**DubboDemo（直连）测试代码介绍；**

1，首先，了解dubbo直连的过程原理：

      就是provider提供一个服务器地址，consumer直接调用已经指定的服务器地址，即可直连成功；不需要注册中心注册地址；

2，环境搭配：

    系统：win10，  IDEA，java 8, Maven :  apache-maven-3.5.3,  tomcat 7:   apache-tomcat-7.0.86

环境变量配置可以参考前面一篇 ” windows10环境搭载 “ 文章；

   检查一下，环境配置的是否正确，可以用cmd命令行测试：

   mvn -version 回车，查看版本号，和jdk版本号；

3， 创建两个项目：DubboDemoProvider和DubboDemoConsumer两个项目，

**DubboDemoProvider项目：**

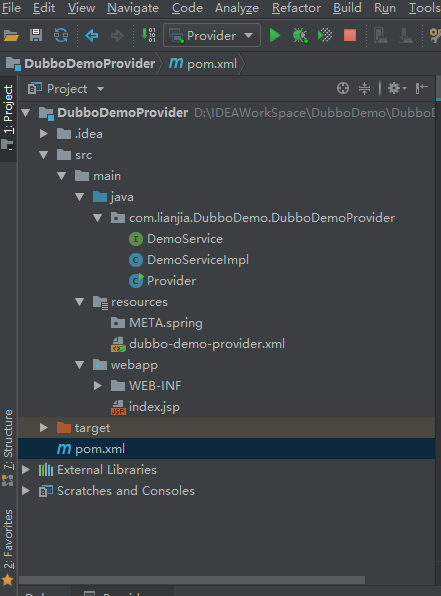
**（1）** 创建一个maven项目，设置配置文件如下：

          需要在maven工程文件pom.xml中加入dubbo的依赖，dubbo版本号选2.5.8.

<?xml version="1.0" encoding="UTF-8"?>  
  
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
  
 <groupId>DubboDemoProvider</groupId>  
 <artifactId>DubboDemoProvider</artifactId>  
 <version>1.0-SNAPSHOT</version>  
 <packaging>war</packaging>  
  
 <name>DubboDemoProvider Maven Webapp</name>  
 <!-- FIXME change it to the project's website -->  
 <url>http://www.example.com</url>  
  
 <properties>  
 <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
 <maven.compiler.source>1.7</maven.compiler.source>  
 <maven.compiler.target>1.7</maven.compiler.target>  
 </properties>  
  
 <dependencies>  
 <dependency>  
 <groupId>junit</groupId>  
 <artifactId>junit</artifactId>  
 <version>4.11</version>  
 <scope>test</scope>  
 </dependency>  
 <!--添加dubbo依赖-->  
 <dependency>  
 <groupId>com.alibaba</groupId>  
 <artifactId>dubbo</artifactId>  
 <version>2.5.8</version>  
 </dependency>  
 <!--添加springframework依赖-->  
 <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-context</artifactId>  
 <version>4.3.7.RELEASE  
 </version>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-core</artifactId>  
 <version>4.3.7.RELEASE</version>  
 </dependency>  
  
 </dependencies>  
  
 <build>  
 <finalName>DubboDemoProvider</finalName>  
 <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->  
 <plugins>  
 <plugin>  
 <artifactId>maven-clean-plugin</artifactId>  
 <version>3.0.0</version>  
 </plugin>  
 <!-- see http://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin\_bindings\_for\_war\_packaging -->  
 <plugin>  
 <artifactId>maven-resources-plugin</artifactId>  
 <version>3.0.2</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-compiler-plugin</artifactId>  
 <version>3.7.0</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-surefire-plugin</artifactId>  
 <version>2.20.1</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-war-plugin</artifactId>  
 <version>3.2.0</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-install-plugin</artifactId>  
 <version>2.5.2</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-deploy-plugin</artifactId>  
 <version>2.8.2</version>  
 </plugin>  
 </plugins>  
 </pluginManagement>  
 </build>  
</project>

注意这里的spring版本号的问题要是4.3.7的！！！然后自动添加依赖，更新索引版本号；

（2）然后，创建各种目录结构；如下图所示；



注意：上述的目录结构要一致，比如：java要设置resource ;  resources要设置resources;  webapp先不用管；

在src/main/java/目录下创建包名：叫做com.lianjia.DubboDemo.DubboDemoProvider此时的包名要与DubboDemoConsumer的包名一致；

