线程解决一些问题。然后第二个常用的是这叫schedule is cute service。那么线程值所创建的线程能够每隔一段时间执行一个任务，它可以执行这种间隔的任务。好，这是jdk自带的两个线程池，我们一会会给大家演示。然后我们使用的是spring框架这个框架，它也帮我们内置了线程池。

那么第一个叫做常用的叫做thread pull task sqt其实它的作用和一样是一个普通的线程池，创建普通的县城。第二个是spread pull task。Schedule它的作用和这个一样，就是说它所创建的线程能够执行定时任务好，这是spring。

但是你要注意就是说我们普通的线程池我们可以拿来用，但是我们定时任务，那么在分布式的环境下，我们分布式部署的时候，这个用的时候可能就会有问題，因此我们其实在分布式环境下，通常用 Class 工具的时候更多一点，那么它适合用来解决分布式的定时任务，当然我们不是单独的使用这个工具，我们是用spring去整合它，然后这个是 cas的官网的网址大家可以去看一下，打开以后大概是这么一个界面比较简洁，然后中间有个按钮叫document， documentation你点进去以后是他的手册，那课上我就不带着大家去看了，因为比较长，但是我一会可以给你演示好了。

那么我们在写代码之前，我想再说一下为什么说在分布式环境下，我用这个东西或者用这个东西做定时任务会有问题，有什么问题，为什么class它能解决这个问题？我们先把了解一下，了解完以后我再依次给大家演示这些工具怎么去用。

好，那么要说这件事我需要画一个图，分布式部署，就是说你这个服务器得有多个节点，做成一个集群，我就少花点，我就花两个，比如说我部署的时候有服务器一然后当然服务器里有什么样的程序，它其实有两类程序，一类就是普通的程序，比如说 CTRL这样的程序，这样的组件它能解决普通的请求，然后它里边还有另外一类程序，就是我们的定时任务，这个定时任务我取名给它叫schedule，好。

好了，总之笼统的概括来说，我们服务器有两这两类这样的程序，当然了我这不只是一个服务器，还得再有一个或者是有更多个，这里我就只画两个。

好，那么我们在集群环境下，我们通常浏览器不是直接访问某一个服务器，这个服务器它的前面需要有一个负载均衡的工具，通常我们会用 nx，那么浏览器发出请求是发给恩杰斯，他发请求发给恩杰斯，然后杰克斯会根据一定的策略，然后把请求分发分发给某一个服务器，假如说这次请求它分发给服务器，一如果这是一个普通的请求注册登录查看首页，那么他肯定是把请求分给了一个controller去处理，处理就完了，这个没什么问题。

虽然说这两台服务器里头的代码一样，但是它同一时刻只能有一个，或者说对于一个请求只有一个服务器去处理，这个是不处理，所以说他一样也没有什么影响，没关系是这样的，但是对于 schedule对于定时任务就有问题。

对CTRL这没问题，定时任务有问题，因为两个服务器都有同样的代码，都有同样的定时任务。

这俩服务器一启动以后，这个定时任务你不用主动去访问它，就自动的在那跑，它每隔10分钟就做一件事，假设你想想这俩代码一样，每隔10分钟他俩都做同样的事情，这不就重复了吗？比如说每隔10分钟我要清理一个临时文件，删掉临时文件，他俩做同样的事情，重复了，甚至会产生一定的冲突，对吧？

所以在分布式环境下，这个sky就不太合适。这就是为什么我们直接使用jdk的 Sky的或者是spring的就不太合适，因为他们没有解决分布式部署的问题，他们是各自为战的。那么如果我把它换成我们最后所说的 cart，工具就能解决问题。好，我把它改成 cart。

好 chaos怎么解决问题？我们要说一下GD k和spring的定时任务组件，它是基于内存的，你配置它说我多长时间运行一次，它配置参数是在内存里，这个服务器一服务器二它内存不共享，所以说它们之间没有办法进行一个数据共享，没有办法知道说你在干嘛我在干嘛，所以说他解决不了问题。

但是快它的就是定时任务的这些驱动的这些参数，它不是存到内存里，它是存到数据库里，而我们的数据库就只有一个，这样的话我先画一下，比如说这是数据库 Db，那么快as它程序运行所依赖的那些个参数存到了数据库里，所以说不管是你有多少个应用，你部署了多少个快儿子这样的程序，那么它都会访问同一个DB。

比如说10分钟访问一次，虽然说10分钟到了这两个程序几乎同时访问这个数据库，但毕竟有先后，即便同时访问，那么他们也可以通过枷锁的方式去抢到，优先抢到资源，假设 class程序优先抢到了 DB资源，然后他是后到达的，他先到达他后到达。

那么对于 class程序他先防了这个表里的数据，他会看一下我这个任务当前的状态，你是在等待运行还是已经运行的？他一看是等待运行，因为是他先访问的，他就立刻把这个数据改成运行状态，然后的话它就运行去了。当 cart程序仿这个表的时候，他一看这数据变成了运行状态，就证明已经有别的服务器已经有别的程序在处理这个任务，他就不去处理了。

所以说通过这样的话，多个科尔的程序，他们能够共享数据，或者说他通过同一份数据实现了一个也可以说排队，或者说也可以说枷锁这样的一种机制就解决这个问题好了。

因此我们最终的目标是要学会使用快去解决定时任务，这是我们最终的要应用了这个方向。

但是我们在学习它之前需要先了解 jdk的线程池和spring线程池，那么首这两种各自的这两种线程池分别怎么用？你对它有一个比较全面的了解，这是我们这节课的完整的目标。好，下面我们就来写一些程序来体会一下这些工具的使用方式。这个程序都是小事例，都是 Test，所以我创建 test给大家演示也比较简单，好，那么我在这个包下新建一个测试类叫做 spread破线程池test，那么我需要写上那三三行注解，我拷一下。

好，那么接下来我就在测试类当中依次给大家演示，刚才我所说的那些个工具演示之前，首先我先室内画一个logo，我们在使用线程做一些事情的时候，最好通过logo去输出一些内容，因为 logo在输出内容的时候，他自然而然会带上现成的ID，并且会有这个时间，所以我们看起来的话这个信息比较完整，适合在多线程环境下去，用它来输出一些内容。

好，那么有了log以后，接下来首先我给大家演示一下 jdk的线程值，那两个先演示第一个jdk普通的线程池，但JD其实好像有五六个县城池挺多的，我说演示这两个是比较常用的，其他的不常用的，我就不演示了，但是你也可以去查阅一些资料做一个补充。

好，第一个比较常用的是 sqt service，好，我直接的把它实例化一下，然后后面的程序你就可以直接用了。那么jdk的线程尺，它都是通过一个工厂来实例化的工厂，就叫sqts然后的话 New fixed spread pull。

然后写一个数我写5，就是说线程池初始化以后它里边包含几个线程，包含5个，它是反复的复用这5个已经创建好的线程，这个我们一会儿通过代码能够看出来。

好，然后再来我再把另外一个也初始化一下，就是jdk他所带的可执行可定时执行任务的线程池，这样可执行定时任务的线程池。

好我也把它实力化一下，这个叫做scatt的sq的service，它取名就叫x Guide you的sq的service，然后也是通过 sq去转化 executors，然后点new skill，然后 straight pull好，我也写个数，也是它里面包含了线程的数量。

那么初始化完这两个线程池以后，接下来我就写两个测试方法来分别演示一下，那么这两个线程池怎么去用？然后呢你要注意我们现在用的是解忧类的测试，在这里面我们写的是解忧类的测试方法，那么解决问题的测试方法它和 Main方法不一样，如果我在一个main方法里我启动了一个线程，那么这个线程如果说不挂掉的话，main会等着它执行，它不会立刻就结束。

但是 test的方法几个类的方法不一样，它启动了一个线程，因为它启动的子线程和当前的线程是并发的test的方法，如果后面没有逻辑的，它就立刻就结束了，他也不管你提供的线程完没完成，有这样的一个问题，怎么解决也好办，那么我在test的方法启动完一个线程以后，我想等一下等他执行完我再结束怎么办？

我可以主线程或者是当前的线程实力的一会儿阻塞一会儿，而斯蒂芬他又老是抛异常，为了吊起来他方便一点，我把斯蒂芬稍微封装一下这样。

好，写一个简单的方法，sleep、参数、浪m毫秒，m是毫秒，好，每次调这个方法就是让当前线程阻塞一下是right，点sleep m当然这个需要处理异常，我就开始一下就完了简单处理。

好，接下来我就演示这两个线程池的使用，然后这是第一个，我演示的是jdk的线程池，jdk普通协同式写一个测试方法，这个方法我叫test，is cute。

Service就演示它的使用，那么现场使用它就是说你需要给他一个任务让他去执行，然后他会分配一个线程来执行这个任务，而这个任务其实就是一个线程体，我们通常都是实现让它包接口来提供线程体来提供这个任务。

所以在开始我实现一个random接口，软的 task，这是一个任务，等于没有软的，我对它直接做一个匿名的实现，然后 run方法里就是写上你要执行的具体的任务的逻辑，这里我简单一点，就log，然后debug。随便写一句话，hello execute service。

好了，好，写完以后我就用这个线程池来执行它，线程池里不是有多个线程吗？我执行多次，每次他都会分一个线程的来执行这个任务，我执行多次，我从0到10，反正我一共执行10次，每一次我就是Diao Sq的servis，点儿 submit，把任务传进去就可以了。好了，反正你调这个方法，subway的方法线程池就会分配一个县城，然后来执行线程体就这样。当然了别忘了死力配一下，要不然这个方法一执行，这个线程还没执行完，他就结束了，我死力了一下，等一会死。

力配1万毫秒，也就是10秒钟就够了，好我就执行一下。好，那么执行完了，其实执行的就是10次，执行的应该是很快的，我们看一下控制台的输出，看这个底就底，然后方块号里边它什么spread？

1spread5其实就是线程池的应该是名字，线程10的名字，你看这就是第一个线程，第二次分配到第五个线程，然后15351342，反正哪个工程就分哪个，你会发现反正就是123 4 5这5个县城来回折腾，就说明确实这个县城是两反复的使用这5个县城。

好，然后输出的内容倒没什么好说的，就是这一个字上这一句话我们主要是看所以说确实看出来这是一个独者程的执行的环境和结果。好，下面我再给大家演示一下可执行定时任务的线程式。再写一个测试方法，就是jdk定时任务线程式，这个方法我叫test，schedule is killed service，Call一下太长了，那么这个线程尺它的功能就是它可以设置时间间隔，然后不断的去执行。

当然了他执行的时候也是你要给他提供一个线程体，线程体的提供方式和这个是一样的，所以我可以 copy，我又定义了一个任务，这个任务还是输出一个日志日志，我说出这么一句话，hello scheduled skto to a service好了，那么我怎么去执行定制任务，我就调用schedule sq的service，然后点一般我们教这个方法 schedule at fixed rate以固定的频率去执行，每隔一段时间执行一遍能执行多次，你也可以调另外一个 schedule with fixed delay，以固定的一个延迟去执行，这个是执行一次，但是执行一次它就结束了，但是你可以指定一个时间，推迟多长时间执行一次这样。

好我用可以执行多次，然后这里边需要传几个参，第一个传入任务，第二个你要传入就是这个任务是你延迟多少毫秒才执行，它并不是说你一调它立马就执行，它可以做一个延迟，比如说我延迟1万毫秒，你看这个参数的含义是initial delay延迟的时间就10秒。

然后第三个参是时间间隔，它可以反复执行的，时间间隔是多少？1000毫秒，周期是1000。第四个是你声明你这几个数字，它的单位是秒还是毫秒，你可以用 Time. Unit，然后点儿美丽塞克斯毫秒就这样。

好，这句话一执行这个任务就连接线程池，就会按照我们所规定的参数周期不断的去执行这个任务，这个就不用启动多次了，因为这一个线程池我们调这一次他就能反复执行，所以说你没有必要调多次，就这一句话他就能反复执行，当然我也需要阻塞一下，然后等一会儿等时间久一点，我等3万毫秒就30秒，这也就够了，我再执行一下这个方法。

你看他没有立刻输出内容，他得等一会儿等10秒，这说明delay参数就有效，10秒以后它开始执行时间间隔是每一秒输出一个内容是吧？然后它一直在这输出的，直到这个方法停止为止。

然后当然输出的内容其实没什么好看的，就是hello那句话，我们主要还是看现成的名字，你看11完了就30秒到了就结束了，你看11213对吧？每次执行的时候，每次调度的时候，它会分一个新的线程来执行 task，所以这这也是一个独者程的环境，也能看得出来。

好了，这是GD Kay的现实比较简单，Sabrina也给我们提供了线程池，然后也比较好用。

我们再来演示一下 spring的形成池，那么spring只要我们引入 spring的核心，它就自动就启用了行程池就可以用，但是我们需要在 application里头对 spring行程尺做一些配置，你要配一下有它启动以后它带几个线程这样的一些数据，所以我打开配置文件，application practice我再加一项配置，那么 spring给我们提供的普通的线程池，它的我们配置的对应的类叫做task，然后 SQL execution，task，sqc process。

好，这里面我们首先要配的是spring点，task点儿，sqc点儿破size等于几？就是你线程池里有几个线程？好，再来。第二个我们往往要配 spring点task，点excuse，然后我刚才这个写错了，不是破-3是这样的，是破点。Com site就是核心线程数量，线程池里边的核心线程数量是5，然后再来这个是破点，max site我设置为15，什么意思？就是说当我线程池创建好以后，它默认的在5个线程，但是如果你调用的比较频繁，你使用的比较频繁，线程不够用的时候，它会去扩自动去扩容，最多扩到15，这个意思。

好再来第三个我们还要配 sq spoor，然后点 q capacity，然后比如说我设置为100，这是什么意思呢？

Q capacity就是队列的容量，那么比如说就是说我这个线程一共已经到了15个最大线程数量了，但是可能还是不够用，你还有更多的任务提交给我，我15个现在不够用怎么办？他会把任务先放到一个队列里，然后当它有空间线程的时候，从队列里取上再去运行队列起到一个缓冲的作用，更大程度的提高线程时的性能，你这个队列一共能缓冲多少个？

任务，这里配的是100个。好，这是普通的线程池配置，那么能启动定时任务的线程池怎么配？它的类是这样的，是task，然后 Scheduling，practice，task，schedule and practice是这个类，然后我们一般只要配一项内容就可以了。Spring的 task，Scheduling gap。Tool点。Site等于5，就是说你线程的数量，线程池里装的线程的数量是多少？是5，有人说你怎么没有说 max什么的，其实没有必要不需要，因为你这个是普通的县城，那么你要执行多少个任务，你需要我执行多少个任务，这个不一定。

这种现成的这种任务它是可以通过浏览器访问，然后主动触发的，所以有多少个不一定，但这种是定时启动的。

服务器通常是服务器一启动它就动了，我们是可以提前预判的，比如说我需要一个任务去清理文件，我需要一个任务去算分，我需要一个任务去做缓存，可能我就需要三个，你这个等于三就足够了，就这个意思，所以我们可以提前预判，看有几个比较合适好了，那么这个参数我就配好了，配好以后我们再回到测试类，然后继续写测试的程序好。

我回到测试类，然后还是我在开头的位置，先把 spring 的两个线程尺给它初始化好。

其实这个初始化不用我们初始化 spring，它框架已经帮我们初始化好了，而且它会把初始化好的线程池放到这个容器里，由容器来管理，所以我们要用的话就注入进来就可以，我把它注入进来。

首先我要注入的是死不用普通的系统池，然后再注入可执行定制任务的行程式。好，先注入第一个 out where 这个线程池的名字叫 thread。Poor。Task is cute。然后变量名我取的简短一点 task，sq 的再注入第二个，这个类的名字叫 spread，pull task 是该得了 task。Schedule。

好，下面我就写两个测试方法来演示一下，那么这两个线程池的使用方式，我做的事情和刚才一二做类似的事情，然后我们做一个对比，看看能不能达到同样的效果。好，这是第三个就是 spring 的线程池，那是普通的 test，类名我拷一下 test 的好了，首先我们也需要声明一个线程体可以 copy 类似的，然后也是打印一句话，打印 hello，spread pull passed sq 等。

然后对于普通的线程池，我要调 10 次，每一次执行一下同样的一个任务，我们看一下它的效果好执行 10 次，每次我调 task，sq 的，然后叫 submit task。当然了也得阻塞一下 sleep 10 秒，好，写完以后我来执行一下，报错以后我们看一下这个提示，它提示说找不到 schedule，不存在我想起来了，我忘了一件事，你要想让 sky 能够这个生效，我们还需要做一项配置，我们需要写一个配置类忘记了，所以说 sky 没有被正确的注入进来，这样我在肯费格目录下新建一个线程池有关的配置类，这个类我叫 spread，然后加上一个注解看微信，然后我少加的东西是这个东西叫 enable schedule。

如果你不加注解，就说明定时任务它没有启动，这个东西比较敏感，默认它不启不启用，这样我就启用了才能用，然后通常我们再加上另外一个注解，盈利宝，那么就是说一会我们可以用，这个是一个注解，注解可以启用的，有什么用我们一会再看。

好写完配置了以后，我们再回来再来执行这个方法。

好，当然其实已经输出完了，但是它程序没有结束，因为它主次在这儿的，然后你看这个输出也是输出了 10 句话，debug 对吧？然后每次输出的内容就是 hello，然后你看现成的名字，14、325，反正一共是 5 个线程，他来回换来回分配也是一个独者程的环境。

总之它的执行的效果其实和刚才这个是一样，但是它的好处是我们线程池配的时候可以配核心的线程数量，最大的线程数量，如果超出这个，它还有一个队列的缓冲，所以说它比 jdk 自带的线程值它更灵活，所以我们要用的话优先用这个比较好了，那么接下来我再写一个方法来测试一下 spring 的第二个线程池能执行定制任务的线程池，这是第四项内容，spring 定时任务现场使用。

因为这些方法其实每个方法它所做的事情很类似，如果不写清楚的话，我怕你课后忘了，所以把它注释写的清楚一点，这个类我也拷一下。Test，thread，pool，task，schedule 的同样的我也需要一个线程体把它拷过来，然后 hello 这句话。好，然后我就调用 task。Scheduler。Schedule as at fixed rate，以固定的频率来执行这个任务。那么它的第一个参数写上 task，第二个参数你要写上，它不是说延迟多少毫秒去执行，它是要写一个具体的时间，一个 data，所以我提前先把 data 初始化一下，data 然后 start 看我就开始时间，开始执行的时间你又对这里我写上 system。

第二 carry on the time maybe 就是当前的时间毫秒数我要加上 1 万毫秒，说白了就是当前时间再延迟 1 万毫秒，其实也是跟延迟一样的效果。这里写上 Start 碳，然后第三个参数写上它的时间间隔 1000 毫秒也就是一秒，然后这个方法它默认就是以毫秒为单位，所以你不用指定单位就可以了，这样就行了，当然也需要阻塞一下实例 3 万，阻塞 30 秒。

好，写完以后我再执行一下这个方法。你看一开始他没有执行，他在等，现在也没有执行得等 10 秒以后才能开始输出，输出挺大的，开始输出了，然后时间间隔是一秒，每次输出这是现成的名字，它是不断的在换，反正每次都换。

当然也有的时候很多时候是一样的，但是你能看到他确实在换，然后直到30秒到了整个程序结束为止，所以这个方法的执行和刚才二一样，好就这样，这是spring的两个线程池，其实这两个线程池我们在使用的时候还有一个简便的调用方式，我再给大家演示一下简便的调用方式，那么简便的调用方式就是说我只要在任意的一个变量里说明一个方法，你该写什么逻辑写什么逻辑，然后只要在这个方法之上加上一个注解，那么他就可以在死认线程池这种环境下去运行，或者说我们在bn中写的方法，我们就可以把它作为一个线程体，而不是说像这样就是一个普通的方法就可以作为线程题，这样就更灵活了，更简洁了。

好，给你演示一下。我随便找一个组件，比如说阿尔法service，以前我们也是在这里写一些测试程序，对吧？我在这里再补充一点东西，随便写个方法，public want这个方法我叫sq一，然后这里边我就还没有初始化这个Lague，我初始化一个Lague。好，那么有了logo以后回到刚才的sq的一方法，我还是就在这里简单打个日志 Logo点大，然后的话写sq的一简单点儿完了。

好，这个方法你这样写，它当然不会以多线程的方式去调，无论怎么写，我们需要前在前面加一个注解。

那么注意刚才我们在配置文件里为什么写这句话，其实就是让注解生效。那么这个方法加了注解以后，能怎么样？可以让该方法在独者程的环境下被异步调用，能够让他在多线程环境下被e部的调用，也就是说我一个线程，我启动一个线程来调这个方法和主线程是并发执行的，是一步执行的这个意思。

好了，当然你这还得有地方调，现在还没调。好，我们在这个测试方法里再去调了一下，我再回到Test，这是我要演示的第五个内容，就是spring普通线程池的一个就是简化的使用方式，他的word test。Test什么还叫这个名字 test，他然后先把简化的方式，当然了我要调的是阿尔法servis，你得把阿尔法servis注入进来，对吧？不然我怎么去调 Biehn的方法，好，我在前面把它注入进来。

好，回到刚才的位置，我在这里我希望调那个方法调10次，那个方法其实就相当于是一个任务，就相当于是一个线程体，现在我要调10次，我就直接循环十遍，每次循环里我就阿尔法service，然后点sq的一就可以了。

那么在程序的底层，spring它会以多线程的方式去调这个方法，它会让这个方法就是作为线程体，他会用线程池去调的。好，我们执行一下看看，那没有输出结果，就是我忘了一件事儿，我还是要在这阻塞一下，不然的话这个线程还没执行，这个方法就已经结束了。

实力培训就是实力10秒，好再来你看很快就输出了这个结果，然后他也是输出了10句话，然后我们看这个线程的名字1245对吧？也是用到了县城治理这个县城也是复用了县城治理的县城，也是独者程的环境，这样的定位方式就更简单了，更方便。

好，那么定时任务的线程值也可以用这种简化的方式来调，我们再来看一下。

我再回到阿尔法，Siri上再写一个方法，这个方法叫 sq二，然后简单一句话，logger dady Bugg excute，完了，那么我要想让这个方法定时的去执行，那么我只需要加一个注解就好了。

下一个schedule的，然后我在注解里可以写一些参数，你搜得类比方说这个方法，我调完以后它延迟多长时间去执行，我希望延迟10秒执行。

然后再来个参数 Fixed，wait，固定的频率就是多长时间执行一回？

1000毫秒也就是一秒这样，然后注解其实还有其他的就是配置的方式，其他的参数，那么我们就不挨个演示了，这种这两个参数是比较常用的方式，你做一个了解就ok了，其他的如果你想更多的了解的话，去翻一下官方的手册。

好这就可以了，如果说我一启动这个程序有了注解，他立刻就按照这个逻辑去执行了，这样我再写个测试方法，通过这个测试方法去驱动，然后让他去执行。

六就是思路定时任务线程尺，当然这也是一个简化的方式。

这个方法名我就还叫后面加一个sample，那么这个方法其实我不需要做什么，因为刚才我们所写的方法它是自动被调的，你不用主动去调它，但是我想让这个程序启动起来，我这样我就直接 sleep30秒，然后那个方法就自动会被调，只要我有任务启动，只要我有程序在跑，它就会自动被调。

好，我们试一下执行，一开始都没有执行，因为它延迟10秒对吧？得等一会儿好，然后10秒以后开始执行。

你看输出的内容sq的二没有问题，然后每次打印的现成的名字123这样的，说明这确实是一个多线程的环境，这种方式就更简单了。好了，那么现在我们就把jdk的线程池以及使用的线程给你做了一个演示，你这两种线程池就都会用了。当然了还是那句话，如果说我想用一个普通的县城，你用就可以了。如果你想用定时任务的这种线程，这个在分布式环境下有问题不太好用，那么我们需要学 class，所以接下来我们来学习 care。

那么刚才我也画了图说了一下，就是说科尔它是依赖于数据库的，那么它有一套表需要我们要用的时候需要提前的去创建，那么这个表我第一章讲买卖s的时候，我给大家提供了一些社科的脚本，大家也进行了初始化，其中有一个文件叫tables刚买circle杠，in，DB的circle，这个文件里的Biao就是cared所需的Biao，你需要打开命令行工具，然后用source命令把它导到我们的 community库里头去，这件事我提前已经做好了。

我们来看一下一共有好像有11张表，这个表都以qrtz ks的缩写来开头，这些表，然后这些表你不需要说把每一张表的结构搞得清清楚楚，我们大概了解它里边存了什么东西就可以大概做一个了解。那么哪几个表值得我们去看看呢？一个是一个是。还有我们通常看这5张表就能看出点意思来。

当然我们在没有看之前，有你一开始你不了解快儿子，你看这个表你也看不懂，我需要先对快儿子给你做一个大概的介绍，你对他有了一些了解以后，再来看这个表可能就有这感觉了，主要是给大家说一下 Class它有它在这个工作的时候，它有几个核心的组件，这些核心组件是什么关系？首先我直接对着这个源码跟你说，首先它有一个核心组件叫schedule就调度器，中央调度器它是整个卡尔斯的一个核心，当然现在我还没法去打开那个类，因为我还没有导入卡尔斯的包这样我表现好了，然后的话我先把 quest的包先导进来，然后我好对着源码给你看那几个类，然后我们再解释这个表。

要倒包的话我们需要打开 maven repost的com，然后我们需要去搜搜什么 spring boot cat好搜一下。很显然这个 Spring boot travels start我点进去选一个 kopje，我把它拷到我们的配置文件里面，泡一点 xml。当然了，因为这是死不用不的提供的包，那么它的父类一定是对版本是有声明的，所以说这个版本可以去掉，就附那种上面的就可以了。

好，那么导入了包以后，我们来说一下快儿子它有几个核心的内容，核心的组件，一个是schedule，孵化器的class，那么这个类是class的核心的调度工具，那么所有的由ks来调度的任务都是通过接口去调，它是核心。

后面我会演示我怎么去直接使用这个接口，先对它有个印象，然后你要知道其实我们定时任务主要是干什么，主要是定时每隔一段时间去执行一个 task，去执行一个任务。

么class是对整个流程做了一个设计，首先我们要用class来解决问题，首先我们需要定义一个任务，那么定义任务通过 job接口进行定义，它有一个job接口，那么这个接口只有一个方法，sq的方法这个方法里头你要写上你要做的事，你要执行的任务。

好，这是我们使用class其实第一步，而刚才的 schedule其实不用我们去写，其实它底层已经实例化好了，我们可以直接用 job需要我们去写。好，然后 job只是声明了我要做什么，那么具体怎么做，我们还需要进行配置，还需要另外的两个接口的配合。

另外还有一个接口叫job detail，job detail它是 job详情，其实它是用它来配置 job的，他要用它来配置照你的名字是什么，你的组是哪个组，然后对这个照片有一些必要的描述，以及相关的一些参数的配置，是对这个照片进行配置的。



然后还有一个接口叫崔哥，崔哥是触发器的意思，那么它用来配什么？这个兆什么时候运行，以什么样的频率反复的运行，所以其实就是我们使用cart，我们编程的主要是三个方面，一个是通过drop接口定义一个任务，第二个是通过dropped kill以及trigger接口来配置这个状，主要是做这三个方面的事情就可以了。

然后我们配置好以后，怎么样配置好以后，当我们程序启动的时候，那么 Cart就会读取配置信息，并且把它读到的配置信息立刻存到数据库里，自动的存到那些个表里，然后以后他就去读取那个表，来执行这个任务，只要数据初始化到数据库里以后，这个配置其实就不再使用了，就这个配置只是在第一次启动服务时用一下，以后就不用了，是这样的一个逻辑。

好，你大概了解这三个组件以后，我们再来看一下这个表。

我说打开的 job detail，就是对 job的一个详情的描述的一个表，我们通过job detail去配置的那些信息就存到这个表里，这里存在什么 schedule，name就是整个任务的名字，job name、这个job的名字，job的分组，job的描述，job对应哪个类等等，就是这样的一些信息，总之这个表存的是job与drop有关的直接的配置。

然后这个表 class simple triggers是还有 class triggers里边存的都是和触发器有关的配置，只不过 simple崔哥这里面存的是崔哥简单的内容，这个是存的比较完整的内容，他也存了任务的名字，然后处罚界哪个组，他反复执行了多少次这样的，然后完整的信息里还存在什么，还说了这个触发器对应的是哪个兆，这个照片是哪个股肉，他也存了相关的照片的信息，然后还存了我下一次应该什么时候执行，我上一次什么时候执行完了这样的一些信息。

然后 class schedule的state存的是定时器的一些状态，纯粹是定时器的实例，定时器的名字，然后检查的一个时间。第二期检查的一个时间，就是说它定时器在那自动运行着，它有没有好好的运行，它有没有问题，那么这个快要在底层的会定期的检查，看你到底是不是ok的，这里边会做一些记录。

最后这个是class，这里面存的是这个组，就是定时器的名字以及锁的名字。那么当多个cart在访问这个表的时候，那么它会以这个名字来对数据进行加速，是这样的。

好了，大概了解一下它底层表的大概的原则它里边存在大概的信息，然后一会我们执行的时候这里数据就会生成，到时候我们再看一下好了。

那么大概了解以后，下面我们就来写一个例子，我们来体会一下用 Class到底是怎么样去解决问题，那么是这样，首先我们第一步要做什么？

按照刚才我所说的第一步我们要定义一个任务，用 job接口来进行定义，我在这边新建一个包叫我把 cart有关的任务都放到这个包下，现在我要创建一个job，这个job其实就是一个事例，没有具体的业务，我叫阿尔法兆，然后它要实现接口 implement，实现它的对应的方法，这个方法做什么？我就是简单的打印点东西，简单点 This out。然后 spread点 current。Spread get name就是我要打印当前线程的名字加上冒号。

然后 it's cute a crass job。就等于一句话说当前线程它执行了一个任务，这job就完成了，就非常简单，反正就是一个方实现一个方法，实现完以后，我们需要对 job进行配置，这样的话科尔的底层才能够读取配置信息，生成表里的数据，才能够让这个任务运行起来。

下面我就来写配置，那么我就在config包下新建一个配置类叫cat，那么当然了配置类也需要加一个注解，consideration，然后写注释就是配置的作用是干嘛的，他不是每次都有用的，配置的作用仅仅是在第一次的时候被读取到，然后初始化到数据库里，它的信息它封装的信息被初始化到数据库里，然后以后 class是访问数据库去调度这个任务，而不再访问配置文件这样。

好，那么我们要配的是两个内容，一个是drop detail，一个是崔哥，所以我需要先写一个b然后 public先配 job detail，job detail。Fact就是be有人会觉得很奇怪，你不是配job detail，这怎么又出来个 factory b一会我再说我先给它取个名字，给这个病取个名字叫阿尔法drop detail。

然后这里我先先给他们闹，要不然他报错难受，接下来还要配吹梗。

我再写一个b然后 simple简单的崔哥，然后 factory bean也带上了factory bean，为什么一会再解释他的病的名字，我取名叫阿尔法学者，那么我们在初始化崔哥的时候，崔哥是依赖于 job detail的，刚才我们看表也看到了，崔哥他是和job有关系的，但是 job detail有一个直接的关系，我们需要把这个病注入给他，所以这里边我上面的参数叫job detail，它需要读取job detail，我把它传进来，而这个job detail变了，名字我这样写。

好了，到这儿你肯定有点晕了。一会儿一会是job detail factor bean，这到底是怎么回事，我需要解释一下。其实 spring底层它有很多地方都有这样的病，都有这样的名字，叫什么fact？就是病 factory并和之前我们一开始是LC的时候，并factory有本质的区别。你还记得那个being factory吗？它是我们整个LC容器的顶层的接口，但是我们所见到的 factory并和就两码事。

这个还经常的是一个面试题，可能是笔试会有这样一个题，所以说并发个人你已经知道了，它就是顶层容器的顶层接口，这个factor bean你在这了解以后你就能回答这个问题，这里我解释一下，我把它写到注释里，这样的话你好，课后的话我复习 fact bean它的作用是简化变的实例化过程就是说 fact并的主要目的是为了简化并的实例化过程，因为有些病它的实例化过程比较麻烦，那么有了发推病以后，它的实例化就容易了，或者说你可以这么认为类的，底层封装了job detail详细的实例化的过程，它对它做了一个简化，具体来说它有几个层面我们需要理解，第一个层面是通过三个病封装了思路，要通过三个病封装了，病的实例化过程，是某一些病实例化的过程。

好，第二个你要知道的是我们可以把 fact bean装到容器里，我们可以将 factor bean装配到spring容器里，那么装配到容器里以后，当然其实我们要的不是这个fact，并我们要的是job detail，但是你注意它底层封装的job detail的一个过程，那么会怎么样？接下来我们可以将 Factory bean装在容器里以后，我们将它注入给其他的病，我们先把它装备到容器里，我们再把它注入给其他的病，那么该病就其他的病得到的是什么呢？

是三个病所管理的对象实例，你看我在这儿我初始化的是job detail factor bean就相当于我把它装配到了容器里对吧？

然后在这儿我这个参数上我需要做个得听怎么办？那么我就把把它这个是变的名字，我把变注入进来的时候，那么默认情况下，我得到的不是factor bean我得到的是它里边管理的对象，job to kill，所以类型是job bean它是这样一个关系。好了，总之他通过这么一个非常感觉有点别扭的方式来实现对病人的简化的过程就总之我们去初始化，这个东西要比初始化装备的q要容易多，就这个意思。

所以这里我这样写好了，我如何初始化照不得跳，发对并的写不出事情，我是主要的目的是要配置 job detail，而这个我要配置的是崔哥把这两项配置解释清楚，再一个那么我们要配 job的q我们要用这个job并简化的方式，我们只能用这个类，然后我们可以崔哥也要用fact并的方式的话，其实我们有两种选择，一个是simple崔哥 factory并，还有一个是crown。Trigger factory。这两个都可以，但是它的模式不一样。这个是简单的崔哥，就是说比如说我每10分钟要执行一次，他能搞定，这个是复杂的。崔哥。它能解决什么问题？比如说我每个月月底的那一天，半夜2点我要做一件事，用这个可以搞得定，比如说每一周五的晚上10点我要做这个事儿，我要出发，他能搞得定，它里边有一个特殊的表达式，它只要用一个表达式就能搞定这么复杂的一个逻辑。

那么你通过官方的手册可以去了解它表达式，或者你去搜 Crown表达式其实有很多解释，然后其实看起来很神奇，但是很简单，好了我们就来初始化这两个病，首先我在这要先使的话，这个照片就反馈好，先实例化对象，然后给它设置一些参数点散照。

Class你要声明你管理的是哪个病，它的类型是什么，我就阿尔法job加class。好，然后再来 factory bean。

第二site name你要声明的 job的名字叫什么？你给这个任务取个名字，我叫阿尔法，job这个名字不要和别人别的任务重复，否则会有问题，然后发觉并点sat group，你给这个任务取一个组，多个任务可以同属于一组，这个组我就叫阿尔法兆 group，这样取个组名。

好，另外 factor been点set, deal ability。一般我们设置为true，这句话的意思是声明一下这个任务它是长久的保存吗？是持久的保存吗？是的。那么就是说哪怕是这个任务将来不再运行了，可能他连触发器都没有了，那么他也一直存着，我们不用删留着。

好，还有 Factor in点sad requests, recovery, 那么它的意思是那么你这个任务是不是可恢复的？如果说你应用程序有了一些问题，然后的话应用恢复以后，这个任务它是不是可以被恢复可以恢复。好，那么这就完了，最后返回fxb总之还并不难，主要是设置这么几个项就ok了。

好，然后我再来同样的方式设置这个水准，当然我首先需要实例化的，然后 Fact to be点儿sad，着得挺，你要设置一下崔哥儿他是对哪个照片做的处罚器，但是你传的要是传这个照片，其实就是这个参数，传进来其实就是它，你要知道就是我们的死人容器，将来我们如果任务多了，job得就发特病，它有很多个实例，或者说 job detail它有多个实例，怎么区分呢？

可以通过名字来区分这个病的名字叫阿尔法jopp dio，我这个参数名叫阿尔法jopp, ditto GB同名，那么是不是会优先把同名的给我注入进来？所以这两个字保持一致，以避免说你实例多的时候它有冲突，名字要对应。好，先传入这个照片，又然后再拿三个病叫赛内，你要给你的崔哥取个名字，这个我叫阿尔法sugar，好，再来发出了病点sat group，我要给崔哥取一个组名，我叫阿尔法崔哥。

就是说这一组好再来 Fact就是been点sat, repeat。Intel就是频率，你多长时间要执行一遍这个任务，我写3000毫秒，就是每三秒我要执行一遍，再来虽然并点赛，知道吧？Data map。New job data map。好，这是什么？就是说崔克它的底层需要存储照顾的一些状态，你用哪个对象来存，你要指定这个对象，然后我就初始化了一个默认的类型，但是你可以自己去写一个新的类型，这里我是默认的就可以了，你就往这里存状的状态。

好最终委托人f2b完了现在我就把 Dropped ill和崔哥也配好了。那么我们使用class的要写的这三个方面就写完了，一个是任务，一个是任务的详情，一个是处罚器就配好了。

配好以后，那么只要我一启动服务服务启动以后，那么因为配置文件默认就会被加载，一旦被加载到以后，那么科尔就会根据这两项配置往数据库里插入数据，数据库里一旦有了数据，那么科尔的底层的调度器是盖德伦，他就会根据这里面的数据去进行调度，三秒三秒就调一次，下面我们就执行一下看看，我就不去执行一个测试方法了，我就执行我们正式的项目，看一下控台，你看是不是？

他已经在执行了，在启动过程中他已经在执行了。你看sq的 Class job这个job就被执行了，每三秒一次，然后每次都是有一个线程在执行，那么这个是线程的名字，34、567这样的。

好，当然了就是筷子它的底层也是依赖于线程池的，然后它线程池有一个默认的配置，那么如果说你想去做干预，你想去主动的把它底层的线程值再进行一个重新的配置也是可以的。

我们需要在 application process里进行一个配置，那么这项配置会比较长，会比较长，那么这样我就不去手写了，我直接从以前的项目把它拷过来，咱们看一下，都已经准备好了。

这一段这么长我就不挨个写了，拷过来以后我们大概再看一遍，拷到配置文件里。

在这它的类对应的类是 quite practice，大概看一下，那么这个是什么？是这个任务它存储的方式它是用jdbc来存的，它底层用的是jdbc然后这个是sky的name，那就是你这个调度任务的调度任务，调度器的名字，佛教community是该做的，再一个它是该得了点instant idea，就是你调度调度器它ld自动生成，然后 drop style。

这个我们任务在存到数据库里的时候，你是用哪个类来做这件事儿，是这个类。

然后你在存的时候用不用gdpc驱动是什么？驱动，然后的话你是不是采用集群的方式？是的，然后的话你用哪个线程池线程池，这是快RS底层带的一个线程池，然后它的线程数量我写成刚才你看它ld都已经到10了，线程太多了，我们用不了那么多，写成5就可以了。

当然如果我们刚才没配的话，它也起作用说明什么？死不认不得，对它做了默认的配置，你这样的话能够覆盖到配置，按照你的想法去配，这样也是比较好的。

好，改完以后我们先不着急再运行，我们去看一下这个表里的数据看有没有变化。

我先看这个detail，那么这个detail你看没有数据，你看崔哥估计也没有数据，为什么刚才其实我说的有点漏洞，如果说我们没有做 Process这项配置，那么我们启动服务以后，那么care的是读取内存中的配置来执行这个任务的。

那么一旦我们做了这样的配置，我要求要把任务持久化到数据库里，这个时候它才会把数据存到数据库里，默认的话它是读取内存的，刚才说错了。

好，那么我们做了配比以后，那么再去执行的话，它会把配置信息存到数据库里，我们要的是这样一个效果，否则的话我们分布式部署就会有问题了，像这样一种方式，其实分布式部署就会有问题。好我就再执行一遍，你看它又在运行是吧？然后每三秒运行一次，然后你这回去观察它线程的名3451。反正你多观察一会会发现它是不会超过5的，因为我们一共就要求他初始化5个线程，这就够了，就这样了。

然后我们再去看一下表里，看看这个数据有没有变化是吧？我们看一下这个点q你看这回就有了数据，因为我已经要求他把数据初始化到这个表里，然后你看调度器的名字是我们配的是吧？Job的名字是我们配的，主名我们配的，类名只我们写的对吧？然后这一代表true0代表false。我们这两个配了true就这样。

然后 simple triggers再查一下，有了数据就是调度器的名字崔哥的组以及后面的一些它执行时所产生的数据，这个是崔哥比较完整的信息，前面是类似的，后面的话它对应的状是什么？

然后的话这是上一次执行的时间，下一次执行的时间有一个记录，还有这个是状态，就是调度器的名字，调度器的一个实例的名字，然后的话他检查的时间，他上次什么时候检查的周期是什么，因为他要看一下你这个调度器是不是在正常的运行着，它要有一个检查，它是记录的是调度器的状态，然后 Locks锁，那么一个是状态的，一个是崔哥的锁，就是说他在检查的时候，他在枷锁的时候，他是对这个状态表加锁，对崔格表加锁，那么锁的名字是这样的，在这里进行了声明。

好我这里我就不想让它再继续运行下去了，我先关了。并且我希望以后我在启动服务的时候，他也别去输出这句话了，因为这句话没有什么逻辑对吧？每次就这么频繁的输出也挺烦的，但是现在你注意这个数据已经进这个表了，只要你一启动服务，它一读这个表有数据它就会运行怎么办？

两种办法，一个是你手动把这个表里数据删掉，还有一个你写一个测试的程序，然后的话调他的 schedule，然后把这个任务删掉。我们还是写程序方便，因为写程序的话一句话他能好几个表的数据都给删掉。

另外我们也知道一下我们删除任务的API是啥，所以我在这个测试类当中我再新建一个测试类，新建一个测试类，结果叫class tests，但是我也需要写上那么几个注解。

好，然后我要删除一个照的，我是要通过他的调度器去做这样的事情，我需要把这个调度器呢注入到当前的便利调度器，就叫schedule了，刚才我们说了。好，然后我写一个测试方法，在这个测试方法里我要删掉刚才的我们用来测试的job，delete job其实很简单，就是schedule，delete job。

然后得力的时候，它需要传一个就是job的key，Job的唯一索引，它需要传一个对象进来，比如说你有一个 job key，这个job key有两个数据构成，一个是job名字，一个是组名，job的名字我们叫阿尔法job。好，然后组名我叫阿尔法show up。Grow。那么由这两个名字能够唯一的确定一个job，它是job的key。当然了我们这句话我们需要传开始，然后这句话它会返回一个布尔值，告诉你是成功还是失败，我接受一下结果，然后我把伪造上的打印看一下。

好，这就写完了，写完以后我执行一下看看。出成功了成功以后，我们去看一下这个表，你看这个缺口没了。没了。是对的。还在，因为这里边存的不是做，这里面存的是sky的人，调度器还是有的 Log。这个还在，因为它也是存在是调度器有关的数据，但是 job以及trigger没有了。

好，当然如果你现在你再启动服务，其实还会有，为什么呢？因为肯费岗他又会执行一遍，又会生成数据，我希望不要再执行了，演示过就算了，我把 b给它去掉，你这样你再启动的时候，这两个病就不会被初始化，不会生成数据就不会被调度，然后配置类保留，因为以后我们解决项目中实际的问题的时候，这个类还还可以再加其他的内容。

