## 1.安装rabbitmq

### $ docker pull rabbitmq

## 2.安装 rabbit带有管理界面的管理工具

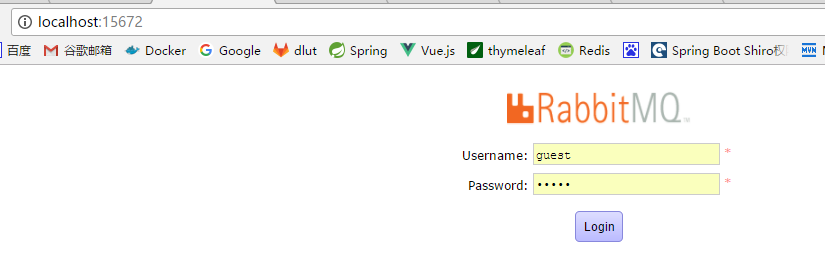
### $ docker pull rabbitmq:3-management

## 3.运行这个容器

**$ docker run -d -p 5672:5672 -p 15672:15672 rabbitmq:3-management**

### 3.1解释5672是消息代理的端口，15672是web管理界面的端口 （后面是容器的端口，前面试docker映射到本地的端口）

### 3.2测试 浏览器打开 <http://localhost:15672/>（用户名和密码默认是guest）



# springboot项目

## 1.spring boot 默认我rabbitmq主机为localhost 端口为5672 ，所以没有必要再properties中配置连接信息了

## 2.启动文件中代码

@SpringBootApplication  
**public class** Ch935Application **implements** CommandLineRunner{  
 *//spring boot自动为我们配置好的rabbit模板* @Autowired  
 RabbitTemplate **rabbitTemplate**; *//1* **public static void** main(String[] args) {  
 SpringApplication.*run*(Ch935Application.**class**, args);  
 }  
  
 *//定义目的地（队列） 名称为my-queue* @Bean *//2* **public** Queue wiselyQueue(){  
 **return new** Queue(**"my-queue"**);  
 }  
   
 *//向队列my-quequ发送消息* @Override  
 **public void** run(String... args) **throws** Exception {  
 **rabbitTemplate**.convertAndSend(**"my-queue"**, **"来自RabbitMQ的问候"**); *//3* }

## 3.消息监听

@Component  
**public class** Receiver {  
  
 @RabbitListener(queues = **"my-queue"**)  
 **public void** receiveMessage(String message) {  
 System.***out***.println(**"Received <"** + message + **">"**);  
 }  
  
}

4.