（3），然后在包下创建一个接口DemoService和一个实现类DemoServiceImpl ，两个文件的代码如下：

DemoService.java

package com.lianjia.DubboDemo.DubboDemoProvider;  
  
/\*\*  
 \* @Aauthor:  
 \* @Date:2018/5/1113:58  
 \* @Description:  
 \*/  
public interface DemoService {  
  
 String HelloWorld();  
  
}

DemoServiceImpl.class

package com.lianjia.DubboDemo.DubboDemoProvider;  
  
import org.springframework.stereotype.Service;  
  
/\*\*  
 \* @Aauthor:  
 \* @Date:2018/5/1114:00  
 \* @Description:  
 \*/  
@Service  
public class DemoServiceImpl implements DemoService {  
  
 @Override  
 public String HelloWorld() {  
 return "HelloWorld";  
 }  
}

（4），然后创建一个Provider类；代码如下：

package com.lianjia.DubboDemo.DubboDemoProvider;  
  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
  
import java.io.IOException;  
  
/\*\*  
 \* @Aauthor:  
 \* @Date:2018/5/1114:03  
 \* @Description:  
 \*/  
public class Provider {  
 public static void main(String[] args) throws IOException  
  
 {  
 ClassPathXmlApplicationContext context = new ClassPathXmlApplicationContext(  
 new String[]{"dubbo-demo-provider.xml"}  
 );  
 context.start();  
 System.in.read();  
 }  
}

（5）dubbo-spring服务配置文件：  dubbo-demo-provider.xml   放在resources 目录下；

<?xml version="1.0" encoding="UTF-8"?>  
  
<beans xmlns="http://www.springframework.org/schema/beans"  
  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  
 xmlns:dubbo="http://code.alibabatech.com/schema/dubbo"  
  
 xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd http://code.alibabatech.com/schema/dubbo http://code.alibabatech.com/schema/dubbo/dubbo.xsd">  
  
 <dubbo:application name="demo-provider"/>  
  
 <dubbo:registry address="N/A"/>  
  
 <dubbo:protocol name="dubbo" port="20880"/>  
  
 <dubbo:service interface="com.lianjia.DubboDemo.DubboDemoProvider.DemoService" ref="demoService"/>  
  
 <bean id="demoService" class="com.lianjia.DubboDemo.DubboDemoProvider.DemoServiceImpl"/>  
  
</beans>

注意上面的小细节：

<dubbo:registry address="N/A"/>这个部分，直连的address是"N/A", 而连接zookeeper的时候要写服务器地址；

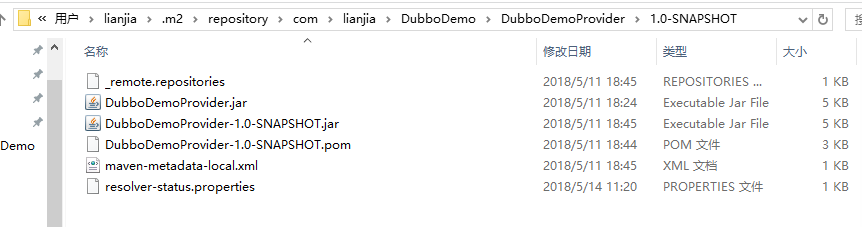
记住端口号：20880；

interface和class要找到对应的路径；

最后将配置好的DubboDemoProvider项目用maven打包好，命名规则：包名+版本号；

放在C:\Users\lianjia\.m2\repository\com\lianjia\DubboDemo\DubboDemoProvider\1.0-SNAPSHOT这个路径下：

如下图所示：



**DubboDemoConsumer项目：**

**（1）** 创建一个maven项目，设置配置文件如下：

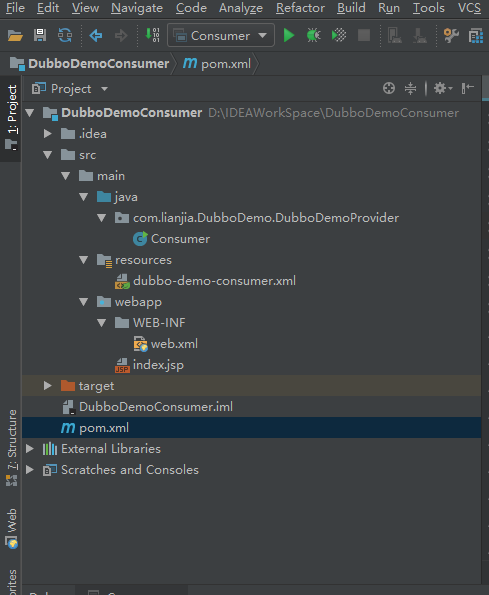
          需要在maven工程文件pom.xml中加入dubbo的依赖，dubbo版本号选2.5.8.