那么我们在学习的时候就是执行过一遍测试程序就可以了。好了，那么这节课我们关于任务执行和调度就介绍完了，那么大家课后一定要对线程池多做一些了解，然后对 class 你也要掌握它的使用方式。好，这次课我们就演示到这里，咱们下次课再见。

## 7.7 热帖排行

49-热帖排行.mp4

这次课我们来实现热帖排行的功能。在首页上我们显示帖子列表的时候，可以以这个帖子的热度进行一个排名，什么叫帖子的热度？其实就是一个分数，只不过这个分数是通过很多指标计算出来的。

那么不同的网站，其实这个帖子也好，还是新闻还是一些内容也好，那么他们都有排名，这个排名其实都是算一个分，而这个分它背后的指标各个网站是不一样的，不过虽然说不一样，但是它是规律可循的，所以说我们来看两个网站他们的计算分数的公式是怎么样的，我们做一个了解，然后的话从中我们得出一些它背后的通用的这些逻辑，然后我们在设计我们自己的公式的时候，好有一个参考。

首先我们看 Hiker news 是国外的一个新闻的网站，那么它排行的内容就是新闻了，那么他这个新闻的分数是这样算的， $t-1$ 除以 $t+2$ 的 $d$ 次方，其中  $p$  指的是投票数，然后  $t$  指的是从文章发布，从新闻发布到现在的一个时间间隔单位是小时，然后  $g$  是一个系数，通常是 1.51.8 这样的数。

当然了我们不用去关注就是说一些特别细的细节，说它为什么减1，它为什么是加2，这个数为什么是 1.51.8，其实都是他通过一些实际的运营得出来这个数据它比较合适，我们加减乘除乘方 log 用这样的一些手段去调整数据，是为了让最终排行这个曲线变得平滑一点，它是这么一个目的。

那么我们从这个公式中可以知道这样一个大致的原则就是说。

我们这个分数是和这个内容受关注的程度，受喜欢的程度有关系的。

那么你越关注这个新闻看的人越多，那么他分就越高。另外这个内容它通常是和时间成反比的，你这个时间就是越往后推移，内容又变得旧了。

它。

很快就过气了，就排不上名次了，而且你看它这个时间外面还加了一个 $g$ 次方，这样的话时间增大一点，就被它给放大了对吧？所以说随着时间的推移，它很快就会下沉。

所以总之通过它也能够大概了解内容的排行和大家喜好程度成正比了，和时间成反比，通常都是这样。

我们再看第二个网站，就是 stack overflow 是国外的技术人员程序员交流问题交流心得的一个网站，或者你可以叫他程序员之家，其实我们都应该了解这个网站，然后它这个公式就特别的复杂，log，然后这里边有  $q_{vos}$  然后还乘了  $4^{q_{vos}}$  什么意思？

它是这个问题的浏览数量，因为交流这个问题，他是发问题回答这个问题的一个网站，所以这是问题的浏览的数量和它成正比，然后  $q_{answers}$  这个是问题的回答的数量，所以说分数和回答数量也成正比。

还有  $q_{sky}$  这是回答的，这是问题问题的。

赞采的差就是你点赞的数量减去点踩的数量差值，然后后面这个是  $a_{scarce}$ ， $a_{scarce}$  是回答的站起来的一个叉，回答问题回答的站减去回答的叉的一个总和，这些内容都是能够提分的和分数成正比的。

然后底下这是除以下面成反比的是  $q_{edge\ in\ ours}$ ，但是这个也是  $in\ ours$ ，这是题目发布的时间差，就是题目发布到现在的一个时间，总之它和时间是成反比。

再一个  $q_{update}$  是最新的回答时间，但这里它也是通过一些加减乘除一些系数来调整曲线的平滑程度，所以你看尽管它这么复杂，其实还是这个原则。

我。

看问题的浏览量，还有一些点赞的数量，这些是成正比，其他的时间成反比，大概是这么一个意思。

好，第三个就是我们牛客我们的这个帖子分数的计算的公式是我们也是通过长时间运营总结提炼出来的一个公式，当然这个公式像这些也是它不是一成不变的，可能随着网站的规模的变大，随着业务的变化会有一些调整系数可能也不断的去调，当前我们是这个样子。

然后是这样的是精华分我们这个帖子是不是有加精了，它会核算成一个分，然后加上评论的数量乘以10，加上点赞的数量乘以2，加上收藏的数量乘以2。

当然这里我强调一下，就是我们的这个项目，咱们没有做这个帖子的收藏，所以说这个我们就把它忽略掉，然后要对这个值求一个log，为什么求一个log的圈？

他这块也是求了一个log么，log它这算是一个技巧，它的好处是它的log它的曲线是一开始前期上升趋势或下降趋势，它的变化趋势非常明显，到后期就趋于平缓了。

这样做的目的就是说你一开始前期做评论，前期点赞，前期去收藏，那么这个权重会高，他很能够把这个帖子顶起来，但后期的话可能你是点了100个赞，可能对得到的效果和前期点10个赞是一样的，所以说我们用log其实是给前期的这些个活动增加了权重，是这样的。

然后后面我们加上了帖子的发布时间，减去牛科技园，牛客成立的时间，你看是不是帖子发的越晚，这个差肯定就越大是吧？

那么就是帖子发布的越新，这个数就越大，它就越容易排到前面去，所以说其实也是让最新的帖子尽可能排到前面去，这么一个原则，用的是一个加法，这也是可以的。

好了，总之这是我们的公式，我们一会会用到，那么不管是哪个网站，其实它背后的逻辑刚才我也说了，我们内容的分数它基本上是和时间成一个反比，随着时间的推移，这个分数是不断的变化不断的降低，到达一定程度的时候，这个分数几乎接近于0了，根本就排不上去了，因为热度已经过了，这都是基本的原则。

当然了那分数和什么成正比，就和点赞关注你的喜好成正比。

当然每个网站它这个曲线不可能是这么的平滑，它肯定是有一些波动的，可能前期分数较低，然后有一个上升，然后又有下降，波动波动之后随着时间推移再往下下来，总之不是这么平缓，这我们可能是画的比较理想的一个状态。

但是这个也不绝对，像有些内容可能就未必一定和时间成反比。你比如说就电影电影评价的网站，那么有的电影很经典，10年20年，它的分数还是特别的高，你像什么肖申克的救赎之类的对吧？然后有些电影你哪怕是你出的很早，不是很新，刚出来的可能分数一下也是非常的低，才六点几七点几甚至五点几这样的程度都有可能。

所以说这个不绝对，但是大部分像文章问答，还有发帖这种基本上都反符合这个规律。

好了，那么大概了解这个公式以后，我们就知道了我们如何去给帖子算一个分，我们应该是以哪些指标，我们要参考哪些指标，哪些指标影响这个分数上升，哪些影响分数下降，这个原则我们了解了，然后公式也给出来，就是能算了具体我们怎么去做，我们什么时候去算这个分。

当然有两种做法，一种做法就是那当我当我做了一个操作，这个操作会影响到帖子的分数的时候，我立刻去算一下，比如说我做了一个评论，我就立刻算一下，点了个赞，我就立刻算一下对吧？我加了个钉立刻算一下就这样，但是怎么说这样的话会效率会很低的，因为比如说你发了一个大号发了个帖子对吧？

可能短时间内它是这个点赞是非常的高频的，点赞转发收藏非常高频，那可能是你他点了有人点了个赞，我们这个服务器正在算，都还没算完，又有好几个人点赞对吧？这个就很不合理，效率就很低。

再一个你算的时候你他点赞，他收藏你触发了算，我时间是在不断的推移，你这个分数还有和时间有关系，是不是？

你总不能说我一秒我就算一下，过一秒我就算一下，这也受不了，因此这种方式效率低，然后也怎么说就是不合时宜，所以说通常我们是定时启动一个定时任务来算，比如说我半个——一个小时算一算完一回以后，热门的比如说前几页的帖子热门，那么它是保持一定的时间的稳定，然后过了一段时间再刷一下，这个也合理，不然的话你时时刻刻的算，比如说我刷新网页，刷一下变一下，这个都给人感觉也非常奇怪，对吧？

也不太合适。

所以总而言之我们采用性能高的方式，采用定时任务的方式来算分数，实际牛客网是每两个小时算一回，但是我们在开发的时候，没有办法把这个时间设成两个小时，太久了，我们测试的话没法测，我把它搞的。

时间短一点。

我们5分钟算一回就好了。

好，然后这是基本的原则，正好我们能把上次课我们所学的定时任务能够用起来，好，我们用定时任务算完以后，有了这个分，当用户去首页上去查这个帖子，当他点按热度排行的时候，我们就给他展现一个结果，至于结果的展现就比较容易了，以另外一种形式排序，在排序里加上死杠不就可以了，对吧？

好了，了解这些原则和大概的思路以后，接下来我们就来写这个代码。好，我打开我的开发工具，当然我刚才说了我要定时的算，但是定时的算的时候，我是把所有的帖子都算一遍吗？比如说5分钟到了，我把所有帖子算一遍也效率很低，随着时间的推移，可能网站有几十万个帖子，你每次都算这么多，没必要，因为有些帖子都没有变化，是这样的，我们可以比如说我点赞加精评论的时候，我们不去立刻算分，而是把分数变化的帖子先丢到一个缓存里，等定时的时间到了，我就把缓存里这些产生变化的帖子算一下，其他没变的底子我就不算，每次算的数据量比较小，咱们效率也比较高，所以说这样是更合理一点。

好我加到把它丢到一个内存里，丢到一个缓存里，丢到哪里去呢？最好是丢到 radius 里比较合适，既然我们想往 radius 里存一个数据，所以说我需要先定义 P 我打开 ready QQ 再定义一个前缀，这里我要存的是帖子，所以我就要 post。

好，然后下面加一个方法，这个方法主要是返回。

统计。

帖子分数的 t 这个方法不需要传条件了，因为我们这个区域这个空间存的是产生变化的帖子是多个，它不是某一个，所以你不要传帖子 ID 进来这里我就直接 return，然后 prefix post 加上 split 加上一个固定的单词是杠就行了，post 冒号是杠，表示帖子的分数好。

可以了。

可以以后下面要做的事情就是在那些个能够影响帖子分数的操作，发生的时候，我们去把帖子 ID 扔到空间里去，比如说我加精的时候，我评论的时候，我点赞的时候没有收藏，我都要去做这样的事情。

好这样我首先打开 list，post server 的 controller，因为这里有加精的操作，其实除了那三个操作以外，我想新增帖子我们也给他算一个分，给他一个初始的分，不然的话他一开始你加了一个帖子，这个帖子没有初始分对吧？

他就排不上去，就上不了榜，这也不太合适，所以说当我新增一个帖子的时候，我在这就做这样一个处理，就计算帖子的分数，因为我要往 rise 里存值，所以说我需要注入 ready stop。

好，然后我就在这儿计算帖子的分数，当然不是立刻算，而是把这个帖子弄到 readiness，首先我先获取 readiness，key YouTube get 更爱去 pose 高 k 好，接下来我就要把帖子 ID，帖子 ID 是谁呢？Post 点 ID 对吧？把 post 点 ID 放到 radios 里，那么 radios 有那么多数据类型，我放到哪个类型里呢？可能的同学会想，我这个放到队列里比较合适，先后发生的顺序放到队列里比较好，但其实放到队列里不是很好，为什么？你想想，比如说我有人给 a 帖子点了个赞，我把 a 放到对这里没有问题对吧？

然后这个时候有人就对b点了个赞，我再把b放在队列里没有问题，这个时候有人可能还对a点赞，然后又对c点赞，又对a点赞。

所以说队列中会存这样的数据，你算的时候比如说你按照这个顺序算，你先算a你后来中间一A又算了一遍a而每次我们都是把a的所有的指标都统计一遍，你相当于这三次算的是重复的对吧？而且我这只是举了一个简单的例子点三次赞没准在几分钟的时间之内，他点了几十个赞甚至更多个赞，你重复算了那么多次，效率反而更差了，对吧？这是其一。

再一个我们用队列来存这个数据，更多的是为了保证一个顺序，但是你看a它有顺序吗？他在最开始也点赞了，最后也点赞了，中间也点赞了，你说他到底应该是先处理还是后处理，这就不清楚了。

因此他有这种重复的操作的情况出现，但我没有只需要算一次，我们要去个重，而且我们又不关注顺序，你说把它存到什么里好，显然不是队列，而是因为site这个数据结构你应该知道它是不能允许重复数据存在的，重复数据会被剔除掉的是吧？

好，所以我把它存到site里，read this template。Ops for value above formula for set，然后add read this key。好，我要存的是post ID。可以了，这是一个地方我处理了。另外我们controller还有一个请求，就是加精的时候分数也会变，也会影响分数，所以我也需要做类似的处理。置顶不用因为什么置顶的话，它默认就会排到最顶上，它不算分的，所以说置顶不是去加分的，只要你置顶排到顶上，它不算到分里来，加丁是要加分算分的。

好，我就在这儿做一个处理，我就另起一行把刚才代码贴过来，也是取Key，然后的话这里边存的是帖子ID，帖子ID是谁？就这个ID删除不需要了，因为这个帖子都删了，你算的分也没有必要，他都不展现了对吧？

不用算了，就是discuss post control了，那么评论的时候点赞的时候也会要做这样的事情，所以说我再去处理评论，他们肯抽了，当然我也需要用到readiness注入，那么当我添加评论的时候，这个帖子分数就变了就应该变，所以我就需要做刚才类似的事情。

然后我是对帖子进行评论，我才去给他核算一个分，所以说这里要做个判断，如果说我评论的类型帖子这个时候我要去计算帖子分数，把刚才代码拷过来，拷在这儿同样的方式获得Key，然后这里存的是帖子ID是谁就这个好，这是评论的时候的一个处理。

然后还有点赞，我再打开like controller也是注入ready template好，然后点赞的时候也是做加分的处理，我就在这一个点赞的方法对吧？就在这最后的位置加分。

当然这个时候你也需要做一个判断，如果我去我是对帖子点个赞，我才去算这个帖子的分数，如果你对其他的内容点的赞我不管。

好，我这样我判断一下，如果Entity tap，如果Entity type等于Entity type，post，这个时候我就要计算帖子分数，我刚才代码拷一下。好，这个时候帖子的ID是谁找一下，是他POS的ID我都传进来了。

好可以了，现在就是我们的项目中，但凡能够影响帖子分数的地方，我都做了这样的处理，我都把贴在地放到了集合里，下面要做的事情就是我每隔一段时间就要算一下，对吧？

我就这要用到定时任务，我们就可以利用我们上次课所学的筷子，而筷子我们在使用的時候怎么用第一步不是得写job对吧？我们先写一个job，我在科尔包下新建一个类叫做post god risk drop好。那么这个类它的作用其实很直接了，就是帖子分数刷新的一个任务。好我要实现这个重要的接口，另外我再实现一下常量接口，一会也是有用处的。Community constant我需要添加Shop未实现的方法，然后对它进行一个实现。

那么在实现的时候，我们定是任务启动的时候，你最好在关键的节点记个日志，这样的话将来万一他出现了什么问题，意外中断了，我们还有一个追查的根源依据，所以我先实例化logo，当前类点class好，logo有了，然后我需要注入一些电，我们计算的数据来源来源于rides，所以我需要把rides他们注入进来。



好，另外我们在计算的过程中可能还要查帖子，查这个点赞的数量，然后计算完以后，我们还要把最终的最新的数据同步到搜索引擎里，因为你帖子数据变了，搜索引擎也要跟着变，所以我们还需要注另外的这些病有关的这些病我注入一下，先注入 `discus`，`post service`。

好，然后再注入 `like service`，然后再注入 `you last take search service`。好，先注入这么几个病，注入完以后先不用着急去实现这个方法，我还要初始化一个常量数据初始化什么。你看刚才我说了，我们要计算留客的分，最终的话你要由发布时间减去留客纪元，纽克纪元就是纽克成立的日子，这是一个常量固定的值，但它又比较特别，因为它是一个日期，所以我们需要对它初始化一下。

好，我在顶部说明一个静态的常量，`final`，然后日期类型 `date` 接口叫它报错的意思这是一个常量，但是你没有初始化，所以它有个提示，我需要初始化，我在静态块里初始化一下，因为常量它只需要初始化一次，好那么我就写了它等于 `New simple date`。`Format` 我要先给一个格式，然后点 `plus` 传入一个字符串。

那么 `simple day` 的 `format`，它能把这个日期转为字符串，它也能把这个字符串解析为日期，但前提是你指定格式匹配的格式，然后得到这个日期以后复制给常亮就 `ok` 了。当然这句话它需要我们对异常进行处理，所以我先再开始一下。

好这个 `past` 里面的需要写上的满足这个格式的日期，那么牛客成立于 2014 年-08-01，然后这个时间就 000 其实我们主要是关注这个年月日不用精确，到时分秒就 00 代替了，然后如果说我捕获到异常的时候，那怎么办？

我就抛我就抛出去了，说初始化游客既然失败，再把原始的议程带上，好这就行了，那么当你正常的初始化完产量以后，他这就不报错了，这里我写了注释就是旅游科技园，一会运算的时候好用它，好有了牛科技园以后，下面我就可以来实现定时任务了。

要实现定时任务，首先我得从 `reds` 里取那些值，对吧？然后我才能一个去算，我得先有 `key`。

`Already is key` `YouTube get post scar k` 然后因为我们是每一个 `k` 都要算一下，我们要反复的去做这样的操作，所以我用棒的 `operations` 来处理，因为它不是一下就能处理完。的棒的 `site operations`。变量名称叫 `operations`。等于 `read this template`。棒的 `site` 和 `ps` 好，然后把 `reads k` 的传进去。`Ok` 了。好，接下来我先判断一下，我们 `Redsk` 对应的空间里它有没有值，万一最近没有，比如说半夜的时候对吧？都没有人访问这个社区，没有任何人做任何操作对吧？你就不用算了，所以看一下有没有数据线。

`Operations`、点 `size`，如果说 `size` 等于 0 表示说没有任何变化，这个时候我们记个日志把任务取消掉就直接入退，不做任何处理了，但这里面要给一个日志 `log` 点 `Info` 就常规日志说这样写任务取消，然后没有需要刷新的帖子，否则那就是需要刷新了，这个时候我在开始刷新之前和刷新结束之后，我都记一个日志，这样将来如果说我们发现好像任务执行的速度挺慢，我们可以具体看一下前后它到底支撑了多少时间，通过日志能看出来，但你前提得有个日志，拉个点因素，然后说任务开始正在刷新帖子分数，然后后面我加上 `operations` 点 `size`。好知道我现在要刷新的一共有多少个帖子，当我刷新完以后也是就在 `logo` 一下，这儿我继续说任务结束，然后说帖子分数刷新完毕，完了我们在这两个劳务之间去处理，当然一个处理你得循环了，我就 `why operations` 点 `size` 大于 0，只要你这个 `rise` 里有数据，那我就算每次算我调一个刷新方法去刷这个分数，我就 `list` 点 `reference`，但这个 `request` 方法是我一会要写的，我先在这里先调一下，一会再补充，然后需要传一个参，就是把帖子 ID 传给他怎么办？

就 `operations pop`。因为 `Operations` 它是集合有个方法叫 `pop` 就弹出一个值，每次弹出一个值就少一个值，直到弹光了，循即这个循环也就结束了，对吧？就是这样。当然了我们这个 `risk` 要的是一个涨数，所以这里我们转型一下，因为它默认的话它以为是个 `object`，我把它转成影推者。

好了基本的任务的代码逻辑结构就是这样的，剩下的事就是我们写一个 `rest` 方法来刷新一个帖子的分数就 `ok` 了。那公式有比较简单，再写个方法。`Private wide at home reference`。好，这里面需要你传输帖子的 ID `Cos l` 好，然后我首先先把帖子查出来，`discuss post`，调 `CS`，然后办的 `discourse d` 白 `d` 传入。

好，然后做一个判断，如果说post等于no，万一说这个帖子有人点了赞，他是要算分的，但后来被管理员给删了对吧？这个时候没查到有这样的情况存在怎么办？我们要给一个提示，然后就不记就算了。终止退，每天之前给个日志我们好追溯，就说该帖子不存在，然后我这里写上不存在ID等于加上post get ID。当然直接POS代替就完了，这样给个明确的提示，然后接下来我就要开始正式算分了，而这个分和什么有关？

它和你是否加精有关，和评论的数量有关和点赞的数量有关，我需要先把这些值先取到，然后我再去用这个公式来统一来串。首先我要得到一个结果它是否精华是否加精，这是一个布尔值，是否加精是个布尔值。Wonderful等于就看 Pose的叫get status。

如果它等于一就是假精的，否则就不是，我就判断它是不是等于一，然后还要取评论数量，这是一个整数，我们从帖子对象中就能得到，他说这就带了，不用查了，然后还要获取点赞的数量，这是一个long 类型 like come等于我们第二like service去查，然后它里边有个方法叫find，Entity like count。

那么我们需要传第一个是Entity type，那就是Entity type，post帖子类型，第二个是对应的实体的ID，就是帖子ID，就得到了点赞的数量，那么这三个值有了以后，我们就要把它运用到这个公式里 log，然后这些值相加乘以相应的系数，然后最后再加上这个时间的一个间隔，好，这样我先把 log之内的那几个相加的值先算一下，其实我们通常习惯于称其为权重，我们习惯于称括号里这部分值是一个权重，然后的话再加一个时间，所以我先求权重的值，注释计算权重，这是一个小数。

W等于。

首先我看一下它是不是精华，如果是精华，我们固定给加75分，给加75的权重值，否则简单粗暴那就是 wonderful。然后如果是加就加75，否则就加零。好，这是精华的部分，然后 Comment com是评论的部分，评论要 $\times 10$ ，再加上点赞 like com要 $\times 2$ ，这个权重值有了以后，我们就要去详细的计算了，计算什么总的分数，你要等于帖子的权重加上一个什么距离的天数？

时间间隔，但这个时间间隔我们是以什么为单位？是小时秒，还有什么为单位？其实是天以天为单位，要不然数太大，好我就算 double死杠等于。

然后你算的时候是权重值要求一个对数，然后的话再加上时间间隔求对数，我们调用 max，然后的话是 log10是以10为底的对数，然后你还不能直接求 w这样不好，为什么？如果万一说 W得到的值是小于一的，你求对数以后这块能得到一个负数，对吧？你没有分就没有分，大不了为0，你不能给他整整出一个负分对吧？这个不太合适。

求对数以后最小值应该就是0，不能出负的怎么办？我们可以这样，不要直接写w而是写成什么？写成 max点，max，然后w一。好，我们先看里边这部分，max点max，w一就是返回这w和一之中较大的数万如果说w大于一肯定就是返回w算w的对数没问题，对吧？如果w小于10~1之间，那就返回一算一的对数，一的对数其实就是0，对吧？那就是最小的值，我们最终就能得到一个0就完了。

好，然后这是求对数求权重的对数，然后的话加上时间间隔和定期太长了，那么我需要获取帖子发布时间，再减去牛科技园，帖子发布时间是post点，get create time这是date，然后我点get time。

得到一个浪字，得到一个毫秒这样好减，然后减去云科技园，然后点它也是对的点，get time两个毫秒相减得到的是毫秒值，但我要的是天怎么办？换算。除以我 $1000 \times 60$ 不是60就是 $3600 \times 24$ 1000这表示换算成一秒对吧？然后一小时是3600秒，乘以它是3600秒对应的毫秒，然后乘以24就是24天。

所以我这样一除的话，其实就是换算成了一共是多少天了，这就可以有了这个公式就可以了。有了以后我们需要把帖子的分数更新一下，然后更新帖子的分数，那么我就调用 Service，然后 update scar，但是我还没有 up的死高这个方法怎么办？

我们还得去补充一个方法。

好，那么我就打开迪斯卡斯boss的map，好，那么在这里就模仿了 update，补充一个方法改分数，这是 update score，参数是ID更改的值是提高，好，然后我们再打开麦克，然后把这个方法实现一下。好，update，scar。Update the distance post said scar等于scar where I didn't ID完了。

好，然后当然了你麦克里有了，对应的service里也补充一下，我再加一个方法，这个方法之内很简单，就是return discus boss的map点，update，sky，把ID传进去，把sky传进去，好，我把这个方法都补充完了，然后我们再回去接着写这个任务。那就是discuss post service点，update，score。

然后把帖子ID post ID传入进去，最新的分数传进去可以了。

然后别忘了我们还要同步搜索的数据，不然的话搜索的时候就出来一个特别陈旧的纸，这就不合适了，我们就同步把搜索的数据好，我就调用 elastic search service要点，它里边有个CF方法对吧？你需要把实体传进去，但是这还不行，因为实体是我们早先查到的，它里边的分还是旧的分，你怎么办？我把实体 Site score把这个指标更新一下，从最新的然后再去保存，那就更新过去了。

好了，这个任务到这我们就写完了，就符合我们的最终的公司的要求，写完以后这个任务要想能够正常运行，我们还得去做配置对吧？我再打开 Class can figure配置的方式和上次课我们写的这两个事例是类似的，我就copy一下，然后在这基础上去改。先写个注释，这个是刷新帖子分数的一个任务，当然这个并注解得让它有效，然后这个方法名得改一下，不然重复了不行，我叫扣死他。

Score，然后with rest drop detail。这个方法它的表达的含义要直观一点，然后这里边我new一个fact been然后这个地方类型要写成刚才的类型 Post，scar，refrax job的Klus，给他job起个名字，我就叫 poste。Scar。Reference。Drop然后分组，我希望我们整个应用正式的任务都是统一的一组就可以了，这块我就叫community drop growth。

然后这两个参数是出没有问题就这样。好第二个还要配一下崔哥，B也需要注释去掉，那么崔哥也得改名字改成还是跟刚才类似。Post Scott reference，trigger。然后他所需要的照片的q是前者，我把拷过来poss的词杠，refresh drop detail。这传的就是他，然后当然崔哥也取名字，Pose的是高瑞福s崔哥。

然后这个组我也希望整个项目正式的处罚器都是同一组，肯定那天催着group好了，这样的话对于刚才这个任务，我这两个地方都配好了，配好以后这个时间间隔再配一下，刚才我说了我希望它是5分钟执行一遍，这里面我们需要传的是毫秒，我这样写1000×60×5，那是60秒，然后乘以55分钟没有问题。

好了，现在这个任务也就完成了，完成以后我们来先不去写展现的部分，我们先跑一下，我先点几个帖子什么加精，然后这个点赞试一下，看这个分他能不能给我算出来。如果说没有问题，那么我们再再继续。

好，那么我就启动一下这个服务，没有需要刷新的帖子，这里面写错了。然后这怎么还有 sq二，我把再重来一遍，太太乱了，太烦了。应该是测试类当中有的注解我没有去掉，应该是 service阿尔法service我没有注掉，这样我一启动它就在运行了，这不需要了把它注掉，要不然太烦了，这样我就把注解注掉就好了。倒无所谓，我不主动掉它是不会触发的，但这个是定时任务，它会自动的去跑，就是烦人把它去掉。

好，然后我再看一下这个表，因为我们刚才启动一下这个服务是吧？他提示有问题，我写错字了，没有需要写错字刷新的帖子，然后我们刚才启动了服务以后，它根据配置应该会创建插入数据到表里，我们看一下表里有没有这个数据确认一下，看这个照片。Post scar reference job没有问题。看崔哥 reference崔哥没问题，ok这两个有就可以，他就能跑起来，好我们再来。

好了，我这个服务就已经启动起来以后，我得打开浏览器，我去对某些帖子做一些操作，这样我登录一下，登录以后，jue的帖子我不管，我先发几个帖子，第一个比如说AA a简单点发布，一个再来一个BBB再来一个CCC好了，然后我一共新发了三个帖子，下面我要对三个帖子做一些操作，你比如说怎么说Cca我不管了，我要操作BBB我给他做一个评论，你好，完了只做一个评论，好，再回到首页，然后对于AAA我要对它做一些处理一些评论，hello，然后再来一个how are you再来一个做了三个评论，所以他的分应该比刚才要多一点，再一个我还给他点赞，除了点赞以外，我还要给他加个精，加精的话我得换个人换一个。

版主21我。

给AAA加个经验，然后我用用户也给他点个赞，也给他做个回复，欢迎。

所以如果算的话应该是AAA帖子的分，就在这三个帖子之内它应该是最高的，其次是BBB然后是CCC因为他们的时间几乎是一样的，在一天之内，其实这个时间的权重是一样的，因为我们那个时间相减是按天来算他们天数一样的，那么其他的地方就是ab分比c高，最终排的话，如果按热度排的话，应该是a在前，b在后，现在是b然后现在是c应该是这样的。

当然了我们得看一下，现在他可能还没有更新，我们查一下差异他们的反叙一下，倒叙这三个帖子，那么他的分数有了，已经有了，可能刚才就触发了是吧？

我们看一下，真的触发了到了5分钟了，你看这刷新完毕，刚才计算了3个，刚刚好就刚刚，然后你看这个分数，AAA1823 BBB18221821是有差距的，但这个差距不大，因为我们并不是直接把这个数量乘以一个几一加让它差距很大，因为我们对它做了一个对数运算，所以让这个数其实差别不是那么大，你得做好多评论好多赞，他的分数才能够遥遥领先，而不让他这个分数突然一下几个赞就特别爆发的那种感觉，这是ok的，现在我们看定制任务没问题，我们做的操作没问题，最后就是剩一个展现了，展现的话我们要做的我们在访问首页的时候，我能当默认是按照这个时间倒序来排，如果我点最热的话就得按照这个分数来排。

当然了页面还是利用这个页面就可以了，只不过我点这的时候需要给服务端再传一个新的参数，让它做就有一个新的排序的模式，所以我们需要对之前的代码做一些重构，让它能够支持这种排序，而这个重构我们需要从这个数据访问层就开始改起。好，你看我们之前查询帖子不是用这个方法查的吗？对吧？

然后这两个是分页的参数，这个是优质ID，它的排序没什么关系，然后之前我们在这里头实现的时候，我们排序的方式是写成固定的方式的，首先根据类型来排，那么置顶的会排到前面去，无疑，然后其次是按照时间来排，现在我们得能支持两种模式，我这样处理，我再加一个要给它加一个参数，叫order，mood就是排序模式，然后先加上 order，mood默认值是0，我就还是按照原先的排，如果你传入一，我就希望按照热度来排，我就按照热度给你熬着办一下我实现的时候怎么办？

也很方便就判断一下你传的条件，然后的话动态的去拼这个奥特曼就可以了，我就得加一个if你的判断，然后这里有写test等于我们得看 order more的，如果说0它=0，我们按照原来的方式来排，不用改。

我再copy一下，如果说这个奥特曼的等于1，比方说我希望按照热度来排，我就order by还是优先按照态度把置顶的排到前面去，其次才按照style，然后倒叙，然后万一说两个帖子分一样怎么办？你再按照时间的倒叙，所以这个时间还保留，就这样一个结果。

当然你这么一改以后，它调用的地方你就得就得都得改了是吧？我们搜一下 Find uses看看哪个地方掉下来了。你看我们之前做了一些测试，那么在测试的代码里我们调了这地方得改，不然的话他就报错了，没有办法。加上一个0，原来是默认是0对吧？加上个0，或者你把它注掉其实也可以，因为这个方法我们已经测过了是吧？其实不再不是那么需要了，然后这里头也是加个0，除此以外还有 Service，discourse，post service这里也掉了，好，我们 service的话也得重构一下奥德姆的，然后把奥德姆的传给迈克就好了。

好到现在我们就把数据访问层还有业务层就成功好了。

然后你还得看一下这个业务方法都被谁调了，掉的地方我们得改，因为现在你加了个参数就报错了，也是搜一下find uses，好在只有一个地方掉，就是home control这里，所以我们只需要改这一个地方就好了，打开 home control了，好，我们需要重构一下这个逻辑，这个参数不能平白无故的来，它从哪来，应该从页面传过来，当我点最热的时候传过来，当我点最新的时候传过来，对吧？

传进来0或者是1，所以我需要给请求的方法再加一个参，就叫我等下，但是你要注意，我们首次访问首页直接敲路径访问的时候，直接敲网址访问的时候，我还没点这两个东西对吧？我就不传参，这个时候怎么办？

这个时候如果你直接这样写就有问题了，你没存在那就报错了，所以我们就要给他一个就是声明，要加上一个request，power注解对于参数加以声明，我声明它的name就是order food，请求参数中outward，另外一个我给他一个默认值得放到外流，等于0，所以说如果你没传进来，默认就是0，默认按照这个就原先的方式时间的方式来排序，这不影响。

好，那么我们在查询的时候就需要把它传给 service 了，那么这个方法就做了一个改动，好做了一个改动就可以了。

另外还有这个地方，我们也需要在路径上把奥德姆的拼过来，就是奥德姆的参数我们是通过公众号传过来的，不是通过请求体传过来，是通过问号传过来，因为如果你通过请求里传，你要用 pose 的请求，用 get 不方便传，get 适合拼一个问号，拼问号的话，我们把它也拼到路径上，这样的话我们路径返回到模板，模板会用到这个路径，在这基础上再去拼分页的参数。

那么你注意你在这上拼分页参数的时候，不要把奥德姆的给它丢了，不然的话就是说你分析以后奥德姆的串了也会有问题，所以把这奥特曼的就拼在这，也当做路径的一部分，就奥德姆 mod 等于然后加上参数，最后我再把奥德姆的装到 model 里，然后传给模板上要用 h h 图，然后的话好点。

好到这儿这个方法我就处理完了，就是一些细节。处理完以后，最后我们就来处理首页 index 点 tml 搜一下，在这个页面上我们找到这个地方最新最热，因为你点它才触发排序的方式的改变，对吧？找一下就在这了，就在这了。

好，我们希望把超链接先处理好，th 冒号，这里我需要练到的路径还是 index，然后需要带上一个参数，就是 order mode，然后等于 0，完了同理最热跟他一样就改一个值就完了是吧？改成等于一就可以。

另外你还要注意，我们页签它默认的是点亮了最新，但是当我点最热的时候，你是不是得把奥德姆的变成了一，你得把它点亮，我再点它又把它点亮，这两个页签把谁点亮呢？把谁点亮就在谁身上加 active，现在不能写死，Active 你要动态的加怎么办？我们可以根据刚才我在 model 里装的奥特曼的装在 model 里，我们在模板上不就可以直接访问吗？我访问奥特曼的判断它等于 0，我就把它点亮，等于就把它点亮对吧？

那就克拉斯动态处理一下就好了，th 冒号克拉斯，然后因为我需要拼一个变量，所以我加上双数线，然后这里边写一个表达式，奥德姆的对，如果说等于 0，那么我就在这儿输出 IQ，否则就什么都不输出，好，这个同理到这目的等于一。如果成立我在这输出 active，否则什么也不输出。

好这样就可以了，处理完以后我们再试一下，我把应用重启一下好了，启动以后打开浏览器刷一下首页，默认是按照最新的方式来排序，CBA，因为我先发的是 a 再 b 再 c 倒叙 CBA 没问题对吧？我们不用看别的帖子，我们只看这三就完了。

好，然后这是最新我点最热，你看这个路径最热，index output 等于一没有问题，然后你看我点分页，是不是分页条件上就带上 order 等于 0，因为当前默认是最新的，也没问题，然后分页参数卡瑞也在，好，现在我点最热点，最热以后我们不用看别的帖子，还有置顶什么的不用看，我们只看 ABC，你看是不是 a 在最前面，b 在其次，c 在最后面，因为这三个分就这么算的，然后你看他这几个帖子排到他前面，因为这几个帖子都是置顶的，优先的，除了这几个以外，那么我很久没有发新帖子了，这个是今天新发的，其他的都是好多天以前发的，所以从时间上来说，越新的排的越排到前面去，它比别的就优先排到前面去，收视率更高一点，就这样好了，那么这个时候你看最热被点亮，然后你看分页的条件，你看左下角路径，index automotive 等于一，是吧？

所以你点第二页的时候，依然是最热被点亮，依然是处于这么一个按照热度排序的模式，末页也是最热被点亮是吧？然后点最新，你看分页条件，奥德姆的等于 0，然后第二页最新依然是保持最新，末页依然是保持最新对吧？说明这两个状态切换没有问题，不只是说数据的排序没问题，另外那么与之相匹配的分页也是没问题，并没有就乱掉。

好了，那么这次课我们就把热帖排行就搞定了，那么。

总之实际上并不难，那么就是背后的计算排行的算法你也不用背，但是你一定要有这个意识，我们要算这个东西，我们需要考察关注哪些指标，背后有哪些基本的原则搞清楚就 ok 了。

然后将来工作的时候，比如说你算某一个内容的分数的时候，可能一开始这个公式设计的并不好，就是这个曲线并没有那么平滑，可能会出现一个断崖式下跌，或者是某一个指标权重过大，很容易被刷分的情况，那么需要你不断的在实践当中去调整这些个参数。

然后的话一是避免刷分，二是避免这个分数有一个大的跳跃，基本上让它更平滑一点，更合理一点就ok了。好，这次课我们就演示到这里，咱们下次课再见。

## 7.8 生成长图

50-生成长图.mp4

这次课我们来学习如何生成长图，那么生成长图这样的功能往往是在这个APP上才有的。通常是在APP我们做分享的时候，我分享一篇文章，我分享一个音乐等等分享的时候，然后我分享的时候把这个内容相当于截个图，然后的话可以发到朋友圈，发到QQ群 Qq空间，发到微博等等这样的。

那么生成长图的手段就有两种，一种是客户端实现，就是APP把它当前的界面截个图就生成长图了。

还有一种方式是在服务端，我们可以用html做一个和APP的界面一模一样的模板，然后我们通过一个工具访问这个模板，然后读取这个模板中的内容，生成一个长图，这是两种方式，那么因为生成长图是非常常见的一种需求，所以说我们在这个课里也给大家讲解一下演示一下，那么因为我们这个项目是一个外部项目，咱们没有APP所以说 APP客户端生成的事我们就不说了，我们只探讨的在服务端，我们怎么利用一个h三模板生成长图，就给大家演示这样一件事，那么我们想把这个模板转换成一个图片，那么需要用到这样一个工具叫wk html图PDF，是这样一个工具，这个工具它有两个功能，实际上就是两个命令，一个命令是这样的，wk html图 PDF，然后写上模板的访问路径，HTTP冒号双曲线什么，然后后面写上你要生成的图片生成的 PDF存放的位置命令是把模板的内容生成一个PDF。

那么下面还有另外一个命令是w Cay html, to, imagine这个命令是把网页生成本地的一个图片，我们要用的是后者，然后这是我们通过命令的方式去直接访问它，那么我们通过Java语言去访问它，不需要倒包，我们就直接用java原生的一句话，run time点get run time。Exec执行一个本地的命令就可以了。反正这工具就是比较小巧，比较方便。那么这个是工具的官网，下面我们就来看一下这个工具，然后的话把它安装好。

咱们用一下试试，然后如果试用过以后，我们了解了它的基本的使用方式以后，我们在模拟开发一个分享的功能，只是模拟，因为没有APP然后的话把后端生成图片的逻辑就利用起来。

好，那么首先我们看一下这个网站，我已经提前打开了，这样然后如果你想看它的相关的帮助手册点这里，然后点这就是它目前它没有非常成体系，详细的文档只有一个生成的一个文档可以去参考。

然后你要下载就点这 Download，这里面有各个操作系统对应的版本，那么如果是windows系统，我们通常选或看你的系统是多少位的，我选的是64位的。

好，我已经提前把它下载好了，稍微还有点慢，但我已经下载好了，那么我把它装一下好，我把它装到d盘work的下面好，然后 instead完成了完成以后关掉就可以了，这就行了。

然后打开d盘work找一下 wk html to PDF，我们要用的命令是在 bean的目录下，这两个exe文件为了我们通过命令行访问它，方便我把它当前的路径配到这个环境变量里，我靠一下，然后在 pass里加一项就可以了，环境变量也配好了，接下来我就给大家演示一下刚才我说的那两个命令，然后我们随便敲一个网址，把这个页面生成一个PDF再生成一个图片，咱们看一下。

当然我们要生成的话，我们需要指定本地存放生成的文件的路径，而且它要求我们提前把路径建好，命令不会自动创建，路径你要自己建好，我就创建一下，我就在 d盘work data之下，我新建两个目录，一个叫wk然后的话杠PDF好再来一个 wk杠image就是说我生成的PDF放这里，生成图片放这里，这就很直接了。

好，那么创建好这个目录以后，那么我就打开命令行试一下。首先我先生成一个PDF Wkhtml to, PDF空格，然后你要写上一个网页的访问路径，比如说百度淘宝是随便什么都行，我写优客网的优客网的首页路径，3w.com这就可以了。

那么这个命令就会解析路径，就是远程下载到它的html的内容，然后的话把它转换成一个PDF文件。

好，接下来空格你要写上你要存放的文件的的路径以及名字，我这是d盘是吧？D盘。Work data，wkpdfs在这里边我希望生成的文件叫一点，PDF简单点。好了，我就写完了，写完以后回车稍等一会儿，它会有一个百分比，最后它告诉你大就完成了。好，那么完成以后我们去看一下有这个文件吗？一点kdf有400多k打开看一下，好。你看是吧，牛客网的首页对应的这些功能，这个效果还是很不错的。

好了，Pdf就生成完了。

好，下面我们再生成图片，生成图片的命令和差不多，我直接写了wk wk然后的话是html to image，然后也是写上一个页面的访问路径，我还请留客网好。

然后还是要指定一个路径，第一盘work data，wkemad然后我希望这个图片的名字叫一点PNG好可以了，我就回车。但完成以后我们看一下对应的路径，data were awk image有一个图片打开看一下，咱可以放大对吧？

就是我们首页的内容生成的图片没有问题，然后这个图片挺大的，我们可以看一下有一兆多非常大，通常我们生成图片的时候不希望这么大，因为你生成图片以后你存的话也占空间，如果你想把它传到云服务器上的话，那也比较费空间，不要浪费钱。

所以说通常我们希望对这个图片做一个压缩，让它小一点，怎么压缩我在把刚才的命令再调出来，然后的话我们可以在 WLK html to, image的后面加上一个参数声明，我希望图片压缩到什么程度，那么这是这样写杠杠，quality就质量75%，就是你压缩到原油质量的75%就可以了，好这个数量可能是比较合适，它是一个性价比比较高的一个值，就是说你往超过这个值以后，那么它的质量没有明显的好转，但低于这个值质量就会下降，但是这个值就是75%这个值，它的效果还不错，还可以，性价比比较高一点。

好，因为我这是重新执行一遍，我想再生成一个新的图片，2点PNG写完以后，然后回车，好完成，然后再看你看二点偏激生成的是吧？然后就小很多了，才700多k然后打开以后，这个内容和刚才一样，好了，当然了其实我们平时在分享功能里一般不会这个网页不会这么大，我们一般分享的都是比较简单的，比较简短的，你像牛客网手机的APP它有一个应该有个打卡的功能，打卡的功能就可以分享出去，一般非常简单，就是一个打卡的数字就完了。

然后的话，所以说一般不会像我们刚才看到这个图片这么大，所以说时间行的时候效果应该会更好一点，这是我们通过命令来访问这个工具。下面我们再写点程序，通过java来访问它。那么我在测试的包下新建一个测试类叫做wk test。那么这个类我只是写个面方法，通过面方法测一下就可以了，这就不用结构类的了。

所以说我也不需要去引用那些注解了，然后我们像刚才PPT上所说的，我们就通过run time去执行一行命令就可以，我先把刚才的命令给它粘贴过来，粘到这个字符串里面，它好粘贴过来，然后把改成三点PNG然后你要注意我们这个命令就是枪在命令行里头你可以直接这样写，我们在这个程序里需要把命令的完整路径写出来，我需要写d盘，work，wk，html to PDF，然后的话是 GB，那么命令叫wk Image，wk html too，里面就最后有了d盘work，wkhtml图pdfb然后 wkhtml图为位置。

