## 反向代理：

先说正向代理，比如要访问youtube,但是不能直接访问，只能先找个翻墙软件，通过翻墙软件才能访问youtube. 翻墙软件就叫做正向代理。  
所谓的反向代理，指的是用户要访问youtube,但是youtube悄悄地把这个请求交给bilibili来做，那么bilibili就是反向代理了。  
在当前教程指的就是访问nginx,但是nginx把请求交给tomcat来做。

正向代理：用户 🡪 vpn(代理) 🡪 youtube

反向代理：用户 🡪 youtube（目标）🡪 但是YouTube把这个请求给了B站，让B站接受请求，呈现内容。

我们现在使用了nginx后，发请求是发送给nginx，但是nginx会把请求转发给tomcat

## 配置代理

配置代理的地方：

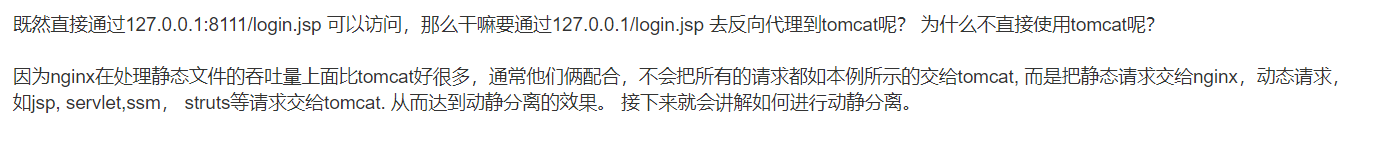


表示所有访问nginx的请求都交给127.0.0.1:8111这个端口的tomcat去做。

现在再去访问nginx的地址127.0.0.1的话，会直接跳到设置的tomcat地址：

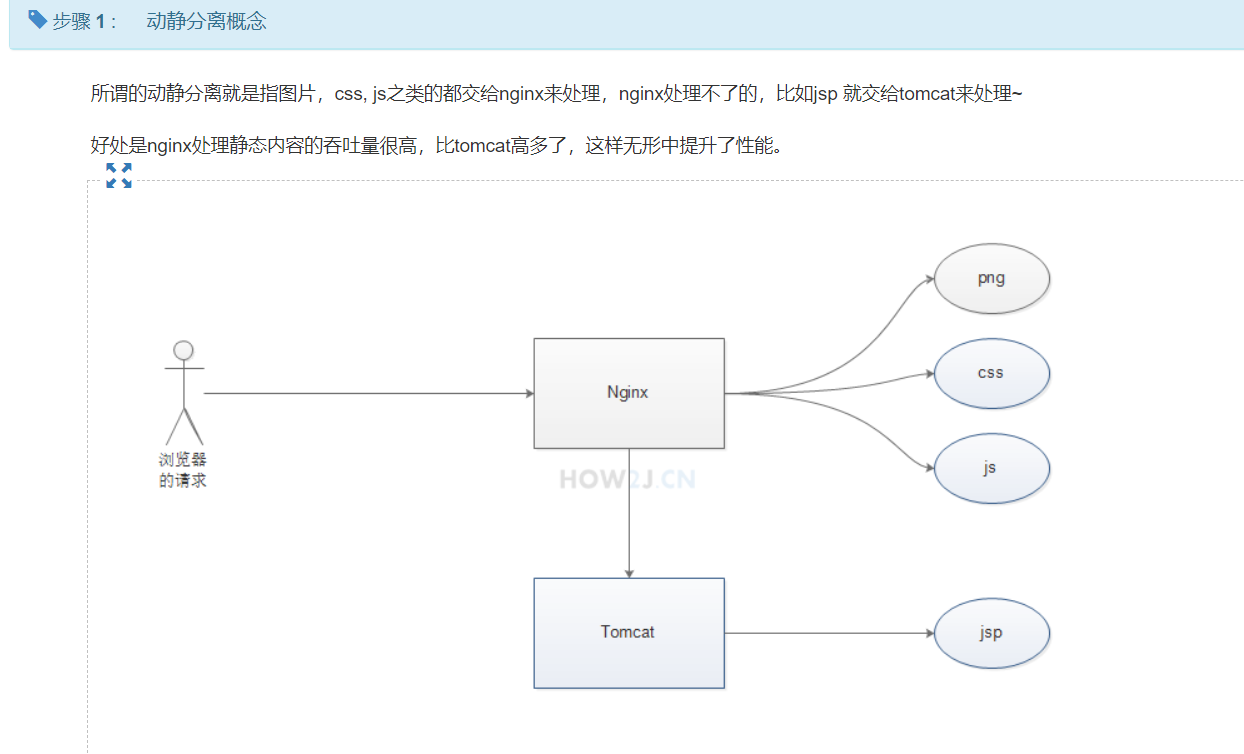