         此外注意的是，要增加对dubbo和DubboDemoProvider工程的依赖；

代码如下：

<?xml version="1.0" encoding="UTF-8"?>  
  
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
  
 <groupId>DubboDemoConsumer</groupId>  
 <artifactId>DubboDemoConsumer</artifactId>  
 <version>1.0-SNAPSHOT</version>  
 <packaging>war</packaging>  
  
 <name>DubboDemoConsumer Maven Webapp</name>  
 <!-- FIXME change it to the project's website -->  
 <url>http://www.example.com</url>  
  
 <properties>  
 <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
 <maven.compiler.source>1.7</maven.compiler.source>  
 <maven.compiler.target>1.7</maven.compiler.target>  
 </properties>  
  
 <dependencies>  
 <dependency>  
 <groupId>junit</groupId>  
 <artifactId>junit</artifactId>  
 <version>4.11</version>  
 <scope>test</scope>  
 </dependency>  
 <!--添加dubbo依赖-->  
 <dependency>  
 <groupId>com.alibaba</groupId>  
 <artifactId>dubbo</artifactId>  
 <version>2.5.8</version>  
 </dependency>  
 <!--添加provider的依赖-->  
 <dependency>  
 <groupId>com.lianjia.DubboDemo</groupId>  
 <artifactId>DubboDemoProvider</artifactId>  
 <version>1.0-SNAPSHOT</version>  
 </dependency>  
 <!--添加springframework依赖-->  
 <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-context</artifactId>  
 <version>4.3.7.RELEASE  
 </version>  
 </dependency>  
  
 </dependencies>  
  
 <build>  
 <finalName>DubboDemoConsumer</finalName>  
 <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->  
 <plugins>  
 <plugin>  
 <artifactId>maven-clean-plugin</artifactId>  
 <version>3.0.0</version>  
 </plugin>  
 <!-- see http://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin\_bindings\_for\_war\_packaging -->  
 <plugin>  
 <artifactId>maven-resources-plugin</artifactId>  
 <version>3.0.2</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-compiler-plugin</artifactId>  
 <version>3.7.0</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-surefire-plugin</artifactId>  
 <version>2.20.1</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-war-plugin</artifactId>  
 <version>3.2.0</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-install-plugin</artifactId>  
 <version>2.5.2</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-deploy-plugin</artifactId>  
 <version>2.8.2</version>  
 </plugin>  
 </plugins>  
 </pluginManagement>  
 </build>  
</project>

（2）然后，创建各种目录结构；如下图所示；



注意：上述的目录结构要一致，同Provider一样，比如：java要设置resource ;  resources要设置resources;  webapp先不用管；

在src/main/java/目录下创建包名：叫做com.lianjia.DubboDemo.DubboDemoProvider此时的包名要与DubboDemoProvider的包名一致；

将打包好的jar的路径也要和上述的路径保持一致；

（3），然后在包下创建一个类Consumer，文件的代码如下：

package com.lianjia.DubboDemo.DubboDemoProvider;  
  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
  
/\*\*  
 \* @Aauthor:  
 \* @Date:2018/5/1114:47  
 \* @Description:  
 \*/  
public class Consumer {  
 public static void main(String[] args) {  
 ClassPathXmlApplicationContext context = new ClassPathXmlApplicationContext(  
 new String[]{"dubbo-demo-consumer.xml"}  
 );  
 context.start();  
 DemoService demoService = (DemoService) context.getBean("demoService");  
  
 // execute remote invocation  
  
 String Result = demoService.HelloWorld();  
  