我再检查一下没错，然后我们的存放数据的目录还是把它放图片还是放到这里，我们来看一下。

现在我这个命令行，有了我要执行的话就是Iran thyme点get right time，exe c comment传入就可以了，当然了它需要怎么处理异常，我就拆开这一下，如果补发的异常，我就把异常抛出来，就把异常打印出来，如果没有异常，那我就输出一句话，ok表示成功的意思。

好，这个程序就写完了，然后我们一边执行的时候一边看一下文件夹里有没有生成一个新的图片就好了。

好，下面我就来运行一下命令方法，然后这边看这个图片你注意有没有看见？这个ok很快就输出了，马上就输出了，然后三是过一会才生成的，为什么会这样？Untime他执行命令的话，他只是把命题交给本地的操作系统，剩下的事就是操作系统来执行。

这个时候我们java程序不会等着说操作系统给我一个反馈，他不等，然后他就直接向下执行下一行。

换句话说，操作系统执行命令和我们当前的主程序它是并发的，它是异步的，因此我们生成一个图片，这肯定需要一点时间了对吧？可能需要几百毫秒，甚至一秒左右都有可能，需要一点时间它就慢，我们整个没有办法往下执行到这儿，这是很快的，所以我们先看到ok再等一会儿才看到图片，你要注意这是异步的，我们看图片对不对？也是没问题。

好了，怎么在Java中去调用这个工具就演示完了，主要是你要注意这行代码和命令执行的先后顺序，因为我们将来在解决问题的时候需要注意到这一点，你要没注意的话，可能会有一些你感觉比较奇怪的问题解释不通。好，那么这个测试代码通了以后，我们会调用这个工具以后，下面我们就来模拟实现分享的功能，咱们把在实际业务中走一下感受一下。那么现在我要做一件事，首先我想对把 wk 这个工具它的命令，你看我们在程序中写这个东西不太合适，最好把它做成可配置的，因为将来我们这个系统上线以后肯定我们换的是Linux，对吧？那就没有d盘，这命令就不对了，把它做成可配的我们好改，这是一个。

再一个我们图片生成图片存放的位置，这也得可以改，因为将来我们要切换这个操作系统，这个都可以可配对吧？而且刚才我们 Wk image是我们手动创建的，你想你上线的时候还得在Linux下手动创建一个这个文件夹，你万一忘了怎么办对吧？

所以最好是我一个是把它做成可配的，再一个我们系统服务启动的时候，我们检查一下这个目录有没有，如果有就算了，没有的话，我把它用程序创建出来，保证它一定会有，避免我们上线的时候麻烦好。

所以我在正式写分享功能之前，我先把这两件事处理一下，我打开 application，补充点内容，这是关于 WPA的配置，这个配置就是我们自己定义的，自定义的我叫wk，一点一妹子，然后.come on。生成图片的命令行是怎么写？其实就是刚才这段代码。

好，然后另外wk点那张生成图片的路径，我叫story存放的位置，那么当前是地盘 work，data。Wk EMAIL好了，这是参数我们在这里配好了，配好了以后一会要用，还有一件事要处理，就是我们服务在启动的时候，我要检查这个路径存不存在，如果不存在我要守我要用程序来创建。为了验证这个程序能不能自动创建，我先把这个目录给删了，我觉得都删了不要了。

好，这个代码写到其实写到哪都行，我就写一个配置文件，专门在配置文件里做这样的事情比较合适。

我在肯费格目录下新建一个类叫wk选这个好，然后这里边写上选这个reason，但其实这里面我并不想定义什么b因为 Double k她和splore也没什么直接的关系，我要做的事情就是在服务启动的时候要创建目录怎么办？

我可以这样写一个post construct，这表示初始化，所以我写一个初始化的方法，盈利，我们服务启动的时候，因为这个类带有configurations注解，所以 spron会以为这是一个配置类，他会先去加载它，先去实例化的，实例化以后就会自动调盈利的，因为有这个主角是吧？

所以在启动服务的时候这个方法就会调用一次，这就够了，这个时机就刚好合适。

然后这样我先提前先把 logo先实例化好，因为我要用好了，另外我还要把那个路径注入进来对吧？因为我们要创建的路径，我们得从这儿去读，把它注入进来，通过value我叫wkenet story好了，在盈利的方法里我只要创建对应的目录就可以了，不是吧？

创建wk图片，至于PDF目录我就不创建了，因为我们项目中其实用不上生成PDF，如果需要的话再说好我就先实例化一个饭等于你然后把路径传进来，这个file不是一个普通文件，它是一个目录，我们判断一下它存不存在 fire点excess是存在加个感叹号表示不存在，如果不存在我就创建对吧？

那就file点，make dir创建路径，然后创建完以后我再给他记个日志，说创建wk，图片目录，然后后续加上路径，因为这个操作比较关键，所以说我们记录下来将来可好通过日志可以追查好。

写完以后我们可以测一下。当前刚才我已经把目录删了，如果说我一启动服务，它就有了，比方说这个程序是ok的，好，我启动一下，你看这个提示已经创建了，然后这个时候我可以看一下目录，wk妹子就已经有了，但是现在这里面没有内容，好这就可以了。



准备工作完成以后，接下来我们就来模拟开发一个分享的功能，因为我这里现在没有 APP，我就用这个 web 请求来代替，但我们这个请求尽量做的简短一些，我们的精力不是放在这个页面上，我们是放在服务端的逻辑上，好，这样我就写一个 CTRL 处理前前端的一个请求，在这个请求之内我要生成图片，生成完图片以后我还给你提供一个请求，允许你访问这个图片，整个服务端的环境就搞定了。

好，那么我新建一个类叫塞尔塞尔肯抽了分享的 controller，那么给它加一个注解，然后我在这里先实例化一下 logo，一会要用。

好，另外我提供一个方法，那么前端的一访问我就要生成一个图片，那么我们生成图片的时候，怎么说它一定是异步的方式，然后因为生成图片时间比较长，一定用 if 的方式，我们一般习惯于用这个事件驱动，这是一个分享事件，controller 只需要把这个事件丢给卡夫卡，然后后续有卡夫卡实现异步实现就可以了，而不是在这等着，因为你等的话时间会比较长，然后的话客户端响应的话会比较慢，这样的话服务器处理的能力就比较弱。

既然要用衣服的方式我们要用卡夫卡，所以说我需要把 invite producer 把它注入进来。

另外的话我们在这里还需要用到什么？用到我们应用程序的域名，还有这个项目的访问路径，我们生成的时候用不上，主要是用户访问的时候他需要一个访问路径，而这个访问路径和域名和这个项目名是有关系的。

所以我们把这两者注入进来，首先是域名 Domain，我再来，然后我先不写，我记不住 k 然后是项目访问名，context pass 好这两个 key，我们可以从配置文件里拷，之前就有你看域名在这，然后项目名防路径。

好，这也是准备工作，然后接下来我们还需要把图片存放的位置也传进来，因为你要你要生成图片，需要用到这个路径也要把它转进来，这个路径刚刚我们配的就是它。

好了，这些准备的数据准备完以后，下面我们就可以写分享的请求了，我就先声明它的访问路径为 class 的卖品，分享就叫赛尔，然后请求方式我就 get 了，咱们简单一点，另外我把这个请求做成异步的反馈精神，这样简单，我就不想返回网页，我们也没有给他准备一个什么反馈的页面，所以加上 response 包点。

好，然后把这个方法声明一下，这个请求需要传什么参吗？显然要传参。

你分享的时候你分享的是哪个功能，你需要把这个功能传进来，我好根据这个功能去生成对应的图片，那么其实我们跟我们实际上如果是 APP 的话，你把功能传给我，我根据这个功能找到与功能匹配的对应的模板的路径，这有一个严格的对应关系，然后通过路径去生成图片，这里我们也没有什么功能，干脆我就直接要求浏览器把路径传过来就得了，你想生成哪个路径你就给我传就行了，我们省略一点事。

好，这里传的是什么？是 html URL，比如说就像刚才我们写的 Knocked 的 calm，像 baidu.com 这样的东西，那么继续我们就来实现这个逻辑。

首先我们生成的图片文件名我们需要随机一下，这个要随机一下，每次都要变，不然的话很容易就重名了，所以说先生成文件名随机，好，我就叫 file，name 等于 community YouTube 点儿 standard uuid 这就一个随机照片儿就可以了。

然后宣传完以后，因为我们采用义务的方式去生成图，所以我们需要构建一个 in mind 好 new in mind。

然后这个事件比较特别，它和什么点赞关注什么的不一樣，我们在这个时间里也不用传什么实体什么的，我们这个数据怎么携带，这样我先把 topic 先指定一下，然后的话再去携带这个携带数据，我们可以用 map 可以传任何数据，但是目前这个 topic 我们还没有加以声明，我还得打开常见接口对它做一个处理加一个主题，在这讲主题这个主题是分享，好，然后回来这里我得实现接口，便于引用常量，好。

这个主题就是 topic sell，然后我要携带一些参数，直接点 site，data 这是把这个数据存到了 map 里，这样就方便一点。

首先要存的就是 html UI html路径把它传过去，然后 site继续再site一个file name，文件名我们在这生成也要传给他，文件名我们在CTRL来指定，然后让卡夫卡按照这个名字去生成，这只是文件名，注意它还得有后缀后缀，你是PNG还是JPG，这个也是我们在这指定 site data surface就是后缀，我希望都是统一的，都是偏近就完了。

好，这个事件相关的内容就指定好以后，我们就触发这个事件一慢不丢色，然后就发一卖，把对象传入就行了，这里写个注释，这是异步的生成长图。这样的话如果说你当前同样打卡分享的人特别多，那么我们这个方法的处理的效率也很高，因为他不用等待，这个内容处理完就可以继续下去好了，然后我们需要给客户端返回点什么东西，返回什么呢？

返回一下就是说我生成完长图以后长图生成完以后你怎么去访问它，我需要给你返回一个访问路径。所以下面我要给它返回访问路径。这个访问路径我们是这样的，我们最终是return应该返回一个Jason字符串对吧？

我就每天肯定那天又跳要 Get精神的时候，咱们第一个可以给一个0表示成功，第二个 now我没有什么提示的消息，然后第三个我给他一个map，就 map里可以额外携带一些数据，所以我把返回的路径放到map里，因此我觉得这样需要16块的map，然后在 map里存上一个路径 map，点put路径我就叫curl好这个路径拼的时候，你注意你可不能拼成这个东西，D盘什么对吧？因为d盘是我们本机的路径，用户远程不能用这样的方式来访问，他需要通过HTTP协议来访问对吧？怎么访问我们就需要拼了，首先是度命，我们这个项目的域名加上contact pass项目名，说白了就是HTTP冒号双曲线，比如说127.0.0.1斜线community这样的路径对吧？然后后面还得加上我们自定义的路径，我们希望下级这个功能路径是什么？是赛尔这是我自定义的，然后 Image，然后image下集跟的是 family就这样就可以了，他通过这样的外部路径来得到这个图片，当然这个路径需要我们实现，我们一会要写这个代码去实现。

当然我们先停一下，我们先做这样的一个事情，就是先把异步请求给它处理掉，现在还没有处理对吧？

所以我们需要写消费的事件，我们去打开英曼特跟萨玛，那么在这个类里头我们也需要注入一些内容，那么首先我们需要注入的就是 command，wk command以及wk的存放的位置，这里边我们都需要用到，然后注入一下，这个可以copy咱们刚才 CTRL那里好像他有 stories把它拷过来，当然我们还需要另外一个 come on的给他补一下，wk里面come好。

拷过来好了，这两个内容有了以后，下面我们就可以来消费，分享实践了，我们就再加一个方法来消费这个主题就好了。好，我就写上卡夫卡，为什么？然后话题等于这里边我只有一个值，就是 topic。Sell，当然了后面他需要跟着一个方法，wide handle。Sell message，然后当然这个方法需要有个参考参考一下，然后这个方法一开始我们也是怎么判断这个参数，怎么去取阴脉，和之前的这些方法都一样拷一下。

好了，如果说我们能够取到这个参数，我们就去处理后续的业务，我们从1万个点能得到什么，把那几个值先得到。一个是html URL，you might点 Get data，这是一个map，然后再get htmlurl这就是我们想要的的数据，当然这里需要转型一下，这是第一个值，后面还有两个值，我们可以copy。

第二个是send name，t也是send name，然后的话是 surface，后缀 k也是一样。好，那么这三个值取到以后，我们怎么去执行这个命令，就是利用这些条件把命令拼出来，然后就执行就好了，对吧？我们需要先拼命令，他不是写死的，首先需要拼关键词，就是wk，盈利com对吧？然后加上跟着的是压缩比，或者说质量杠杠，quality，然后75别忘了前后有空格。

好，然后我再另起一行再加上就是html的访问路径，htmlurl然后再加上一个空格，再加上图片存放的本地位置，就是wk位置，stories对吧？

然后他后面这只是目录，你还得加上图片的名字和后缀对吧？然后加上斜线，加上final name加上后缀。

好，连起来就是wk html to image。尴尬quality75。空格。Http什么？空格。D盘如何对吧？就是这样  
一个完整的命令。好，有了命令以后怎么执行就 get run time点exec对吧？然后的话传进来这个come  
on就行了，当然我们需要对这句话处理异常。那么这样如果说没有问题，这里我记个日志，log点Info说  
生成图成功，然后把 come on的把命令给它也记下来好。那么如果失败的时候，给个也记个日志，还  
有生成图失败，然后这个时候我把失败的信息就把它记录下来，就这样。

好这样就可以。

这个事件就消费事件的逻辑就写完了，写完以后我们再回到刚才的CTRL上，这里我们只是说能够把这个  
图片生成，那么你给用户返回的访问路径我们还没有处理，这个处理就比较容易了。

因为之前我记得我们做过上传头像，上传图像应该是在 user controller里对吧？然后头像的访问路径我  
们不是处理过吗？其实跟这个是类似的，在这里我们就类似的逻辑我们再写一遍，获取成熟为快速的卖  
品 pass，我叫sir，然后 image后面跟着一个family，其实你看我们这个是不是这样写的吗？

我们的路径前面不说了，项目里面不说了，然后赛尔盈利成本是范围内对吧？所以我们需要按照这样的  
方式去处理或者说实现这个路径，然后 master等于 get好have it？

这个方法就不能简单地返回一个，它不是返回模板，而是直接给浏览器返回一个图片。我们需要通过  
response直接处理。好这个方法名儿，我叫get sat eat。然后首先我就要把路径中的参数先得到单位内  
部，先得到用 pass variable。好，另外我需要用 response，那么准备好以后，接下来我就在这里用这  
个risk向外输出这些内容。在一开始我先做个判断，判断一下这个参数。String YouTube点is blank。  
File name。我先判断一个参数，如果参数为科目我就抛异常，这没法返回说文件名不能为空，好，那么  
不为空我们就要输出了，手动输出内容的时候，首先你要声明我输出的是什么 set，content type那就是  
你妹去偏近，好，EMAIL表示我输出的是图片，后面偏近的表示的是格式，然后继续我需要我输出的  
话，我需要读取本地的文件对吧？

我先实例化一个file，new Fein这个路径是wk midge storied对吧？加上文件名加上斜线加上文件名，  
file name对吧？加上点评级固定的。

好，然后我们就要输出了，输出的话我们因为这是图片是字节，所以我需要获取 output stream，通过  
response获得好这句话说要处理异常才开始一下，那么如果有异常，我们给个记个日志说获取长途失  
败，后面加上详细的错误信息，好，那么如果没有18我们继续我觉得他要读取这个一边读取这个文件，  
一边往外输出要读的话，我要实例化一个file，file input stream等于new file input stream。

把file传进来，把这个文件转换成一个输入流，然后的话每次读取我们需要有一个缓冲区大一点，8分等  
于new，but你长度自定义我定义为1024个字节，然后在上面的游标，然后通过while读取，如果说b等  
于fis点read8分，如果这个值不等于-1，每次我读读取到了读取的数据都读到8分里，然后的话把它赋值  
给b如果不等于-1就表示我确实读到数据了，这个时候我就把它写到输出给目标，如果说没读到 while就  
结束 os点。

Right，8个。

然后的话从0~b因为你最多能读缓冲区这么多值，但有可能最后一次达不到这么满，所以说你要有一个范  
围声明，好，这样就可以。我们读取图片的方法也就完成了。好，接下来我们就可以测试了，我把这  
个项目重启一下，好勒。重启以后打开浏览器访问一下，House。我先访问个首页，然后这个就不需要  
登录，咱们也没有什么限制，谁都可以访问，然后的话style Cr后面你要带一个参数，带的是什么？带  
的是 Htmlui对吧？

是一个外部路径。你想把哪个网页做成图片，就把这东西传进来，我就问号htmlurl等于我就写留客网吧  
好，这个路径写完整好，完整以后回车，回车以后他给我一个响应，我这搞错了，你看按理说这个地方  
应该是一个HTTP什么，但是他想象这个说明什么？

他我这参数注入的时候写错了，估计看一下确实写错了，这里应该写个到了括号忘写了，图片生没生成  
没生成，看看报没报错，报错了。

然后说找不到指定文件，我估计那个地方我也写错了，看一下，确实这地方也写错了，就是说你看他的反馈，他反馈说明明这个地方应该是一个具体的值，但他写的是k那就很显然你k没有被成功地替换出value，那就是你注入的时候就写错了，所以这个是很明显的。

好，改完以后还得再重来一遍，肯定是f9重新编译，好写完以后打开浏览器再来一遍，我刷新一下，回车，怎么还是不对？再刷一下。我这儿也没写对，我说怎么还不对，再来一遍。Ctrl f9。这个图片生没生成看一下。图片已经生成了，这样我把它删掉重来。好已经重新编译重启了，我再回到浏览器回车，你看这回这个路径就对了，虽然长一点，但是它格式是对的。

看文件生成了双击打开，确实是一个图片，那么我通过它返回的路径能不能访问，再打开个页签粘过来回车可以访问是吧？这次课我们生成长图就演示完了，然后怎么去模拟开发一个分享功能，在一个分享功能，你把它用起来，咱们也给大家做了演示。好，这节课我们就演示到这里，咱们下次课再见。

## 7.9 将文件上传至云服务器

---

51-将文件上传至云服务器.mp4

这节课我们来学习如何把文件上传到云服务器。

那么上传。

到云服务器有两种方式，一种方式是通过客户端上传，我们在网页上在表达里直接把数据提交给云服务器，而不是提交给我们的应用服务器。要改变提交的目标，我们的项目中有一个功能刚好适用于这种方式，就是用户头像的上传，咱们用户登录以后不是可以在用户设置里改头像对吧？

那么上传的时候我们可以直接将表单提交给云服务器就可以了。

然后第二种方式是服务器直传我们的应用服务器，把这个数据把文件提交给云服务器。

那么。

我们刚刚做过的功能就分享的时候，生成的图片因为是在服务端生成的，所以说如果你想把它传给云服务器，那就得通过这种方式的，所以那么一会儿我们这两种方式都会给你演示，然后我们就把这两个功能重构一下，把这两个功能所提交的文件都传到哪呢？

我们用的是七牛云的服务器。

那么当然了。

主要是从我们学习的角度来选择云服务器的，因为其中云它能免费的试用，比较方便，所以我就选择76云大家都好用。

然后。

七。

牛云它的官网是7牛.com，然后我们要用的话，你需要先对这个网站先对这个产品有所了解，我们来看一下。

好，这就是它的官网，那么如果你想使用的话，你需要先注册，然后注册的时候你需要选择这个账号的类型是个人还是企业，当然你现在需要选择的是个人的类型，然后注册好以后，你还需要进行一个实名认证，如果不进行实名认证，不能用，而实名认证的时候需要这个上传你的身份证的正反面的照片，另外还要绑定一个支付的方式，我绑定的是支付宝，所以说注册以后它有提示你要注意，然后你提交了认证的资料以后，它有一个审核的周期，理论上是三个工作日完成，但实际上是比较快的。

工作日的话也就一两个小时就搞定了，比较快。

好，那么我已经把账号提前注册好了，所以说我就直接登录了。

那么登录完以后我们可以去看一下他的手册，因为7诺云怎么用，它的API怎么调，它是有比较详细的手册，而且是中文的，所以说你自己去看也能学得会那么如果你想看手册的话，我们是在这里点好点管理控制台，然后右上方有一个文档，然后有开发者中心，在开发者中心里，那么我们用的产品它有很多产品一个系列的都有，那么我们用的是对象存储，就是纯图片纯音频视频这个都可以，我们选所以我看一下这个产品怎么用，他这是他详细的手册，这个内容比较多，他的介绍手册比较多，那么你在看的时候，我觉得大家有必要看的一个是关于产品简介里面的基本概念，你要了解这个云服务，就是对象存储它能给你提供什么样的帮助，你要有所了解。

然后你。

要看一下存储区域，它有哪些区域你选一个离你近的，这样的话上传的速度会比较快一点。

再一个你要看编程模型，就编程的时候它的一些基本的原则，或者它底层的一些基本的架构大概是怎么样的，你对它有所了解，对整个机制有一个了解。然后再看一下安全机制，那么就是我们上传资源的时候，我们需要用到两个。

两。

个k两个密钥，然后的话这怎么用的？为什么要用这两个密钥有一个介绍。最后你再看一下上传资源，就上传资源的话，它分为几种形式你要有所了解。

好，以上是对它整个产品做一个了解，那么从编码的级别的话角度来说，你最好再看一下 Sdk，DK里边有一个Java SDK，就是我们用Java语言去怎么去访问它，这里也有比较详细的介绍。

好了，这些内容就是说我们课上不去一点一点看了太长，那么大家课后可以自己。

有选择。

的去看，那么课上我会把主要的内容给你做一个演示，那么你照着我去做，也能把这个功能实现了。

好了。

我们要想使用七牛云，首先我们的选择这个产品服务，选择对象存储，点进来，这手册我得点到点回主页，这还是开发者中心，我回到这个地方就关掉了。

回到就是七牛云控制台，在控制台里你可以我们要用的是对象存储，你可以点这立即添加，立即添加什么？

添加一个对象存储的空间，你创建一个空间，好来存东西，当然你也可以这样在这点对象存储，因为我不是每次都要添加的，比如说我添加过以后下以后我就从这儿进入，所以说我习惯了在这儿进入，进入以后其实我已经创建了两个空间了，一个叫header，一个叫cell，header是用来存用户头像的这么一个空间，cell是用来存分享时生成的长图的这样一个空间。

总之你要存什么类型的数据，我们通常一种类型的资源创建一个空间。好这是我以前创的，我先不用了，我创建新的，你要想创建新的空间就点新建存储空间，然后这里边你要给空间取个名字，然后取名叫做 community hide，叫这么个名字。

就是说这个空间的名字就意味着我往这个空间里要存的是用户的头像，然后你这里可以选一个存储的区域，我选华北，因为我的位置处于华北，这样的话传的速度会稍微好一点，再一个访问控制是公开的还是私有的，那么它的区别就是什么？

就是我们上传的时候一定是有限制的，但是我们读取就是传完以后资源是不是公开谁都能读，还是说只有你自己有这个密钥才能读，我希望所有人都能读。

因为我们传的图片之后，其实是每个人的头像是吧？是谁都可以访问的，这个是不做限制的公开的，然后的话点确定好这个空间创建好了，创建好以后你要注意它这里有个提示说，它会给你分配一个测试的域名，我们怎么访问这个空间，就是利用这个域名来访问，但这个域名它不是永久的，它是30天它就回收了，所以我们域名只能用30天，如果到期以后怎么办？

你再创建一个新的空间，再来一个新的域名改一下就可以了。

然后它每个域名我们试用免费使用，然后的话它是每日限总流量10g我们学习用肯定不会超过这个数量，所以说基本上就是免费的挺好的。

好，然后的话，当我们传了内容进来以后，我们可以在内容管理里在这里边有个列表能看到，现在的话我刚创立的什么都没有，所以目前没有任何内容，当然如果说你有自己的独立域名，你可以在这里绑定你的域名。

好，那基本上我们主要是通过空间的概念了解测试域名就可以了。再一个我们通过内容管理，一会能够看到这里面传过来的内容就可以好。这是创建的一个空间，我再创建一个，因为我还要存分享的生成的图片了，对吧？再创建一个我觉得挺牛，那天。虽然然后我还是选择华北公开的空间，创建好了以后，这个空间又有它的一个独立的域名，这两域名是不一样的。

我之所以强调这个域名，是因为我们一会使用的时候，我们要访问这个空间，你得用上，用这个域名还需要用到它这个空间的名字，这两个地方都要用到好了。我们做了这样一个基本的操作以后就可以用了，我们就可以通过 Java的API来访问这个空间往里传东西了，传完以后这边就能看到。当然你也可以通过 URL直接就是远程的访问到它。

好，下面我们就来讲解一下我们在项目中怎么去利用青牛云把资源传到这里来。我们主要是两个功能，咱们一个来做，第一个我们先来处理就是传头像。好当然那么我们在做这两个功能之前，首先还需要做一点准备的工作，你得倒包对吧？青牛的 SDK你要访问的话，你需要把包倒进来，然后我们在配置文件里再做一点点配置，方便后面的程序的开发。

所以我就打开 may every postcard com，然后我搜一下青牛，就搜青牛的拼音就可以了。

然后第一个就是气流，Java，SDK我就选择它选一个版本，然后 copy一下，好，下载完了，有了包以后，接下来我们再做一点配置，这个配置其实不是七牛云，就是硬性要求的配置是我们自定义的配置，是为了让我们这个程序更灵活，配什么呢？

主要配两个内容，一个是配密钥，一共有两个密钥，我们来看一下这个密钥是什么？我们从七牛云的空间里能看到看哪呢？这就是你的用户的头像，然后这里边有一个密钥管理，然后这里就有两个密钥，一个叫这个叫ak就是aqcess SK，这个叫SK，Secret key你可以显示其中前者密钥是用来标识你的用户的身份的，就是说我们要往这个空间里传东西不是谁都能传的，你只有有密钥就表示说你是当前这个空间的使用者，你才有权限传，如果你没有密钥是不能乱传的，不然这个空间就被挤爆了对吧？

这个是用来标识用户身份的，SK这个是用来我们在上传具体内容的时候为内容加密的，所以需要两个t这两个t我们最好能把它配到配置文件里，万一说你将来换了这个空间的话，或者说这两个key有所变化的时候，你好去改动，而不是把它写死在这个程序里。

好，我把它copy一下，把它配到 application process里，好在这儿七牛，然后因为是我们自己的配置，所以说你 k随便写，这里我就写七牛点k密钥，然后点access，这个是用用户身份的标识的密钥，我把 key贴过来。好再来一个7牛点key点，secret这个是用来对文件内容加密的密钥，我把刚才的自拍拷过来，好了，这密钥就配好了，配好密钥以后还要配什么？

我们再配一下我们刚才所创建的空间，我后退一下，我们刚才创建了两个空间，那么我们要访这个空间，一个是要指定空间的名字，第二个是要指定这个空间的域名，每个空间的名字和域名我们最好做成可配的，因为万一将来要变的对吧？

所以把它配置一下，这样我先配黑的，这个名字好记，主要是一个域名copy，然后他说了你这个域名访问的时候，你只能用HTTP协议访问，不支持https，所以这个时候你要注意好，这里我写犀牛1.8k的桶。

其实就是空间的意思就是这个云空间的意思，存储空间的意思，它76云管存储空间叫bucket。

然后点还有点。

这个黑的是我存放头像的空间点，name的是它的名字是什么，空间的名称是什么等于可没有那天还有点，然后再来七牛点儿，8k的点儿，a的点儿，URL就是用来存放头像的，访问路径是什么？是hgdp冒号双切线手敲上，然后把这个路径写出来贴过来。

好，同样的我们另外一个空间也做这样的配置，主要是考一下域名7牛点8k的点，这个空间用来存放分享的文件塞尔 name搞错了这个name。K没有，那天赛70.8。K的点赛点12。好了，主要就是配其实就是两类数据，一类就是密钥，一类就是空间，因为我们有二个空间，所以说配了两份，就这样了。

好，配置完以后，接下来我们就来处理上传头像的逻辑，首先我们找到这上传头像请求是在哪处理的？

是在user controller里，所以我就打开 You there controller，然后我先干什么，我先在这里注入一些刚才的那些个参数，我把这两个key注入进来，另外把跟头像有关的空间的内容注入进来，因为一会儿写代码要用好，我先把道路括号写上省得忘了。

首先我要注入的是第一个key access key copy一下。

第二个我要注入的是secret key，然后再注入的是空间的名称，最后注入的是空间的路径。

好了。

然后这些k我就考点一面写错了。

好了，接下来我们在这里要写什么代码，我们主要是要处理上传的逻辑，而这个上传的逻辑我们之前实现过不漏的，对吧？现在我要把它作废掉，为啥？刚才不是说了我们上传头像，因为有表单，所以我们可以采用客户端上传，表单直接提交到76云，就不走这个方法了，因此这个方法就要作废，这里我就不把它删了还保留，但是我写上说要废弃，你注意不用不要使用它就好了。

好，另外我们上传完的头像在我们本地的一个路径之下，我们怎么访问，我们是提供了这样的一个方法，可以让人访问。现在你是把这个文件传到了七六云上，这个方法显然也就没用了，对吧？它也是比较废弃不用了。

好，我先把这两个该废弃的废弃掉，我们需要做的事情是什么呢？我们在这个路径是打开用户设置的路径，打开上传头像表单的页面，这个时候我们需要在这里生成上传凭证，把凭证写到那个表单里，当我们打开这个表单的时候，表单里应该有凭证，所以我们需要在这里生成凭证，然后传给你模板，然后表单把凭证配好以后，提交的时候才能提交给齐能源，否则他不认。

好，所以我在这里补充一些内容，主要的内容其实就是主要一步就是生成上传的凭证。

好，让汽车云能够识别你的身份，而上传凭证它主要是它需要一个是需要上传文件的名称你要提供，每次传的文件不一样，凭证它也就不一样，再一个你可以指定，那么当我把资源传给齐能源以后，他给我们一个什么样的响应，你期待他给你一个什么样的响应，你可以把响应做一个规定，他按你的规定给你做出响应，这样我们好知道成功还是失败，所以我们还要设置响应信息，然后我们根据这两个内容再去生成凭证。

所以第一步我是要生成上传文件的名称。

然后第二步我还要设置响应信息，这两个设置弄好以后再去生成凭证。

好上传文件名，我们每次就生成一个随机的就可以了，而且你最好生成随机的。

当然有人可能会想这个好像不需要，因为是用户的头像，比如说我这个文件名就是用户的ID点评级不行吗？用户又在那么点偏激，肯定跟别人不一样对吧？怎么说第一个如果你这样做的话，比如说我一开始传了一个头像，后来我又传了一个头像，后者会覆盖前者，它不是一样的名字对吧？它会覆盖。那覆盖以后，如果说将来有一天我们想说开发一个什么类似于足迹那样的功能，想给你显示出来你这些年都做了哪些关键的操作，你以前传过什么头像，那就没有历史记录，这是第一个。

第二个如果说你是同名的话，我们把这个文件传到云服务器上，然后我们去访问这个云服务器，你注意它是立刻会传上去，但是我们访问的话你会发现它在一段时间之内不会立刻生效，因为云服务器它都有缓存，所以你名字不便传上去，你去访问它，因为还是同名的它有缓存，你不会立刻刷新，所以我们通常就会生成一个随机的文件名，每次都变，这样保证的话它缓存没有效。

再一个我们历史的数据也可以做一个存留，也不会真正的删掉这样的。有人说这样不是耗费云服务器的空间吗？其实空间如果你交费的话，其实也没多少钱空间是最便宜的，硬盘是很便宜的，没有关系。

好了，那么我就生成了一个随机的文件名，然后生成完以后我还要设置响应信息，怎么生成？这就是新能云它做了一个规定代码，就这么写，我们要生成的响应信息存到一个叫string map的对象里，然后这个我叫 policy，然后你又死去，好，然后的话 Policy往里存东西点，put其实就是一个卖方点，put return高点就是响应的体，你希望响应什么内容你注意。

我们使用汽车云客户端直传的话，通常会采用异步的方式来传，因为异步的话它给你返回一个杰森这个事儿的比较方便，你如果是用这个不是异步的方式同步的方式，他给你响应html对吧？

我们就没法处理，你怎么说我们目前在牛客网的网站上，然后你现在把数据提交给7676云，给你返回一个网页，这就乱套了，所以我们最好是他给我们反馈一个Jason，只要告诉我们成没成功就可以了，后续我们自己来做进一步的处理，所以通常都是异步的。

因此我希望他给我返回一个Jason字符串，告诉我成功与否就可以了。这里我就肯定有 Youtube的get Jason0。

好，这句话最终会生成一个Jason字符串，就是扣的冒号0，就是说我希望成功的时候，他给我返回扣的冒号0就行了，如果失败了，你只要是是不是这个值，我就认为是失败就可以了。

好响应信息我们也做了一个声明，最后我们利用刚才这两个内容去生成上传的凭证，上传凭证他这个对象叫 auth然后的话我们给它实例化一下auth点，用create方法来实例化的时候，你要传入两个key，第一个是与身份有关的access key，第二个是与加密对内容加密有关的 secret key。

好了，然后我们上传 aoth对象以后，我们就可以利用这个对象来生成，上传生成的凭证其实就是一个字符串。

我叫阿不露的。

ok还有。

th点up low的投给，然后第一个参数写的是。

我们上传的空间的名字的8k的内容。

第二个是我们的文件名 file name。第三个你是指定一个过期的时间，就是 key它有效期是多久，多长时间以后过期，好这里我写的是3600秒一个小时。好，最后把响应信息传过来，这样就可以了，你把这4个信息给他，他就会给你生成1个字符串，因为我们的名字每次都不一样，所以这个字符串每次都是有区别的。

好了，生成完凭证以后，我们需要把凭证发给模板上好用，我就model点l的，这里我还没有声明，model声明一下，ID然后 up copy好这是上传凭证，那么另外我们再把这个文件名也传给模板，因为模板要向76云提交数据，这个文件名在模板上也有用的。h股，然后的话filename内容这就可以了，那么我们页面上总之就利用这些数据模板上就利用这个数据去构造重新构造表单，然后并且我们把表单的提交方式改为异步的，提交给700就可以了。

但是我们先别着急去处理这个表单，我们在这个control当中再加一个方法，再加一个什么方法。当我们在表单里把数据提交给新能源以后，异步的新能源给我们反馈了一个消息，比如说扣的0成功了，成功以后怎么办？我们是不是要把我们优则表里的 Hi的URL做一个更新为青牛云的路径，对不对？如果你不更新的话，我们表里的路径没变，你白传了是吧？



这个需要我们的应用服务器来做，而我们之前的话没有专门搞一个方法去处理这件事，我们之前就是在阿波罗的方法里就顺带做的，但现在我们需要单独提供一个方法给页面去调用，而且这个需要是异步的，我们把这个方法写一下，就更新头像的路径，先声明一下，访问路径pass等于。

结果叫hider。用。

然后 MR的等于request点post，因为你要图提交数据更新数据进来，所以说post然后一步的

方法名我叫update high的UI好，那么页面它想更新路径，它需要传入什么呢？它需要传入的是文件名family，我这单词写错了，好，他把翻译内容传进来，传进来以后，我们得判断一下它是不是为空的话，不行，你文件名为空我就没办法去更新了，可以先判断一下。

如果说 File name是空值，那我就直接因为是异步请求，我就直接返回一个json，字符串给它做一个提示。Community YouTube点get接生死卷一表示错误，然后的话说文件名不能。

为空，

如果不为空的话，我们需要拼一下 URL。我们访问76元的URL是什么呢？固定的。刚才的空间的域名，因为我们把数据存到了嗨的空间里，而空间是有个域名的对吧？

所以说我们就把这个空间域名斜线加上final name，这个文件的访问路径，好，我先拼一下 Url等于空间的名字房名叫hider，8k的uil对吧？

加上你得加个斜线，然后再加上非常内容，这就是这个文件的从76元上的访问路径，然后的话我就更新 user service update，header，user ID就是当前的user对吧？我就host older get user get idea好，然后 URL传入进来，这就可以了，最终你要给页面一个反馈，community YouTube的get杰森。是这样，0成功就完了。

好，现在我把 user control已经重构好了，主要是我们废弃了两个方法，然后对原有的方法做改进，另外的话还写了一个新的方法。

那么这件事完成以后，接下来我们就可以去处理对应的表单了，我就打开对应的表单setting点太慢。这个表单找一下他，因为现在我们是表单里要多一些内容，因为你传新能源的时候需要加就凭证对吧？要多一些数据，另外这个表单还不是普通的提交，你还要做成异步的。

所以说改动较大怎么办？我就干脆把原有的表单给它注掉，然后我把提前复制好，我们重新写一下，这样的话如果你想找到以前的代码也有一个留存，好，我把之前的 Form注掉，然后接下来我再到后面再把刚才我复制的内容贴过来。

然后这里我写个注释，原有的内容是上传。

到本地。

现在这个表演讲示的是上传到七牛云。

好，那么这个表单我需要做一些处理，首先什么my sir等于POS的EMC type等于什么action等于什么，我就把它注册掉了。因为我们现在不是同步上传，我们是异步的，是要通过js去编写上传的逻辑的，为了js写上传逻辑的时候获取 form方便，我给它加一个ID，ID等于up load form。这就行了，一会他提交的提交给谁，用什么样的方式提交，我们都通过GS搞定，这是其一。

其二这里头样式不需要了，我把这个去掉，样式还不能都去掉，就把 th动态的部分去掉。

好，总之恢复成原状，我们之前的逻辑就不需要了，然后 ID该保留是保留 name，注意我们传记录员的时候，这个name需要写成就叫fan，把它改成叫fan。好，然后后面的话，

错误提示这也不需要了。好，就这样把之前的一些处理方式给它清理掉，然后我们重新来过。好。那么这一步我们处理完以后，我们还需要补充两个数据，这两个数据是传给青牛云的时候它需要的，我们就加两个黑的，因为这个数据不需要展现 input，tap等于黑的，然后 name等于固定的叫token，上传的凭证传给青州云的时候，这个名字必须叫token。

这个token的值是谁？就是我们CTRL传过来的阿波罗的token value等于 Upload token还需要另外的一个黑灯，就是我们还有另外的一个值要传给青龙云，那个叫k而 k的值。那个文件名大家好，这两个是700要的，他要求的，我们就这样处理就好了。

好了，那么现在我们这个表单处理完了，处理完以后我不说了，我们需要异步的提交对吧？得写js，所以说我们看一下当前的页面，它所引用的js文件是什么？Aaah是Globo的GS，我们把这个代码写到这里来不太合适，所以我们给这个页面单独创建一个js文件，Copy一下，这个文件我叫citing点GS，然后我在这个时代 GS目录下新建这么一个文件，塞停点减s这个文件的逻辑我们需要从头开始写，首先写什么？

就是我要给刚才的 form不是按钮，因为按钮一点 form就提交了，我是给 form定义一个时间，当我们点submit的按钮的时候，会触发表单的提交，其实是触发表单的提交事件，我需要对这个事件做一个定义，我定义这个表单具体怎么提交？

好，所以我在GS里面我这样写，到了括号 function，表示说页面加载完以后我要去调 function调匿名函数，页面加载完以后我要干什么，我要给那个方式给 form定义一个事件，就是重写它的事件，提交事件我就写先获取它的ID叫upload放它的时间是提交时间就submit。

然后。

upload upload是一个方法名，但是我还没写马上就要写，意思是当我点击提交按钮触发表单的提交事件时，这个事件这个由阿波罗的函数来处理。

好了，下面我就来写阿波罗的函数，好。然后但这个最重要，你别忘了要return false，如果你不return false，尽管说你这里能够做一些逻辑处理，但最终它还是会尝试提交这个表单，但你又没有写action，这里就会有问题。我们return force意思是你就不要再往下提交了，我们上面的逻辑已经把请求给处理了，已经把这个事件处理了。Return false表示说事件到此为止，不再继续向下就执行你默认底层原有的事件了，这个意思。

然后我们要想一步的提交表单，这个表单它还比较特别，它里边包含的不是普通的数据，是这个文件，那么所以我们还不能写到了点post，因为到了点post是一个简化的方式，有些参数没有办法进行设置，我们需要写一个能够设置更多参数的更强大的就是异步请求的方法，那就是到了点的价值。

其实到了点get，到了点post其实是对这个方法的一个简化，我们现在不简化，用原始的稍微麻烦一点的方法，但是它功能强大，然后这里边写一个大括号，里边可以包含多种ky6，每一种ky6对应的是一个参数，你可以定义帧，就是所有的参数都可以在这里进行定义。

好，首先我定义的是URL，我这个请求要提交给谁，我要提交给秦流云，我这个路径从哪里找呢？那么七牛云的手册里就是文档开发者中心对象存储，然后它不有一个存储区域，存储区域，华北这个区域它的名称叫g一，然后的话如果你是客户端上传，它的路径就是它up6的杠z1.76up.com，我把它copy一下，省得写错了，路径就是它了。

你HTTP或者是加s都行，我就不加了，就这样，当然你前后你得加上引号，因为这是一个字符串，这是一个参数值。

好第一个写上URL，第二个写上message，我请求的方式是什么？是post。然后第三个比较特别了，你看我没见过 process, data false。这句话的意思 False是拒绝要不要把表单的内容转为字符串。那么默认情况下，我们提交表单浏览器会把表单的内容转换成字符串提交给服务器，但是你想我们现在是上传文件对吧？不应该把它转成字符串，所以写成放肆，然后还要写。

content tap first。

按理说 contentType应该写什么？你是html还是Jason写写的是你传的数据的类型，但是我写成force什么意思？是不让解carry去设置上传的类型，那么浏览器会自动的去进行设置，为什么这样呢？是因为我们在提交文件的时候，那么浏览器就是文件和别的数据不一样，文件它是二进制的对吧？它和别的数据混编在一起的时候，它的边界怎么确定？其实浏览器会给它加一个随机的边界字符串，好去拆分这个内容。

如果说我们这里指定了content type，那么且夸人就会去自动的去设置类型，那么边界它设置不上，就会导致我们传这个文件反正就有问题，主要是边界的问题，我们通过手动设置的边界 key又不行，key需要用浏览器随机生成，所以说就得设置成force，你要注意。

好，然后再来 data就是我们要传的数据是什么，这个数据需要特殊处理一下，我们叫new form。

data。

这是一个js对象用来封装表单的数据，当我们传文件的时候需要做这样的特殊处理，然后你把对象form传给传到对象里就可以了，而form是谁？Form不是他吗？Upload form对吧？

然后你注意这是我们用解决选择器选中的节点，我们得到的是解决对象，但是这里这是GS，它要的是GS对象怎么办？方块0，因为解决为对象其实就是盗墓对象的速度，这是它的本质。当我们从这个数组中取到某一个值的时候，就得到了一个动，所以我取d0的值得到一个down就符合要求了。

好，然后再来success，表示成功的时候我们做怎样的一个处理，我们需要说明一个函数去处理响应的信息，那么函数的参数就是响应的具体的值，这里我做一个判断，如果说 data存在。

并且带头点。

扣的等于0，对它存在，并且对它的扣的等于0就证明成功了，对吧？然后的话否则就是失败，失败怎么办？给个提示，提示说上传失败完了，成功怎么办？成功的时候，我们需要更新头像的访路径对吧？

