

## 20、将springboot服务---->打包成jar---->构建成镜像---->运行镜像

1、创建一个springboot的demo服务，编写一个helloworld程序，能正常访问即可

The screenshot shows an IDE with a project named 'demo' at 'C:\Users\Administrator\Desktop\demo'. The project structure includes 'src/main/java/com/example/demo/controller' with 'HelloController' and 'DemoApplication'. The 'HelloController' code is as follows:

```
1 package com.example.demo.controller;
2
3 import org.springframework.web.bind.annotation.RequestMapping;
4 import org.springframework.web.bind.annotation.RestController;
5
6 @RestController
7 public class HelloController {
8
9     @RequestMapping("/hello")
10    public String hello(){
11        return "hello,kuangshen";
12    }
13
14 }
15
```

The console output shows the application starting successfully on port 8080:

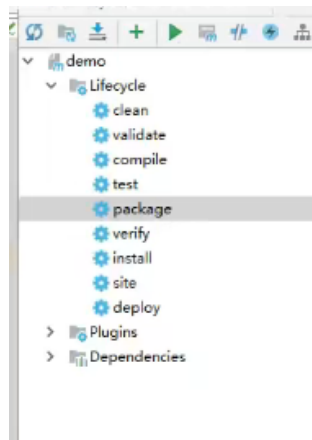
```
Run: DemoApplication
Console
2020-05-15 23:53:58.431 INFO 8340 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.34]
2020-05-15 23:53:58.432 INFO 8340 --- [main] o.a.catalina.core.AprLifecycleListener : Loaded APR based Apache Tomcat Native library [1.2.23] using APR ver
2020-05-15 23:53:58.432 INFO 8340 --- [main] o.a.catalina.core.AprLifecycleListener : APR capabilities: IPv6 [true], sendfile [true], accept filters [fals
2020-05-15 23:53:58.432 INFO 8340 --- [main] o.a.catalina.core.AprLifecycleListener : APR/OpenSSL configuration: useAprConnector [false], useOpenSSL [true
2020-05-15 23:53:58.448 INFO 8340 --- [main] o.a.catalina.core.AprLifecycleListener : OpenSSL successfully initialized [OpenSSL 1.1.1c 28 May 2019]
2020-05-15 23:53:58.533 INFO 8340 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2020-05-15 23:53:58.533 INFO 8340 --- [main] o.s.web.context.ContextLoader : Root WebApplicationContext: initialization completed in 938 ms
2020-05-15 23:53:58.686 INFO 8340 --- [main] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService 'applicationTaskExecutor'
2020-05-15 23:53:58.856 INFO 8340 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
2020-05-15 23:53:58.858 INFO 8340 --- [main] com.example.demo.DemoApplication : Started DemoApplication in 2.036 seconds (JVM running for 3.595)
```

http://localhost:8080/hello

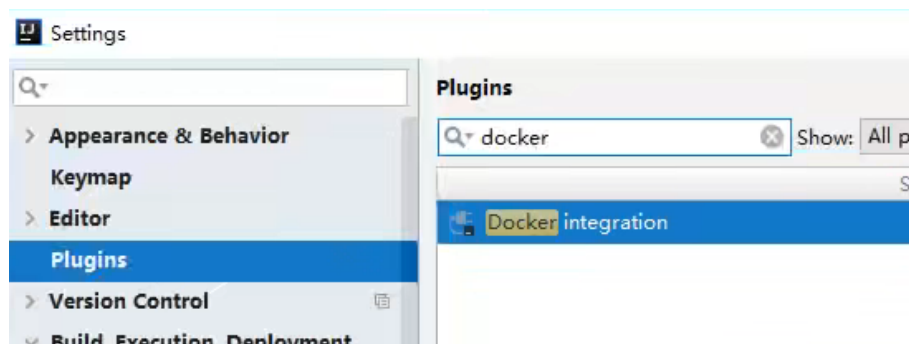
收藏 狂神说 基金 官网 工具 文章

hello,kuangshen

## 2、将demo服务打包成jar



## 3、安装docker插件



## 4、编写好Dockerfile脚本文件，并将打好的jar和Dockerfile文件一块上传到linux的创建好的/home/idea目录下

demo C:\Users\Administrator\Desktop\demo

- > .idea
- > .mvn
- > src
- > target
  - .gitignore
  - demo.iml
  - demo-0.0.1-SNAPSHOT.jar
  - Dockerfile
  - HELP.md
  - mvnw
  - mvnw.cmd
  - pom.xml
- > External Libraries
- > Scratches and Consoles