 // show the result  
  
 System.out.println(Result);  
 }  
}

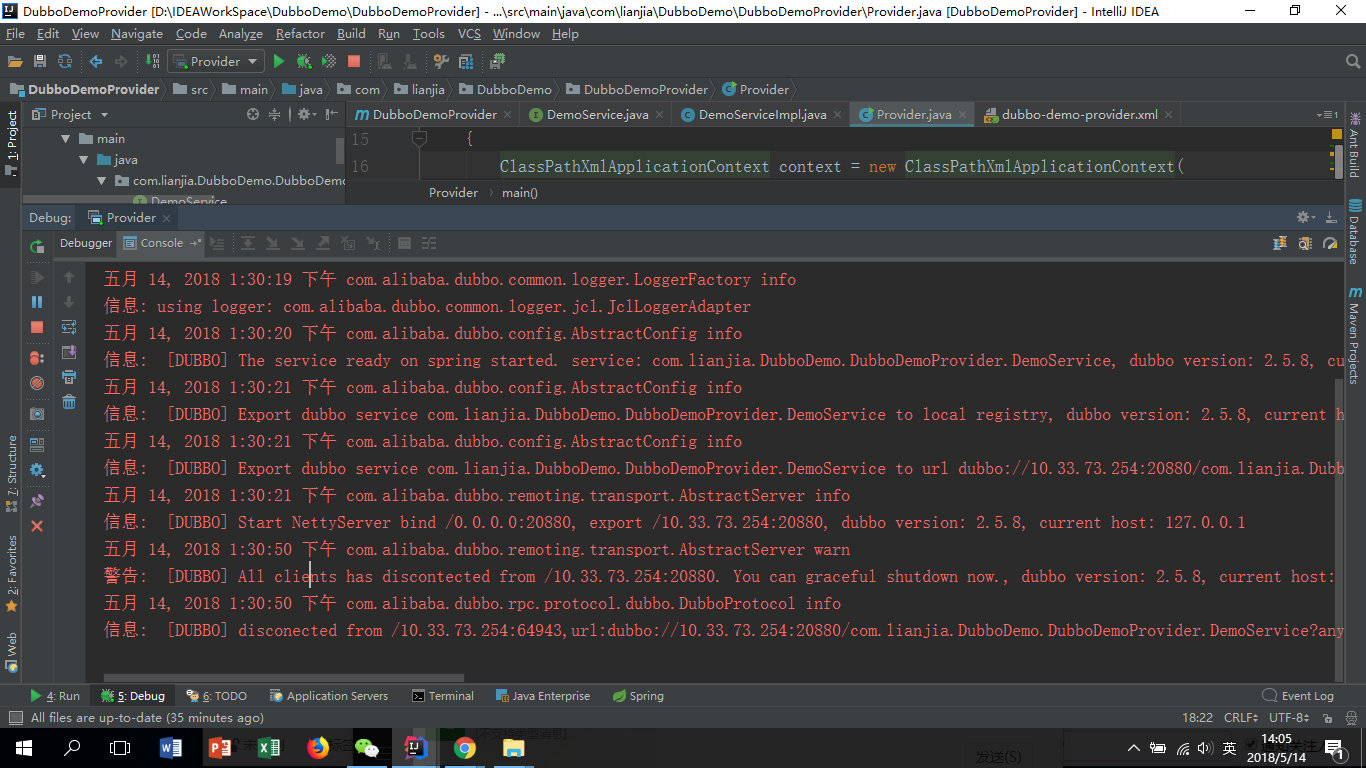
（4）spring服务配置文件 dubbo-demo-consumer.xml  放到resources下：

<?xml version="1.0" encoding="UTF-8"?>  
  
<beans xmlns="http://www.springframework.org/schema/beans"  
  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  
 xmlns:dubbo="http://code.alibabatech.com/schema/dubbo"  
  
 xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd http://code.alibabatech.com/schema/dubbo http://code.alibabatech.com/schema/dubbo/dubbo.xsd">  
  
 <dubbo:application name="demo-consumer"/>  
  
 <!-- <dubbo:registry address="multicast://224.5.6.7:1234"/>-->  
  
 <dubbo:reference id="demoService" interface="com.lianjia.DubboDemo.DubboDemoProvider.DemoService" url="dubbo://10.33.73.254:20880"/>  
  