我们需要还是在这个页面上用eve的方式去访问CTRL更新头像，所以好这里再接着写更新头像访问武将还是异步的，到了点pose，因为提交给我们自己的应用，所以说我就直接用简便的方式来写了。然后路径是context，pass加上user，然后 hide URL，看一下确认一下，hide URL咱们类上有优点对吧？没有问题。好，然后再声明一下我要传的数据的名字，我们这个时候要传给 Controller的是 found in。好，冒号 final name咱们可以从表单里取对吧？

看一眼不就是他吗就黑的，这个黑的它的特点是它有name，name等于t我可以利用这个特征把它取到，当然你加个ID也行，这里我就懒得加ID，我就直接通过内部来取了，到了括号，Input元素选择器选择所有的元素，然后再翻括号，属性选择器选择name等于k的属性只有一个，所以我就得到了对应的黑的，然后点vi得到了它的值传给了服务端，服务端处理完以后会给我响应，我要接着处理。这是我们服务端的响应，我需要怎么说对它做一个解析，我们服务端返回的是普通字符串，我需要把它解析成json。

到了点pass。

接着刚才为什么这个地方我们没有说做一个解析，就这个地方新能源给我们它是返回的杰森格式的数据，而我们的服务器我们统一返回的都是字符串，普通字符串并没有声明为杰森，但是我们知道格式是Jason，所以我们可以把它转型为Jason的对象，好转完以后判断如果带头要扣的等于0就表示成功了，这个时候你刷新一下当前页面，让头像变一下，你一看就知道成功了，就window location点别漏的刷新当前的页面好，否则否则就是失败给个提示 Alright，data点。

message。

好了，到这儿我这个代码就写完了，关于头像上传的逻辑，接下来我们就可以做一个测试。好，我启动一下服务，启动以后我打开浏览器，然后先访问一下首页，因为目前没登录，所以我也看不到头像，我得登录一下 CC，好我就登录了。目前的话 Cc的头像是一只小猫，我们看一下表里的数据，看一下它当前的路径是什么。好，你看它的路径是我们本地的服务器的访问路径，路过house对吧？我们一会要传的话就是要把它改成齐。

刘云的，好我就传一下，我就点这个账号设置。这里我选一个文件，改成它第一个，然后我要传了立即上传，传完以后页面刷新了，你看头像变了对吧？然后我先检查一下表里的数据，看看有没有变化，这样我倒叙一下倒叙排列方便找到这个数据你看 Url就变成了七牛云的对吧？

七牛云上有没有这个图片应该是有的，不然的话这也看不到，我们看一下空间概览，点 header，然后点内容管理，你看有了一条数据数据上传的时间在这，然后的话你也可以点这对吧？

做一个预览，没有问题。

好了，现在我们上传头像这个事就解决了，这种只是我们上传到云服务器的一种方案，只是通过客户端进行上传。

下面我们再给大家演示一下服务端直传，就是在我们的分享的时候，我们在后台生成了一个图片，我要把生成的图片通过服务器，通过我们的服务器直接传给760的服务器，在后台实现。好，我们就来重构分享相关的功能。再回到我们的程序里，我先打开下肯特勒分享对吧？好。然后首先我也是要传入注入一个参数进来，注入什么？注入分享的空间的URL，因为你看这里面我们没有直接在这个方法里实现分享，没有直接实现，我们只是把它丢给了一个队列对吧？

让队列去实现，但是我们在这需要给浏览器返回一个路径访问路径，因为你要把它底层改为七牛云，所以这个路径得改，那路径改的话，是不是你得知道分享的空间的URL对吧？用它来拼。好，所以我要注入一个东西，这个可以copy url这个是嗨这不是嗨的赛对吧？然后 Key我也copy，以免写错了。

好，整个方法其他的逻辑不用动，我们文件名还是随机的保持不变，然后我们还是通过异步的方式把它事件提交给队列，让队列去处理，这个还是不变，最终返回给浏览器的路径。

那么我们需要做一个改变，就不是原来的路径了。

我就map点to。

curl现在的路径是76元的空间的URL塞r8k的url然后加上一个斜线加上一个final，这是新能源空间的路径，就是这样的，你可以去新能源空间看，你看刚才这个图片它不就是这样格式，前面是这个空间的u二，后面是图片的名字，好这就可以了，这就改完了。

然后我们之前还有写一个方法，这个方法是从我们本地把图片取到，然后传给客户端，这个方法不需要了，因为我们一旦传给京东云以后，你是通过760去获取图片，所以这个方法也就废弃了，我做一个标记，但不是真正的删除。

好，然后剩下的逻辑就是去写跟萨玛就是去写消费者，因为我们的逻辑是在消费者里进行一个体现的，所以说我打开。

invent跟summer。

然后我们需要在消费者里头消费这个事件，然后像七牛云传东西，这里面我们也需要传几个参进来，包括那两个key，包括上传空间的名字。

好，我那两key我可以copy，因为刚才这里不是写过了吗，把它copy过来。好，另外再把空间的内容传进来，这口叫下。

打。

k的内容，然后它的key copy一下，好，那么我需要的参数就已经注入完了。

那么除了这几个参数以外，我还要注入一个组件，先写上就是什么 spread pull。task，schedule取名叫task。Schedule为什么需要注入这么一个东西，这是啥呢？这不是一个线程池，一个可以执行定时任务的线程池。

好，你可能记得我以前说过说我们通常新这样的定时任务都是通过class去执行，之所以通过class去执行的原因是因为要考虑分布式部署的问题，为什么我们还要在程序里用这个东西，为什么可以用做一个解释，我去我要写代码的地方做一个解释，消费分享实践。

好，我们之前已经把逻辑写好了，这个代码一执行，这个图片是生成了，现在要做的事就是把它传到70云上去，怎么传注意这里可不能直接传，直接传会有问题，有什么问题你记得上次课我写的代码的时候，就是说如果就这句话和命令谁先执行完的这句话，因为命令是生成图片，它是一个比较耗时的过程，它可能需要300毫秒500毫秒甚至一秒都有可能就看你图片的大小，你想这命令假设500毫秒以后才能执行完，那么我们当前的方法的主线程不是等着他，而是继续向下执行。

他从这句话跳到这句话，可能连一毫秒都不用对吧？所以这句话先于他执行，也就是说这句话后面的逻辑比他执行的还早。

所以你在比如说我要把这图片上传到青龙云，你这句话执行的时候，那图片还没生成，你怎么传对吧？不行，好我们怎么去解决这个问题？我们只能等，但是你怎么等你硬把这个程序阻塞在这吗？性能也太差了怎么办呢？

我们可以用这样一个办法，就是我在这人为的我启动一个定时器，我每隔一段时间看一下，比如说我每隔半秒钟我看一眼你生没生成，好，你没生成我在等着，再过半秒我再看了一眼，你生没生成，直到有一次我发现你生成了，ok我就去传，如果你一直不生成，我等了很久，比如说我等了30秒，我认为你肯定是有问题了，这个时候我就取消，我不传了，我认为你肯定生产有环境有问题，所以说我们这个地方就是要用到一个定时器，主要是来等命令的执行完成。

那么当然了我们这里可以用 Schedule了，为啥？因为你想我们这个逻辑并不是每一个服务器都会执行的，虽然说我们每一个服务器上部署了同样的肯萨姆，但是你注意消费者它是有一个抢占的机制，比如说我有5台服务器，5台服务器部署了5个肯萨姆，那么如果说一旦这个消息发出来以后，只有一个可能他们能抢到他去处理，别的可能他们不会去处理的，他是有一个天然的这么排斥的这个作用肯萨姆，所以所有消息队列都有的一个机制好了，所以说这个方法只有某一个服务器上会执行，其他服务器就不执行了。

在这一个服务器上我启动一个定时器和其他的服务器并不产生什么关联，不会产生什么影响。它这个地方和我们之前所说的定时器不一样，我们之前的定时器是服务器一启动，它就自动暂停运行了，对吧？你5个服务器启动就服务器5个服务器都运行，而这里不是这里是谁抢到了这个消息，只有这1个服务器会启动定时器和别的不发生关系。

好这里我要把做的事再写个注释，这里我要启动启用码定时器，然后主要是监视该图片，然后一旦生成了，图片生成了，那么我们就上传至七牛云，如果你不生成我就监视着，就这个意思。

好，既然我要用定时器，我是不是得有一个线程体？当然我可以做匿名的实现，但这个逻辑比较复杂，所以最好我单独写一个类，这个类我单独写个匿名的，因为别的地方也用不到。

好，我写个class，然后的话up load task上传的任务。Deployments。软的，当然我需要在这个类当中实现软方法，好还没完，这个类我希望给它加几个属性，第一个属性是文件名称，再来一个属性是文件后缀。为啥加这两个属性？因为你想你要调这个任务的时候，让他去传这个文件，你不得告诉他文件名是什么，文件的后缀是什么，然后它对应到对应的目录下去找到这个文件才能传对吧？

这个任务只管传，文件是谁，他不知道，你要告诉他你要传进来对吧？

好，所以我加两个属性，这是file。

name。

文件后缀。

Cervix,

那么你在实例化任务的时候是必须要把这两个属性传进来的，所以说我在这给它生成一个有参的构造器，强制它传这个内容，开始抓它，生成一个带有两个参数的构造器。

好，那么当然一会我要实现乱方法，我先不去实现这个地方代码怎么写呢？但这就容易了，就实例化task对吧？Upload task task等于6。

upload。

task。然后传入 family之前，我们有 family service对吧？好，这个任务构造完以后我需要调 task，schedule，然后 at fixed rate，然后把任务传进去，然后希望每隔500毫秒半秒钟执行一遍，这样就可以了。这个不需要太特别快，500毫秒也就够了。

然后的话你注意，就是我们这个定时器，你一定要考虑它什么时候停止，你这样搞的话，这么直接写的话它就永远不停止，这是不对的，我们从那以后要停止，定时器停止肯定不是在这停止，你在这只是触发定时器的执行，在哪停止，一定是在入院方法之内，我们当某条件达成的时候我才停止，任务完成了我才停止对吧？好，我怎么停止定时器，其实是我启动定时器的时候，它有一个返回值，叫filter，future里边封装了任务的状态。

除此以外 filter它还能够用来停止定时器，我们就要用到它。

你看这是他调了这个方法以后才返回 filter，这个filter怎么给这个round包用呢？也好办，我可以在这个类当中再加一个赛的方法，要求你实例化完以后，16画完以后，你把最后再给我传进来，用这样的方式来做好。

所以我这里再加一个属性。

首先启 feel就是启动任务的返回值，它可以用来停止定时器，好，那么我在这儿再生成一个赛的方法就可以了。

有了site方法以后，你看这儿我是先实例化了 task，然后我去启用了定时器，然后紧接着我赶紧把这个定时器，因为它500毫秒以后才执行对吧？马上下面这句话是先执行，然后才到500毫秒的。所以下面的话我赶紧就是task点set，标准把这个结果传给他，然后他在这里边在执行的过程中就可以根据情况调用它去停止定制器了，就这样。

总之我们在这个任务当中主要来写什么逻辑，就是上传在上传的时候，我们一定要考虑可用性的问题，这是定时器一定要保证它完成任务以后就关闭。那么就像我刚才所说的，有可能有些极端的情况下，他完不成任务永远都完不成。一种情况是什么？命令执行的过程发生了失败，图片就一直没生成，如果你这个任务他一直没生成图片，你肯定没法传对吧？传不了你传不了以后永远都成功不了，那你就无法停止。怎么办？对吧？这是一种情况。

还有一种情况就是有可能他图片生成了，然后我传给70云失败了，因为此时我的网络有问题，或者此时恰恰欺凌云的服务器它就挂了，万一出现这种情况你永远都传不进去，你一直在这传也不行，这样服务器时间久了，这服务器就被你给撑爆了，用这样的无效的行为对吧？

所以我们一定要考虑到这一点，我们一定要考虑一个兜底的方案，就是一旦出现这种极端的情况，我也无论如何要把它停掉这个定时器，所以我们可以做这样的一个处理。

就是我在这里再加一个加两个属性，一个是开始时间，我在实例化 Task的时候记录一下，开始时间任务执行的时候，我在得到当前时间我减掉开始时间算一下我这任务跑了多久了，我一旦发现这个任务跑了超过30秒，我就认为什么这里有了问题，我就不再执行了，我就强制关闭，一般30秒足够了，30秒都没解决，那就是解决不了了，我就把它强制关闭。

这是第一个，所以我加一个开始时间，这个我就直接用浪方便运算，叫start time。

好，我再来一个叫上传次数，因为这个时间比较长，30秒其实往往见分晓到底成功还是失败，不用30秒可能是几秒就够了，但我们这个几秒又不好界定，我们再来个上传次数，比如说我发现上传次数都达到三次了，这个是什么原因？一定是我上传到青旅的时候失败了，因为如果不失败一次就成功了，对吧？上传次数变成三，就表示说有失败的情况，然后再把次数累加了，如果一共上传三次，可能这三次发生在5秒之内，我也强制停止掉。

因为我认为我传三次新能源都没传成功，很有可能是网络要不就不行，要不就是新能源服务器挂了，我就不再传了，三次尝试就够了，所以我再加一个上传次数，upload times。

好，总之我们有这两个一个时间一个次数的保障，一定能保证我这个任务最终会停止。当然我们不希望这件事发生，最好是他能成功，但一旦出现意外，这有一个兜底的方案，就这个意思。好了，我这个时间得初始化一下，次数的话默认就是0，这个项目就不用初始化，这个时间你要初始化对吧？时间的话我可以在构造器里初始化，那么这里写this点start time等于system点current，当前的时间，系统当前的时间完了。

好，那么最后我就是要来写乱方法来处理上传的逻辑，这个方法其实是反复调用的，我们处理它的原则和递归差不多，这方反复被调用对吧？每个500毫秒被调用一次，我们最担心它不能停止，所以说和递归的处理原则是类似的，我们先去判断它终止的条件。

好，所以说第一种情况就是生成图片失败。

生成图片失败我们就会执行很久，我们会等很久，可能会等超过30秒，这个看时间超过30秒通常就是生成失败。

好，我判断一下，如果说sister点当前时间减去start time，这是毫秒大于3万，3万毫秒就30秒大于30秒，这个时候我们认为生成图片就有问题，我们强制他停止，我记个日志给他 Logo点l说执行时间过长，然后我就终止任务，并且我把这个file name拼到这个字符串里，我好知道是哪个文件我给它终止了，是哪个文件我终止的这个事儿。

当然了这只是记录日志，你怎么停止定时器调 feel去点看守，这就可以停止，然后树这就可以停止好。既然停止了，后面的逻辑不用执行了，return。

好再来，这是执行时间过程，通常是由于无法生成图片导致的。还有一种情况是上传失败，就说我传7度云失败，由于网络和某种原因，这个时候我看次数 Up, load times如果说它大于等于3，反正最多三次，我就不再传了，这时候再给一个错误提示。Logo点l说上传次数过多，终止任务也是我把非常内部提上来，也是停止定时器并且未退。如果这两种情况没有发生，那就意味着我就可以继续向下执行下面的逻辑。

好向下执行的时候，首先我要从我本地的目录里找到那个文件对吧？本地的路径是什么，我得把它拼出来。本地的路径是wk，什么image？Story，这是我存放WTO文件的目录对吧？然后加上斜线，加上file name，加上surface of后缀，这是我本地存放文件的完整的路径和名字，对吧？

然后我要看一下它存不存在那么实例化一个file。好把 Pass传入，然后做一个判断，file。

点儿。

exist，如果文件存在，那我就开始传，如果文件不存在，我什么也不做，什么也不做，过一会又掉一次，直到它存在，我再执行这个逻辑对吧？就是这样。

好既然存在，我这里要开始传了，我打个日志，因为这件事比较敏感，我们将来出了问题好追溯，所以打个日志音符是吧？我要对增票格式化，我要死定一点。

说。

卖的说开始d百分号第几次，这是一个数字，需要替换第几次上传，然后后面给个分号，百分号s这是需要一个字符串替替换，好，就等于这么一句话，我需要后面需要替换。

第一个百分号d需要用 Upload times替换，而且 upload times还得加一对 upload times字正一好。

然后第二个是用 file name替换，文件名替换，记完日志以后，下面我就要正式传了，传的时候也是你要生成凭证然后才能传，而生成凭证又需要设置响应信息，又需要文件名，但文件名其实已经有了，所以我就不用生成文件名了，我就直接设置响应信息，还需要使劲map，policy等于new使劲，然后的话 put return包。

点。

返回的数据的内容，数据内容也是个Jason community YouTube点get杰森死卷零，我希望在成功的时候还是给我返回，扣的0就可以了。

好，然后有了这个信息以后，我要生成上传凭证，生成的方式和我们刚才那里写的是一样的，我再写一遍，create也需要存入两个key对吧？然后分别是access。

security key。然后。

凭证是一个字符串，upload都肯等于它点upload。

Ok那么第一个参数是上传的空间的名字，sell back的name，第二个是文件名，非常累，第三个是过期时间，凭证的过期时间3600秒一小时，然后是响应信息完了。

好，然后我们要传这个时候我们这里要和表单不一样，表单我们是在GS里设置一个上传的地址对吧？这里我们需要通过Java程序指定上传地址，或者说指定上传的机房，要指定上传的机房。你可以写它的域名也可以，它里边有一个静态的工具可以直接得到华北的机房，我写一下就是阿布露的我们需要用到阿布露的manager，在服务端的时候，manager等于new upload manager。

然后这两项利用一个consideration，新能源的配置配置里存的是机房的信息，它有一个类要用点一用一，我只能华北机房，然后的话我要开始上传了，上传的时候我们需要对上传的代码处理异常 try catch，k是到的是七七牛的XP，我先把异常处理一下，如果说我捕获到了异常怎么办？

我就记个日志，然后的话我们就等着一会还会再调一次，我们再来一次，我们可以就给他一个机会，万一说网络不好或者怎么样的，我们传世之外的对吧？

有这种情况我们就再来一次，但是我们一共三次机会，三次你都传不上去就算了，这个意思，但我这里要记个日志，log点因数。

再一个我们记个日还有一个好处就是6776云它有一个政策，就是说如果说你的上传的成功率低于百分之多少的话，他会给你退钱，低于99%退多少钱，低于95%退多少钱，它有一个可能性的保障。

所以你记个日志，然后的话一旦说你发现这个错误率很高，你去找他退钱，这也是一个有必要的的一个手段。好，那么log点infer好，这里我还是要格式化字符串，Spin点format说第多少次上传失败，后面再给一个，再加上 final name。

好了，然后我后面需要替换，用 up low的 Test进行替换，用这个范围内替换那个字符串替换这两个参数，然后对异常处理了，这个地方也得再处理一下，你看如果说我们发现这个文件存在，我们就做这样的处理，它不存在怎么办呢？不存在当然我们进行下一轮的下次调用到这里我也记个日志，我们把日记得完善一点，因为这个地方一旦出了问题，我们好通过日志去追踪，这我记得是说等待图片生成，因为你不存在就意味着图片没生成，我就等着我就记个日志对吧？

三个内容。好，这里我也是打了个日志，好，日志记完了以后，接下来我就处理端里边的内容，我就开始上传图片了。

因为。

凭证已经构建好了，上传图片，它返回一个响应信息叫 response，返回的是response，然后我们调用Manager，然后点 put方法，那么这个方法它可以帮我们去传图片，但是你需要把这个参数传给他。

首先我传的是pass我本地的文件的路径，然后是file，name，文件名，然后是upload头啃，这是上传的凭证。

然后下一个参数通常是now，然后的话下一个是一内置，然后讲一妹子加上后缀上传的文件类型，最后一个给个force。

好，它这个参数有这么多参数，最后一个检查什么东西我并没有细看，反正一般都是force。这里是传的时候需要携带什么额外的参数吗？没有，大概就是这个意思。这是我们从它官方手册上找到的，它的一个代码的格式就是这样的。然后这句话执行完以后，我就处理响应结果。

好，那么响应的结果，因为我们之前给的要求是一个Jason对吧？所以我得到的是一个Jason的数据，我就但是我得到的从这个数据得到的直接得到的是一个字符串，我需要把它转成Jason，我就这样处理。

Jason object，然后点pass object，然后从response里获取数据，点get不是点get点到。

点4卷。

我就得到了返回的阶层字符串，把它转为阶层对象，我得到的是一个Jason和Jack。



好，然后我要判断一下 Jason 他有没有值，如果说 Jason 等于没有值，或者说 Jason get 扣的等于 no 没有扣的，或者是 Jason 点 get 扣的，点 two string，点 echoes，0，我加一个感叹号表示不是这样的。

好，这是什么意思也很明白了，就是说如果说我返回的数据，要么是空，要么里面扣的没有，要么扣的值不对，我就认为不对，因为我认为正确的时候一定是有扣的有 0 的对吧？这不对怎么办？记个日志，错误日志这个方式和这个是一样的，就是说第几次上传失败，然后失败的是哪个文件，完了 Else，否则的话怎么样，否则就是成功了对吧了。

第二因为什么？给个记录说有点格式化，死定要卖的说 d 几次上传成功，这些后面是 s 我这写错了，因为第二个是字符串，好，然后 Up load times family。我这儿又记录了第几次传的成功了，在成功的时候这个任务就可以结束了，对吧？我就 filter 点看守书，任务结束到此为止。

好，总而言之，你看整个的逻辑就是说如果时间过长，强制结束，如果失败超过三次强制结束，那么还有成功的时候结束，那么其他的情况没超时，失败次数没达到，我就再运行一次，争取让它成功了这样的。

好了，这个逻辑我就完成了，总之这个代码其实并不长，逻辑也并不复杂，但是主要就是考虑的比较多，一定要重点考虑说这个任务他不完成的时候我怎么办，主要是解决这么一个问题，有点麻烦。

再一个为了能够追溯这个过程，所以说我们到处记日志记得也比较繁琐，主要是麻烦在这儿了，好了，那么完成以后咱们试一下，我把这个项目重新启动一下好了，启动完以后我打开浏览器，我把缩小一点，这样我能够直接的看到控制台提的提示，这样我扫那些首页，现在刷新一下，好，然后我要分享，因为这个功能是通过分享来触发的，对吧？

是。然后 html 等于然后 code.com 就它了我不改了，回车看控制台，等待图片成功，第一次失败，第二次上传成功，完了。

好，这样空间里有没有看一下？内容管理，有了点来看就有这么个图片，那么如果图片越大的话，其实成功率就是失败的概率是越高的。

那么小一点的话成功率会高一点，这可能是跟网络有关系，或者是也可能是跟我们免费的账号有免费的使用有关系，所以说他给我们分的机房可能就是没有那么性能那么好，也有这样的可能，反正是他有经常能看到一些失败。

好，我再来一下再刷新一下，刷新成功了，再来第二次成功了，如果换个网站，baidu.com 成功了，再来成功了，你看百度成功率就高，因为百度的首页它内容少，牛客咱们首页的内容多，我们实际在分享的时候，我们分享的页面一般模板都比较简单，所以说成率还是比较高的，这样好。

当然了，如果说你想在商用上让它云服务器更稳定，你可以选择

比如说阿里云选择付费的这种服务性能就会更好一点，更有保障一点。好，这次课我们就把上传到云服务器的这种方式给大家做了演示，这次课我们就进行到这里，咱们下次课再见。

## 7.10 优化网站的性能

---

52-优化网站的性能.mp4

这次课我们来学习如何优化网站的性能。那么优化网站的性能其实有一个非常行之有效的手段，就是加缓存，那么但是加缓存它是有一定的策略，有一些原则的，也有一些方式方法，我们需要对这个如何加缓存有一个系统的了解。

再一个我们之前也加过缓存，加完缓存以后我们是想象着说它的效率会提高，但是具体来说它到底有没有提高，它的性能到底提高了多少？我们没有一个明确的这么一种感受，或者是没有一个明确的证据。

所以这次课我们在优化完某一个功能的性能以后，我们用压力测试工具给他测一下，当然我们之前也测一下，这样的话两者做一个对比，你一看就明白说

加上缓存和没加之前，那么它性能差多少？

那么你把这个缓存加上去，并且做了——一个详细的测试，有了一个详细的数据，那么你面试的时候去跟面试官讲这个是有理有据比较让人信服的，而且你自己从这个数据上也能够感受到这个缓存能够给我们带来的一个价值，所以说你以后在开发的时候也就主动愿意去用这个缓存了，你也会主动的多往这方面去考虑，这也是你一个经验的提升。

好了，那么缓存我们在使用的时候有三种，第一种叫本地缓存，我们把缓存数据存到我们的应用服务器上，那么这样的方式性能最好是因为客户端发请求发给的是应用服务器，应用服务器一看我本地就有缓存的数据，我就直接给你返回，对吧？

它最快。

然后把数据缓存在本地的话，常用的缓存工具有 eh catch，还有瓜味儿，还有咖啡因，然后那么一会儿我会给大家演示咖啡因，因为目前咖啡因它是性能最好的一个缓冲工具，我们就用其次是分布式缓存，分布式缓存是把缓存的数据存到一个note和数据库上，那么它的优点就是能够跨服务器，而前者像本地缓存就是有一些数据是不适合存在本地缓存的，你比如说登录凭证，因为这个数据它和用户有直接的关联，那么如果说用户访问 a服务器在分布式这个系统里，他访问a服务器，得到一个凭证，这个凭证存到了a本地，下次请求它会被打到了b而b上就没有这个缓存，那么可能就影响它登录的状态，所以说这种情况不适合存到本地缓存。

而分布式缓存它能跨服务器，因为我们 ABCD多个服务器都访问同一个 No circle数据库，所以它的优点在于这儿，然后它的性能肯定是比这种my circle数据库要好，但是它的性能比本地缓存略低一点。其实低在哪？低在网络开销上，因为我们请求发给了应用服务器要访问。

No circle的话，它往往是在两台机器上，所以说它之间有网络开销，网络开销主要是有一个影响，所以它比略低一点。

然后常用的缓存的工具具有 Memory开始还有 radius，我们的项目中用的是 radius，而实际上 rise我们怎么用它缓存数据，咱们都用过，我们不是用rise缓存过验证码的头啃对吧？也缓存过user等等，我们缓冲其实好多数据了，这个其实我们已经会了。

另外我们也可以把本地缓存分布式缓存把它们结合在一起，形成一个多级缓存，那么多级缓存通常是这样的，通常我们是有两级，然后当请求到达服务器以后，服务器看一下我本地缓存作为一级缓存有没有，你要的数据有就返回，如果没有的话他就会去访问，峰值缓存也就是二级缓存，看一下你这里有没有缓存数据，有返回，如果没有怎么办？

最终去查数据库。那么如果查到数据库查到数据以后，它会把这个数据存到二级缓存里，存到一级缓存里，最终再把数据返回给你，在这次请求当中数据库得到了数据更新了缓存，返回的数据给你，是这么一个流程。

好，总之这种做法是尽一切可能尽最大的可能就是避免你直接访问数据库，或者说我们这种方式，因为我们在DB之前加了两级缓存，它能够避免缓存雪崩，什么叫缓存雪崩？就是缓存失效。如果你只有一级缓存，比如说缓存在radius上，万一 Radius它瘫痪了挂了是吧？这一下子所有的请求都会到达DB，可能就一下就把DB给弄瘫了，是吧？或者说如果你所有的缓存都在万一本地缓存，由于某些原因不行了失效了，所以说也有可能请求直接就到达了 Db而到达DB大量请求到达DB，对DB是一个有毁灭性的影响，所以说这个是要尽量避免的。

我们有了两级缓存以后可能一级缓存挂了，还有二级缓存，二级缓存挂了还有一级缓存对吧？两个缓存同时挂掉的概率就比较小，所以说那么这样的话就能够提高我们系统的可能性。

好，这是我们对缓存大致做一个了解做一个介绍，然后我在写代码之前我想在介绍的更直观一点，因为这个是从文字从口头说明的时候，可能有的地方你就是理解的不是很到位，所以我画个图再给大家再说一说，很有必要说得更清楚一点。

好，我们说的是我们将来这个系统它的就是最终的部署，现在的网站都是分布式部署，都是多台服务器组成集群，那么我也不画那么多服务器，我就画两个服务器，一好服务器里头我们有的是什么是，有的是我们的一些组件，当然我们的核心其实是 service 是业务总监，我们最终的 control 是要叫 service，当然 service 会去访问数据库，这里我画的是 service 表示我们要处理的业务，它部署在了我们的服务器里，或者说这样我画成 APP 就是我们的应用部署在了这个服务器里，好。

然后那么一级缓存服务器的缓存就是服务器的一个内存空间，这个就是本地缓存，然后这是我第一台服务器，那么我们实际部署的时候可以有多台，比如说我再换一台，这是第二台，这两台服务器上它的代码 APP 一样的，然后它的缓存都是占有它各自的内存也是就类似的。

然后我们的请求，比如说用户第一个请求恰好被分发到了服务器一，他访的是这个服务器一的 APP 就应用，然后应用要应用缓存它怎么办？它会去缓存里找这个数据，然后看一下这个数据有没有，有就给你直接返回就不用去访问数据库了，这样效率就高，我现在的假设它没有没有怎么办？但是我现在假设只有这一级缓存的情况，我们就换个图先说一下本地缓存它的就是工作的场景。

好，假设我暂时先没有二级缓存，没有分布式缓存，所以说当你从本地缓存中取不到数据的时候，你就直接需要访问 DB 了。所以这个时候服务器就会去访问 DB，然后得到数据以后，它会把数据更新到本地缓存，就有一个更新的过程，然后这个数据就被返回给客户端。那么刚才我说了本地缓存的局限性就在于如果说本地缓存的时候用户息息相关的数据，那么你看刚才这不是第一次请求吗？如果第二次请求这个请求被分发给了第二台服务器，这个就糟糕了。

因为此时你缓存的是和用户有关的数据，可能是关乎他是否登陆的一种状态，这个时候 App 会从缓存中取数取数据，然后取不到他可能就认为你没登录，比如说我们存的是缓存凭证，这个时候他可能就不会访问这个 DB 了，他就认为你没有登录，根本就是 word，所以说这样的话如果是和用户息息相关的数据缓存到本地缓存，那么会不合时宜，但是如果是其他的数据，比如说我们缓存热门的帖子倒可以，我们第一次请求访问 APP 这个应用，然后它没有数据访问数据库，然后更新缓存，第二个请求不管是谁访问 APP 应用，它也是同样的道理，它也是访问数据库，然后根据缓存，你这样的话这两次请求完成，完成以后后续的请求，那么无论是你打到服务器一还是打到服务器二上，这个缓存里就都有数据了，整体性能就被提高了，只要数据和用户不是有直接的强关联，那么我们在每个服务器上都缓存一份，也可以也是 ok 的。

那么 readiness 缓存它的好处就是刚刚能够解决刚才我说的那种和用户相关联的那种问题。

我们在琢磨一下，比如说我第一次请求访问服务器一他看本地缓存没有数据，如果说我们这时候用的不是本地缓存，而是 ready start，那么假设我们用的是 ready，比如说应用看 release 里有没有数据，这里得改个名字，一看没有怎么办，他就访问 DB，访完 DB 以后他需要把这个数据同步传到 radios 里，然后把数据返回给客户端。

如果用户下回请求被打到了第二台服务器上，第二台服务器它也是我先看 release 里缓存里有没有这个数据，一看有那就直接从这里取到这个数据，返回就完了，就避免了访问 DB，所以说这个 readiness 它可以跨服务器，ready 这里你是缓存什么样的数据都可以，它就没有局限性。

所以说我是希望通过画图的方式给你直观的让你感受一下就是本地缓存和 reds 这种分布式缓存它的区别在于哪里。

另外你也能看得出来本地缓存的效率要比 reds 要高，因为本地缓存应用直接去访问，在同一台机器上对吧？这个速度很快没有网络开销，而 ready 我服务器要访问它，两台服务器中间有网络开销是吧？

所以说各有优劣，但总体来说 ready 是适用的程度更广，本地缓存有些地方不能用，你要注意好了，然后再画一个图，再画图说什么说一下。

我们整个缓存在使用的时候详细的使用的过程，假设我部署的是两级缓存，本地缓存和 release 我都用这个过程是怎么样的，我这里我就不去花两台服务器了，我就是用一台服务器来说也能说得清楚。

好这是我的服务器，然后这里面部署的是应用要写上APP，然后当然我的服务器有一部分的内存空间是被划分做成了，缓存的空间，它就是本地缓存，也就是那么当请求抵达这个服务器访问应用的时候，那么应用首先它不是上来就去访问滴滴，它是先去看本地缓存里有没有你想要的数据，如果有就直接返回给你，这是第一步。先看一级缓存，本地缓存里有没有你要的数据就返回，如果没有怎么办？

再看二级缓存，二级缓存呢就是redis，当然我们的应用里是redis，你也可以用别的，所以如果说第一次看本地缓存没有数据，就真正的看redis里有没有这个数据，然后如果有数据就给你返回，如果没有数据怎么办？假设此时恰好就没有，这是第二次进入，第二次查看缓存没有，这个时候就需要访问DB了。

好，所以这个时候他要访问DB，访问DB肯定是有你要的数据的，这是第三次访问，访问到数据以后，这个时候它需要对数据进行同步，同步到缓存里，这样的话下次缓存里还有数据好能用，所以这个时候就第三次访问以后，应用会把数据同步到二级缓存里，进而再同步到一级缓存里，然后再给你返回，是这么一个过程。

所以当你下次再去访问这个服务器的应用的时候，那么本地缓存就有数据了。

当然了我们通常的缓存会设置一个自动的过期时间，万一说本地缓存过期了，或者说我们这个缓存一个是要设置自动过期时间，还有一个还会设置它的总的大小，本地缓存一般的大小会比较小，因为它要占有服务器的内存，服务器还要运行应用，它分不出太多的内存给你，它都空间总的空间比较小，能存的缓存的数量比较少，所以有可能你这个数据被其他的数据给挤出去了，有可能访问的时候一级缓存里没有数据，而你访问redis里面redis里有，所以这就是两级缓存的意义了，从这方面也可以去理解它的风险小，而它的空间很大，能存很多缓存的数据。

所以当你下次再访问的时候，看一看本地缓存有没有有就返回，如果没有这份 readiness，通常这两个一般就会有，万一实在没有再访一次数据库，但有了这两个缓存以后，访问数据库的概率就会大大的减少，那么整个的服务器的处理能力就会大大的提高，速度就会提高很多的。

好了，这是我对缓存做一个大概的介绍，然后刚才也提到了缓存它都有一些这个基于大小，基于时间淘汰数据的这么一个策略，你要设置缓存里能存多少个数据，超过这个数据就要淘汰一些，有的是按照先进来的数据先淘汰，有的时候是按照他用的使用率，他使用的不频繁，就淘汰缓存，它有底层的一些淘汰策略，然后再一个也是有基于时间的淘汰，不管你使用的频不频繁，反正我比如说到了10分钟我就一定要淘汰你，也有这样的一个情况，这样的话好腾出空间给其他的这个数据有一个公平竞争的机会，这样。

好了，大概了解了缓存的这些个原则以后，下面我们就要通过缓存来解决我们项目中的某一个功能的问题，提升它的性能，并且我们通过工具去测试，看一下提升多少，咱们有个直观的了解。

我们的这次课的目标是什么？是要优化热门的帖子列表，我们都游客社区的首页是个帖子列表对吧？它默认的话是按照这个帖子的时间的倒序来排的，但我们还有一个按照这个热门程度来排，我们对热门程度的列表数据进行一个缓存，你提高它的性能。

有人说为什么你不是对默认的时间倒序的，按照时间排序的帖子里边去设置缓存，因为不合适，什么样的数据适合缓存，一定是数据变化的频率相对较低，数据变化的不那么频繁，而我们的默认的首页按照时间倒序排，经常有人发布新帖子，经常有人去评论去改这些帖子相关的数据对吧？

所以说会导致第一页的数据变化的比较频繁。

所以你缓存的话缓存老变，你经常要去更新缓存反而效率会低的。而按照热门程度来排序，我们热门程度上算分的，分我们是过一段时间算一回的是吧？所以说它是一——隔一段时间更新一下，它能够保持一个比较长的时间几乎不变，所以我们可以把数据缓存到内存当中，所以基于这样一个原因，所以我要修改要提升功能的性能，大家注意，好这是我们的这一次课的目标了。

然后我刚才也说了，我们要用什么来实现呢？那么本地缓存咖啡因可以用本地缓存，因为我缓存的是帖子，谁都能看，你登不登陆都能看，和用户其实没什么关系，所以我们就是用本地缓存咖啡因来做。

至于分布式缓存我就不演示了，因为我们如何把数据缓存到redis，大家都会了，我们之前很多功能里都演示过，这里我就不去做了，但是我写程序的时候会告诉你，你看在这个地方我们可以先查一下release，然后如果没有再查买色口这样就实现了。

二级缓存我会告诉你那个位置你可以自己去做一个尝试。

好，然后我们使用咖啡因实现本地缓存，可以单独用，也可以用spring去整合，其实这些这几个缓存都可以用spring整合，但是我不建议咱们用spring整合缓存的工具，为什么？

因为spring整合缓存它是用一个 catch manager一个缓存的管理器去管理所有的缓存，我们程序中可能有很多缓存，a缓存缓存帖子b缓存缓存，其他的数据有很多缓存，那是不愿意用一个缓存管理器管理所有的缓存，然后的话把它的过期时间缓存空间大小是统一进行设置，这个不太合适，因为每个缓存缓存的业务数据的不同，它的要存多少个数据，它的强制数据淘汰的时间应该有所差别，不可能是一刀切的。

当然我们用spring能不能就是说每个缓存给它单独一套设置也可以，你需要多个缓存管理器，那样反而麻烦了。既然他麻烦了，我们还不如就直单独使用这个工具就算了，所以我就不用死人去整合了。那么关于咖啡因，他的手册你需要去get up上去找你在 Get up，然后去搜咖啡因，搜到的结果就是这样的。

第一个点进去以后，它后面会说你要看手册，点链接，好，这是他详细的手册在这里。

看好这是我们手册从这里找，然后的话我们要使用它的包，我们还是去妹妹repost那边去搜，比较方便。

我搜一下搜咖啡因，好，那么第一个这就是这里少说了一点，其实之前很长一段时间瓜娃很流行，那么现在是咖啡因更流行，但其实它俩使用方式几乎一样，咖啡因就是仿照着刮过去做的，但是他想到最新的算法，那么它的效率要优于优比它提升很多，所以现在我们就使用第一个我点进去，选择一个新的版本把它拷过来，粘贴到我们的配置文件里，很快就下载好了，下载好以后包有了包有以后怎么办？

接下来我们还需要设置一下参数，这个参数是我们自定义的参数，并不是说咖啡因强制要求的，为什么要自定义这个参数？

因为我们在缓存帖子列表的时候，我们要声明能缓存多少个数据，我这个缓存空间能存多少个数据，然后多长时间要把这个数据释放掉，好让新的数据进来，还有一个更新的自动更新的这么一个机制，这个参数最好做成可配的，因为我们现在在本机是这样的，我们将来可能部署到这个服务器上服务器性能好的时候，我们可以把这个数调大一点对吧？

所以最好是能可配的，我就打开 practice配置文件配一下，配置是与咖啡因有关的，也就叫咖啡因，然后的话这是我自己取一个key咖啡因点，pose。为什么叫pose字？帖子我们要缓存的是帖子列表，是pose是缓存的名字，如果你后面还要缓存其他的数据，比如说你要缓存评论你可以叫咖啡因点，comment or posts点。

max size。

等于多少？

就是说缓存空间里一共能存多少个对象，我写的少15个，有人说这也太少了，你要知道我们缓存列表，我们是缓存列表是缓存一页一页的数据，因为用户看的话看的是第一页第二页第三页，我们每次缓存缓存是第一页第二页第三页的数据，你有必要缓存那么多页吗？对吧？

因为你可以想象得到我们即便是热门的帖子，大家也不可能看到几十页之后去，你可能看个两三百页就差不多了，当然最频繁的肯定是第一页，对吧？能够缓存15页基本上就够用了，咱们没有必要浪费那么多内存空间，所以我就缓存15个数据，但缓存15页它未必就是前15页，那没准说有的人就前5页看的人多，然后最后也未页看的人多，末页反而被混合进去了，不是15对吧？

也有这样的，可能总之就是混合15个数据。好，然后再来咖啡因点，posts。Xp。Seconds。就是过期的时间，单位是秒，我写成180秒，三分钟就是这个数据不管怎么样到三分钟，就是你存到这个缓存里的数据到三分钟自动就会被清理掉。

然后当你下次再访的时候，它会从数据库里又查一遍，然后再更新到缓存里头去。为什么要这么做？因为其实缓存它的数据的淘汰有两种机制，一种是主动的淘汰，比如说我这个帖子数据发生变化的时候，我把这个缓存清掉，当你在缓存的时候就重新加载了。

再一个就是一种自动淘汰定时的，而我们有的一般我们很多的功能可以两个都用，数据变的时候我也淘汰，然后定时也有个淘汰，我们只有定时淘汰，为什么不主动淘汰？因为我缓存的是一页数据，如果里边某一个帖子数据稍微发生变化，我把整个这一页都给刷了，这个挺麻烦的，对吧？

不太合适，所以说我们就没有这样的一个淘汰机制，你这一页某一个帖子，比如说他被做了一定的修改，多点了个赞，多了个评论，我不去更新缓存，然后的话到三分钟刷新一下，所以可能是你看到的这一页的帖子，它里边的点赞的数量一些数量可能会有一定的延迟，但这个并不影响使用，没有关系好了，这个参数也就设置好了，设置好以后我再去优化我们查询的方法，我们优化什么方法，通常都是优化 Service，业务方法，因为你无论前端访问哪个controller，最终你要查这个帖子肯定是调 service的对应的方法去查对吧？

我只要把service优化了，你去调性能就高了。

所以我们去优化 Service，这个service就是 discuss。Post service好，首先我先初始化一个logo，因为我在一会儿写代码的时候要用到 logo，记一些日志。好，另外把刚才我上面的那两个参数注入进来肯定是要用的，对吧？第一个就是max。

size。

缓存能存多少个数据最大的数量。好，第二个是x片second过期时间，我把这两个key拷进来，好。

参数有了，有了以后，我们现在是要使用咖啡因去缓存帖子列表，同时我也把移动的总行数也缓存一下，这个也可以缓存，因为我们每次去查询帖子列表都会调这个方法调的比较频繁，我们也可以把它缓存一下。

为什么可以缓存？因为总的行数它影响的是我们算的总页数，我们利用它算总的页数，如果说你去访问这个帖子列表，总的页数稍微有点延迟，在你看来少了一页，其实问题不大。

再一个其实你看牛客网，总的页数它运营一段时间以后帖子特别多，页数特别多，那么总的页数它最多限制到100，你永远看都是100末页，其实还有有更多的不让你看了，因此其实总的行数我们做一个缓存也是ok的，然后用咖啡因做缓存，咖啡因的核心的组件接口叫catch，就核心接口叫catch，就代表一个缓存，然后的话它有两个子接口，有好几个子接口，有两个比较常用的，一个叫做loading。

Catch，还有一个 A sack loading test，那么其中 loading test比较常用，它是同步缓存，比如说我有多线程，同时访缓存里的数据同一份数据，那么它会但是缓存里数据没有怎么办，他会让这些人都等着他去，数据库取完以后再给你们返回，他是让你们去排队等着的，是这样的。