1 FROM java:8

2

3 COPY \*.jar /app.jar

4

5 CMD ["--server.port=8080"]

6

7 EXPOSE 8080

8

9 ENTRYPOINT ["java", "-jar", "/app.jar"]

将target中打好的jar拷贝出来和Dockerfile文件同目录，只是为了好一块上传到linux服务器

将和Dockerfile同目录的jar复制到linux的根目录下，起名叫app.jar

```
[root@kuangshen idea]# ls
demo-0.0.1-SNAPSHOT.jar Dockerfile
[root@kuangshen idea]# ll
total 17228
-rw-r--r-- 1 root root 17634295 May 15 23:58 demo-0.0.1-SNAPSHOT.jar
-rw-r--r-- 1 root root      120 May 15 23:58 Dockerfile
[root@kuangshen idea]#
```

5、通过docker命令执行Dockerfile文件，将jar包构建成为docker镜像（kuangshen666）

```
[root@kuangshen idea]# docker build -t kuangshen666 .
Sending build context to Docker daemon 17.64MB
Step 1/5 : FROM java:8
8: Pulling from library/java
5040bd298390: Pull complete
fce5728aad85: Pull complete
76610ec20bf5: Pull complete
60170fec2151: Pull complete
e98f73de8f0d: Pull complete
11f7af24ed9c: Pull complete
49e2d6393f32: Pull complete
bb9cdec9c7f3: Pull complete
Digest: sha256:c1ff613e8ba25833d2e1940da0940c3824f03f802c449f3d1815a66b7f8c0e9d
Status: Downloaded newer image for java:8
--> d23bdf5b1b1b
Step 2/5 : COPY *.jar /app.jar
--> 45204313a0d8
Step 3/5 : CMD ["--server.port=8080"]
--> Running in b4427fd7ffd7
Removing intermediate container b4427fd7ffd7
--> dc70e3c8e0dc
Step 4/5 : EXPOSE 8080
--> Running in 466ca222eeb2
Removing intermediate container 466ca222eeb2
--> cd9692a255dd
Step 5/5 : ENTRYPOINT ["java","-jar","/app.jar"]
--> Running in 689ale241071
Removing intermediate container 689ale241071
--> 7a60f84797b1
Successfully built 7a60f84797b1
Successfully tagged kuangshen666:latest
[root@kuangshen idea]#
```

#### 6、通过镜像运行出容器，然后访问容器服务的hello接口

- 给别人的是一个镜像，而不是jar包，别人要用的话直接运行镜像即可，即以后一般是将jar构建为镜像，发布到公有云镜像仓库中，供别人拉取，运行镜像即可。

```
[root@kuangshen idea]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
kuangshen666        latest             7a60f84797b1       5 seconds ago      661MB
tomcat               latest             d03312117bb0       2 days ago         647MB
redis                5.0.9-alpine3.11  3661c84ee9d0       2 weeks ago        29.8MB
java                 8                 d23bdf5b1b1b       3 years ago        643MB
[root@kuangshen idea]# docker run -d -P --name kuangshen-springboot-web kuangshen666
611c89108c6964271261e658aea2dbf2a9049f39657fcaaaabab14ea6356c5f2
[root@kuangshen idea]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
611c89108c69        kuangshen666       "java -jar /app.jar ..." 8 seconds ago       Up 7 seconds        0.0.0.0:32779->8080
/tcp kuangshen-springboot-web
[root@kuangshen idea]# curl localhost:32779
{"timestamp":"2020-05-15T16:00:45.370+0000","status":404,"error":"Not Found","message":"No message available","path":"/"}[ro
ot@kuangshen idea]# curl localhost:32779/hello
hello,kuangshen[root@kuangshen idea]#
```