</beans>

4，启动项目测试！

  （1）启动项目：DubboDemoProvider

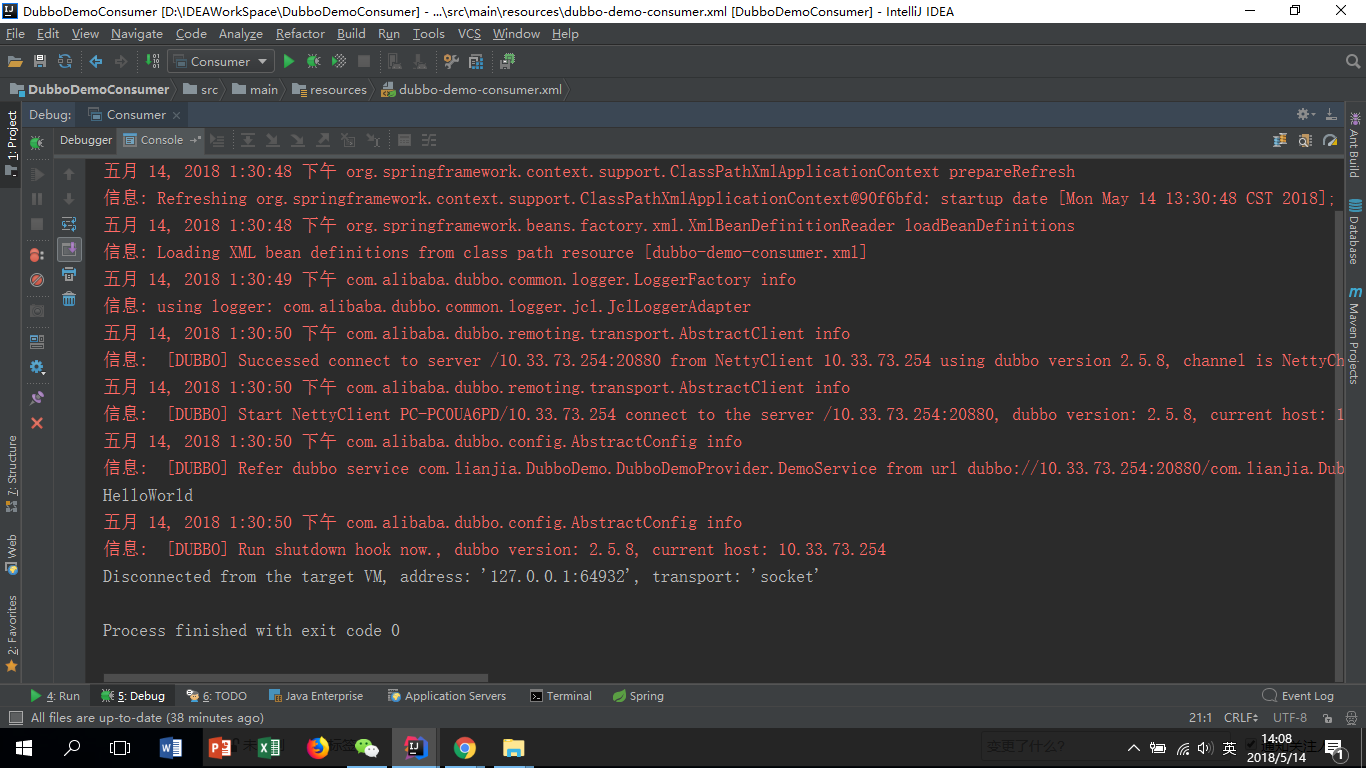
          运行 Provider.java  之后；出现如下界面：



D:\ProgramFiles\jdk8\bin\java.exe -agentlib:jdwp=transport=dt\_socket,address=127.0.0.1:64916,suspend=y,server=n -server -Xms800m -Xmx800m -XX:PermSize=256m -javaagent:C:\Users\lianjia\.IntelliJIdea2018.1\system\captureAgent\debugger-agent.jar=file:/C:/Users/lianjia/AppData/Local/Temp/capture3210.props -Dfile.encoding=UTF-8 -classpath "D:\ProgramFiles\jdk8\jre\lib\charsets.jar;D:\ProgramFiles\jdk8\jre\lib\deploy.jar;D:\ProgramFiles\jdk8\jre\lib\ext\access-bridge-64.jar;D:\ProgramFiles\jdk8\jre\lib\ext\cldrdata.jar;D:\ProgramFiles\jdk8\jre\lib\ext\dnsns.jar;D:\ProgramFiles\jdk8\jre\lib\ext\jaccess.jar;D:\ProgramFiles\jdk8\jre\lib\ext\jfxrt.jar;D:\ProgramFiles\jdk8\jre\lib\ext\localedata.jar;D:\ProgramFiles\jdk8\jre\lib\ext\nashorn.jar;D:\ProgramFiles\jdk8\jre\lib\ext\sunec.jar;D:\ProgramFiles\jdk8\jre\lib\ext\sunjce\_provider.jar;D:\ProgramFiles\jdk8\jre\lib\ext\sunmscapi.jar;D:\ProgramFiles\jdk8\jre\lib\ext\sunpkcs11.jar;D:\ProgramFiles\jdk8\jre\lib\ext\zipfs.jar;D:\ProgramFiles\jdk8\jre\lib\javaws.jar;D:\ProgramFiles\jdk8\jre\lib\jce.jar;D:\ProgramFiles\jdk8\jre\lib\jfr.jar;D:\ProgramFiles\jdk8\jre\lib\jfxswt.jar;D:\ProgramFiles\jdk8\jre\lib\jsse.jar;D:\ProgramFiles\jdk8\jre\lib\management-agent.jar;D:\ProgramFiles\jdk8\jre\lib\plugin.jar;D:\ProgramFiles\jdk8\jre\lib\resources.jar;D:\ProgramFiles\jdk8\jre\lib\rt.jar;D:\IDEAWorkSpace\DubboDemo\DubboDemoProvider\target\classes;C:\Users\lianjia\.m2\repository\com\alibaba\dubbo\2.5.8\dubbo-2.5.8.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-beans\4.3.10.RELEASE\spring-beans-4.3.10.RELEASE.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-web\4.3.10.RELEASE\spring-web-4.3.10.RELEASE.jar;C:\Users\lianjia\.m2\repository\org\javassist\javassist\3.20.0-GA\javassist-3.20.0-GA.jar;C:\Users\lianjia\.m2\repository\org\jboss\netty\netty\3.2.5.Final\netty-3.2.5.Final.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-context\4.3.7.RELEASE\spring-context-4.3.7.RELEASE.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-aop\4.3.7.RELEASE\spring-aop-4.3.7.RELEASE.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-expression\4.3.7.RELEASE\spring-expression-4.3.7.RELEASE.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-core\4.3.7.RELEASE\spring-core-4.3.7.RELEASE.jar;C:\Users\lianjia\.m2\repository\commons-logging\commons-logging\1.2\commons-logging-1.2.jar;D:\ProgramFiles\IntelliJ IDEA 2018.1.2\lib\idea\_rt.jar" com.lianjia.DubboDemo.DubboDemoProvider.Provider  