然后三个录音开始它是义务的，可以怎么说？可以支持并发的同时取数据，但是我们希望是这样的，我把这个数据取好以后，再去给你们返回，你先不要着急做并发的取，所以一般我们会用 Loading开始。

好，那么要用 loading开始，我需要利用它把两个缓存声明好，一个缓存是缓存帖子列表，一个是缓存这个总的行数，其实我们需要两个环节能够把它先声明出来，然后再做初始化。

好，第一个是帖子列表的缓存声明一下，类型就是loading。

Catch。

t是string值list。Discuss。Post是一个列表的集合，那么这个名字我叫post list。

当然。

这个开始需要初始化才能用。好，然后另外这里说一下为什么 Loading开始需要两个值，一个是 key 一个是 value，其实所有的缓存它都一样，都是按照key缓存value，都是按照k缓存外流，这是一个k为什么是个死病？

咱们一会再详细再说。好再来接下来我还要说明一个缓存是缓存帖子的总数，帖子总数是吧？也是楼顶开启。Key我用inter，value也是inter，然后我就要post rose。

catch。

好，这个缓存我们什么时候初始化，它不需要初始化多次，在我们服务启动或者你首次调 service的时候，初始化一次就够了，我可以这样，我可以给当前的类增加一个初始化方法，在初始化方法里它唯一的调用一次的时候，我去初始化这两个缓存，post开始转，在这里我要分别要初始化帖子列表缓存，另外还要初始化帖子整数的缓存，把它们一次初始化了。

好，具体怎么初始化咱们一会再写详细的内容。我后面先告诉你怎么去调，这样你好对 key有一个了解，比如在这里怎么去调。首先我们说一下这个方法，因为是多种场景都会调，首先我们访问首页，按照这个时间的方式来默认方式来排序，那么回调其次我们按照热门帖子的热门的方式来排序也会掉。

再一个用户有一个查看自己发布的帖子的功能也会掉，这个方法多多个地方掉，但我只想缓存的是热门帖子，也就是说好的mood等于一的时候它不等于一的时候我不缓存，再一个我是缓存首页的，首页的首访问首页的时候，uz ID是不传的，uz ID是0的对吧？

所以当uzid等于0，当它等于1的时候，这个时候我才缓存，而我缓存的是一页数据，而这一页和谁有关？是和这两个条件唯一确定一页，所以说t应该是什么？应该是它俩的组合对吧？我怎么组合其实怎么组合都行，把它俩加起来，中间隔个逗号，隔个冒号都行是吧？我就隔一个冒号就以它俩为k因为它俩的标识一页，而和另外的两个参数无关，甚至我还要对它做限制好，所以我做个判断，User ID等于0，并且order mode等于一，这个时候我才去启用缓存，否则我就不用我启用缓存怎么用，我就直接return post list，catch点get，然后再把key传进去。

Key刚才不是说了 offset加上一个冒号加上厘米对吧？以此为k就行了，我直接从缓存里返回结果。如果是这种情况下，否则我就还是该访问数据库还是访问数据库，我就不启用缓存了。

当我访问数据库的时候，我记个日志 log debug咱们debug一下漏的post历史，让 DB当访问数据库的时候，我这记个日志说从数据库查数据了，这样直观一点，这样我们一会写测试案例的时候，能够通过这句话能看到我这数据到底是访问数据库取的还是从缓存取的，对吧？如果有这句话从数据库取的，没有从缓存取的。好，这就完了。

再看第二个方法，这个方法也是因为我是缓存的，是我要缓存的是帖子列表总行数对吧？其实我这个方法主要是针对帖子列表来用的，用户查看自己的帖子的时候，他传的用户ID进来，这个时候我是不走缓存的。

好我这样我就判断一下，如果优质ID等于0的时候，这个时候是首页查询我要缓存种的数量。Are you return? Post? Rope. Catch the gate。写一个t其实我们当前就没有，其实不用tea，不用什么条件，因为和use ID没什么关系，他不就在地一定是0，我才去走这个缓存，但它一定要求有一个t怎么办？我就干脆你就是ID作为t它T永远都是0，那也没关系，然后也是记个日志 log对吧？

然后这里我写漏的post，rose不让DB好，调用的时候是这样，通过调用的时候我们说说明白了 k到底是怎么回事。

再一个我们是这个方法并不是说所有场景都缓存，有限的场景它是走缓存的，好，剩下我就是要对这个缓存进行一个初始化，那么初始化的方式也是固定的，你去咖啡因的官网它有这个实例，我就直接给你写出来了。

Post list cash等于咖啡因，这是一个相当于是个工具类点，new builder这个builder用来构建 new builder点，max，maximum size。

意思是。

你这个缓存你最大的数据量是多少？就是参数max对吧？然后再点 XP after, right? 当你把缓存写入到缓存空间里头去以后，多长时间它自动过期还是参数？Spr seconds, 然后它的单位是什么 Time? Unit 点second单位是秒，然后点就设置好这两参数以后，这build点调用build的方法，build的方法之后就是让参数生效，同时返回 loading开始对象，build返回loading开始对象，而build需要传一个参进来，这个参数是一个匿名的是一个接口，我们可以通过匿名实现，这个接口需要叫做开始楼的，我匿名实现 new。

catch load好。

它自动的把这个方法就给我加上来了。

New catch loader, 当然 key你看key是string, y6是集合，其实和 k是保持一致。接口是干嘛的就是说当我尝试从缓存里取数据的时候，它会咖啡因会看一下这个缓存里有没有数据，有我给你返回，如果没有怎么办？没有的话他得知道我怎么去查这个数据，然后把它装到缓存里，所以你要给他提供一个查询数据库，得到初始化数据的办法，漏的方法就是实现查询的这么一个一个办法。

一个方法在这里，你要实现我怎么去访问数据库查数据，或者是这个数缓存的数据的来源是怎么来的，你要在这里解决。好了，然后它参数就是t把它改名叫k就是我们常说的k你要返回 K对应的值。

好，我先判断一下，如果说k等于now或者说 key点less等于0，就是说key是空的，我觉得抛异常了，就不允许为空，就直接说参数错误再来，然后如果不为空，我要解析这个key不是两个数字拼在一起的，我就split一下对吧？

P点 speed用冒号来切割，切割完以后得到一个string字符串，这是我们要的参数，然后你也判断一下，万一说切完之后不是你想要的，parents等于 nor或者是tyrants点lance，它不等于二，我一定是两切切出来两个值，如果不是两个值就不对，这个时候我还是抛异常。

好，就一定这个参数是对，我才能查到我想要的数据对吧？然后凯尔姆斯里两个值它应该是整数，我把它转型一下。第一个是off site等于integer管理office, 0同理 Limit interval of parents e那么有了这两参数以后，我就可以在这委托我就可以查了，有这两条路我就可以查查我想要的数据了，调什么查调迈克尔。This cast post my program select, 然后 uziuzid是0，然后这个是off site，然后是limit，然后是1为什么第一个参数和第四个参数是固定的？

因为我们只这个缓存只适用于这一个场景，其他场景我不管，这两值此时就是固定的就可以了。

然后在这里我告诉你就这个地方我们在访问数据库之前，你可以自己加一个二级缓存，你可以访问 release, 如果rise没有你再访问my circle, 你这块的逻辑我是直接访问数据库了，你可以把它改成这样子，我先访问release看一下有没有数据，如果没有再访问my circle, 这样的话又多了一级缓存，这个就比较容易，反正就缓存到red就可以了。好了，然后这里又访了数据库，我再打一个日志，只要访了数据库我就记个日志，好这就可以了。

同理我们在初始化第二个缓存初始化的方式和刚才一样的，post rose点6x等于咖啡因点new build, 然后点max最大的多少个数据，我就和刚才第一个可能保持一致了，我还是利用这两参数就是缓存15个数据，然后的话180秒，其实这个也够了，我们缓存一个表里的总的多少行数，其实我们1个就够，但15个也无所谓，他也用不上，15个就15个，反正用不完够用就行，这块我写成max, 然后再点。

This is pair after, right? 然后 x pair seconds, 然后 time unit there are seconds。

最后点build。同理里边也是有一个匿名的实现，new catch。low点，那么这个事情还叫key, 好这里边这个就比较简单了，我们也不用做判断，反正这是一个整数，我们直接就利用它来查就可以了，我就return this cast post map。对 c let rose把k传进来。可以了。然后在查询之前记个日志，好这就完成了。

那么完成以后我们要看一下这个缓存有没有效，我们调取这个方法，它会不会就让缓存能用起来，我们写一个测试类来试一下，其实测试类我在上这节课之前已经建好了，叫咖啡因test, 为啥我提前建这个测试类，我主要是在这个测试类中写了一个测试方法in it data for test。



我想给这个测试初始化一些数据，因为一会我要做压力测试，压力测试的话你最好数据库里数据多一点，然后让访问数据库的性能差一点，这样的话我们好和访问缓存做一个对比，如果你数据特别少，就一两百条数据，其实访问数据库也很快的，突出不了缓存的价值。

当数据越来越多的时候，在数据库里堆积越来越多的时候，访问数据库慢了，缓存价值就越来越大，所以我把它初始化了30万条数据，我就是循环了30万次，然后的话每一次我就另一个post，数据都是一样的，然后插进去，这已经执行完了，他需要跑好一阵，因为30万的数据也不少，然后你看一下我的表里执行现在是30万零一百零一百五十九条数据还好。

好，那已经初始化好了，初始化好以后不用管了，我不跑它了。跑不起，然后的话我在写一个测试方法来测一下这个缓存叫test。

Catch。

怎么测非常简单，我就直接查询调用post service点find this cast pose我就只测这一个方法，这方法没问题，Find rose有没有问题？

好，然后传零0 10—就是不关注用户，然后是按照热门排序，然后0代表着第一页10条数据对吧？我一共查三次，但我们能猜到第一次缓存里没有数据会访问一次数据库，访问完以后缓存里就有了对吧？这个时候这次访问就能够取缓存中的数据，这次访问又能取缓存中的数据，只访问一次数据库，我们只打印一个日志，所以三次访问只打印一个日志那就对了。

好再来一个第四个我这样我就写一个0，你看这个是查询这个帖子列表，但是我是按照默认的方式排序，这个是不走缓存的，它也会访问数据库，我们看一下是不是我们缓存对特定场景是生效，对这个场景是无效这个意思，写完以后我就来执行一下，看一下看输出结果，主要是看日志，你看漏的一开始漏的post from DB访问了一次数据库，然后后面还打印出了对应的circle，然后输出了第一个列表数据，它是一个数组，它打印的是一个数组形态，然后第二次输入第三次输出直接输出值了，就没有漏的，说明没有访问数据库，这个数据从哪来，一定是从缓存里来对吧？

它不可能凭空造出来。然后最后一个输出一条数据输出之前你再看是不是又漏了一次，然后的话又有一个circle，所以跟我们预期的一样，说明这个缓存是有效的。

另外查行数逻辑是一样的，我就不测了。当然了我一会压力测试会测，现在就压力测试，好，那么当然我压力测试之前我还得把这个缓存去掉，我先不加缓存去测一下，然后加缓存再测一下，我先把它去掉。初始化缓存的代码好不容易写的不用它把这几行除掉就可以了，就不走缓存直接查数据库对吧？这也是不走缓存，直接查数据库，再有一个我提前还把Aspect除掉了，因为要不然每次访问service都打一个日志，我们压力测试有频繁的访问特别烦，我得先提前的把这俩注解注掉，你注意一下。

好，那么压力测试我们需要找一个好的工具来测，咱们不能靠眼睛去观察，或者是我们自己写代码比较麻烦，有好用的工具。那么我选择的工具是解麦特，其实还不止一个工具能搞定，我用的是解麦特，这是阿帕奇提供的一个软件，那么在哪下载？在这个网站上叫解麦，特点阿帕奇点org然后下载的话是点这当漏的，有历史，然后在这里边他有一些版本，那么我下载的是Zip，压缩包已经下载好了看一下。

在这这个工具很简单，只要把它就是解压缩到哪，然后直接就能用，也不用做什么特殊的配置。好，我就解压缩一下。D盘我去d盘work看一下就在这，然后要用我们就进入b目录，然后的话双击解麦特点bat就可以启动。

当然启动以后这样我需要先把这个服务启动起来，因为你压力测试是利用那个工具模拟客户端去访问这个服务器，然后看一下它的性能，而且是模拟很多人同时访问的是吧？

所以你得先把它起来，我先把这个服务启动一下，好，启动以后我再回到解麦特，为了大家同时能看到解麦特又能看到跑的时候又能看到控制台，我把它弄得扁一点，这就够了。

好，然后的话我们不能上来就运行，我们还需要给它一就做一个测试计划，然后做之前我先点这儿点，选择语言，选择中文简体，这样看起来舒服一点，好。

我就建一个计划添加，这个软件的功能很强大，它能做各种各样的测试，我这只是一个比较基本的简单一点，添加线程组，因为你要模拟大量的用户访问这个数据库，你肯定是要模拟出多个线程来，我需要创建一个线程组可以给它改个名字，我这里就叫index，community index，社区的首页，然后你要模拟多少个线程数，我这里模拟100个，不要搞得太多，因为你的我们是家用电脑性能没有那么好，100个已经不少了，就100个。

然后什么rap up时间，就是这100个线程是在多长时间之内把它创建好一秒，默认就这样。然后再放大大一点这个地方选择永远我们是持续执行的，持续执行你也不能一直执行，你要执行到什么程度，我们这里选择一个调度器，然后这里写成60秒，反正我一启动以后，他就创建100个线程就在那跑，就在那访问这个服务器，一直访问直到60秒这么一个意思。

好这是线程组，然后在这个线程组织还要创建一些内容，添加取样器，HTTP请求，你要访问谁对吧？你用什么协议访问谁，这里我们选择的是HTTP请求协议填HTTP这个地方的IP地址，你可以写logo house或者是127.0.0.1端口号，8080方法get，因为这是一个查询对吧？路径 Community。Index。注意我们测的是热门帖子对吧？所以 order food等于是吧等于1，然后编码-8utf-8，好，这是对应请求的一个设置。

然后接下来点线程组我们再添加一个定时器，添加一个统一随机定时器，就是说如果你没有定时器的话，你一启动这个就是压力测试，它就不间断的去访问你这个服务器，很快服务器就瘫痪了，它中间没有间隔，我们CPU处理不了，没有间隔，我们需要有一定的间隔，你可以中间有固定的间隔或者是随机的间隔，我这用的是随机的间隔，比较模拟自然的状态，因为自然情况下我们访问一个网站也不可能说是一点间断都没有，中间肯定是有一点间隔是随机的，然后随机多长时间，我这里写的是1000，它是随机从0~1000毫秒之内的一个数，是这样，0~1秒之间的一个随机的数，它中间有这么一个间隔，好，再来点线程组再添加一个监听器点聚合报告，最终测试结果我们通过这个报告能够看到，当然它有很多形式的报告，我们看这个就可以了，而且这么多数据我们不用都看，我们主要是看什么吞吐量吞吐量就是什么，每一秒这个服务器能处理多少个请求，这就很能说明问题。

比如说我没有压力测试没有优化之前，它每秒只能处理10 10个，请求优化完以后它能处理100个就很显然它性能提升了不少10倍甚至更多对吧？好，这就配好了，配好以后我们就可以运行了，我点这个按钮就运行，启动时他问你要不要把这个计划保存一下，yes or no都可以，yes吧？

然后它默认存到 b目录下，我就c保存，保存以后它就开始执行了，这个时候你看控制台他就开始走，每次请求他都访问数据库，他就打印 circle对吧？

然后你看这个就卖头，看这个聚合报告，看这个吞吐量大概就是在789不断往上涨，但是可能涨到10甚至多一点就差不多了。

我们跑完一分钟看一下，应该就是10左右，每秒10左右个请求，这是它的处理能力。

那么这个服务器在处理请求的时候，就是说你们在测的时候也不能把线程数测的太少，如果你线程测的太少的話，那么你的电脑就是一个比较清闲的状态，它没有满负荷，这个时候它的性能没有激发出来，还没有到达它的瓶颈，所以它处理的是比较有游刃有余的。因此

它的吞吐量比较大。

然后当你随着你给它加线程数以后，它的当这个线程数到达了它能处理的一个接近瓶颈的时候，它的性能会下降，最好你能找到性能下降的点，比如说你可以一开始给50个线程，然后80个然后100个，然后你看一下，如果说吞吐量一开始还比较大，后来慢慢反而小了，当你线程越多它反而小的时候，就说明它的性能开始下降了，就说明到达它的瓶颈了，我们就用这个点去测。然后的话我们在没有优化之前记录一个值，你看我一分钟完了以后，他怎么1分07秒，因为到一分钟以后他开始释放线程，让线程释放掉，释放这个线程或者说它是线程走完，执行完花了7秒钟，最终是9.5，不到10的吞吐量，每秒处理大概10个请求这就比较差了。

好了，我们就把刚才的优化的代码放出来，我们再执行一遍，当然你为了准确你可以多执行几遍，因为我们间隔时间是随机的，你多执行两遍，反正也大概就是这个数不会偏差太大，我就执行一遍了，然后把刚才的注释去掉以后重启一下，启用缓存以后我们看一下执行结果会怎样，那么再回到这个工具，那么因为刚才你跑了一下，这里面有一些数据，你要把它清掉再跑，不然的话它会累加的，所以你点 SARS 扫把清一下没了，完了以后还是100个请求对吧？

我没有变，然后的话你看控制台这回他是执行了两个circle以后就不执行了，因为这俩数据都被缓存，你以后再去访问数据库的时候，再去访问这个服务器的时候，它基本上直接就返回这个数据就不用走数据库了，这个时候你看它的吞吐量直接飙到了160多，还在往上涨，而原来都不到10，差了10甚至20倍左右，那么他处理的请求这是请求的数量比原来快得多了多得多。

然后有的同学可能会发现什么呢？这个时候它可能会有一定的异常出现，有可能就是我电脑处理的性能还可以，它可能没出现，有的电脑就是性能比较弱一点，它可能会这异常是表示什么？不是我们服务器报错了，而是说他们看到他处理不了这么多请求，因为你现在处理请求的速度太快，他处理不过来，然后有一些请求就丢掉，丢失了他就没有响应了，是这样的。

或者你一开始执行的时候没有这么多，执行几遍以后就这样了，那说明你 Cpu就已经跑不动了，跑了一会儿你看这回到一秒他释放马上就释放完了。

100个线程不用1分07秒，然后的话最终的是吞吐量的定格在188.9，差了大概十八九倍的样子。

那么经过这样的测试以后，你就很清楚的知道我们优化和不优化加了缓存和不加缓存它有多高的一个性能的差别，无论是你从自身对缓存的认识上，还是你去从面试的角度来说，这都很有好处。

以后我们在工作的时候，

遇到那些性能比较有瓶颈的，访问量比较大的，然后的话这些功能我们都可以用这种方式去解决这个问题。那么一个缓存解决不了的，我们尝试再来个二级缓存，那么可能性就大大的提高，而且性能也会很好。好了，那么这次课我们就演示到这里，咱们下次课再见。

## 第8章 项目发布与总结

---

### 8.1 单元测试

---

53-单元测试.mp4

到现在为止我们已经把外部开发相关的核心的技术讲解完了，然后我们这个项目中的功能已经做得比较完善了。那么这一章我们主要是对整个项目做一个好的收尾，收尾主要体现在两个方面，一个是我们要讲一下我们项目上线时，项目上线前后我们要做什么事情，包括上线之前我们怎么去做单元测试，上线的时候我们怎么去部署，以及上线以后我们怎么去对这个项目进行一个有效的监控。

然后除此以外，我们对整个课程做一个总结，做一些概括，另外我们再讲解一些面试题和技巧，那么这个是我们这一章的主要的目标。那么我们第一节先来讲这个单元测试，就是说我们项目做完了要上线了，那么在此之前我们一定要做好单元测试，让这个项目质量就是有保障。

当然我们不只是在项目上线之前做电源测试，其实我们在平时的开发的时候，每个功能开发完都要做单元测试，这个单元测试是贯穿着我们整个开发与维护的过程当中的，因为我们这个项目是一个spring布的项目，所以说我们利用了spring布的它对单元测试提供的支持，具体来说就是我们依赖了super starter test包，当然这个包里包含的内容比较多，所以它的功能很强大，它包含了g unit以及 spring测试相关的内容，还有断言等等，它包含很多内容。

当然了，我们现在的项里已经把包依赖好了，而且我们之前做过很多次这样的测试，我们已经很熟悉了，已经用过很多次了。

不过还有一些内容需要再介绍一下，我们在完善一下对这个内容的一个认知和使用，我们所写的测试的方法，我们通常叫测试用例，那么我们编写测试用例的时候，这个是一定的要求的。

其实主要的要求除了能够把我们要测的能测出来以外，我们还有很重要的一点就是你要保证这个测试方法它的独立性，也就是说我这个测试方法不能依赖于别的测试方法，可能我们之前做测试的时候，我a方法插入了一条数据，b方法对这个数据做了修改，这两个方法就产生了依赖，这是不合适的，为什么不合适？

因为我a方法插入的数据，b方法去修改这个数据，我们现在当前执行没有问题，过后可能是过了一段时间，你当初插入的数据被删掉了，你平时把它删掉了，删掉以后比如说这个功能在日后有了一些重构的机会，或者需求发生变更，你把它改了，改完以后怎么办？你是不是得再做一次单元测试，对吧？

你再做一次单元测试，最好的方法就是我之前写的代码，我就直接立刻就能运行，看一看结果就完了，而不是再写一套单元测试方法对吧？你这个时候你想你再去运行测试的修改的方法，可能就测不出来东西了，因为它所依赖的数据在之前很长一段时间之前你把它删了，就有这样的问题。

所以你要注意我们单元测试这个方法要保证独立性的原因是它不是一次性的去执行，我们开发完这个功能要执行，以后我这个功能发生变化，我还要去执行，赶紧验证这个变化对整个业务有没有产生负面的影响，那么上线以前我可能还要执行他，要经常执行，所以说你要保证它独立性，随时随地它都能执行，执行下去，而不是依赖于别人的发生变化，对他产生影响，是这样的一个目的。

那么要保证一个测试方法的独立性，我们可以按照这样的步骤来实践，那就很容易就能保障我们在执行这个测试方法之前，我们就初始化数据，就为这次测试单独初始化一份数据，然后我在执行这个测试代码就利用这个数据，然后执行完以后我去验证测试结果对还是不对？有没有问题？

然后验证完以后，最终我再把初始化的数据清理掉，这样的话你下次再执行这个数据又重新初始化，你又去用又清理掉，你每次执行这个数据是单独为你初始化的，就是为这个测试而服务的，所以说这个测试方法它就能独立了，它不依赖于别的方法，它也不依赖于数据库里的某一条特定的数据，这就比较好。

那么当然了我们要实现这样的这个步骤，我们可以在测试方法里去写代码，我先组织发一条数据，然后去执行测试再删除，但这样比较麻烦，对吧？而且很有可能是什么？我要执行好几个方法，它所依赖的数据，其实我们用同样的数据也可以，难道说我初始化数据删除数据，这个代码我要写多次吗？其实不用 Sprint，为了避免我们写多次重复的这样的代码，那么他给我们提供了几个注解，能够帮助我们实现这样一个步骤。

那么这几个注解一个是before class，一个是after class。其实它的含义其实已经很直观了，before class的意思是在类测试类加载之前，这方法就执行被注解修饰的方法，在类初始化之前执行一下，after class是在类销毁的时候执行一下，所以说如果你有一份数据是整个在类初始化的时候，我们就要初始化一次，你可以在这个地方去实践在这去销毁。

然后还有比方和after这两个注解，before这个注解是当你调用任何一个测试方法之前，它所修饰的方法都会被调用一次，after是你调用任何的测试方法之后，它所有的方法都会被调用去。好，我这样说可能是不直观，我们写一个小例子，通过例子一看就非常的明白。

好我就打开 Idea，然后我在 test包下新建一个测试类，这个类我叫死不认不得 test，那么我需要那么几个注解拷过来。

好，然后我说了我们要验证那几个注解的作用，那几个注解都是用来修饰方法的，我就先写一个方法，public注意study，work，这个方法叫比报克拉斯。

那么在这个方法之前我加上注解，你放克拉斯，因为这个注解所修饰的方法是在类初始化之前执行的，它只执行一次，它是和类有关的，所以说它必须是静态的。好，我在这里打印就是一个单词 Before class，一会我们看一下它什么时候执行你就明白了。

好，同理我再来一个 after class。这也需要是静态的方法，我这里也是简单的打印一个单词 after class，然后还有另外两个方法，那两个方法是我们每调一个测试方法，前后那个方法就会被调用，而这个before class after class是掉只掉一次，那个是掉多次，好，我写一下before这个就不用是静态的，因为每次调方法它都要执行，它就不能是静态的好，然后我也是输出一个单词，同理再来个after好了，那

么写完以后我需要再加几个测试方法，因为没有测试方法的话，这个效果不容易看得出来，对吧？

我加几个测试方法，然后也是随便说出一个单词，这是第一个方法，再来一个。

好两个就够了，然后我就执行一下，你不要直接执行这个带有特殊注解的方法，他们是一个初始化数据和销毁数据的方法，他们不是用来执行测试代码的，我在 test 我执行 test 一看一下是什么效果。Run test 一号，你看我在 runtest 之前掉了，比方放之后掉了 after，就这个方法之前之后，分别调 before 和 after class，方法没调吗？

其实是在我们启动就启动服务的时候他就已经执行了，你看一下，在初始化的时候就执行了，我们看这个地方的消息，比方克拉斯最后是应该是 be a couple 的位置 after class，然后同理我们叫 test 二，其实也是类似的执行一下，就在服务启动的过程中，在这个程序的初始化的过程中，before class 以及它就调用了，然后在方法之前之后分别调了 before 和 after。

当然你这样看的话看不出来 before 和 before class 有什么区别，反正都是之前调了对吧？看不出来有什么区别。那么我换一种方式去执行这个测试代码，这个效果可能就会明显一点，怎么执行这样。我选中整个类，然后的话右键软你看我之前执行的是选择一个方法，要单独执行某一个测试方法，我直接选择这个类，然后右键 run，其实是执行这个类当中所有的设置方法。

好执行一下看看，你看这两个测试方法都被执行了，那么 test 一执行之前之后，比方 after test 二前后 before after，就是说每个方法在调用的时候，前后 before after 都会被调用一下，然后我们去看整个程序加载的过程里面输出了什么，before class 你往后找只有一次。

这样的话执行就充分证明了刚才我所说的现象，就是说这两者只在类加载和销毁的时候执行一次，这两者是在我们调每个方法的时候分别会执行一次。

好了，所以说我们就可以在这样的方法里去初始化数据，然后在 after 里销毁数据。至于说你在这里初始化还是在这里初始化，那就看看实际的需要了。好，这样我们用一个特定的需求来演示一下到底怎么用，比如说我现在写好了帖子的 Service，然后的话我要测 service，我就把 service 注入进来，好，然后我希望在调每个方法之前，我都初始化一个新的数据，之后也初始化一个新的数据，然后这个数据就为这个测试方法所用，因为每个方法之前我要初始化一份新数据，所以这个数据在比方里说实话比较合适对吧？

好我就在比方里初始化一份数据，所谓初始化数据就是我们实例化一个 discourse post，然后把它插到表里，这样的话当前执行的测试方法就可以用这个数据了。

好，那么我就这样，我就 discourse post，我先实例化这么一个对象，然后给它设置一些数据，但是你看我这样实例化完以后，我在 test 里怎么去用它对吧？这样不方便用。所以通常我们会把变量挪到外面去，声明为成员变量，这样的话我改变的是成员变量，在 Test 方法里就可以调用成员变量，所以我在之外声明变量结果叫 data，就是一个要出要测试的数据，data 这里我就 data 等于 new discourse，然后我需要给它赋值，data 点 site，优质 ID，比如说 111 data 点 site 抬头，这个我叫 test title，点 set，content，test，content，然后 data，点 site，create time。

然后当前的时间，好，那么其他的属性我们可以不赋值，因为其他的属性不是整数，就是小数，它默认值就是 0，所以默认值就好了。然后我调用 Discourse 的 service，然后添加把这个测试数据加进去，你注意这个笔画方法在每个测试方法之前都会执行，比如说我调 test 一它会执行，那就有一个 data，这个 data test 一用，用完以后我要 after 的销毁，删掉，然后调 test 二之前又来一遍这个数据给 test 二用，但是这样的话其实效率会比较低，每次都新建一个数效率比较低，但是好处是这个方法它不依赖于别人，它依赖于自己的数据，独一份的数据，这样的话独立性好。

好，别忘了我 after 的时候把这数据删掉，不然的话表里最后搞了一堆垃圾数据测试数据，那么你打开页面一看，好多垃圾数据很让人奇怪是吧？把它删掉。

好那就删，我就调用 discus post service，然后点咱们没有真正的 delete，只是把它状态改了。Update status，好，你要删的是谁呢？它这测试数据对吧？那就是 data 点 get ID ID，然后把它的状态改为二，就代表删除了。这里请注释就初始化测试数据，然后删除测试数据。

好，那么这件事完成以后，比如说我要再写两个测试方法去测一下 service里其他的方法，加一个测试方法，test比如说我要测一下查询方法就find ID test find by ID，好，我就直接掉discuss post service，然后点find discuss post by ID。

那ID是谁呢？Id就是当前你初始化的数据的，ID我就查这个值。

好我当然了我会查到一条数据 post，这是我们执行测试代码，就是执行这个测试方法得到一个结果，我要验证这个结果对不对？这个时候你就不是说我system点out或者是logo打印出来看，你注意我们这个单元测试是我整体执行整个类，然后再挨个跑一遍，然后看他报不报错，报不报错，如果不报错每一个方法之前，就像我们刚才看的都会打个勾，如果一看都是勾，比方说这方法都通过了，我就不用细看就完了。

如果有差，我们重点看有差的方法是什么原因，怎么样才能够让我们怎么样才能够判断出这个结果，对不对？从而给他是通过还是不通过，那么用的一般都是一个断言的方式，断言就是说我判断一下你当前这个条件成不成立，什么也不发生，不成立就抛一场，我们要判断的是什么？判断一下我们查到的结果和我们初始化的数据是不是一致的，对吧？你查到的和他一致那就是对了，否则就不对，我就判断一下，首先我这样写，assert点a assert not。

Now这是判断传入的值是不是非空。我判断一下查询结果是不是非空，因为因为数据初始化的数据肯定是有，你一定是得非空才对，如果抛异常表示为空那就不对了，所以我先判断它是不是非空，然后再判断assert点assert一口子。

好，然后这一口子两传两个数据进来，一个是你期待的数据，你期望它是多少？第二个传的是你当前得到的数据，比如说我先判断得特点，get抬头，这个是post点get开头，这句话比较一下 Title 相等不相等，我查到的title和我初始化的 title是不是一致的。

好，同理我可以再判断date点，get post点get content，看这个内容是否一致，以此类推。

好，我就不挨个写了就这样好了，当然了其实你也可以怎么着，你也可以把这个discuss post这个类里边的兔斯顿方法重写一下，然后把两个对象比较相等，做好，处理好以后，你 e cos直接比较两个对象就好了，我现在是没有重写，我随便找两字段这么去处理。

这样然后先不着急，我只写完一个方法再来一个，比如说我要测试就写错了，我要测试一下 update就是修改分数的方法是否正确。

好，我就调用discuss post service点update，scar，我改的是哪个数据还是初始化的测试数据，这个测试数据我们通过data变量引用对吧？Data点get ID，我改的是 ID，然后比如说我要把它改成2000固定2000分，好当然了改这个方法执行完以后，我会得到一个修改过的行数，然后它应该是紧的，肯定是一行对吧？

所以我就多判断一下是特点一口子，我期望期待的值它是一行，我判断的结果是 rose看它是不是等于一，如果不等于它就会报错。

好，另外我还想判断一下，我改完以后，这个数据它那个值到底是改成了2000没有那怎么办？我得把这个数据再查一下，那就可以这个方法再执行一遍。我把测试的ID传进去，得到一个post，我看一下数据库里当前的post他 Score变没变对吧？也判断一下判断它是否相等，它不等于2000，然后 post叫get strong。

注意，那么我们判断两个小数它有第三个参数，第三个参数是精度小数，因为你要知道其实计算机底层它是二进制的，它只有整数，然后小数其实是一个近似的值，是浮点，浮点表示法表示一个近似的值，所以很多时候小数并不精确，你很多时候没有办法准确的比较两个小数准确的相等，那么因此它这里面给我们一个精度表示这是两位小数，就判断它两位小数到两位小数这个地方它等不等，因为有可能小数是无限循环或者不循环的，对吧？

我只判断两位等不等这个意思。

好了，写完以后我要执行，对整个类做测试，我就乱一下类。好，当然它test一test二这两个test都执行了对吧？然后它都通过了。你要注意就是说我test find body执行之前，就是创建了实例化的一个data在before里对吧？然后这个方法执行以后，data被销毁，调这个方法之前又一个data，所以虽然说它俩这里都用到了同一个变量名，但是在执行的时候它变量所引用的对象是不同的，你要注意。

好了，这个我们平时写单元测试的时候，尽量按照这种方式来写，你写完以后这个方法比较独立，那么我们在以后改了代码以后，直接一执行这个类，通通跑一遍，看看报没报错，马上就测完了，所以说以后的效率会很高的。

虽然说我们当时测的时候这样就有点麻烦，以后就会很快好了，当然了 spring boot test它其实还有很多其他的功能，平时我们其他的功能用的比较少，我们能做到刚才我所说的这种程度就差不多了就ok了。好了，那么关于单元测试我们就演示到这里，咱们下次课再见。

## 8.2 项目监控

---

### 54-项目监控.mp4

这次课我们来学习项目监控，就是在我们这个项目上线以后，我们怎么去对这个项目进行一个监控，好知道他这个是不是健康而稳定。那么我们所开发的框架是 spring boot，像这样的框架所开发出来的项目，我们项目监控就更重要了，为什么？因为我们用spring布的去开发的时候，它的开发难度是降低的，我们很多地方它都不用配，或者说简单的配一配就可以就能用就能开发。

但是程序在执行的时候，它底层做了很多的事，我们表面看不出来，所以我们程序中一共有多少个bin呢？并之间是什么关系，有多少个日志对吧？有多少开始我们的请求这个路径合并之间是什么映射关系等等，这么一系列的内容，我们都看不见摸不着，这个很让人这个担忧，担心他们是不是会有什么问题。

所以说我们在系统上线以后很有必要去关注这些个点，看一看这个程序是否像我预期的那样去健康运行。

那么 Spring boot的开发者他们也就是想到了这一点，所以说它提供一个很好用的工具来帮我们做这样的事情，这个工具叫spring boot act Twitter是专门用来监控十分像的，那么它的功能非常的强大，然后也非常的灵活，那么它弥补了4分部的项目，它在简开发简单的同时，带来了底层的就是隐藏所带来的一些就是隐忧，他帮我们解决了。

而要学习 step的艾克推特，我们需要主要了解这么一个概念，叫on, the points。

翻译过来就是端点，那什么是端点呢？这里我写的很清楚，就是监控应用程序的入口，我们通过端点去对我们刚才所说的那些个方面去监控，这个端点是多个 sprint里边它自带了很多个端点，如果你觉得不够用你也可以自定义，所以它一个是功能很强大，同时又很灵活。

然后我们如何去访问端点去看这些内容有两种方式，一种是我们通过HTTP的方式，当我们应用启动以来以后，我们通过浏览器写一个符合HTTP规范的路径，就能够看到就能够访问端点看到一些内容，或者是用 gmx我们jdk安装以后，它里面带了一个工具一个副客户端也可以，我课上就演示前者比较方便一点，我们用前者要访问这个端点这路径怎么写，路径是这样的，前面的话你该怎么写怎么写，如果house的8080 community，那么从项目往下，下级路径叫active统一的，然后下级这是端点的ID health，这是举个例子。

那么sprint内置了20多个端点，你就访问哪个端点就写他的ID就可以了。

当然了因为它的 idea特别多，我们不用说每一个都给你演示一遍，大家每一个都了解都懂。那么当你这个工作的时候，然后的话，你需要监控什么？你去手册上找一下 sprint部的手册上，就专门有一章写这个艾特，然后那边有多少个端点，他写得很清楚了，你去找一下，然后去看就可以了。

但是我们在加了这个功能以后，你要注意的一件事就是我要按需去配置，要暴露的端点，那么怎么说就是说如果你这个端点就从来不用你就别暴露了，因为暴露的话一个是影响性能，再一个的话也有被别人窃取到的一个风险。

再一个的话我们要对所有的端点进行一个权限控制，因为如果你不控制的话，那么谁都能访问到信息，它就能直接访问你应用的底层的内容，这个是很危险的对吧？所以说你要给它加上权限控制，像这样的请求，只有管理员能访问这样好一点。

好了，大概做了介绍以后，下面我们就来写一些例子来看一看这个东西怎么用。首先我们要用这个功能，你得先打包，我就搜一下这个包打开妹妹被抛弃的人，然后的话我搜是不瑞不疼对头。

好，就是 Spring boot activity不是 start，我没有start用 start好点进去，然后选一个版本，然后这copy我把它粘到我们配置文件里，这个版本我去掉，这样就用副泡沫指定的版本，很快这个包就导进来了，导进来以后，你只要这个导入了这个包，它立刻其实就有效了，这个时候我们启动应用的话就可以访问它的它的一些端点。

当然它默认的话不是所有端点都这个暴露出来，是这样，它默认有20多个端点，其中几乎所有的端点都是启用的，只有一个端点，默认是禁用的，端点的作用是用来关闭服务器，我建议你就不要把它启用了，因为什么？你通过端点去关闭服务器这个事儿有点儿儿戏，你要关闭服务器去服务器上去关，你为什么要通过端点去关呢？对吧？

而且它关闭服务器的方式还得用POS的请求对吧？就给人感觉这是留了个后门，你非要走后门，这是这是很危险的一件事，你就别启用了，对我们来说默认就所有的端点都启用了，但是默认情况下它只暴露了两个端点，只允许你访问两个，那么其他的端点你要访问的话，你需要声明加以配置，一会我们看怎么配，好，这样我已经导入这个包，我就把程序启动一下，我们来访问一下这个端点它暴露的端点看一下，好启动一下。

好了，服务已经启动了，我打开浏览器，然后访问一下，我先访问首页，好，然后我要访问这个端点，下级路径 Twitter，然后它默认暴露了两个端点，一个叫health端点返回的是当前的服务健康的程度是否健康的状况，然后它返回的是一个阶层格式的数据，返回的是status，up表示这个状态是ok的。

挺好。好，然后还有一个端点是默认是开放的叫音符，其实返回的是一个服务端的一个一些相关的信息，但是现在他什么也没返回空的。好了，总之我们现在先尝试一下，确实能访问，只要我一装上这个工具，它立刻就能够暴露出来一些信息给我们，对吧？当然这两个只是一个概况，更有效的我们更关注的内容，默认的话它没有给我们暴露出来，怎么暴露，还需要配我们配一下，我就打开配置文件。

Application, practice, 这个配置你可以从思政部的官方手册去找，然后这里我就直接写出来了，这是x的相关的配置。

然后的话通常会这样配，首先是manager问题，点on, the point就是端点，然后点y X pose点include等于什么意思？就是说我要暴露哪个端点，比如说我要暴露病死，这个端点我要查看所有的病你可以这样写，比如说我还想暴露日志的端点，你可以选logs，当然是20多个端点，比如说我都想暴露怎么办？你就别一个一个写了，你可以写个星，就所有的端点都暴露，我都可以访问，但是可能有个别几个我不需要，我想把它禁掉，我先都暴露，然后再把个别几个禁掉，这样不省事对吧？

所以说我看可以这样写，再复制一行management。

第二and points Web expose，然后这个地方就排除 include等于，然后写上你要排除的端点，比如说我要排除音符，然后我要排除cat，比如这个是缓存相关的信息或排除，其他都保留，这就是一个例子，将来你可以自己完全自定义，但是我是给你做一个示例示范。

好，那么写完以后试一下CTRL f9，好，那么它就重启了，重启完以后我打开浏览器，我这回再访问Info，你看它提示404不暴露了，隐藏了，然后我访问catch返不了隐藏了，而其他的health可以的对吧？



然后还有一个beans，BS是返回我们应用中，我们当前这个容器里所有的病，他是一个超级大的一个内容，当然有人说你返回这样的内容我怎么看？

他只是给你返回这个内容，这个内容的分析，你可以写一个程序，或者是你想你把这个内容拷起来，写一个程序解析一下，然后再去细看，或者是你安装一些和 act有关的一些工具，它能够帮你去分析这个结果。

总之我们看到这是一个接生的对象，它里面包含了所有的病，有这些东西我们就怎么说就放心了，然后的话需要用的时候再详细的分析，对吧？毕竟有这么一个渠道好。

然后再看比如说我要看logs，这是所有的日志的内容等等，总之二十几个端点你可以挨个这样去看，每一个端点暴露出一方面的内容，你需要看什么，你就去这样去访问就可以了。

好，以上是我们演示的是它内置的端点，比如说我想监控的内容，这里边没有，我想监控的内容其实可能和业务有关系，要监控个性化的东西，你可以自定义端点，下面给大家写一下怎么去自定义一个端点。

我们会自定义端点以后，可能你对艾特的底层就有一个更清楚的了解了，我把这个端点单独建一个包，我在这个肯定得等一下再建一个包就叫艾特。

好，然后这里我要创建一个类，代表的是一个端点，我先说一下我建这个端点要监控什么，比如说我要监控数据库，看它的连接当前是不是正常的。

好，所以端点这个类的名字我叫data，贝斯数据库，然后 and point。好，那么这个类我们需要就把它交给容器来管理，所以我加上component它不属于任何一个层，它是公共的一个组件。

另外我们还需要在端点上再加一个特殊的注解。On, the point。On, the point里你要写上ID等于什么？这就是给端点取一个ID，好，让我们将来通过ID能访问它，你这个ID是什么这个地方我们就通过ID来访问它。好，这个我就叫好，然后这里边怎么写？首先我先实地画一个logo，一会有用。好，实例画完log以后，我们怎么看数据库当前的连接是不是正常的？你可以在端点调用端点的时候，你尝试去访问一下数据库，你尝试去获取一个连接对吧？如果说这个连接能取到，那就表示ok的，如果取不到就是有问题的对吧？

所以说我们尝试获取一个连接，我们尝试获取连接其实有多种方式，一种方式我们可以访问连接池，你把data source注入进来，然后 get connection，或者用更原始的方式，我们直接把连接参数注入进来，然后我通过专卖店去访问都行。这样我就简单简单一点，我就通过连接池去访这个连接，咱们来看一下，好，那么连接尺其实是由 spring容器来管理的，所以说你直接把它注入进来就可以了，连接池的顶层接口就叫data source。

好，有了连接池以后我就可以尝试获取连接了，我就能看我能不能得到连接。接下来我需要写一个方法来去做这件事儿，我定一个方法，这个方法我就要check，connect这方法之前需要加一个注解，我叫read。Operation什么意思？就比方说这个方法我们是通过盖的请求来访问的，是一个盖的请求，如果是其他的请求，你要写你要访问其他的option，其他的好像有我记得好像是right option。

你看有write option那就是 post或者是put提交数据，我这是get请求，所以用 Read option，也就是说这个端点你只能通过get请求来访问，它表示的get请求的意思。

好，然后这个方法之内我就尝试获取连接，我就 data source点get connection。当然这句话我们需要重新开始，这样我把这句话写到小括号里，为啥写在小括号里？因为在小括号里我们初始化的资源，那么编译的时候它会自动加finally把它关闭掉，这样我就不用手关闭了。好，然后那么在这里我如果是程序进行到这就表示没报错，那我就记个日志，不是记日志就直接返回一个结果，我们要返回一个什么结果？