## 动静分离，配置nginx的意义：



1. 动静分离

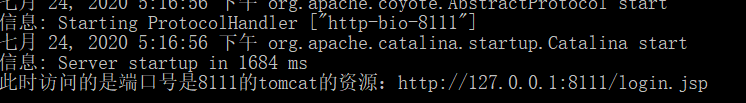
动态资源和静态资源的分离，会使web应用的加载效率提高很多，让nginx管理静态资源的吞吐，tomcat管动态资源的分配，这样“双线程”加载返回数据会很高效。



配置好nginx管理静态资源：



之后重启tomcat和nginx，就可以看到tomcat的log中，请求只访问了jsp文件了，其他的js,css,png资源请求都是nginx处理的

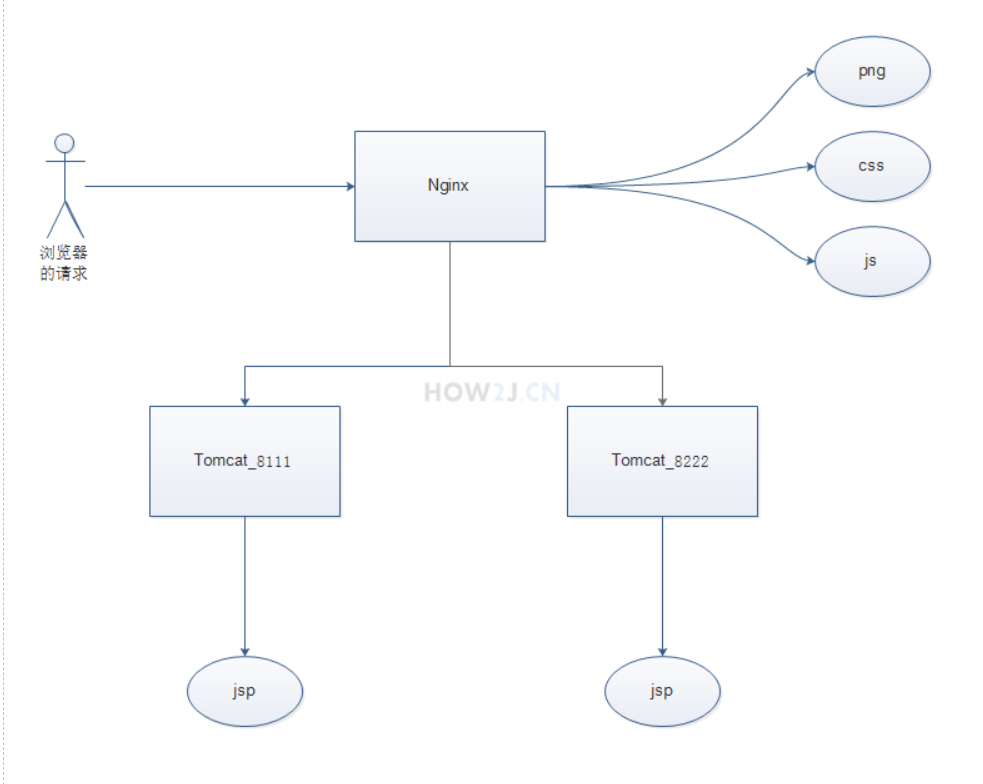


## 负载均衡：

顾名思义，访问量大的时候就要配置多个tomcat“替身”来分担访问压力。

原文：

负载均衡的概念就是当访问量很大的时候，一个 Tomcat 吃不消了，这时候就准备多个 Tomcat，由Nginx按照权重来对请求进行分配，从而缓解单独一个Tomcat受到的压力



配置一个tomcat地址群对象，表示参与nginx分配的多个tomcat地址:



多次访问页面时就会出现两个tomcat都会有反应：



## 5.Session共享