Connected to the target VM, address: '127.0.0.1:64916', transport: 'socket'  
五月 14, 2018 1:30:19 下午 org.springframework.context.support.ClassPathXmlApplicationContext prepareRefresh  
信息: Refreshing org.springframework.context.support.ClassPathXmlApplicationContext@327471b5: startup date [Mon May 14 13:30:19 CST 2018]; root of context hierarchy  
五月 14, 2018 1:30:19 下午 org.springframework.beans.factory.xml.XmlBeanDefinitionReader loadBeanDefinitions  
信息: Loading XML bean definitions from class path resource [dubbo-demo-provider.xml]  
五月 14, 2018 1:30:19 下午 com.alibaba.dubbo.common.logger.LoggerFactory info  
信息: using logger: com.alibaba.dubbo.common.logger.jcl.JclLoggerAdapter  
五月 14, 2018 1:30:20 下午 com.alibaba.dubbo.config.AbstractConfig info  
信息: [DUBBO] The service ready on spring started. service: com.lianjia.DubboDemo.DubboDemoProvider.DemoService, dubbo version: 2.5.8, current host: 127.0.0.1  
五月 14, 2018 1:30:21 下午 com.alibaba.dubbo.config.AbstractConfig info  
信息: [DUBBO] Export dubbo service com.lianjia.DubboDemo.DubboDemoProvider.DemoService to local registry, dubbo version: 2.5.8, current host: 127.0.0.1  
五月 14, 2018 1:30:21 下午 com.alibaba.dubbo.config.AbstractConfig info  
信息: [DUBBO] Export dubbo service com.lianjia.DubboDemo.DubboDemoProvider.DemoService to url <dubbo://10.33.73.254:20880/com.lianjia.DubboDemo.DubboDemoProvider.DemoService?anyhost=true&application=demo-provider&bind.ip=10.33.73.254&bind.port=20880&dubbo=2.5.8&generic=false&interface=com.lianjia.DubboDemo.DubboDemoProvider.DemoService&methods=HelloWorld&pid=3584&side=provider&timestamp=1526275820828>, dubbo version: 2.5.8, current host: 127.0.0.1  
五月 14, 2018 1:30:21 下午 com.alibaba.dubbo.remoting.transport.AbstractServer info  
信息: [DUBBO] Start NettyServer bind /0.0.0.0:20880, export /10.33.73.254:20880, dubbo version: 2.5.8, current host: 127.0.0.1  
五月 14, 2018 1:30:50 下午 com.alibaba.dubbo.remoting.transport.AbstractServer warn  
警告: [DUBBO] All clients has discontected from /10.33.73.254:20880. You can graceful shutdown now., dubbo version: 2.5.8, current host: 127.0.0.1  
五月 14, 2018 1:30:50 下午 com.alibaba.dubbo.rpc.protocol.dubbo.DubboProtocol info  
信息: [DUBBO] disconected from /10.33.73.254:64943,url:<dubbo://10.33.73.254:20880/com.lianjia.DubboDemo.DubboDemoProvider.DemoService?anyhost=true&application=demo-provider&bind.ip=10.33.73.254&bind.port=20880&channel.readonly.sent=true&codec=dubbo&dubbo=2.5.8&generic=false&heartbeat=60000&interface=com.lianjia.DubboDemo.DubboDemoProvider.DemoService&methods=HelloWorld&pid=3584&side=provider&timestamp=1526275820828>, dubbo version: 2.5.8, current host: 127.0.0.1