其实你也看到了端点要给浏览器，返回的是一个Jason对吧？

所以我死卷要返回的是Jason的字符串，我这里写委特community有推广get杰森死菌，然后的话扣打0给个提示说获取连接成功，比如说连接没问题，如果捕获到异常了，这个时候我最好记个日志说获取连接失败，把这个错误消息也带上，然后的话这时候也返回一个结果，给浏览器平面的YouTube get杰森斯卷一获取连接失败。

好了，这样就可以了，写完以后我们不用配，因为这个并是交给容器来管理的，直接就可用，然后我们直接重新编译CTRL f9。好了，完成以后我们打开浏览器访问一下，这块你就得写data，base你看获取连接成功没有问题对吧？那就表示说这数据库连接没问题。好，如何自定义端点我们也演示完了，总之你是用它自带的还是自定义的都可以了。

最后别忘了一件事儿，就是我们端点的路径一定要做权限管理，不然的话谁都能访问，这个很危险对吧？而做权限管理的话非常简单，因为我们现在有这个secret，所以说我们直接在Secret里做一个简单的配置就可以了。找到配置类，这个功能应该是只有管理员能用，其他人都不能用，所以说我把它配到这儿。这个路径是艾特杠，星星只要以艾特打头的，那么都是管理员才能访问。

好，再重构一下，再重新编译一下，编译完以后再来。比如说访问一下首页，看一下我当天有没有登录，没登录我访问艾特Twitter，然后health不可以你得登录对吧？我登录一下CC。好，再访问艾特还是不行，因为权限不够，这不是管理员。好，我用管理员登录11再来访问艾特尔斯，这就可以了，再访问Database。好没有问题。那么如何使用speak Twitter进行一个项目监控，就是这样的一个操作。好了，那么关于这项内容我们就演示到这里，咱们下次课再见。

## 8.3 项目部署

---

55-项目部署.mp4

项目打个包，然后部署到应用服务器上去。当然了我的应用服务器选的还是Tom cat，它和我们之前所使用的汤派的是有区别的。我们之前用的是spring布的，它内嵌的汤派的，而这个是一个正常的普通的一个Tom cat，不是内嵌的。

Tom cat它是一个应用服务器，它是一个软件，它需要装在硬件之上，这里我选择的硬件是阿里云的云服务器，然后配置是我选的是2核的CPU，4G的内存，这个系统的版本是三头s7.6，那么如果你想尝试的话，你也可以去购买这种云服务，或者是你自己在本机搭一个虚拟机也可以，但是要注意的是系统版本的问题不无所谓，问题不大，主要是配置内存不能小于4个g如果小于4个g的话，我们装的这些东西就跑不起来了，这是一个最低的配置的要求。

好了，然后我通过这个图主要是想给你展现一下，我们一会儿部署的时候要干的要干这个活，我们要装哪些东西，这些东西之间是一个什么关系，我把它给你理清楚，直观的呈现给你，这样的话我一会装你不至于说乱了，我们随时保持一个比较清醒的思路，那么在云服务器上我需要装这些东西，首先有NDS那么它负责分发请求，浏览器访问的是ndx而不是直接访问Tom cat，为什么要这样做呢？

其实这个Angel就是Tom cat的相当于一个代理，我们通常称之为反向代理，那么怎么说，如果是正向代理指的是他代理的是浏览器，反向代理指的是代理的是服务器，你要访问服务器，必须通过ndx访问，那么将来如果说我Tom cat部署多个节点，ndx可以平均的把请求分发到某一个节点之上，对吧？

如果说没有ndx你直接提供了三个Tom cat，你说我访问哪一个就就很混乱对吧？Ndx能够起到这样一个分流的作用，好，所以要装ndx然后装Tom cat。

Tom cat它本身是一个用Java开发的一个服务器，它自身的运行就需要依赖于GRE再一个它里面运行的是Java程序，也需要GRE，所以说我们在此之前还得把GRE装好，掌握运行环境。

然后我还要装一个maven，maven的话主要是用来编译打包的，为什么在服务器上装没问呢？

就是说我希望在服务器上打包程序，我们写的源码发给服务器，服务器打包，然后再把它放到摊开的之下，有人可能会想你为什么没有在你本机上打好包，然后传到这个服务器上来有两点原因，第一个如果我在服务器上打包的话，这个打的包里边会包含所有我们依赖的账，这个会很大的会有一两百兆，那么你在本地把打好的包传给这个服务器，那就比较慢了，当时间这是第一个。

第二个还有可能我们本地的电脑有一些环境和应用服务器有可能有区别，比如说我本地装的是jdk12，这里边装的是GRE假设是11或者假设是13，那么万一有这样的差异的话，你本地打好的包在这上运行可能会有一些问题，而你在服务器上直接编译打包，肯定就是能够直接在这上运行起来，出于这么两点的考虑，所以服务器上装一个没问用来打包程序。

然后我们的应用依赖于这么多个工具，包括my circle，red is卡夫卡elastic search，还有 wkhtml图PDF这工具，那么这些工具是他们开的，在运行的时候会去调用他们的，我们都要一次把它装好。

当然了我一会装的时候并不会说严格的按照我刚才所说的顺序来装，反正按照我习惯的顺序，或者说按照他们依赖的顺序去装，那就一会再说了，总之要装这些东西。

然后这里我想说一下，就是说我们现在是在教学学习阶段，那么我们所拥有的服务器资源是非常有限的，我只有一个阿里云的服务器，所以说他只能装这些东西，如果是一个真实的生产环境的话，其实很多个工具是可以部署多个实例的，tomcat可以部署多个实例，数据库可以多个实例 release，卡夫卡几乎每一个工具都可以部署多个实例，让它处理能力更强，并发能力更好。

但是同样的你要梳理多部署多个实例，很显然你一台服务器就不够了，你得好几台服务器，我们学习的时候不方便，没有那么多资源怎么办？我们就每一个应用每一个应用只部署一个节点，只部署一个实例，所以说我们做的是个比较单一的这么一个体系这样的。

好，然后我要把这些应用装到阿里云服务器上去的话，我需要一个工具能够访问阿里云服务器，其实访问这个就是Linux系统的客户端有很多，这里我用的是免费的铺贴，用它来访问云服务器，然后去操作服务器装东西，这样好了，大概介绍到这以后我们知道要干什么了，以及我所做这件事的环境知道了，接下来我就开始安装这些东西，我们先安装，安装完以后我们再去对代码稍作调整，然后把它部署上去运行起来。

好，首先我需要安装铺贴，因为我要访问这个服务器，得先把它装好，铺贴它的官方网站是铺贴一点org打开以后比较简陋，然后的话你点这儿可以下载，然后这儿就是 Windows的版本，安装的工具，那么你可以我选的是64位的，我这里已经提前下载好，并且我都已经装好了，为什么这个工具不给你演示安装呢？

因为就是双击，然后下一步一路默认就完了，这里没有任何的一个需要注意的，特别的细节，默认安装一路下去就可以了，所以我提前装好了，然后装好以后我先启动一下，然后访问服务器看一下，给你看一下这工具就它然后这里我之前已经就访问过，所以说它有一个记录120.55.53.217，这就是我所购买的阿里云的服务器的IP地址，然后默认端口是22这样。

当然了，如果你要是做了域名解析的话，你将一个域名绑定到 IP地址上，你通过域名也可以访问我这里，我没有买域名，没有做域名解析，所以说只能通过这个IP来访问了，你就把这个IP当域名也可以。好，然后别着急再点一下 Connection，然后这个地方改成10，如果你不改10的话，可能是你连上这个服务器，然后你稍微过了一小会儿，没有动作它就断开了，改成10以后它很久也不断开这个意思。

好，然后 open以后他说了log in as，你以什么样的身份去登录这个服务器？

Root就是最高级别的身份 root用户，然后这是密码，这密码是我通过工具随机生成的，我也记不住会copy，然后到这怎么粘贴过来，你可以按键盘seat加insert，这里它没有任何显示，但是你 safety insert以后回车，这就登录上来了，好，那么我把窗口变大一点，我现在就连上了这个服务器了，连上以后接下来要做的事情就是刚才我PPT上所展现的，我要装这些东西。

那么装这些东西我再多说两句，因为我用的是sent os这个版本，所以我们安装的时候优先采用亚母命令来安装，因为亚母命令我们装的时候它很方便，也非常稳定，通常不会有问题，而且装完以后它会自动的配置好环境变量，自动安装好启动的服务，所以说它一切都是自动的，只要一个命令就什么都装好了，你都不用手动去配，很方便，但是它有一个缺点就是亚母命令它一般给你下载的工具的版本稍微旧一点，不过有些工具稍微旧一点也无所谓，也正常能用，所以说我们能这个命令就用这个命令，那么要么命令它是从一个中央服务器去下载安装工具，这个服务器就像没问的服务器一样，它是汇集了所有

经过大量测试的稳定的版本让你去下载去安装的，没有经过测试的不稳定的版本是不会在这里出现的，所以它安装的工具一定是非常稳定的，有这样一个好处。

好，那么总之第一个原则就是我能用要么安装的，我就用命令安装。

第二个那么有一些工具它这个样的库文件实在是太旧了，但我想用一个比较新的，然后恰好这个软件它的官网又提供了亚马仓库的更新的文件，我可以把这个文件下载到服务器上，然后我把这个库文件先装一下，这样的话我就可以用亚母去装新版本的软件。

你比如说my circle就是这样的，就默认要么仓库带的马赛克版本比较低，我从它官网下载一个仓库文件把它安装好，这样的话就可以用亚马逊装一个新的my circle，这样比较方便，因为my circle要是你要配相关的服务的好，

我刚才所说的库文件，我们从my circle的官网上怎么去下载，我给你看一下，因为之前没有讲过这个环节。

这是my circle官网，my circle.com，然后你可以点 download，准备下载，然后你看这有一排下载什么东西，我要下的是什么 young repost，样仓库的资源点，它点了以后，这里边它就默认了，就是下载最高版本的文件，这里面提供了几个链接，分别是针对不同的操作系统的，那三台os我要选选然后点 download以后，然后点这就是我要下载就可以了，但你这是下载到本地，我如果说我想在服务器上下载，我需要的是这个地址怎么办？

你可以右键复制一个链接地址，然后在服务器上通过这个地址来下载就可以了。

好，所以如果说这个软件我能用要么安装，但是版本较低，我去看一下它的官网有没有对应的仓库的下载，有的话我就还是用这种方式先下载库文件，然后有一些软件它是不支持用亚麻安装的，仓库里没有这个文件，这个时候怎么办？我们就别用样了，我们就去它的官网下载针对Linux的安装包，是后缀是踏点gz的安装包。

好，这里我举个例子，比如说没稳，我们之前在windows下我们是下载安装过，如果是我要在Linux安装没稳，我应该下载哪个版本，我们看一下梅文点阿帕奇点org然后点左侧的download，然后后面之前我们windows上不是下载 zip对吧？你现在是 Linux下载 tar点gz这个包，当然我们需要做的是右键复制这个链接的地址，然后在我们的服务器上通过命令把它下载到服务器就可以了。

然后我们课程里还有别的工具也是类似的逻辑，你要去官网下载包，但是那些工具我就不挨个给你演示官网了，演示怎么去下载的，因为我们上课的时候都告诉你卡夫卡官网是什么对吧？

以来txt官网是什么，然后怎么去下载，告诉你怎么找过，那你找到那个界面，只不过换一个包而已，我相信大家都能够搞定好了。

以上是我们在服务器安装这些软件的时候，我所建议的原则，就是能用要么就用，要么不能要么就下载安装包，然后有些工具我们需要手动配置一下，那就手动配置一下。

接下来我给你演示一下，比如说我要在服务器上下载一个工具，比如说没问我应该怎么去下载，刚才我不是已经把 maven的链接已经复制了一下，我回到这个客户端它连的是这个服务器了。

好我要把这个软件下载到哪，我就CD到目录下，CD到其实我当前就在 root目录下，我就在这里，目前其实有一些内容就是我提前下载好的，我课堂上我刚才说了在哪去下载这些资源，大概怎么下载，然后具体我课上就不去给你演示一个下载了，因为很费时间的，然后现在我只是做一个示例，说我要下载怎么办？做一个示例，这样我先创建一个test文件夹，然后我CD到test目录之下，现在什么都没有，我在这我要把美文的包下到这里来，我给你演示一下。

非常简单，W get杠a杠c然后把刚才的路径拷贝过来，你可以右键粘贴或者是 safe的加insert粘贴都可以。好，这就是我刚才复制的没文的下载的路径，然后回车。当然了这需要等一会 maven的包还算比较小，现在还算是比较快，所以我用他的给你演示，其他的包就不是这样的。好我们等一会下载完了，我们再往后继续。

好了，这就下载完了，你可以看一下当前的目录，确实有这个文件，但其实我不需要，其实我之前都已经下载好了，你看上一级 root目录已经有了，没问的包，我只是给你做一个演示。

另外像这个一类是txt，还有它的中文分词的插件，还有卡夫卡，这几个文件我也已经提前下载好了。他们的官网在哪，之前我上课讲过，所以说你自己去找。然后买circle的库文件，刚才我说过，我也已经提前下载好了，一会就不再重新下载了。另外还有 Tom cat之前我们用的是内嵌的版本，没有去单独下载它。Tom cat我也下载好了，在哪下载，我告诉你这个地方它的官网是 Tom cat点阿帕奇点org，然后你点左侧的 download登录的里边有一个选择一个版本，你可以选择一个9最高的版本。

好，然后踏gz现在这个就可以了。

好，总之这些我们课上一会要安装所需要的这些包，我都提前下载好了，不在这个范围之内的，就说明我要用arm去安装，就不需要下载包。

好，那么做好了这些准备工作以后，我还要做一件事，我们不只是要从官网上下下载一些资源，我们也需要从我们本地向这个服务器传一些东西，你比如说我们初始化数据库的脚本，以内的circle点GIP要传进来，对不对？

然后的话我们项目的源码最终是要传进来，是这样的，我们如何从本地往服务器上传一个文件，我明天给大家演示一下，首先我先找一个文件，好，我就找这个文件，肯定那些盈利的circle，就是我们初始化数据库的文件，好我要传的话把它压缩一下，压缩成一个GIP文件怎么传，我需要通过本地的命令行，我打开命令行，然后找到这个路径，CD到d盘work之下，然后看一下就 GIP文件，然后传的话是用命令 ps CP pscp然后的话写上你要传的文件的名字，我这个文件的名字叫 community in it circle点接IP。

好，然后你要传给哪个服务器，这里我写root at，然后 IP地址120点55点53.217，root表示说我要以root用户的身份往这个服务器上传东西，后面跟的是这个服务器的IP地址，然后的话冒号，后面斜线跟的是你要把它传到哪个路径下，我是要放到root test目录之下，好，写完命令以后回车，这时候他要求你填写入他的密码，好，还得考虑一下，回车，然后很快就传过去了。

这个时候我回到铺顶，回到这个服务器上，然后我看一下 CD到root，test目录下看一下有了，所以说我们本地的资源是这样传过来的，tot这工具怎么用，大概我们就了解了好了，做好这些准备以后，接下来我们就开始安装刚才我所说的这些东西。

那么在安装这些东西之前，我们还要安装一个小的工具，是一个解压缩的工具叫安工具，因为什么？你看我们本地往这个服务器上传的文件，我们是压缩成zip的包，这个包我们得用安zip来工具来解压缩，而我系统下默认没有这个工具怎么办？我们可以用 Em的方式来安装。

好然后在root下不要用去要么去搜资源，搜不到我CD到根路径下，然后我搜一下一样的去一样的库里搜一下资源，这个命令叫an z然后星比如说以an z开头的资源，我都要搜一下回车，很快就收到了，就一个按这个点 x86-64复制一下，你可以右键复制或者是CTRL加insert，复制完以后用这个样去装一下一样，然后的话is大杠y杠y的意思是说你在安装的过程中，如果需要我选择是与否一律帮我选成是好，然后把刚才的名字粘过来，然后回车，然后稍等一会儿，很快就装好了，这就可以了，一会儿我们会用到，反正现在装好了。

好接下来我们就安装 PPT上所展现的这些工具了，首先要装什么呢？装 GRE，因为我们大部分的工具都依赖于它，所以我先装 GRE，我也是搜一下erm list，Java星你需要以Java开头来搜，它叫Java什么，然后一下搜了一大堆的东西，有很多个版本，1.6开始到1~11，然后后面还有一个 latest，我要用的 latest就最新的版本也就是12号，我就把这个copy一下，let it open jdk x86-64就它好，然后采用要么命令把它装一下一样，instead跟刚才那个一样，然后杠y把这个名字粘贴过来，可以了，然后推车。

好需要稍等，因为我们只是访问了库，要从库里下载资源才能安装，这需要一点时间，不过还好比较快。那么装完以后你可以试一下12瓦word，就可以直接用版本是12没有问题。那就是说我们用arm去装工具以后，他帮我们已经把该配的东西都配好了，我就可以直接拿来用了，很方便。

好那么GRE装好以后再来装maven，Maven的话不能用arm装，我们必须通过踏包来装，而这个包我已经下载好了，我CD到root目录下看一下就是它了。

通常我们会把安装程序放到opt的目录下，所以说我只要把包解压缩到opt下面就可以了，这是一个套文件，所以用套命令来解压缩套杠zvx然后的话阿帕奇 maybe什么然后杠c斜线opt回车。

好，这就解压完了，解压完以后你去看一下 CD到opt下，看一下有这个文件，我进去 CD到美版的目录下，然后 pwd我要看一下当前的路径，绝对路径是opt阿帕奇每本3.6.1，我把它copy一下，为什么要copy？因为我要配环境变量，因为我们是解压缩的方式把它解压缩到一个目录下，但是系统是识别不了它的命令的。我为了用方便我给它配个核心变量。

么Linux系统下环境变量的配置方式有好几种，我只讲一个比较常用的，我们把这个环境变量配到etc profile目录下， Etc文件夹profile目录下，我就用 vim工具 etc profile，然后放到其实我们要配的这个pass本来就有值，它就包含了一些参数，我现在需要把每本也追加到 pass目录的，后面怎么追加，我在这儿安一下。

插入模式，然后的话写上export，然后大写的pass等于dollar pass，dollar pass代表的是pass原本的值，然后后面冒号， pass拼的时候，它每一段中间用冒号隔开，冒号，然后粘贴过来我刚才拷贝的路径。

那意思是说我要把pass原有的值后面加上冒号，每晚的路径拼在一起，做成一个新的值，再写回到pass变量里 pass就被更新了，就这样好。然后那就保存一下，打到q保存一下，保存完以后现在还没生效，我们需要再通过命令让文件生效一下，让配置生效一下。

Source provide我写错了，少写个 source，profile这就生效了，我们可以这回可以打印出 pass变量，看一下它有没有维稳的路径，用一口命令来打印，pass到了pass，你看后面有维稳，这就可以了，到这你可以测一下每本高word，你看他说command not found，这个命令不存在，但我没写错，没文章我怎么写错，为啥不存在？

我仔细一看这块配错了，我们配的是命令存在的目录，其实是阿帕奇命令BIM的目录下，我这里忘了写BIM好吧，我重新再来一下vim etc program，然后插入模式，这里加个b保存一下，保存以后还得让他立刻生效，然后就SARS profile。好，这回我再来没本word你看这一回他就输出了，这回他就输出了，没文的版本3.6.1没有问题，好了没？文装好以后你还要注意，我们需要改一下密文的配置文件，然后把它镜像设置为阿里云。

好，我现在当前在哪个目录下，没文的目录下我看一下它的配置文件在conf路径下 CD过去，配置文件是sites编辑一下。

好，那么镜像要配到美容的内部这个代码，我们可以从本机copy好，我就打开了我本机的没有 config settings。好了，然后找到镜像文件的配置的位置，把这个代码贴一下，但是这儿我得插入模式，然后暂停到这儿。好这就行了，然后的话保存一下这就可以了。

这样的话我们一会通过maven的去编译，打包的时候，它会下载第一次比较慢会下载插件，这个时候它访问阿里云就会快很多了，好，没文章好以后，下面我想装 My circle，目前你看我们装好了什么呢？Gre每本对吧？

然后我再装 my circle，这些东西，那么my circle我们可以通过让仓库来装，但是你看我CD到跟录音像一样，历史，my circle先搜一下，它资源比较少，而且你搜到的马赛克的版本是比较低的，我们现在都更新到8了。

因此我才一开始在路的目录下提前的下，好了， my circle的最新的库文件我需要把它装一下，我能够在亚母仓库里收到最新的 my circle的版本，好装 my circle，我先把文件装一下，怎么装，也是用样本来装样本，然后的话意思到杠y my circle什么回车，好很快装好了，装了以后我切到根路径下，然后的话我搜一下就买circle一样，类似的。

My circle行。

好，这回东西就很多了，但是我不是说我都要我要装一个master或者服务器，它 copy一下，它的版本是8，后面买CK80是吧？好，然后就装一下一样，你知道y把这个路径把这个内容贴过来回车，那么他就会从仓库里去下载my circle的安装文件，然后自动帮我装好。那么my server的文件是比较大的，需要一个挺长的时间，我们就等装完以后我们再继续。

好了，这个过程比较慢，不过终于安装好了，安装好以后我把 my circle的服务启动一下，那么我们用亚麻安装的工具，如果是可以启动的话，我们直接用 system CTL这个命令就可以启动的话，用 start，然后后面跟的是你的服务的名字， circle d马斯克d是马斯克后台的服务的名字，回车，他需要稍微等一会儿停顿一下，因为这个启动也是需要一点时间的，好，这就启动完了，这个时候我可以CCM ctr，然后的话Seder's买Soady，看一下当前服务的状态，my circle d服务的状态它是Xu running正在启动，没有问题。

好，然后这个时候我们需要做什么？就是我们在安装的时候，我们之前在windows上也安装过，在安装的过程中，它会给我们生成一个临时的密码，那么这个密码在哪？他在my circle的日志文件里，所以这个时候我们需要找日志文件把密码找出来，好能够登录它把密码给改掉，我用命令 great，然后的话搜这个单词pass word，从哪个文件里搜 What？

Log，然后 my circle d点log，从 log文件里去搜password这个单词，我这里一下搜出来4个，其实我要的是最后1个，因为我在这个服务器上其实装了好几遍，装了33个装好几遍，最后一个刚刚生成的，它的密码很复杂，就这个了，这根本记不住 copy。

好，copy完以后我们就用密码登录一下my circle，然后把它密码设置为我们自己能够记住的 my circle杠u root杠p密码粘贴过来，登进来了。

好，那么登进来以后我们修改一下 my circle的 root，用户的密码，auto user root at look host。

I don't find后面写上这个密码，你注意 my circle，Linux版本它要求密码要求的比较高，它必须有大写字母，必须有特殊符号，必须还得有数字，所以说你不能说简单的123456不行了，我这样我写now cold下划线123用这个密码，好回车，回车以后可以了，我就先退出一下，然后用我新的密码重新登录一遍 my circle杠u root杠p now hold下划线123。

可以的。好了，My circle我就装好了。

装好max以后，一会我们用的时候它需要有初始化的数据，咱们的测试数据，这样我们跑这个项目的时候才能跑得起来，所以说我需要导入数据，我先CD的word目录下，你看这里边我之前不是传了我们用到的4月份的文件对吧？

好，我现在先把它解压缩一下，解压缩这个文件就需要用到IE命令按然后杠d解压到哪个目录下，root目录下，然后文件名你c口点g回车好，解完以后再看一下，有了一个目录我进去不用进去，就是这里边三个文件我直接写了，我需要登录my circle，然后执行这三个文件导入数据密码登录进来以后我先建个库，叫community和我们之前的保持一致，瑞德贝斯 community，ok然后又是community，现在我要导入数据了，用命令SARS root in it。

Gun circle。然后第一个文件名叫in it god。Game点server，好，导入完毕，然后再来。

第二个是in it data点circle。完了就第一个是我们创建所有的表，什么帖子，优质这些表。第二个 data是测试数据，然后还有一个 cart，定时任务的工具所需的表也需要导入进来。它的名字叫tables。My circle。In the DB点circle。打完以后咱们看一下，so tables这些表都是我们熟悉的对吧？然后这里需要注意优点，我们需要稍微调整一下，为啥要调整？主要是调整数据，不是调整结构，为什么要调整给你看一下，谁来ID有人类，either you are ill，you are user。

这里边有一些用户的头像的路径是logo house的是本机，对吧？我们本机有这个服务器上你logo house的访问的服务器，本机在这上运行的话，那服务器上没有这个东西对吧？所以说路径你要是用这个路径的话就找不到这个图片了。

所以我们需要把这样的带有logo house的单词的路径改成这样的方式，好能找到这个图片，能找一个图片，至于找到是哪个图片都无所谓，要不然看不到，很难受，我随便copy一个，然后的话更新一下User。Update user。Site, hider。12等于粘贴过来都等于一个固定的，我换一个不316了好吧，666t点PNG好，都改成同一个头像，然后给个where条件 where either you are like local house。就是说你路径里带local house的这个单词的，我就把它换成这个路径可以了，然后你再查一遍，最后就没有找本地的图片，这样就好了。

好，那么my circle搞定，我退出去，接下来不要装radius，这个radius我们可以利用这个样来安装，然后版本不是最新的，但是也够用了，我试过。好，我就搜一下一样的类似的release。行，就一个就是它Copy以后我装了一样的。In, store杠y他这个包比较小，很快就装好了，装好以后我要启动ready这个服务还是 system CTL start ready? 启动以后看一下状态对不对？绿的对的。

好 Release就可用了，我们可以测一下，read it cri反映他就是解密客户端，然后你kiss行，啥也得不到，但是命令能执行说明ok的可以了，推出好，那么rise我们也就装好了，再来再装卡夫卡，那么卡夫卡我们不能用亚麻安装，我们需要用压缩包，那么有再去入的。

看一下它，好我还是把它解压缩到opt的目录下，它杠zvxf然后的话卡夫卡点它tgz然后的话杠c OBD解压缩完以后我们去 OBD下看一下，看看有了那就进去再看。它有一个肯目录里面装的是配置文件，我们要看一下配置文件有没有必要改。

Cd肯定搞错了，CD肯定是好，他这有很多配置文件，其中我们之前也讲过，我们要看的是主keep还有Siri，我们先看主keep，你看他这已经配好了，就data dir主keep所产生的数据会存到路径下，temp tmp主基本这可以这个路径我们不用改它，临时文件都放到tmp目录下，在Linux系统下没有问题就这样，然后退出了，然后再看 server，点process的break，ID就是服务器的ID节点的ID是0没问题，然后再往后看它也有一个路径，log drs等于 camp卡夫卡洛克斯这也可以，我们不用改这路径，就是符合Linux的规范，好退出默认配置就可以了，我就不改配置了。

那么接下来我就要启动这个组keep，然后再启动卡夫卡，那么我们需要敲命令去启动，就现在我在哪个目录下往上退一格，退一级。

好，我在这个路径下要启动，组keep，但你注意我们之前启动组keep怎么启动的，敲命令，然后的话在命令行敲命令启动的这不假，但是我们是用前台的方式启动的。那么如果说我把 come on的命令窗口关了，那么这组paper就死掉了。这是前台的启动的方式的一个弊端，你窗口一关掉他就挂了。这是服务器，我不希望窗口一关掉它就挂了，它应该一直在那跑着，除非服务器死机的对吧？怎么办？

我们需要用后台的方式来启动，我们之前没有尝试过如何采用后台的方式启动主keep也很简单，b然后Super server start, sh我要执行这个命令，然后后面加一个参数，drmdaemon我用后台的方式来启动，然后后面再跟上，我要使用的组keep的配置文件回车启动了，好，然后用类似的方式来启动卡夫卡，但启动卡夫卡命令和它还不一样，那卡夫卡后台启动是这样写，no hard。然后 B卡夫卡server start执行卡夫卡的命令，然后用哪个配置文件是config? Server点process，后边还得跟着这样的一句话，一大于号dev杠 now然后还得写二大于号 and一，然后再来一个and固定的写法，你必须这么写，它能够后台启动比较别扭，它每个方式还不一样。

好，然后回车启动这是它的 ID进程ID，好，那么启动完以后我们测一下怎么测，当然你可以给卡夫卡的创建个主题发个消息什么的，你可以测我就不那么麻烦了。我尝试执行什么，就是我尝试查看一下卡夫卡它里边有哪些主题。看一下。

B卡夫卡topics, sh还有杠杠list, 杠杠, bootstrap杠, so然后我要访问的服务器 local, host是本机的929092，那么卡不卡它默认的 IP默认端口是9092，然后回车，稍等一下，最终他什么也没打出来，为什么？因为我这个卡不卡刚装好，他什么主题都没有对不对？

所以打印出一个空行来，但是他没报错，那就意味着这命令是可以执行的，而且他卡了一下，他确实在搜没搜到而已。对了。



好，这是卡夫卡。安装好了，下面我再来安装 Elastic search，这个工具也是通过解压缩的方式来安装 CD到root下，我已经提前下载好了安装包，you let it take search把它解压缩，然后还有一个分词插件，把这个分词插件解压缩到 plus目录下，我们要解压缩两次。

好，我先解答说你来stick search杠zv xf然后以less take search，然后是6.4.3，踏点gz杠杠c然后的话 opt好了，回车就解压缩过去了。

你别忘了我们还有这个插件，也需要解压缩到 us text search目录下，它是zip的包，这个插件没有踏包，只有zip的包，我就按z杠d安，最普的命令是后面跟的是你要解压做的目录杠d目录 opt less take search6.4.3，然后普拉蒂斯，然后注意普拉蒂下面我要建个ak目录，把这个插件放到 ak目录下，当然我普拉丁Cr默认没有，ik你这样写的话它会自动帮你创建好这个目录。

好，然后你解压缩哪个文件是以last search杠analyse，好了，然后回车，完了以后去看一下 cdopt you last take search，这里有这些东西，我们进到普拉丁之下看一下，用ik进到ik下看一下，有内容没问题，好，然后以last设置，我们需要对它进行一些配置，我再重新拿CD到obt以来CT目录下看这个目录下有它的配置文件，CD到看这个目录下看一眼好。

首先我们要配一来stake设施点yma这个文件打开它打开以后插入的模式我们需要配什么这个地方，我们需要配集群的名字，我把它改成老抠的，我们之前开发的时候也是这么干的，和我们的开发的代码的保持一致，后边的话有 pass点data是它所产生的数据要存放的位置，我希望它和卡夫卡什么的保持一致，我们把这个数据都存放到tap目录下。可以改一下你看。

然后里边建一个e list这个目录，然后再建个data子目录，把这个数据放到这里来。当然这个目录还没有，它会自动创建，好在后面还有一个这 pass点loss，就是 Elstx设施所产生的日志文件存放到哪里这里我也改一下，放到天p less take locks目录下好了，这样就可以了，其他的地方我们就一律默认就完了。

另外我再改一下 jvm office，就是说我一来是take设置它占用的 Java的内存空间占多少，我们需要给它改一下，它默认占一个g太大了。我们一共4G内存，它自己占一g的话，其他的就不够用了，我试过它占一g我们这些东西又跑不起来了。

好，我打开 Gvm options，你看它这默认了，占了一g最小启动时最小占一g最大也占一g就改一下改成 256，启动时它占256g内存，然后的话最大512，这是256512兆。

好可以了，其他的不用动，保存一下。

好到这一来是设施我就装好了，装好以后你要注意，你来CK设置它的要求比较，它不允许你用root用户去启动，你必须用普通用户启动，怎么办？

我还得建一个普通用户，我就group艾特先增加一个用户组叫脑壳的，然后 user at脑壳的一密码 123456，然后组脑壳的就是我建了一个脑壳的组里边建了一个用户叫脑壳的一密码是123456，好了，完了以后还得给用户设置权限，他得能访问opt目录，能访问到 e list的设计，还得能访问到tap目录，能访问到他的日志文件数据目录，我CD到opt目录下，然后 chown用命令杠r然后扣点e冒号，我扣的后面写个星，比如说用户他在对目录拥有所有的权限，他可以访问目录下所有的资源。

行，我就省点事就这么配了。

好，然后同样还得配 tap目录，cdtmt然后也是刚才的命令，脑壳的一脑壳的，然后行，好了，那么配好了权限以后，我们就切换到脑壳的一用户上面去，su杠，然后后头一你看我这脑壳的一什么已经切换，切换完以后我就CD到高PT，以less take search目录下，还是用后台的方式启动它，后台启动以less take设置怎么办？这样并 you let it take search，然后后面跟个杠d就可以了。

好了，执行完以后我们需要也是做个测试，这样我测试之前我先切回到 Root，用户那就ICU杠后面不用写root，杠默认就是root，你需要输入密码，拷贝一下。好，我切回来root用户这回我要测一下，我就发出个请求访问一下。

Elastic shirt服务器 curl杠x get然后的话，local house冒号9200斜线，health v就是通过命令访问一下，一来txt它的服务器是否健康，回车，你看green健康着已经启动了，可以了好了。这样的话我 es也就也装好了。

好，这一排工具，还有一个就是wkhtml图PDF我再把它装一下，我再回到命令行，这个可以用样本安装，我就先切换到根目录，然后搜一下一样的 list，wkhtml to，PDF后面跟个星回车，很快就搜到了它wk，html to，PDF x86-64，好，我就安装一下 erm in store杠y粘贴。

好可以了。

可以以后你要注意这工具默认在我们windows上，这样就可以，但在Linux上还不能，因为我这个Linux是一个纯服务器，它没有gui程序，它没有能处理界面的程序，所以你wk这个命令是用来生成，PDF用来生成图片的它是和gui有关系的，它需要gui的支持对吧？你又没有怎么办？我们需要装一个模拟的。Gui程序 Gui服务器。好，所以我还得搜一下一样的。

然后历史星 xvfb行搜一下，你看这工具它是叫xorg杠什么server？Xvfb它是在中间的，所以说我需要前后加个信用卡就装它了，复制，然后再装一下，这样 instead粘贴安装，很快就装好了，装好以后咱们测一下，看一看我们利用这两者能不能生成图片，我在哪生成 Root？Test我要在 test目录下生成图片，怎么写命令的这样写？

Xvfb-y你得先指定虚拟的 server，声明什么？后面加参数 Server，UPS，杠杠serve杠UPS，然后等于双引号杠screen。空格0，然后再来1024叉768。叉24。这块其实主要是设置我生成图片的时候，那么我采用的分辨率是多少分辨率，指定是分辨率，屏幕的分辨率这个意思。

好，然后后面跟的是我要运用的命令，我用的命令是wk，然后 Html to，因为我们项目中直接生成图片了，然后指定你要给哪个网址生成图片，HTTP s wwwbaidu.com，生成图片叫什么名？一点PNG完了回车，好完成以后看一下，有了，但是这个名称这样写太麻烦了，我们每次都这么写，太麻烦了。

我们可以再把它包装一下，我把它copy一下。怎么包装？就是说我们自己写一个脚本，然后的话它把这句话封装一下就可以了。

我两个CD到opt目录下，我在这里自己写一个脚本，我这样写 Wimwkhtml to，image点sh，这是一个脚本的名字，打开以后这是我们新建一个文档，里边什么内容都没有，你把刚才的等插入模式，刚才代码粘贴进来，xvfb杠run什么wkhtml to MH然后再来后面你要写一个引号，然后到了这就是说我们通过命令可以后面传一个参进来，这是表示参数的意思。

好，然后保存完以后你能看到这个文件已经生成了，但是他有读和写的权限却没有执行权限，我们需要给他执行权限。Ch mode加加x然后的话wk什么？回车再看，那么它有了执行权限，既然有了执行权限，我们就再测一下我在CD的root test目录下，这回我要调用的是opt下面的wk工具是我包装好的工具，然后给它传参HTTP，s三w百度com二点PNG回车好也生成了，看一下二点片机有了，这个工具也可用了。接下来我要安装的这些零碎都安装好了，接下来我就安装Tom cat，然后最后再安装NDS这就快了。

好那么Tom cat也是采用解压缩的方式安装 Cd到 Root目录下，我已经提前从官网下载好了这个文件，我把它解压缩一下，它 cvxf然后的话阿帕奇他们看，然后把它挤压缩到也是opt的目录下，那么解压做好以后，我们需要给他们开的配这个环境变量，因为我们可能可以随时随地去启动它，去关闭它，而不是非得切换到那个目录下。

好，所以说我就先CD到 opt目录下，然后阿帕奇 Tom cat我先CD到这个目录下，然后在CD到b目录下，因为它的命令是在b目录下，对吧？我pwd一下把它的目录拷一下，拷完以后也是配环境变量vim然后的话etc profile改这个文件也是在pass后面加一个路径而已，我和类似 Export pass等于到了pass冒号，然后把路径贴过来。

就是Tom cat并目录贴过来，别忘了保存，保存完以后要想生效，还得执行 `SARS` 命令 etc program，然后我们可以打印一下一口糟了，pass你看有没有放开的油，但是打印的时候会发现 maybe有重复，这个其实不影响使用，然后的话如果你把这个窗口关掉，重新再次连接的时候它就ok了。因为我们这次实力没有重连，所以说它就会有这样的一个情况，重连以后就好了，这不是问题。

好了，既然他们开的已经安装好了，我先不去安ndx我先把它启动一下，直接访问他是如果没问题再装index，用nx的分发请求给他，我在访问index是 Tom cat的启动命令，我可以这样写start up点sh回车start其实这需要一个过程，需要一个过程，稍等一会儿。

我这样我可以CD到opt阿帕奇Tom cat目录下看一下，那么它有一个con conf目录这里可以对它做一些配置，它有一个lab包里边是它自带的依赖的一些炸包，还有lock里边是它启动的一些日志，所以如果说你的tomcat启动完以后访问不了，你可以看看它的日志是不是启动的时候有什么问题。

还有一个很关键的目录是web，APP我们部署的项目就放到这里来，我们新装的他们还在这里是有一些默认的东西的，我们看一下这里面每一个文件夹就是一个项目入的项目，manager项目，excellence项目，DOC思想等等，它默认带了几个事例，带了几个事例，我们可以访问。

好应该是启动起来了，我打开浏览器，但我得通过IP访问它 copy一下hgdp冒号双曲线，那么Tom cat默认端口是8080，我就访问80 80回车，这是tomcat的默认的一个界面，这界面是从哪来呢？咱们刚才不是看了吗？它里边不是有几个默认的项目吗？那么我们访问他们开始他默认会访问 root下面的首页，其实我们看到这个界面是root的首页，就是这样的。

好，看到这个界面说明什么？我们刚开始装好了没有问题，好，我们再回去接着装最后的一个软件nexus。我还是回到根目录，然后我搜一下 Ndx也是可以压门安装，类似的恩格斯行就是它了，ndxx8664把它拷一下。好，然后就安装一样，y粘贴很快就装好了，装好以后 nx怎么样才能分发请求？把请求分发给汤太的，这需要配。

好，我们来配一下 vimetc配置文件的在etcndx目录下，配置文件具体叫ndx点conf插入模式，n这一项，它的配置文件的结构比较复杂，那么大家如果说你没有了解过，你可以课后找点资料自己看一看，前面比如说 events什么include，这些都是一个公共的一些参数，这里我们不管不用动，然后他它配置文件的结构是分块的，是分层次的，最外层的是HTTP，然后下面有一些公共的信息我们也不动，然后后面是server，其实我们主要是配 server，配这个server是配什么？

配一下，我们是配一下什么呢？我一共有几个Tom cat，真实的服务器有谁？那我ndx再配一个虚拟的服务器，把请求分发给真实的服务器，所以其实我们配什么？就是配虚拟服务器和真实服务器之间的关系。好，这是它默认给的一个示例，我们不用这个示例，我把它注掉，别删注掉你可以做一个参考。

稍微小心把它注掉，当然我们一会配的时候会比这个简单，然后它有杰克斯有很多功能，那么我们只是利用它的一点点功能做一个请求的分发以及负载均衡，当然其实我们只有一台服务器其实也均衡不了，其实主要就是请求分发或者叫反向代理，用恩杰斯来代理他们开的代理服务器的服务，我们叫反向代理，那么代理浏览器的叫正向代理或者就叫代理，好回来，我把这个重要以后我要配自己的内容，首先我配什么呢？

Up steam my server，这个是配我真实的服务器，我真实有多少服务器，那么请求就会分发给真实的服务器，我这里只有一个，但其实可以配多个 server，然后的话1127.0.0.1，就是本地本机端口是8080，其实就是本机的。他们看对吧？

然后后面写上max filth=3，什么意思？如果说杰克斯向汤姆开始分发了三次请求，都没有得到正确响应，他认为他们的挂他就把它移除到服务器的集合里，好，然后再来fail。Time out。什么意思？如果说他把这个服务器移除了，那么过多长时间，他会再去看一下你活没活过来，我这里给的是30秒。如果说三次访问你没有响应把你移除，那么过30秒再访问你一次，看你能不能活过来这个意思。

好，这是上面是我配好的真实的服务器，后面得配一个虚拟服务器能够分发到真实的服务器，这一堆服务器里，当然这里只有一个怎么配12万。

好，lesson80，比如说我虚拟的服务器使用的是80端口，监听的是80端口，那么所以你只要敲120.55.53.21780端口是默认的，直接仿的是谁呢？

我虚拟服务器好，然后接着写 Name，server下划线name120.55.53.217，这里我写的是IP地址，那么因为我没有域名，我只能写IP，其实也可以写域名，就是说当你通过这个IP以和端口访问的时候，它能够监听到它能够接收到，就由他来处理他怎么处理 location，实现意思是你访问访问的是根路径，这个就是12120.55.53.217，80我认为是根路径，斜线表示根路径，这根路径我由谁来处理，由他来处理，这怎么写很简单，pricing pass，然后的话HTTP冒号双斜线，myself意思是如果有请求访问的是它的80端口，那么我就把这个请求分给这里边的某一个服务器，当然平均分分给这里边的某一个服务器去处理，现在只有它了，就送给汤太太处理了，这就是分发了，这种行为就叫反向代理，你访问他把请求给他，就这样。

好，然后保存一下。好，保存完以后我测一下，我要启动一下，因为我是用要安装的，所以说它的启动服务已经自动装好了，我可以用system CT start来启动它。好，那么它出现这么一句话，说failed表示失败了，失败的原因基本上我刚才配置的有什么错误，所以我需要检查一下，看看刚才哪配错了。

好，我们看一下就所有改的代码，首先是这里有一些注掉的代码有没有搞错，再看一下我刚才写的代码，那么上面这都注掉了，这没什么好说的，主要是看这两个代码需要注意的是什么，就是这里边括号里边配的每一句话，后面都得有分号结束，而且是英文的，然后那就没有都是分号都有，我就看一下写的代码有没有错误。

Time out这个单词写错了，好，再检查一下upstream，好就这样我们试一下 wq然后我再次尝试启动恩格斯这回它没有什么提示表示应该是启动成功了，你可以再看一下 system ctr status，index状态是running没有问题，好，这个时候我尝试去访问一下它，回来我把这个端口去掉，直接访这个IP，直接把这个IP和你写上80是一样的，因为默认就是80。这个由谁来监听？就由 Ndx因为我们配的回车，一回车看到的还是Tom cat的这个界面，为什么？因为他把请求分发给了他们的助理，但这个地址没变，依然是这个IP。

就是说你感知不到它的背后是有另外一个服务器，你就以为是 Ip处理的，但其实是另外一个IP另外一个端口处理的，是吧？是这样，好了，现在我就把我装的这些东西都装好了以后，那么接下来要做什么呢？就是要把我的代码部署下去，部署上去了，主要是把我的代码部署到 tomcat的web APP的目录下，那么在部署之前我还要说一下就 ndx他们开的他的一个怎么说更更细致一点的关系。

再一个我还要说一下 Tom cat和部署项目之间的关系，你知道我访问了恩杰斯，恩杰克斯和tomcat什么关系？Tomcat怎么去访问他下面的项目，这个关系搞明白以后，我们一会再配的时候，有的地方你才能明白，这个我重新画个图，画个简单的图。

好，首先我先画一个方块，假设这就是我的方块。好，然后他们看一下，刚才我们也看了它里边默认带了几个项目对吧？我把它写一下。第有一个项目叫root，然后好像还有个项目叫examples，当然了我们后面要部署的项目也是放到这里来，也是部署到这里来。

然后我假设我 Tom cat它的访问路径，它的IP地址端口号是什么？假设是1.1.1.1，好8080这样我简单好记，好这是Tom cat，那么Tom cat之前有什么不是有 Njx对吧？我再画一个方块用来代表 Nx，谁来访问ndx当然是浏览器客户端了对吧？我还得换一个浏览器，那么浏览器他发出请求访问的是ndx然后 Ndx他把请求分发给汤盖，我们在这里只有一个唯一的汤盖。

那么你注意恩杰斯它是有它自己的 IP的，我假设虽然我们是一个服务器，但是我做个假设，它的 IP是1.1.1.0，冒号端口是80，这是ndx浏览器要访问tomcat，他需要通过nx访问他的地址，它的域名写成什么呢？他的IP得写成或者是把80去掉，也可以因为默认80或者是我们买了域名和 IP绑定到一起了，做了域名解析，这个时候我敲域名也可以，比如说域名是not code.com，当然我没买，只能敲IP了，实际上买了以后就可以这样了，你敲这个域名，通过域名解析找到了对应的IP就访问ndx然后 ndx把请求分发给真实的服务器淘汰，是这么一个过程，这么一个过程。

然后我们看到的我们刚才是这样做的，我们访的是 ndx然后他分发了给了 Tom cat，然后我们看到了一个界面界面，这个界面刚才我也说了，他们开的访问是谁，默认访问是入的项目下的首页，就是说你访问他们开的，他默认会访问入的项目，这就有一个小问题，如果我要访问张老师这个项目怎么办？

你得这样写这个路径你这样写脑壳的.com，直接写这个域名是访问root，如果你要访问一个张老师，你就加个下级一个张老师，就是说你这样写是访问root，你这样写是访问张投资就有一个问题，如果说我要部署我们的项目，不是叫肯定那天好，如果我这样部署的话，你看防火的项目是不是得写一个脑壳的.com斜线，肯定那天你这样不别扭吗？

对吧？你看哪个网站，我们访问的时候还得敲个下级，我们只能记住淘宝.com，baidu.com对吧？脑壳的.com下级是很难记的，所以我们肯定是希望什么？我希望我想脑壳的.com你就访问肯定的对吧？

好要想达到这一步，我们需要做一些处理，怎么处理处理方式，不只是一种其实有很多种办法，我们习惯于这么干，首先我先把 Root删了，没有root，然后我把肯定的项目打包的时候就打包成root，但是打包它是个包包root点，那么他们在一启动一运行，这包就被自动解压缩成入的目录，明白吧？

这样的话我们写脑壳.com就访问了，我们肯定的项目，通过这个办法去解决这个问题，我们把这个项目打包成root点y好，当然了，我们这个项目原始的名字是叫community对吧？是把它打包成了 root点，这里有个打包的行为用的是package命令，然后这就完了吗？也不是，你看如果我这样的话，处理完以后我先拿code.com，他就访问了 ndxndx分发给1.1.1.8080，那么它默认就访问了community，你注意我们直接访问肯定那天会得到什么呢？

我们访问肯定那天的跟路径会得到什么什么也没有，因为我们在control里从来就没有处理过这个路径，对不对？

所以我们还得想办法再处理，还有一个根路径，不然的话什么也得不到，我就得处理根路径，根路径是什么？斜线这个路径其实我们是希望直接访这个项目，它会返回首页对吧？我得想办法让它等价于index，在做这件事，把它搞明白就可以了。

怎么样才能让根路径等于index有两个办法，一个是我访问斜线，就重叠项的windows这就等价或者是转发过去也可以，那一会我用转发，这样路径这样就不用发一个新的请求，这个请求就可以解决问题。

好，然后还要注意，你看我们访问 Root的时候是不用写个斜线root的，也就是说 root这个项目它没有访问名，root项目没有访问名，既然can没有那天我们把它做成入的项目，他就应该没有访问名，所以我们之前项目中加了访问名，在这个时候需要把它去掉变成空的值。所以我们一会儿要对项目基于这个规则要做一个修改，就是把这个肯定的项目能改成空的值。

另外还要加一个对根路径的一个处理，让它访问index。好了，说完这个逻辑以后，接下来我们就来处理刚才我所说的这几个点。首先我要把 Root先把他们的自带的这个东西删了，好我回到服务器，我CD特别犀利，我先把他们给关了，你要删东西把它关了，SARS down点sh把它停止。

然后 CD到opt阿帕奇Tom cat目录下看一下，这里边有web APP进去看一下这里边有好几个项目都不是我们要运行的，是我们自己的项目，而不是他带的事例，干脆都删了算了对吧？你就 remove杠rf行。好，再看什么都没有了，删掉了全删了。

全删了以后，接下来我要调整代码了，我打开我的idea对代码进行调整，首先把项目的访问名改成空值改配置文件了，application practice在最顶部，这不写了肯定的地方把它改成空值，注意就等号后面什么都不写，就是空值就是空不是闹就是啥都没，有空值空字不串。

好别忘了，还有一个地方需要改，我们前台的logo点GS里也定义了项目房名那地方也得改搜一下个logo点GS，这地方也得改，但是你这个地方满足他的语法改成控制不串就可以了。

所以这就是我们当初统一定义的项目名要改的时候的便捷之处，不然的话你把这个项目名拼到各个角落改就麻烦了，这还好就改两个地方好了。改完以后还不行，我们要想部署的话，还需要对这个项目做更多的一个改造。还有什么？这个跟路径的处理，我回到我的idea，我打开 Home很重要，我们在这里定义的首页不是这里有首页，我再加一个路径的处理根路径根，路径就是斜线，我叫root跟路径跟 root，

然后 return 转发或重新下我转发了 index，所以你访问它就相当于访问 index 等价，好这个改完以后还不行。

我们打包的话，你看我们之前的配置文件，home 之前就是我们没有进行声明，他默认打包会把这个项目打成炸包，但我们现在是外部项目需要打成外包对吧？你需要对它做一个处理，你要在这写一句话，叫 package，然后我还希望你打的包名字叫 root 对不对？怎么去声明外包的名字，需要在最底下 Build 编译的地方去构建的地方去处理，在这里面需要写上 final name，最终打包的名字叫什么？叫 root，这样就行了。

这样的话我去。

怎么说。

我去打包这个项目就能够得到一个名为 root 的花苞，这就行了，还不行还差什么？

我先把这些先关掉，还差什么配置文件，你看我们项目中有两个配置文件，一个是 application practice，一个是 look back，那么这两个文件你看这里边的路径 d 盘，我们的阿里云的服务器它是 Linux，它可没有 d 盘，你这得改，对吧？这里头也有类似的现象对吧？包括一些 IP 地址，包括一些路径你得改，但你不能直接改原文件，你改它的话我们部署没问题了，我们本地执行就有问题了。对吧？难道说我把它改过去，然后我在开发的时候再改回来吗？那也太麻烦了。

所以 Spring boot 它有一个好的解决方案，它允许我们项目中存在多套配置文件，我开发时一套，部署时就上线时生产环境一套，测试是 1 套，你可以 3 套 4 套几套都行，然后通过一个开关去启用，哪一套配置文件，这里有个简单点就两套，开发一套，然后生产环境也就是部署的时候一套，好这样我先关掉，这样我把这个文件 copy 一下 CTRL c CTRL v 然后改个名字需要有一定的规则，必须后面写个横线，然后写个单词，我叫得拜了，很显然配种件是开发使用的 develop，然后再把它拷贝一下 CTRL c CTRL v 然后这个是 love back spring 横线对外的，好，就是说这两个文件是给开发时用的，这两个文件需不需要改，因为我们之前开发时做的配置就是给开发使用的对吧？

不用改，但是有了就可以了。

好再来我再点它 CTRL c CTRL v 再给第二套，就是生产环境用的 produce 把它复制一下，可能 c 可能 v 也是杠 produce，好我先这两个肯定要改的，我先不说这两个怎么改先不改，这是两套配置文件，一套是关键词是 develop，一套关键词是 produce，都满足固定的规律是以杠这个关键词结尾的，我怎么让 simple 的去加载某一套配置文件，我们需要这样它默认的配置文件中做一个开关，默认配置文件其实可以内容都删了，因为我们有了两套，另外对吧，这里需要做一个开关，怎么做也是很简单的，那么就一句话，这句话主要是对他支持叫 profile，它这种模式叫 profile，我们配的是和 profile 有关的配置，这样排 spring 点儿，profiles 点儿 active。

我们目前激活的配置文件是哪一套？如果我写对外的，那么就是以对外和结尾的配置文件被加载，如果我写 produce，那么以 produce 结尾的配置文件被加载，因为我一会要部署到 Linux 上去，所以说我写 produce 文件就被加载，另外这个老板该怎么办呢？还得配上 log back。得这样配了，log 点儿挺费等于 class pass。

Spring 杠它点儿 xml。什么意思？就是说我 Log back 的配置文件加载的是克拉斯帕斯之下名为 log back 杠 spring 杠，这个开关所对应的名字的 xml 文件，我现在是投丢，所以说它加载的是配置文件。那么总之我们在以后我们开发的时候就把它改成 develop，我们要部署的时候就把它改成 produce，这样就可以，就非常方便好了。

那么除此以外，最后还要处理一个地方是什么呢？你看我们的启动文件，我们之前是直接 run Main 方法对吧？把它当做一个普通的 Java 程序来启动的，但现在我们是要把它部署到 Tom cat 里，把它作为外部项目来启动。他们开的就是由 Java 来做的一个软件，它本身就有 main 方法，不可能就是一个 Java 程序，不可能有两个命令方法对吧？

不可能命中掉命，所以说我们现在就不能执行命了，脱不开的调哪个类从哪个入口去执行这个程序，我们需要给他一个入口，我们需要给他提供一个接口，所以我们就在当前的路径下新建一个类，实现一个接口在目录下，这个类的名字我叫community so late，然后的话一定是来着，当时写没写错，就是说这个单词写错也无所谓，但是最好是别写错没错，然后需要给它实现一个接口。不是是继承一个父类，通过继承父类的方式间接的实现一个接口。我们继承的是spring boot，然后的话写错了，spring boot。

然后 So late it is lies继承于它，然后我们重写它里面的一个方法，这个方法叫做config。这个方法有个参数叫builder，这个方法内部的逻辑我们要重写，很简单就一句话，我就直接 return builder，resource source。谁 community？Application点class。

就是说 Tom cat会访问类的方法，通过这个方法作为入口来启动来运行我们这个项目运行的时候，我们这是声明，我们的主配的文件，我们的其核心的配置文件是谁是肯定的，txt这里指的是配置文件，所以还是从它来开始加点配置好了，现在我就把我们这个项目需要调整的地方调整好以后怎么办？

我们就需要把这个项目部署到他们的下，把它拷贝到他们的下，我们怎么做？我先把这个项目的源码拷贝到这个服务器里，然后通过maven对它进行打包以后，把它搬到 Web APP下，复制到那里去就可以了。

好，看一下我们当前的这个项目有target，我要把这个target移除掉，为什么要移除？这样我打包的时候小一点，传的快一点对不对？好我就没问颗粒，你看这个target就没了，好，没了以后我找到我的work space，把这个项目打包一下，我然后再上传好，压缩好了以后我就要传了，那就再打开本地的命令行。

我现在是处于d盘work之下，我要去的是 work space这里边我要传的是肯定没有内贴点GIP还是用明了pscp没有贴点GIP root艾特，然后的话是120.55点53.217，冒号斜线就我把它传到目录下，回车需要密码，拷贝密码粘贴过来比较小，很快就传过去了，如果你导出的话带着炸包就大了，这就很小。

那我再回到服务器，CD到root之下看一看它了对吧？我需要把源码解压缩一下，我好去编译它，我去打包它，我这是一个zip安，然后杠d放到就放到入得下，肯定得点击IP再看，有的肯定的进去，写错了进去，然后这里边没有他盖没有编译好的代码，我需要编译好，不只是编译还要打包，但你注意我不要测试，一测试我们的测试类都走一遍，而我们的测试类写的又不标准，我们之前后面我单独讲的测试写的什么标准，所以说你执行的话会很麻烦，不要执行测试怎么办？

我们这样写命令 Maybe可令，如果你已经编译过你就肯定一下，如果你没编译肯定可以省略，这里我就写一下拍k上打包，然后的话杠d加个参数没稳点儿，taste点儿，skip忽略，等于数我要忽略测试不要测试，然后回车。

那么如果你是第一次安装没问去执行的话，它可能会下载一堆东西比较慢。我的其实已经执行过它本地有缓存的那些个包，应该不会太慢，好，咱们稍等一下，成功了看一下。Cd到to get目录下，有了root点y对吧？然后我就把root点y拷贝到它们的下，就算部署了，我就 move移动到 tomcat支架，他在哪？

在 opt阿帕奇tomcat放到 web APP的下面，我这个写错了少了一个文件名，这样不对，你得在后面跟着文件名然后再写文件夹，文件名是root点，然后的话是 Tom cat web APP的目录回声，好移过去了，你看当前的目录没有了，然后我CD过去，好看一下，有了root点就只有这一个，这个时候我就start up sh启动他们看一下，但是他马上提示说started其实还没有，因为我们的项目加载需要一点时间，没那么快，你可以这样，你可以去看一下loss，它里边有一个凯特琳的点，什么log，你可以看日志点，我好像有点问题，你看他报错了，为啥报错了，我突然想起来了，退回去，我突然想起来了。

刚才我是把这些处理好以后，我刚才才是做到这一步，复制了两个 produce，这个文件以后我没有改对吧？这个里面没有改，所以说这里面参数不对，所以说这里执行不起来，这样我再把它再得当一下，忘了这一点了。

好我回去再改，好我再重来，先改 application，produce点practice，从上往下走一遍。8080这个没问题，然后这地方注意改成处就是我们部署环境一生产环境缓存可以启用，然后这里是look，house没问题，为什么？

因为我们Tom cat和 My circle他们是同一台机器，他们开的就是访问本机的my circle，所以说这里边写logo house没有问题，root然后密码给改了，其他的不动，这个地方不动，这儿没有，这儿不动，这儿得改。

读命，你得把这个IP改成我们现在的这个服务器的IP，它的IP就是HTTP冒号上线127点55。不是120记错了，copy一下120.55点53.217，后面的端口不用写，默认就是80，然后你上传如果上传文件的话传到哪去？我们不能写d盘了，那就是TNT，然后的话 Abalos，aaah传到这里来，好再往下看，Readies也是本地的，没问题，卡不卡？

也是本地的，端口没问题，然后es也是本地的端口，没问题，这个task没有问题，看他没有路径，这不用改，wkwk这就有路径了，这个就必须得改了。

Wk我们访问wlan对吧？然后是我们自己编的叫wk html to，image点sh对吧？然后wk文件存放的位置，我们换一个tmp wk杠。Image可以了。好，七流云的配置不变，后面不变改好了，然后look back，别的地方不用动，该当然你也可以比如说把控制台打印去掉，比如说日志级别可以调，这里我不调了，我们必须改的是什么呢？是路径，对吧？

你不能放到d盘，没有d盘，我放到TNT下，可没问题下面好，这就可以了，这回才算是改好了，刚才有点问题，我得重新上传删了，重新打包一下，压缩好，然后我再重传一遍，还是执行刚才的命令Pscp什么？

回车，密码好充完了充完以后回到服务器，然后CD我是不是杀的到杀的到CD到root下看一眼，这肯定都是新的，我把原来旧的给它删了。

好，然后重新解压缩一下。那就是on，zip杠d root。放到root下肯定没有内存点zip。

好，看一下。Cd进去，好，我去看一下确认一下这个是有变化的。Cd的c下面下为SARS之下，然后我就vim application杠produce practice。你看这个开始变成true了对吧？然后这个域名变了，说明我们重新传，没有问题，好了，我需要去打包了，好就是这个路径。

然后我就重新打包一下，没问不写颗粒码还可以查看d没问题，Test。Skip two，好，然后回车。好，看一下，打完了进到target里面去，我把root重新移动过去，root点，然后放到opt阿帕奇Tom cat外部APP目录下，他说已经有了，你要不要覆盖？好，当前没了。到了唐凯的家，我重启唐凯的start sh好，我们可以CD到OB t阿帕奇tomcat，外APP下看一眼，确实有这个文件，然后的话这个时间是更新了。好，然后你看它有一个root文件夹，就是我一启动它们看的它就会把包解压缩到这儿来。

好，然后我再看一下日志，Cd到logs之下，vim卡特琳娜2019什么回车，我们看后面你看他又报错了，他又报错了，他报错了说什么是有个d好像刚才我也看到一个d的文件夹，可能是刚才我配置没改，它生成了一个错误的东西，刚才造成的影响还挺大，我再去把它弄一下，再set down一下。

然后再当前是洛克斯的CD点CD y的APP，它有个地冒号这么一个项目，这个肯定是有问题的。Rm杠rfd冒号杀了他，Rm杠我干脆把Ruetz文件夹给删了，只保留vau，然后重来start up，再去看他的日志，就是我们部署到服务器上以后，我们就没有办法用工具来调试了，我们一切都只能看日志，所以说一定要习惯于多看日志。

当前是外边APP CD点点，CD plus看日志，再往后翻。

好，最后它会有一个过程，然后会有一个提示，刚才那个错是刚才的，它会有一个提示，它提示说started server音多少毫秒对吧？

然后再看看再往前你看它就会提示，它就是解压缩了袜包root点儿袜，这是汤派的它启动的一个日志，我们退出去，我们再看我们自己的日志，我们自己的日志我们配了是在探探目录下cdtmp你看这里都是生成的一些放日志的文件夹，community就是我们自己声明的它自动创建的CD community。



好，然后你看这里边就有什么各种日志 one和era没有只有infer没有错是吧？好我们就访问一下，现在已经启动起来了，访问一下，我回到浏览器，直接访问ndx然后分发给我的Tom cat写这个路径就可以了。然后原来访问是猫的界面回车，好，现在就访问到了我们的项目，这是首页返回的内容对吧？然后你可以点注册点登录，你看他没有肯定的在一起直接 log in什么的，对吧？

比如说我登录aamvbs。

登录没问题是吧？然后看一个帖子没问题。

发个帖子也可以。

进去看点个赞也行是吧？

然后再回个帖也没问题，给回帖点个赞对吧？然后你这个点赞以后这里有消息了，有通知了对吧？谁评论了你谁点了赞，你也可以看就过来了，你看我点能点赞，说明release方面没问题对吧？然后我能发消息能发通知表示卡不卡？没问题，能不能搜索，刚才我是写的是测试部署是吧？测试搜到了对吧？

可以，但只收到一条数据，因为什么？我们 Estx设想我没有往里初始化那些，以前的旧的数据只有这一条，能收到说明es没问题，换个头像，换头像说明我们访问新能源，没问题，也说明wkhtml推灭命令没问题，基本上就是我刚才装的这些东西，数据库肯定没问题，ready是卡夫卡es这个工具我都能访问对吧？都没问题。

好了，这样的话我们终于就把这些内容都部署完了，非常的繁琐，但是你要知道你确实你在做这件事的时候，是需要经历繁琐的，而且中间会遇到各种各样的坑，如果你自己去做的时候，所以你们在自己尝试的时候，我建议你最好选的操作系统的版本和我一样，最好是3层os7，你跟我的命令都一样，这样的话就不容易出问题。

然后你要知道我们之所以每一个工具只部署一份，是因为我们的服务器是有限的，只有一台服务器没法部署那么多份对吧？实际的项目中可能是实际的环境中可能是他们开的各个工具都会部署，多数都会分布式部署对吧？我们是在这里就没法做到那一步了，因为太耗太耗这个福气了，没办法弄。好了，那么这次课我们这个项目部署就演示完了，这次课我们就进行到这里，咱们下次课再见。

## 8.4 项目总结

---

56-项目总结.mp4

到目前为止我已经把我们这个项目当中所有要演示的内容给你演示完了，那么这次课我们来做一个全面的总结。这里我画了一个图，第一个图反映出了我们这个项目中所开发的绝大部分的功能。

第二个也反映出了我们这个项目中的用到的绝大部分的技术，并且我让这两者之间产生了一个匹配和关联，然后让你能够直观的回顾一下我们这个项目中用哪些技术解决了哪些问题，就整理一下我们的思路。

好，我们先从技术来看，那么我们整个的技术是构建在 Spring boot之上，所以说我把它画到最底下，那么其他的技术都是依托在死人布的之上。那么当然了spring boot其实它不是我们整个技术的核心的东西，它只是起到一个辅助的作用，它能够降低我们其他的技术的使用的难度，仅此而已，它是起到一个帮助的作用。

而我们整个所有的技术的核心其实是 spring，我们所有的技术几乎都是围绕着spring来展开的，那么对于重点的技术你需要关注的内容，那么我画了一个下划线，其他的也是这个意思，然后在死不认不得之上，我们用到了它的很多模块，比如说spring mvc解决了前后端请求处理的问题，或者说它交互的问题，然后我们用到了死不用买betis，然后可以访问数据库。

再者我们还用到了 sprint，然后用它来管理我们项目中的安全层面的内容，管理我们项目中的权限。

当然了这三个技术是构建在使用之上，由使用进行整合的，所以说我把它们画到了spring的上方。那么 spring mvc my bed is还有 secret，我把它们放到了一块去，意思是什么？它这些内容它是我们整个项目的基石，就是说我们这个项目当中几乎绝大部分的请求都是用这些技术来实现的，或者说他们都是基于这些技术的，所以它是我们整个项目的基石。

然后当然我们还用到了其他的技术，这些技术不同的技术解决不同模块不同层面的问题，那么我们看一下其他的技术以及这些基础，他们都分别解决了哪些问题。

首先我们做的是登录模块，以及后面的话我们又做了授权，其实我把它们合并在一起，称之为权限模块或者说权限管理模块，那么我们在开发权限模块的时候自然是用到了这些东西，其他模块也是除此以外，权限模块还用到了spring EMAIL，我们发了邮件对吧？

除此以外，我们还用到了spring mvc当中的拦截器 Intercept，那么这两项技术当中 intercept大家要重点的关注一下，因为它能够拦截所有请求对吧？能够解决一些通用的问题，但凡能够解决通用的问题的这样的技术，那么它涉及的面比较广，它影响了请求比较多，对吧？

所以我们一定要慎重，一定要关注，所以拦截器大家要多注意。然后权限管理的模块，我们主要是开发了这么多功能，注册登录退出，没什么可说的。

然后是状态，就是说我们在每个页面上怎么去显示这个登录用户的用户名头像，然后我们开发了用户的设置的功能，主要是可以上传头像，然后还能修改密码等等，最后还开发了授权你不同的类型的用户登录进来，我能让你访问不同的功能，当然这个是由 Skype来实现授权，然后这一些功能怎么说我们开发是开发完了，在这些功能当中我们还要关注一个比较重点的话题，就是绘画管理，这里我写了艾特绘画管理，你关注它，或者说你复习的时候多去看看这方面的内容，因为我们整个权限管理其实是构建在规划管理的基础之上的，没有规划的话，我们现在开发不出来这样的应用，如果没有会话管理前后端就是后端记不住前端的话，那么我们只能开发出一个类似于看新闻那样的网站，而要长期的交互，或者说连续的交互，要开发这样的逻辑，那就必须有规划管理。

那么关于规划管理你需要了解这个cooking session他们各自能发挥什么样的作用，它们的区别是什么？然后的话还有我们项目中我们后来规避了session不用session，这是为什么？是因为我们考虑分布式部署的时候，那么session可能会有一些问题，当时我也讲了，你需要去回顾一下，然后我们不用session以后我们是怎么去解决这个问题，对吧？我们是把数据存到了res里，另外我们还用到了时代的logo对吧？那么这些细节我总结课上不去讲这么细的东西，因为之前讲过，但是我提到了希望你课后自己去，如果你忘了自己去再看一下，再回顾一下。

好，总之权限管理从这个技术上我们关注一下 intercept，从逻辑上或者从业务上我们重点去关注绘画管理好。

那么然后我们又开发了项目的核心的功能比较多，包括首页、帖子评论，还有私信异常和日志其实都差不多，尤其是首页帖子评论私信其实都是基于crmvc实现的，而异常和日志这个怎么说是用到了一些通用的技术，这个需要重点的关注。

那么从这个业务角度来说，我们在开发这些功能的时候有两个重点，第一个你要关注敏感词怎么过滤的，主要是算法前缀数的结构以及相关的算法。再者你要重点关注事物，因为我们在这里解决了一些事物管理的问题，事物是怎么回事。再一个这个事物的隔离级别又怎么样，对吧？然后我们用 spring，我们是怎么去管理事务的，这些东西你要做一个回顾。

当然了模块我们是基于这些技术实现的，除此以外我们还用到了otherwise控制器的通知，我们用它来统一处理的异常对吧？

另外我们用到了LP学到了LP，用LP统一记录了业务层的日志，当然给人感觉是好像这个功能有点有点薄，然后的话好像我们其他的地方就没有用到LP，但是平时我们听到LP也听到别人老说，所以OB很重要，其实确实很重要，因为其实其他的很多功能它依赖于up，你像事物的底层就基于aop所以这op挺重要的，你要重点的去复习这方面的内容。

再一个就是事物方面的内容好，然后再往后我们开发了怎么说，我这里说的是一个性能相关的模块，对性能要求高的模块，这里包含了一些社交网站中常见的一些功能，包括点赞关注，需要高频的访问，需要高性能，我们是利用release开发这样的功能，但readies不只是能够开发这样的功能，它也能够做出一些更高级的事情，比如说我们统计了网站的UV还有活跃用户对吧？

用reds的特殊的数据类型去做了统计，那么你也需要去回顾。最后网站中有一些地方我们觉得性能不够好怎么办？我们用rise做了缓存，然后提高了它的性能。那么就这个模块我们重点关注的是reds这项技术，详细来说我们关注的是rise的数据结构，Rise有那么多种数据结构，每一种数据结构适合解决什么样的问题，适合混成什么样的数据，这个你要搞得很清楚，你要自己去回顾一下。

好，然后我们又开发了通知模块，其实就一个功能系统通知，那么我们在这里用到了消息队列的一个框架，卡夫卡这个框架它在使用的时候非常的简单，特别简单，那么我们在复习的时候，你关注的不是这个框架的使用，因为谁都会用，只要你学了谁都会用对吧？

我们关注的是这个框架背后的一些通用的原则，就是它的模式，生产者与消费者模式。

我在写这个框架之前，咱们不是自己写了一个自己实现了一个生产者消费者模式，对吧？那么大家把代码也回顾一下，重点回顾与这个模式有关的内容。因为只要你理解了什么是生产者与消费者模式，你才有可能明白我什么时候哪些功能我可以用它来解决这个问题。银行来解决，你要解决的时候，你自然就会想到我就用卡夫卡就可以了。

好这样往后我们开发了这个项目中的搜索功能，当然我们搜索是针对这个帖子能做一个全文的搜索，它各个字段，我们用到的也是一个新的框架，Us take search。

那么这个框架在使用的时候也比较简单，我们在复习的时候关注的是什么呢？关注的其实也是它的数据结构，因为以来stake设施它也要把数据单独存一份，而它存的方式和数据库又不一样，它是以索引的方式来存的，你重点要关注索引的结构，我建议你去找一找相关的文章去看一看 Us take search索引相关的内容。

好，最后我们又开发了其他的一些功能，包括帖子排行，包括上传文件上传到云服务器，再一个我们也对服务器进行了一个本地的缓存，那么我们在开发这些功能的时候，我们基于这样的一些技术，包括cat可以做定时任务，包括咖啡因可以做本地的缓存。

这里咖啡因又画了一个下划线，表示说在众多技术当中它也算是比较重要，因为它能解决缓存的问题，它能够提高我们整个应用的性能。

另外因为它是本地缓存，它在分布式部署的时候它有一定的局限性，你得知道对吧？所以说你要把它和release结合在一起去做一个回顾。然后从业务层面，我们在这里面重点还要去关注线程池，其实就是class它背后的一些原则形成时，然后关注缓存这两个话题。好了，以上是我对整个我们这个项目从技术到这个业务所做的一个总结。

那么还是那句话，就是说我只是大概的帮你梳理一下我们整个的思路，帮你画一画我们整个开发的这些功能所学的这些技术当中的一些关键点和重点，然后让你在复习的时候能够找到一个这个切入点，能够抓住重点去复习去回顾。

而我们每一项内容，我们在课堂上都已经做了很大篇幅的讲解和演示，所以课上我现在我不再跟你再讲一遍了，因为太浪费时间也没有必要好了。这个是我们从业务从技术的角度来说去对这个项目做的一个回顾，我们理解了技术和业务他们之间的一个应用场景，这就说完了。

说完以后其实还不够全面，我再补充一个，我们再换个角度，我们从一个运维的角度，我们整体的了解这个网站架构，这样的话可能你会对这个项目理解的更好一些，更透彻一些。当然有人说从运维角度，那不就是把它部署以后，那个角度你上次课不是已经部署过了吗？我上次课确实是部署过了，但是我上次课是采用一台服务器去部署的，对吧？然后每一个服务我只部署了一个节点，为什么？因为我们资源有限对吧？不可能说那么浪费，搞那么多服务器部署那么多节点没做不到，那么这次课我给大家画一个图给你，演示一下我们在真实的环境下，真实的情况下，我们到底是怎么部署的？

哪个服务它到底是部署了多少个节点，它们之间到底是一个怎么样的关系，把说说的再透彻一点。

你从一个更宏观的角度，从一个网站的角度，而不是从这个代码的角度再去看待这件事儿。那么可能是怎么说你理解了这件事以后，可能你就会理解面试官他所关注什么，你就更能够理解我所画的这些重点都在哪，为什么是这些是重点。好了，要了解网站整体架构，我给大家画一个图，现画提前没有画好。好，首先我们先从这个客户端开始说起，就是我们客户端要访问这个网站，那么我们这个网站去怎么去处理这个请求，然后他一步一步需要调用谁，我们再往后再去演化。

好，首先我先画一个方块，这方面代表客户端，这个客户端它未必是浏览器，也可能是APP对吧？所以说我这里就统称叫客户端。好这是客户端，然后客户端它向这个服务器发请求，其实请求分为两类，一类他请求的是静态资源，像什么css文件，js文件图片这是静态资源，不会有任何变化。

一类是动态请求访问的是controller，要求返回动态数据。那么这两者这两种请求，我们其实在服务端部署的时候，是由不同的服务器去处理的。

好静态和动态，其实我们可以从牛客网去看一下，拿code的卡，咱们就以Logo为例，右键检查，logo它是一个超链接，超链接你看对应的这个是有个样式声明，超链接的背景图片，你看这个图片叫static点脑壳的.com，他是从时代脑壳的.com域名得到的数据，然后的话我们整个网页是脑壳的.com，就是说我们网页和我们网页中所包含的静态资源，它是两个域名或者说它是两个服务器返回的，为什么要搞两个服务器？

其实是为了性能，你要理解好，我就把这个画一下。

首先我先说一下我们的动态资源，是我们服务端部署的 ndx去处理的，因为我们上次都不说了，我们部署的时候是在应用服务器之前部署了一个部署了 ndx然后 ndx可以对服务器做反向代理，如果你有多个服务器它还能做负载均衡，对吧？是这样的。

好，所以说我们在服务器是部署了 ndx然后 ndx其实通常情况下我们会部署两台，部署两个，一个是组，一个是从平时是组服务器处理客户端请求从服务器只是起到一个备份的作用，万一某一刻主服务器挂了，从服务器顶上他是这个意思。

是为了就怕它主服务器发一台服务器挂掉，如果单节点挂掉那就糟糕了，所以说就是部署一个组一个从那只有写上恩杰克斯主，再来一个ndx从。

好，那么当然了我们默认情况下，如果他没挂掉的话，我们肯定是把请求发给 n这个主服务器，他俩其实就是形成了一个小的集群了，好，然后这是动态请求，如果是静态请求不发给他，不由他处理，我们是把静态资源部署到了 cdn的上面去。

就cda缓存服务上面，你这个缓存服务你可以买，比如说阿里云买，然后它的好处是因为缓存服务的提供商，他在全国各地都有这个服务器，那么你把静态资源发给他以后，他会把静态资源部署到全国各地的服务器上去，然后我们用户如果去cdn上加载一个资源的话，它会就近加载，全国都有服务器就近加载，这样速度就快了。

所以为什么我们要把静态资源部署到第三方的服务器上去，因为它有得天独厚的一个优势，这里我就画了三份，比如说这个是多，其实是更多的，我只画三个表示任意多个，所以你看我们在一开始处理资源的时候，就有这样一个巧妙的办法，这是为了提高性能。

好了，那么所有的动态请求都发给了 ngx服务器以后，ngx它是起到了一个负载均衡的作用，那么它会当你把请求分发给某一个应用服务器，好，那么应用服务器我们在真实的部署的时候一定是多台，除非你这个网非常小，只要稍有规模一定是多台，我们这里是多台多个server，我也是画三个表示任意多个server，我们这个server里部署了什么，我们就部署了我们的项目代码，对吧？

我们的项目代码是什么呢？Community当然我们实际是把它改名为 root，但是我们还是按照这个业务名来说，要不然感觉别扭肯定没问题，这是一个。

再一个我们 server里还还有本地缓存，因为我们通过代码给它分配了一些缓存，本地缓存就缓存，因为我们画在server上自然就是server本地的这样一个缓存，就这样。

好，然后 n 这个字我们访问它以后，它会把请求分发给某众多 server 中的一个按照一定的算法去分发的，比如说就发给了 Siri，调用了它里边的事先部署好的代码，这个代码首先它会如果你访问数据，我们是用了缓存，它就会上缓存里去找，看有没有这个数据，它会访问本地的缓存，他不会访问别的思路的缓存，所以他们之间是这样一个关系，再往后然后我们除了 Siri 以外，Siri 要调的东西很多，我们还部署了数据库对吧？

那么他几乎所有的数据来源都是数据库，买 circle 我们通常是部署了两台这个叫 DB 然后往往是要做读写分离的，就是一台数据库负责只负责读，一台数据库只负责写，他们的分工是非常的明确，这是通常的一个做法，r 但是读 dbr 表示读，再来一个 dbw 表示写。

实际上是这样的就是说如果说我这个 server 里的代码访问这个数据库，如果是要写的话，他就会访问写入数据的 DB，然后写入数据的 DB 会把数据写入以后会同步到读取的数据库里，然后我们在获取数据的时候是从数据库里取，总之只往这里写，只从这里读，是这样一个一个逻辑。

好，这是数据库的一个部署的情况，读写分离，这样的话既有备份，然后的话效率也会高，你只往一台数据库里只写，这个数据库只读，那么效率会高一些高很多。

好了，当然了我们应用还会访问的是 radios，那么 radios 我们可以部署任意多个做一个集群，好所以这个 ready 是多份，当然了我们 ready 其实也是起到一个缓存的作用，那么我们本地有缓存 rdc 也有缓存，其实这是形成了一个二级缓存了，那么本地缓存有直接返回本地缓存没有看一下 ready 是有没有，如果有返回效率还是高的，实际上没有再去访问数据库，这样的话避免最大程度的避免请求直接访问数据库，大大的提高了效率好。

然后我们还部署了卡夫卡，那卡夫卡其实在应用当中其实也可以部署任意多个，当然了这个 server 也可以访问其中的某一个来获取相关的数据，然后我们还部署了什么搜索引擎服务器 us take search，这个其实也是可以部署 n 多个服务器访问其中某一个，搜索到相关的数据就可以了。

那么总而言之就是这些服务这样去部署，他能都是为了能够尽可能保证的高性能以及高可用，有一台服务挂了，另一台服务器顶上，那么为什么数据库我们只部署两个，没有部署那么多，因为数据库如果说你跟他们一样做分布式部署的话，那么就要去处理分布式的事物，这个是比较麻烦的。

所以说一般我们尽量避免这种情况，而且通常的话我们业务没有大到那种级别，说需要一定需要做分布式部署，因为你看所有的很多数据，我们都可以从缓存里取，其实从他这里取数据的几率就大大的降低了。

所以按照读写分离的方式已经能够满足绝大部分情况的需要了，就没有必要去做那种集群式就部署了。

当然了还有一个除了这些以外，我们还有一个文件服务器，有人说我没部署文件服务器，其实不用你部署，因为我们文件服务器用的是第三方，我们用的是七牛云，对吧？七牛云也是云服务，它也是有多台服务器的，我们是将数据存到了某一个服务器上，然后从这个服务器去获取，是这样一个关系。

好了，我们最终的部署的方案就是这样的。

这里边 Nds 主从数据库读写分离两台，那么其他的每一个服务包括 server 文件服务器，瑞丽斯卡夫卡一来是 txt，甚至 cdn 都是任意多台，具体是多少就看你的业务量，可能我们最初部署两个，有一个备份就可以了，后来业务发生变化，就是数据量大，再加一个不够再加一个，这就是慢慢演化的一个过程，每个网站都不一样，所以这个没有一个固定的数量，但是我告诉你这些地方可以部署任意多个，所以你看作为网站的建设者，或者说作为你的面试官，他要考虑的是这样的一个问题，他不只要考虑说我这个代码怎么写的问题，他更关注的是这个网站能不能正常的运转起来，能不能很性能很高的运转起来，能不能某一个服务器挂了，我还能够提供继续提供服务，而不至于让整个网站瘫掉，所以他最关注的什么问题？

第一个是性能问题，第二个是可靠性的问题，以及安全的问题，这些是他非常关注的。至于说你把这个功能实现了，把代码写出来，这只是一个最基本的要求，这只是我们建设网站的第一步而已，是吧？所以说那么你再回顾一下我刚才说给你画的那些重点，对吧？其实都是落在了性能上，落在安全这些方面。所以说你在回顾的时候，你把功能层面的你自然是要掌握这些方面，提高性能方面，提高安全性的

方面的内容，你是尤为要重点来关注。

好了，现在我已经从技术的角度，从运维的角度给你介绍了，回顾了我们整个项目的一些内容，最后我再给大家一些建议，你准备面试的时候，你怎么去准备这方面的建议？我是从面试官他要考察的内容的层面给你一些意见，那么你要知道就是面试官他面试你，他希望考察你的什么内容什么方面，我总结的是三个方面，第一个方面面试官想看看你的职业素养，看你能不能具备作为一个程序员的基本素质，这个主要是看你的数据结构算法，看你的基础。

这些内容其实只要是计算机或者是相关专业的，基本上大学我们都学了。说白了面试官想看看什么，他想看看你你大学的时候你是在打游戏搞对象，你还在真的在学习，他当然想找一个真真正正扎扎实实在学习的人，这是他的一个目的。

好，第二个面试官想考察你的一个项目经验，看看你当前你掌握的技能的栈技术栈全不全面，你来到我这个项目组以后，你能不能直接胜任我要求的这份工作？说白了我希望你来了就能干活，而不是说我得找个人带你对吧？因为我找个人带你，你得拖那个人后腿，你整天问他问他拖那人后腿，哪怕就说你不要工资，你也会拖他后腿，他的效率就低了，对吧？是这样一个逻辑。

而你来了就能干活，你就少拖人家后腿，我说这是我们想要的。然后如果说你的职业素养ok，你的项目经验也不错，那么你这两个方面已经传达给面试官以后，你基本上成功的概率就很大了。

但是还有一种情况就是说万一有多个入围者，基本素质都ok，项目经验也都差不多，这时候怎么办？面试官还要看你第三个能力，就是你的钻研能力，他想看看你的长板而不是短板，也就是说你最擅长的领域是什么水平？

我想观察一下你的专业能力，你的独立解决问题的能力，所以往往面试官在面试的时候，他会经挑你就说你哪熟我就问你，我问的尽可能的深，我看你到底水平有多深，就挖到你的底为止，我看你的底到底有多深，为什么想看你的长板？

它背后的逻辑是什么？就是说还是我希望你来了，虽然说你的技能站ok，你能干这个活，但是你干活的时候你肯定会遇到各种各样的问题，我希望你遇到问题的时候，你能独立自主的解决问题，你不要来问我，或者你少来问我，因为我很忙，还是还是这个方面的问题，就是你们不要拖人后腿。

如果说你的长班很长，比方说你的专业能力足够，你平时遇到的问题基本上都能解决，很少会问我，然后哪怕将来有一天说我需要用到引入一个新的技术来解决项目的问题，我让你去研究一下，你也能搞得定，你的对团队的提高价值大了去了。

所以我建议大家从职业素养方面，你的数据结构算法你还是得复习，当然我们这个课不是解决这方面的问题，这个是靠你自己去复习。

第二个项目经验，我们这个项目给你带来了一个我觉得比较可观的一个项目经验了，只要你把它搞扎实了就够了。

然后第三个专业能力，你得看你自己的了，我们有这么多个方面，你可以挑某一个领域某一个方面你做一个深入的学习和研究，你比如说我仔细研究rise，比如说我就仔细研究数据库的读写分离，我就深入研究一下Nx等等，你想要什么都行，当然我去研究使用源码，对吧？

我去研究是不认识快乐，我去研究一级缓存，我去研究Tom cat都可以。

好了，当然了我们不要贪多，就是把你把一个研究透了，研究得非常到位，长板显得非常长，你的专业能力非常的突出，这就够了。然后比如说我研究这俩然后都一般般要好，所以说你挑一个研究的足够透彻，这个是我所建议的。好了，那么这次课我们就从这么几个方面给大家做了一个项目的总结，然后总结完以后只是指导你去做一个复习，希望大家能够按照我的这些个一些指引，你去把你的复习的工作做好。好，我们这次课就讲到这里，咱们下次课再见。

## 8.5 常见面试题

这次课是我们的最后一课，那么这次课我来给大家讲解一些常见的面试题，那么说是面试题，其实我不是真的给你讲题，我是给大家总结一下，那么面试的时候有哪几大块是面试官的比较常问的，那么我把这几块重要的知识点挑出来给大家就是过一遍，然后有些知识点是我们课程中已经讲过的，那么我就是一笔带过在这里，只是做一个提醒的作用，表示说这个东西重要，你课后的话做一个回顾，然后还有一些内容，课程上我们没有讲过，或者说没有详细的讲，那么我会做一些补充是这样的。

然后这些内容你都搞透了，面试的时候无论说他从哪个角度问他是出什么样的方式去考你，那么你明白这个原则基本上就都能够把这些内容给它回答清楚，这是我们这个课的一个目的。

然后我给大家整理的面试的常用的知识点有三个方面，一个是my circle其实问的是挺多的，第二个是reds问的就非常的多。

最后是spring，我们整个的应用是构建在spring之上，当然了他问的也比较多，然后当然了我们这三块的内容合起来，他知识点是特别多的，我这里挑了一部分非常重要的非常高频的给你做一些总结，这里列举了这么多内容，是我一会要讲的内容，我就不一个一个来念了，我们就一个的去去过。

首先我们先来看一下 my circle，那么我们先来说一下 my circle的存储引擎，其实my circle它是一个基于存储引擎的数据库，它有很多个存储引擎可以供我们选择，然后我们选的是in rdb因为自从 my circle5.1开始，它默认的引擎就是inner DB，而从5.1之前是 isam那么为什么要切换成因特DB？因为它好在哪？它有一个比较典型的特点，突出的特点就是你看这个是它知识的内容， Transactions，事物 in the DB是支持事物的，而其他的引擎不支持事物，尤其不支持事物。

你想我们要不要用inner，DB自然是要用的，所以绝大部分的场景我们都会使用 inner DB引擎，所以我们才把它设置为默认的。

当然了还有一个引擎也是支持事物的 ndb但是ndb我们用的不多，因为 nd GB引擎它是在集群当中使用的一个引擎，其他的引擎是不支持集群的。

我们之前讲过部署，就是说我们 my circle我们通常能不做集群就不去做集群，因为如果你做集群以后，你还要考虑就是分布式的事物非常麻烦。

好了，当然了因特DB引擎还有一些其他的特点，比如说它是它支持外建，然后之前的就不支持。

再一个我这里提存储引擎的目的是因为我们一会儿要说事物，要说索引，其实我们都是要基于引擎来说的，因为不同的引擎它机制不一样，而我们又没有必要说把每一个引擎所有的机制都了解到，我们只要了解因特DB它是怎样的方式就可以了，因为我们用它好，总之存储引擎我们就做这么一个大概的了解就可以了。

然后接下来我们看这个事物，我记得我们第三章有一节我们专门讲了事物，包括我们回顾了事物的4个特性，再1个我们重点讲的事物的隔离性，如果说你没隔离好，他会有什么问题，对吧？然后我们在做隔离的时候有4种隔离级别，他们分别各自能解决哪些问题，我做了详细的阐述。

最后我们那节课上还讲了，这个spring他怎么管理事务，这些内容因为我之前的课当中已经讲得非常的详细了，所以说我在这次总结课里就不给大家重新讲一遍了，这里是给你提个醒，这个内容非常重要。

经常问尤其是隔离级别相关的内容，你一定要好好的复习，然后我们那次课也讲了说要实现隔离性怎么办？需要枷锁，而当时我在讲枷锁的时候，锁我们只是一笔带过，并没有讲得很详细，这次课我希望把这个锁给大家讲的是详细一点。

那么 my circle的锁我们从它锁的范围上来说，它主要有两个范围，一个是表及锁，就是锁整个表，那么这种锁它开销小，开销比较小，然后加锁的速度很快，但是它有一个缺点就是发生冲突概率高，然后并发的程度低，然后它不会出现死锁的现象。

然后还有一个是行级锁就是加在一行上，那么它的开销比较大，加锁速度比较慢，然后发生冲突的概率低，病发程度高，有可能出现死锁。那么 isam的引擎它默认的是表级锁，它都不支持行级锁，所以说它的锁的力度是比较粗的，它的并发能力就一般，然后 inner DB它是支持行级锁的，而且它默认就是行级锁，所以它并发能力比较强。

好，然后我们就说一下 inner DB，inner DB它所的一些机制，那么对因特地比来说，它从类型上来划分，分为这么几个类型，一个是共享锁叫s锁，这是行级锁，然后就是在读取一行的时候可以加锁。

第二个是排他锁x锁也是行级的，当我们更新一行的时候需要加锁，也就是说当我更新这一行的时候，加上锁以后，你别人就不能读也不能更新了这样。

然后它还有一种锁叫意向锁，意向共享锁is这是表级锁，那么在我给一个数据加s锁之前，我需要先给这个表加意向共享锁，他的意思是我准备加共享锁，所以再加共享锁之前先加，然后还有意向排他所。

ix也是表级锁，它的意思是准备加排他锁，也就是说我在加排他锁之前需要先加上 x这个锁，好。

当然还有最后1个，一会我们再说，那么这4个锁我画了1个图1个表格，我们来看一下，其中这一列表示的是事物a它的的级别代表是事务二，那就是说我事务一加了一个，所以后事务二能不能加其他的锁，那就是说我事务一加了锁以后，它和事务二的某些锁是冲突的这个意思。

那么如果事务一加了意向共享锁，比如说我想读取这个表里的某一行的数据，这个时候事物二它不能加排他所，因为你加排他所意味着你要写了我正在读，我要读你却正在写，那就冲突了，所以你不能不能写，但是你可以准备读，你可以读，你也可以准备写，这都可以。

所以这个不冲突，那么事物一如果是我加了意向排他锁，事物二你就不能加s和x锁，因为我准备写，这个时候你就不要读了，你也不要写，但是你可以做准备，你可以准备读准备写。

好事物一如果是加的是共享锁，比如说我正在我要读这一行，马上就读这一行，这个时候事务二就不能写，也不能准备写，但是你可以准备读或读，如果事务一加了排他锁事务二什么也不能做，就排他所和任何的行为都是互斥的。

所以通过这么一个例子对比，你就能理解这个锁加上以后别人能干什么不能干什么，就是这样你自己的客户再把这个表格好好的琢磨一下，对比着看一下。

然后除此以外，还有一个锁叫间隙锁叫nnk那么这个锁也是行级锁，它是发生在查询的时候，当我们使用范围条件查询时，比如说ID大于几小于几，我们一查是查多条数据对吧？使用范围条件时我们对范围内不存在的记录加速，比如说我想ID大于100，可能比如说我这个表里有101 102 103，然后没有了，那么会怎么样？

就是这个锁会加给那些不存在的技术，104 105他现在又给加上了，为什么要这样做？

它是为了防止读，如果说你不加锁，你说我查ID大于100的值，然后我正在查的过程中，比如说有人插入了，他为了防止这样的一个读，再一个它是为了满足恢复和复制的一个需要大家恢复的数据的时候，复制的数据的时候也需要用到这样的一个锁。

好了，这是因特DB它的集中锁的一个这个概念，然后我们再说一下我们怎么去加锁，那么首先我们在增加行级锁之前，index会自动给表加意向锁，所以意向锁是自动加的，但这个意向锁是意向共享锁，然后我们在执行。这个意向锁，不是这个意向锁，就意向锁所有可能是意向共享锁也有可能是意向排他所，就统称为意向锁。

然后我们在对某一条数据执行dml语句的时候，那么in the DB会自动的给这个数据加排他锁，所以排他锁是自动加的，意向锁也是自动加的。

那么除了这些语句之外，还有查询语句，我们在执行查询语句的时候默认不加锁，那么如果你想加锁需要用特殊的语法，比如说我想给查到的数据加共享锁，s锁，那么我就需要在 Sql语句的后面加上lock CM的，这样就加上了共享锁，然后如果我想加排他锁，我就在这个查询语句的后面加上for update，就加了排他锁。

然后间隙锁是针对查询而言的，那么上面的查询语句如果我们使用的是范围条件，那么英德DB就会对不存在的记录自动加间隙锁是这么一个原则。好了，最后加锁的话，咱们一开始说引擎的时候也说了，就是说行解锁它有可能可能会出现死锁的现象，那么我们说一下什么情况下会出现死锁以后我们应该怎么办？



首先我说一下这个场景，比如说事务一他做了这样的事情，他更新一张表，t表条件的是ID等于1，先更新了ID等于一的这行记录，然后再更新ID等于二的记录，而事务二的操作与它相反，先更新ID二，再更新ID一，有可能有这样的情况就非常巧，失误先执行一半，更新了ide的数据。与此同时失误二更新了ID v二的数据，它俩是与此同时发生的，然后这个时候会怎么样？因为你这是一个dm二语句，Ide就被事物e加锁，idea就被事物二加速，这个时候往再往后事物一想，更新idea一看它被别人锁住了，他就等着，而事物二又想更新ID，——看也被别人锁住，他就等着那他俩就互等。

如果说没有一个人让步的话，这就一直等下去就死循环了，这就是死锁的一个场景，那么这个解决方案其实也好办，就是说通常情况下因特尔DB它比较聪明，它能够自动检测到这种行为，而且它能够自动处理，它能够让一个事物回本，另一个事物继续所以通常它会自动解决。

再一个我们要注意就是说需要设置参数 in the DB lock, wait time out, 设置一下等待时间，如果一个事物它在执行的过程中超过这个时间，我就自动让它回滚，因为有可能是死锁，也有可能是出现一些其他的问题，通过这个可以做一个兜底的这么一个方案，但是你要注意其实出现死锁很多时候是我们的逻辑写的有缺陷，你最好避免这样的逻辑，这完全是可以避免的。

我们更多的应该是避免写出这样的代码，怎么避免就是说我们在不同的业务访问并发访问多个表的时候，我们应该约定你最好以相同的顺序来访这些表，你不要换着来访问。首先表的顺序你要以相同的顺序来访问。

第二个当我们以批量的方式处理数据的时候，你应该先对数据排序，然后保证各个线程按照一个固定的顺序来处理数据，就是保证顺序一样，你不要混乱的顺序就有可能容易思索。

然后第三个，那么我们在事故中如果要更新记录，你最好就直接申请足够级别的锁，也就是排他锁，你上来就申请排他所，而不是说你先申请个共享锁再转换成排他锁，这样的话也有助于避免死锁的现象。

好了，以上是关于死锁的一些说明，那么刚才我们说的是my circle它的锁的机制，那么买circle锁加的这种锁，我们通通的称其为悲观锁，当然不只是买这种数据库，加的锁都是悲观锁，就是他天生他认为这一定会出问题，我一定得先加锁，他是持一个悲观的态度。

那么如果说我们不用数据库的锁机制，我们自己实现也可以我们自己实现，我们可以实现，用乐观者的方式自定义乐观锁，那么乐观锁通常有两种实现机制，一个是利用版本号，这个麻烦在哪？

我们每一张表都得加一个word版本，你得改这个表，而且每个circle都得把 Word带上，他对于蛇口的表的侵入性比较强，但是它确实很有效，或者说它效率比较高，乐观所天生我认为这不会发生什么问题，但一旦发生问题我又能解决。

当然悲观所的话就是什么？如果你更新数据很频繁，你最好就悲观所如果你根据数据不频繁查询数据的情况多，你乐观锁效率就高，是这样的。

好，我们再看一下这个版本号的机制，就是说我给这个表加了版本号以后，那么我在更新数据的时候，我要把版本号加一，然后更新，这是必须的，每次更新版本号必须加一，然后 Where条件里不管之前条件带上什么，我都需要加，再加上版本号这个条件word等于word，原先参数问的是什么呢？

是我之前我在更新之前先查到这个数据，然后去改，改完之后再更新，我之前查到的 word，也就是说我刚才查到的word，比如是二，现在word把表里的word也等于2，这个条件匹配我就能更新成功。

如果我这个word刚才我查到的是二，但你现在的表里的数据变成了三，就意味着别人改过了，产生了并发的冲突，这个时候条件不匹配，更新失败，整个更新语句最后的反馈结果是个0，对吧？失败了那一失败我就知道这有冲突了，就能避免这个问题，很有效。

那么除了这个以外，还有一种常见的机制叫做cas算法，它是一组单词的缩写，叫compare and swap比较和替换，然后这种算法它是一种无锁的算法，就是我们利用一个无锁的方式来实现一个有所的感觉。

那么这个算法它底层涉及了三个操作数，三个数，一个是内存值v一个是旧的值a一个是新的值b然后算法的基本原则就是这样的，当你内存里的数据等于旧的值a的时候，我才采用原子原子方式，然后用b的值就是新的值去更新v就是说我内存里的值和旧的值相等的时候，我再去更新 a否则更新 v否则我就不更新了，是这个意思。

好，然后算法它通常采用的是自选的操作，所以叫自选锁。什么叫自选操作？就是说我a线程对这个数据加了锁，我b线程一看你加锁了，我怎么办？我不是阻塞的等待，而是循环着等我循环一次看，解没解锁没解锁，我再循环一次看解没解锁没解锁，它是不断的循环去等着你解锁，这种不断的循环等着你解锁，因为循环中间是几乎没有时间间隔的，它不是说我等10毫秒，它是连续没有时间间隔的，所以它效率很高，你那边解锁的我立刻就能感知到。

而反过来如果你阻塞，比如说我阻塞个两秒还是多少秒，可能是或者我每隔100毫秒看一下，可能中间的间隔比较大，效率就没有循环的方式那么快，是这样的。

但是 Cas算法的这种这种自选锁它也有明显的缺点，这缺点你要了解，第一个是 ABA的问题就是说如果我一个线程将 a的值改成了b然后我又改回了a那么对CAS会认为这a没有被改过。

它是这样的，就是说因为它是判断你值变没变，你这个值没变我才去更新的，但其实你变了，但又变回来了，他认为你没改过，他照样去更新，所以他可能会被误导，他可能会被欺骗，这是 ABA的问题。

然后第二个就是自旋操作，刚才说了它采用循环的方式来实现的，万一说某一个线程它对数据加的锁的时间比较长，处理时间非常长，这个时候我另外一个线程在那循环等它可能会循环好多次，这个循环没有间隔，会给CPU带来很大的压力很大的开销，CPU就受不了。

最后一个就是说 CS它这种算法只能保证一个共享变量的原子操作，如果有多个共享变量不好办。好，这是关于我们如何实现乐观锁的一个介绍。那么关于 My circle，最后我们再说一下它的索引，这也是非常常见的一个问题。如果说事物是为了保证数据的一致性完整性，它是为了保证数据的安全的这么方面的一个话题，所以是为了提高查询的效率，所以这两方面就是面试官最关注的问题。

然后这里我们这里面我们也是说说 inner DB这个引擎，inner DB引擎索引用大家应该都会用都非常简单，然后的话当然你可以测一下，你要测索引的效率高低的话，你需要数据多，比如说你要造个100 200万的数据，至少的效果可能会比较明显，然后你不加索引去用字段去搜，加索引再去搜，你去看一下它这个时间差别效果是比较明显的，我们课上就不去做这个事了，我只是跟你说一下索引背后的原理。

Inner DB引擎它是采用 b加速的算法来实现的，索引我们主要说一下 b加数这种数据结构。那么我概括了几句话，首先说我们在存索引的时候，这个数据是分块存的，而每一块一叫一页数据，然后索引的所有值，它是按照顺序存储的，是有序的，并且每一个叶子到根的距离是相同的，这会这个一会有图，咱们看看图会更直观一点。

那么非叶子节点存的是数据的边界，而叶子节点存的是指向数据行的指针，然后通过边界可以缩小数据的范围，从而避免全表扫描，加快查找的速度。说白了索引其实就跟我们读书那书的目录一样，我想读每一块我通过目录一找，很快就定位到它在哪一页，我上那一页去找，就这段内容这不就快了对吧？就这样的。

然后其实跟类似然后我们看这个图，刚才说了索引是分块存的，这是一块一页，这是一块一页这是一页这是一页。

那么这些黄色的部分它是整棵树的末级，是叶子节点，所以它叫叶子叶，那么叶里存的是指向实际数据的指针，比如说我们是用户表，我是在邮政类目上加了索引，这个叶子节点会存的是一个指针指向整个某一个优质的那一行的数据的指针，而它的叶子节点的上层是对数据的一个key，或者说它是对数据的一个索引，然后叶子节点之间是有联系的，就是每一页数据它和另外一页数据之间是先后首尾相连的，首尾相连的，然后的话叶当中的这些数据是经过排序以后的数据了，那么它们连在一起相当于整张表的数据，在这里按照所有的字段它就做了一个自然的排序，整个表都在这一级做了一个排序，按照索引的字段，然后一上一层是对整个下一层的一个边界的一个索引，或者说是下一层的边界的限制，比如说我

一看我要查的数据是小于这个  $t_1$  的，这个地方它有一个指针指向了这一页，比如说我这个数据是在这一页当中的，我就上这一页里去去找去遍历这一页的数据找到我想要的值。

比如说我一看我的索引，我的数据也是位于这个范围之内，是位于  $k_1$  和  $k_2$  之间的，我这个数据就是相当于位于这一页。所以上一层的页里的数据，我们通过它是很快速的能够找到我的数据的，具体在哪一页，具体是哪一个，就是底层的页，然后找到这一页去编列这一页就可以了。

这样的话我就不说我去遍历整个表里所有的数据，避免对整个表进行遍历，这样的话就通过这样的方式就很快找到这一块，提高了效率。然后你还要注意上面上层的节点，它不是一个它可能是多个，而且可能还有更高的层次，这个数的深度，这个是看表的数据量而言的。表的数据量少的话可能有2层3层，再多的话可能有4层甚至更多的层次，这都是有可能的。

好了，总之这就是 B+ 树的一个数据的结构，大家把这个结构要做一个了解。

好了，那么 my circle 的这些方面的知识点我们就说完了，下面我们再说一下，首先 readiness 我们重点要关注它的数据类型，当然了那数据类型我们在项目中大部分都用过，其实这个好说。

然后的话我们在大学的时候也学过这个数据类型，所以说基本上比较容易理解。那么你除了要知道，所以说 ready 它支持哪些数据类型之外，你还要知道每一种类型它的长度是多少，它最多能存多少个数据，这是面试的时候他可能比较常问的一个话题。

这里我做了一个总结，我们首先把这个数据分为 key 和 value，对吧？Key 也说一下，这 Key 就是字符串那么最大 512 整，然后只有这么多君西等等，那么使用字符串 512 兆，然后 list set 都是能存多少数据？

是二的 22 的 32 次方减 1 这么多数据，那么它的单位是这么多个数据，这么多个兼职队，这么多个数据，这么多个无序数据，然后有序集合官方手册里没有写它到底最大的限制是多少，这里我也没有写。

没有不确定就是不确定。

然后 beta map 他其实本质上它就是字符串，所以说它存的数据最大也是 512 兆，而害怕 log 它是最大 12k 最坏的情况是 12k 那么总之 readies 它支持的数据类型比较丰富，功能比较强大，那么你需要关注的是每一种类型它最多能存多少数据，我们在使用的时候要知道它的内存的使用情况以及它的一个极限，这样的话在某些情况下能够可能规避一些问题。

好，第二个我们来介绍一下 radius 的过期策略就是说 radius 如果我们对 key 做了过期的设置了过期的时间的时候，那么他会把 key 放到一个单独的字典里，然后在 key 过期时，它不会立刻把 key 删掉，因为他如果每个 k 都看着过期，马上马上就删掉，效率低，然后它会通过两种策略来删除过期的 k 第一种策略叫做惰性删除，就是说你过期就过期，我还在那放着我不删。

当客户端访问在访问 key 的时候，那么 Redis 总会检查一下 key 过没过期，如果过期就删了，这样的。好，但是如果这样的情况下就会出现一个问题，有可能有一些 key 老也不会访问，他一直就不被删除这样的，key 慢慢累积的多了，对内存是一个很大的消耗，对吧？肯定不能一直这样下去。

那么所以它还有第二种机制叫做定期扫描，那么它默认会每一秒执行 10 次过期扫描次数可以配在配置文件里，它有一个 Hz 的选项通过它来配，然后具体的扫描的策略是这样的，他每次扫描的时候从过期字典中选，随机选 20 个 k 然后把这 20 个 k 当中已经过期的 k 给它删了。

然后再判断一下你过去的 k 的比例超没超过 25%，如果超过 25% 就认为他就认为什么？你字典里过期的比例已经很高了，我得赶紧再删一下，那就赶紧再执行，再重复执行步骤一再来一遍，这是它的过期的策略。

除此以外它还有一种淘汰策略，淘汰策略指的是什么呢？当 readiness 它实际占有的内存已经超出最大的限制的时候，我怎么把这个数据删掉一些，因为你超出限制了，哪怕没过期我也存不下别的数据了。

最大的限制我们可以通过 max memory 选项去设置，然后如下这么几种策略，我们可以通过 Max memory policy 指定你要用哪一种策略。

好了，总之我们选择任何一种策略的目的都是为了让 `riders` 它丢弃一些数据，或者说淘汰一些数据好，腾出新的空间我们继续玩下去，继续提供读写服务。他所提供的淘汰数据的策略有这么多，第一个这是默认的他可能对可能导致增大内存的命令，直接返回一个错误的消息。

比如说我要往 `Reds` 里继续写数据，他认为你看我内存已经达到最大值了，你再写不下去了我就报错，大多数的写命令他都这样处理，但是得利的处罚，因为得利的是把数据删了，他允许好，这是默认的方式就是这样。

然后第二个是这个方式是指的是对于设置了过期时间的 `key`，那么我会选择剩余寿命比较短的可以把它淘汰，因为剩余寿命比较短，就说明你快要过期了，我现在存不下了，你就提前退休，提前把你删掉了这样的，那么剩余寿命叫 `TTL` 所以说它策略的后缀 `ttl` 然后是在设置过期时间的 `key` 当中，他会选择最少使用的 `key`，将其淘汰。

那么最少使用的 `key` 它怎么选择最少使用 `k` 它用一个 `lru` 算法，我们一会再说，这个也非常合理，就是说我内存已经满了，然后你那些设置的过期时间的 `k` 当中，我不管你时间新和旧，只要你用的频率最少的 `k` 说明你不怎么用，我就干脆把它淘汰也很合理。

然后还有一个这个是在设置了过期时间的 `k` 当中，我们随机选一些可以把它淘汰，不管你是时间新旧，也不管你用的多与少随机。

好，那么还有两个这个是其实这两个和这两个是类似的，只不过他选择的范围不一样 `all case` 所有的 `key` 是在所有的 `k` 当中选，最少使用的也是利用这个算法把它淘汰，而这个是在过期的 `k` 中选，这个是在所有的 `k` 中选，随机选一些把它淘汰，这个是在过期的 `k` 里选，所以是很像的。

好，整个这些策略当中，一个是涉及到 `ttl` 选择剩余寿命这个好办，因为我们是给 `k` 设置了过期时间，你直接看时间就知道剩余寿命对吧？这也没有什么需要特殊的算法。

我们如何确定一个 `t` 是否是最少使用还是最多使用，这里涉及到一个 `lru` 算法，这个算法你要知道官方手册上也有说明，这里我做了一个提炼，什么叫 `lru` 算法呢？

就是说这个算法我们是通过一个链表去统计，你是使用的多还是少的，我维护一个链表，然后我原来按照顺序存储访问过的 `key`，那么当我访问一个数据的时候，那么我会把刚刚访问的 `key` 移到表头，比如说数据我访问过它就在链表里，它处于中间的位置，我刚才又访问它一遍，我就把 `k` 从中间移到表头去，那么如果是我访问一个新数据，我就直接把它放到表头。

总而言之我通过这样的操作，就是新的 `key` 挪到表头，我就不断的往表头放，新的 `key` 挪新的 `key`，最后表尾剩的一定就是最少访问的 `key`，那么我淘汰的时候就从表尾去淘汰就可以了，所以是用一个链表去维护 `k` 使用的新与旧。然后但是 `release` 它不是直接采用原始的 `lru` 算法，它是自己做了一些变通，我们称之为近似 `lru` 算法。

这个逻辑是借鉴了 `lru` 效果也和你差不多，但是逻辑实际实现的方式就变了，它是这样实现的，但是每一个 `k` 给 `k` 维护一个时间戳，我淘汰的时候随机选5个 `k` 然后我从中淘汰到最旧的 `k` 因为时间戳我就知道谁最旧淘汰掉一个，如果内存还是超出限制，我再继续这样的操作，再随机一遍再淘汰。

有人说这样有什么好处，它的好处是这种方式。特别 `lru` 节约内存，因为 `lru` 你会把所有的 `key` 都存一下，反馈的 `key` 都存一下，它很占内存的，但我这种方式它比较节约内存，而效果和 `Lru` 非常的相似。右边这个图你看这个图就是 `ll` 算法的一个图绿的，比方说比较新的数据，深灰的就表示比较旧的数据，浅灰的已经被淘汰的数据，最终是这样一个效果。

然后其他的三个图是 `Release` 不同的版本，用这个算法能达到的一个效果。我们就看这个3.0，这是比较新的一个效果，而且它是每次选5选10个数据，那么它的效果和这个图已经非常的接近了，已经非常的能够满足我们的需求了。所以说这就是瑞丽斯他采用了一个近似 `lru` 算法来实现的这么一个淘汰策略，然后其他两个版本我们就不说了。

好，那么因为 `rides` 我们是内在内存中存数据，我们是缓存数据，所以我们非常关注缓存它释放的及不及时，清理的及不及时，这是第一个。

第二个我们还关注什么？缓存失效的问题，如果说你缓存一下子失效了，那会造成一个很恶性的后果就是说我请求就直接抵达了数据库，存储层，对数据库造成过大的压力，可能把数据库给压瘫痪了，数据库一瘫痪，我们整个应用就歇菜了。

好，那么我们在使用ready的时候，怎么说我们要避免三种与缓存相关的问题，我们来说一下。

第一个是缓存的穿透问题，什么叫缓存的穿透场景就是这样的，说有的人他故意使坏，黑客故意想黑你，他访问的时候你看我可以直接敲个路径访问这个网站，比如说我敲discus，然后斜线写一个帖子ID，如果我写这个帖子ID是-1的话，我根本这个数据就故意查不存在的数据，你release里一定没有这个数据，这个时候我就请求直接就访问了数据库，我把这种请求搞得多一点，你对你数据库就造成很大的压力，有可能就把你搞死搞死机了，就是这个意思，这里边画了一个图，就是客户端故意查，没有的数据-1，访问缓存的时候肯定查不到miss，然后的话又去访问存储层，其实也是miss存储层就会给你返回一个空的结果0的结果，我再来一次又返回零0，很频繁，数据库就被这样玩死了，怎么解决其实也很好办。

第一个办法缓存空对象，如果说我存储层没有命中数据miss了以后，我仍然把空值存到缓存里，你下次再访问的时候直接给你返回一个空值缓存给你返回，你别访问数据库了，就在这个地方我访问数据库的时候查不到我不要return吗？Return之前我先把这个数据空值也存到这个缓存里，当你下次再访问这个没有的数据，直接就给你返回空值，别走数据库就好了。

然后还有第二种办法是不能过滤器，这是release自带的一个过滤器，那么它可以将所有存在的key，我们需要把所有的key提前存到过滤器里，然后把过滤器部署在缓存层之前，在这儿在缓存之前加上过滤器里存了所有已知的key，那么它会起到一个拦截的作用，如果请求防缓存层先经过过滤器，然后过滤器判断一下你这个请求的key有没有，如果没有的话，直接就给你返回一个空值就完了，就连缓存都不让你访问了。

这样好这是缓存，穿透它的特点是我访问一个根本就不存在的数据，别管是缓存还是数据层，没有。

第二种情况是缓存击穿，场景是这样的，我有一份热点数据访问访问得非常的频繁，访问量还很大，然后那热点数据我们肯定会存到redis缓存里，对吧？

但是你要注意，如果说我们在失效的那一瞬间，因为我们不是设置了过期时间，它失效的那一刻，那么大量请求一看缓存里没有数据，他们一下就涌到了存储层上面去，如果这个请求特别多，上千上万，有可能就把数据库搞崩溃了。

那一份热点数据有可能非常量大，有可能出现这种情况，那么这种情况我们怎么办？主要两种办法，一个是加互斥锁，就是说你对数据访问的时候，对redis里数据访问的时候，我给你访问加速，当某一个线程访问这个数据的时候，它就加了锁，其他线程就只能等待你就等一会再访问，而这个线程访问过以后，你不是数据失效了吗？

我访问的时候没有数据，那就访问数据库，访问数据库得到数据以后缓存被更新对吧？或者说缓存被重建这个时候我访问完了，我把手释放了，其他线程它就可以从火车里取值了，就可以避免同时抵达数据库的这么一个问题。

然后第二种办法是永不过期，我不给这样热点数据设置过期时间，你就不会出现上述的问题，那么这是物理上的不过期。

其实还有一种层面，我们也可以给每个value设置一个给value，不是给key给value设计一个逻辑的时间，然后我们有一个线程去检查这个东西，一旦发现一个值逻辑上是过期了，我单独启动一个线程去重建缓存去把缓存更新掉，这也可以好。

这是缓存击穿。

然后第三个还有一种场景是缓存雪崩，雪崩可能是由于某种原因redis缓存不能再提供服务了，这个时候请求就直接访问了数据库，有可能造成存储层的宕机，这种方式就是整个缓存层整个整体不可用，这叫缓存雪崩。

刚才说的缓存击穿是一个热点数据，是某一个数据，而这个是大批量的数据，这个范围不一样。那么这种情况下我们怎么解决问题？第一个你要避免同时过期，就是说缓存成不能提供服务的未必是ready是挂了，有可能是你有很多key在同一时刻失效了，对吧？这个时候访问这么多k的那些请求，直接就到达了数据访问层，所以我们应该避免同时过期。

就是说我们给 Key设置过期时间的时候，你不要10秒20秒30秒，这样很容易就凑到一块去，整数就过期了，你最好是我在10秒20秒一分钟2分钟之外，再给它加一个随机数，这样的话每个k的过期时间它肯定是不一样的，错开了就好一点。

好，第二个我们在构建瑞丽斯缓存的时候，你应该构建高可用的reduce缓存，就是说你部署多个实例做集群，那么一个节点挂了，其他的节点还能用，避免说出现这种情况高可用。

然后第三个万一高可用也不行，你最好是构建多级缓存，你把本地缓存加上去，我们这样的话本地缓存加ready缓存，两级缓存对吧？多一层屏障，那么你就可以降低请求直达存储层的一个几率，这也是一种办法。

再有一个如果说本地缓存也不行对吧？Redis它也挂了，请求就是要去访问数据库，这个时候怎么办？我们也不能干，等着我们对数据库也要启用限流和降降级的措施。对存储层增加限流措施有对应的框架，你去利用它去限流。

然后当超出这个限制以后，其他的流量怎么办？我们就也不是不给他提供服务，我们给他提供降级的服务，降级你自己定义，但通常其实就是给他返回一个默认值，返回一个空值，或者是给他返回一个错误的信息，这样的话其实比如说一共有5000个请求来，我只能处理2000个，这2000个我还正常处理，那3000个我给你返回空值或者是返回的错误消息，但我整体服务还在运转着，我没有挂掉它总比挂掉强。

所以以上是我们缓存雪崩的一些方案。

当然了就是说我刚才所说的这些缓存的问题，这解决方案只是一个热门的方案，大家公认的比较有效的方案，其实也有可能有一些其他的方案，你也可以自己想一些方案，这都可以，那么这只是给你做一个就是参考或者是准备。

那么关于redis你还要了解一个内容，就是分布式锁这个话题，面试时间问的是极多的，因为怎么说，我们通常的互联网的项目都是分布式不足，绕不开这个话题。

首先你要理解什么是分布式锁，这个场景是这样的，就是我们修改数据的时候，我们通常设备需要要把这个数据先读取到内存里，读到内存里干什么可能要做个判断，你数据合不合适，你比如说我要买东西，我读这个数据，我看一下它的库存够不够，对吧？

这样的，那么我首先要将数据读到内存里，然后改了，存回去往往是这样的一个操作。

那么这个操作我们是单体应用，没什么问题，但分布式应用就有可能什么多个进程同时做这样的事情。因为你分布式应用server有多个，你有可能是多个server同时做这样的事情，同时改同一份数据，读同一份数据然后要改对吧？然后你但是你要知道读和改这两个操作它不是原子操作，所以多个进程同时做这样的事情，就可能会产生不一致的现象，就会产生冲突。

分布式锁是为了解决这类问题，而分布式锁它的基本原理是什么？其实和我们Java中的同步锁原理是差不多的，我们回顾一下同步锁，同步锁其实就是说我们解决多个线程访问同一个数据的问题怎么办？

我就加锁，所谓加锁我在这多个线程都能访问到的一个地方，其实就是堆内存打一个标记说当前这个数据归属是谁哪个县城，你别人就别碰了，就这个意思。

好，这就是同步锁的原理，而分布式锁原理跟它一样，但分布式锁是进程是多个进程，你要加锁你要打标记，你需要把标记放到多个进程都能访问到的地方，那就不能放到这个内存里了。你放到servo一的内存里，servo二防不了servo一的内存对吧？怎么办？具体解决方案有三种方式，我们可以把数据存到数据库里来，实现分布式锁，我把锁存到数据库里，你多个server可以访问同一个数据库对吧？

就解决这个问题，放到他们都能够得到的地方，或者是我把这个锁放到rids里也可以，或者说我把这个锁放到组keep也行，三种主流的方案，那么通常使用rides实现的比较多，所以我们是把这个内容放到 Rise里来讲，或者也经常是我们问readiness的问题的时候，愿意问分布式说，其实分布式说不是瑞丽斯独有的一个能力，好，我们使用rise实现分布式锁的原则是什么呢？

有三个原则，第一个我们要考虑安全属性，锁必须是互斥的，在任意时刻只有一个客户端能持有锁，你不能两个人都有锁那不行。

第二个是考虑它的 a活性b活性a指的是你需要无死锁，就哪怕说你有锁的客户端崩溃了，或者网络被分裂了，其实就是断网了，那么这个锁仍然可以被获取到，不然的话你说我加了锁，然后我挂了，这个时候不就死锁了对吧？

好，活性b指的是它的容错的一个特性，就是说只要大部分的risk节点都活着，那么客户端就可以获取和释放锁，并不是说百%的，因为分布式的系统很有可能某一个节点挂掉，这是很正常的。所以说它只要大部分都活着就可以继续，这是它的三个原则。

好，再往下看，那么具体来说我们怎么去利用rise来实现分布式锁分两种场景，一种场景是我们单个reds实例，我们怎么实现？当然我们通常都是多个实例，但是你要理解这个单个实例，因为多个实例也是在一个基础上进一步实现的。

单个实例它实现分布式锁，首先它需要它加锁的时候，或者说它获取锁的时候使用这样一个命令set，然后 resource name其实就是key。

My random value其实就是一个值，但这个值通常建议是随机的，你每一个server产生的随机值都不一样，后面 nx是not exist，不存在，它的意思是只有在 k不存在时它才成功，如果k已存在，比方说别人用了 k别人已经加了锁，我就失败。

然后 px后面跟的是一个时间，只是说设置锁的自动过期时间，万一说你挂了，我们有一个自动过期时间的机制让能够解锁这个意思。

好，然后我们主动的解锁你加锁以后，当你执行完你的操作以后，你要把所释放好，让别人去积蓄对吧？一定要有释放所的一个行为，或者说把锁删掉的一个行为就删掉这个数据。

释放所 Reds官方建议是用卢瓦脚本去执行写这么一个脚本，我获取 k判断它和我之前的 k是不是相等，如果是相等，我就把它删了，就说白了如果是我的t我就把它删了，这个意思好。有人说为什么要用卢瓦去删呢？我们赛特的值我们完全可以通过release命令去删，那么主要是为了避免这样一个问题，它可以避免删除别的客户端，获取成功的锁，什么意思？

就是说如果我们用命令reds命令来删锁，可能会出现这样的情况，你看 AA服务a客户端，我对数据加了锁加锁以后它阻塞了，然后在它阻塞的过程当中锁超时了，自动被释放了，注意它是阻塞，它没有死掉，被自动释放了。

你所一释放以后是不是b这个客户端也可以就可以加锁了，他就加上了，加上以后这个时候a很AA从这个阻塞的状态恢复回来了，他恢复以后他下一步要干嘛，他要释放锁他就释放锁，然后但是他其实之前已经被自动释放了，他又执行了一遍释放锁的代码，结果把b的锁给释放了，所以说就有这样一种情况，就是我释放锁，我把别人的锁给释放了，那么用这个撸脚本不走 rise客户端，那么可以规避这样的问题，所以说利用这个办法会更好一点。

好了，那么如果是多个rise实例，怎么实现这个分布式锁，其实也是要利用刚才的命令，但是他对命令又做了封装，又有了一些新的要求，那么这个要求称之为 read log算法，这是 reads的作者又发明了一个算法，那么这个算法它有现成的实现，不用我们自己去写，那么Java版本的实现叫ready，如果你要用的话就下载这个包，然后这个算法大概的原则或者步骤是5步，第一步我想获取当前的 UNIX时间，就是从1970年到现在的的一个毫秒为单位，然后我依次尝试，因为是分布式多个rise实例，我依次尝试从n个实例使用相同的k和随机值，获取锁就串联锁，然后一定要注意你要设置一个响应超时时间，因为你有n个 release实例，有可能哪一个不给你响应怎么办？

你要设置超时时间，如果说我服务器没有在规定时间内得到响应，我怎么办？我赶紧尝试去访问另外一个 ready 的实例，我就不用傻等着。我们搜房 ready 获取锁的时候要挨个获取，要判断响应时间，如果超时我就下一个，这个是它的原则。

然后第三个就是客户端使用当前时间减去开始获取锁的时间，这就是开始获取锁的时间，我客户端其实就是 server 对于 riders 而言，server 去访问它，server 就是客户端使用当前时间减去开始获取锁的时间，那会得到一个获取所使用的时间，我获取所也是需要消耗一些时间的，然后当期仅当大多数的 release 节点都取到了所一，多半取到了所，并且使用的时间小于失效的时间，锁失效的时间，这锁我才算是取得成功了。

你看就分布式是非常麻烦，我要考虑的是多个节点，大部分成功的情况才算成功，你要这样去协调。当然了你得一多半节点我都拿到了锁我才认，这是第一个前提。

第二个前提我拿到锁这个时间小于所失效的时间，如果我拿到锁以后这锁就失效了，那就白拿了对吧？所以这两点都满足，我才能够认为枷锁是成功的。那么如果说我得到了锁，而 key 的这个时候我就可以用锁了，Key 它真正有效的时间我们注意是等于什么？是等于它原本的你设置的有效时间减去你获取锁的时间，你要有一些损失的，而如果你获取锁失败怎么办？你就需要对所有的 release 实例解锁，因为你刚才获取锁其实就是枷锁，如果一旦失败了，你要把那些锁都解掉，对所有的 ready 的实力实行执行。

解锁的代码，你别管他加没加锁都要去进行解锁，因为你也不知道哪个成功哪个失败这样。

好了，那么刚才呢我们是解释的是 Rise 相关的一些知识，总之其实数据库层面的面试问的是比较多的，数据库，一个官营数据库，my circle 一个就是 reds，这两个问的都比较多，然后我们整个项目的根基在于 spring，spring 也问的比较多，但是 spring 它内容比较多，重点是在于三个方面，一个是 IOC 因为所有的功能都构建在 LC 之上，第二个是 aop 那么 AOC 解决不了的问题，aop 可以解决，而很多的功能又构建在 aop 之上。

第三个是 spring mvc，因为我们外部项目关键就这儿。好，那么首先说 IOC 其实 IOC 我在第一章有一节专门讲了，其实讲的也比较详细了，这里我不再重新讲一遍，但是我做一个补充的是并的作用域。我当时讲的时候说讲 b 是有作用域的，我们默认是单立的，我们可以通过 scope 注解去改它，我可以改成 product type。

所以大家对于单立的作用域 product type，应该是明白，我们之前讲过默认是单立的，但是因为当时我们还没有学习外部开发，所以我就没有讲到外部应用当中这些作用域，其实还有别的跟针对外部应用的作用域，有 request，级别的作用域，就是说我对于某一个病人而言，我每次请求都给他建一个新的实例，就是 request，级别，塞神级别、作用域是针对某一个病，每个会话我给他建一个实例，global session 是全局的 session，给他建立一个实例，但其实现在不用了，这只有在普特赖特这样的应用中才有效，不用了。

然后还有 application 表示整个应用为整个外部应用建一个实例。

这个大家了解一下，其实其他的这些个中医我们几乎不怎么用，几乎不用，我们往往用的最多的一般都是心口疼，一般都是默认，你看我们那么多功能，我们哪个去改了对吧？但是面试时他可能会了解的就会说到这个话题，但是你只知道这两个就有点有点少有点薄，所以说其他的你也了解一下就好了。

那么 Sebring 第二个重点是 epopee，aop 我之前在第三章好像是最后一节，也详细的讲过，所以这里也不再进行重复，但是我想告诉你的是什么？就是 aop 在使用上它很简单的，但是在理解上就比较麻烦，所以说你要吃透它的每一个术语，你要理解这些组件之间的关系，你要理解说你要理解它的原理。所以当时我画了这么一个图，大家把这个图各个环节之间的关系，各个术语把它吃透，你做一个回顾就好了。

最后是 spring mvc，面试官很关注你说你到底能不能做外部项目，你做外部项目，你整个流程你能不能衔接起来？通过哪能看你对整个流程熟不熟，通过问 Super mvc 的，刘成你是不是 mvc 的流程熟了，至于说你 CTRL Diao servis service Diao deo 那个是简单的，这就好办了。



好，这里我又画了一个图，把整个spring mvc它的核心组件做了一个串接，把它们的顺序串在一起，让你从一个更彻底的一个角度去理解 surprise，分析它处理请求的过程。当然了，如果你在 spm分析底层的代码里打断点，也可以跟到这个顺序，我建议你去跟一下。

整个spring mvc的核心是despatcher serval eyot组件之前我们也说过，所以当客户端发出请求访问服务器的时候，访问这个super mvc的时候，由谁来处理这个请求由他来处理，然后它会调用这么一系列的组件来解决这个问题，顺序是我标的这个步骤。

首先第二步，那么他会先调用组件叫handle卖品映射，组件能够根据你的访问路径找到这个路径，处理这个路径的组件，就说白了找到 CTRL了，然后他会给你返回给 dispatch server的返回对象叫handler excuse。

这里边封装了能够解决这个请求的control。

除此以外，还封装了能够对请求实现拦截的拦截器，就这里包装的是拦截器加CTRL a给了despatcher servitor，然后 dispatcher serverlet就去从这里得到comptroller得到拦截器去掉。第五步他去调谁，他不是直接调CTRL了，他是调handler的adapter，CTRL是封装在这里的，他从嵌里得到这个adapter去掉，然后 adapter里边内部掉了controller这样的。

二 this picture在调组件之前是先调所有得到的拦截器的，play很多方法，依次调拦截器，列表的plan的方法，在此之前 handle can handle adapter里边掉了了以后CTRL了最终会返回一个mode and view，把 mode and view返回给dispatch。

那么再返回 modle之后，dispatcher会调一下poste，handle方法调拦截器的post handle方法，所以就在这儿 dispatch，把拦截器controller调完了，然后最终得到了model and view，结果这里面封装了模型数据，封装了视图，然后 dispatcher调组件will reserve，视图解析器，他把 model and will给他，让他去处理数据的显示，而view reserve会根据 view找到对应的模板引擎，模板给你把数据给模板，然后由模板引擎向客户端做渲染，做展现，整个流程就是这样。

而模板引擎在向客户端输出数据以后，dispatcher又会去调拦截器的after completion这个方法。这就是整个 Spring mvc的处理请求的过程，这里边什么时候调拦截器的代码，什么时候去调CTRL，什么时候去调，这个模板顺序就一目了然了，如果你跟一下的话，这个效果可能会更好一点。

好了，我们这次课就讲这么多方面的内容，然后我们整个课就到此告一段落就结束了。

然后我们课程虽然结束了，但是课后如果说大家在学习的时候，你有问题你继续在这个群里去问，或者你在这个我们课程后面不是有个讨论区吗？去问咱们再去进一步的去交流。然后在这个课程即将结束的时候怎么说，也预祝大家能够找到一个自己比较满意的理想的工作，也希望我们这个课能够帮助到你。好了，我们这个课到此就讲解完了，咱们跟大家说一声，再见，咱们后会有期，拜拜。