（2）启动项目：DubboDemoConsumer

         运行 Consumer.java之后，出现如下界面：



D:\ProgramFiles\jdk8\bin\java.exe -agentlib:jdwp=transport=dt\_socket,address=127.0.0.1:64932,suspend=y,server=n -javaagent:C:\Users\lianjia\AppData\Local\Temp\capture4889jars\debugger-agent.jar=file:/C:/Users/lianjia/AppData/Local/Temp/capture3257.props -Dfile.encoding=UTF-8 -classpath "D:\ProgramFiles\jdk8\jre\lib\charsets.jar;D:\ProgramFiles\jdk8\jre\lib\deploy.jar;D:\ProgramFiles\jdk8\jre\lib\ext\access-bridge-64.jar;D:\ProgramFiles\jdk8\jre\lib\ext\cldrdata.jar;D:\ProgramFiles\jdk8\jre\lib\ext\dnsns.jar;D:\ProgramFiles\jdk8\jre\lib\ext\jaccess.jar;D:\ProgramFiles\jdk8\jre\lib\ext\jfxrt.jar;D:\ProgramFiles\jdk8\jre\lib\ext\localedata.jar;D:\ProgramFiles\jdk8\jre\lib\ext\nashorn.jar;D:\ProgramFiles\jdk8\jre\lib\ext\sunec.jar;D:\ProgramFiles\jdk8\jre\lib\ext\sunjce\_provider.jar;D:\ProgramFiles\jdk8\jre\lib\ext\sunmscapi.jar;D:\ProgramFiles\jdk8\jre\lib\ext\sunpkcs11.jar;D:\ProgramFiles\jdk8\jre\lib\ext\zipfs.jar;D:\ProgramFiles\jdk8\jre\lib\javaws.jar;D:\ProgramFiles\jdk8\jre\lib\jce.jar;D:\ProgramFiles\jdk8\jre\lib\jfr.jar;D:\ProgramFiles\jdk8\jre\lib\jfxswt.jar;D:\ProgramFiles\jdk8\jre\lib\jsse.jar;D:\ProgramFiles\jdk8\jre\lib\management-agent.jar;D:\ProgramFiles\jdk8\jre\lib\plugin.jar;D:\ProgramFiles\jdk8\jre\lib\resources.jar;D:\ProgramFiles\jdk8\jre\lib\rt.jar;D:\IDEAWorkSpace\DubboDemoConsumer\target\classes;C:\Users\lianjia\.m2\repository\com\alibaba\dubbo\2.5.8\dubbo-2.5.8.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-beans\4.3.10.RELEASE\spring-beans-4.3.10.RELEASE.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-web\4.3.10.RELEASE\spring-web-4.3.10.RELEASE.jar;C:\Users\lianjia\.m2\repository\org\javassist\javassist\3.20.0-GA\javassist-3.20.0-GA.jar;C:\Users\lianjia\.m2\repository\org\jboss\netty\netty\3.2.5.Final\netty-3.2.5.Final.jar;C:\Users\lianjia\.m2\repository\com\lianjia\DubboDemo\DubboDemoProvider\1.0-SNAPSHOT\DubboDemoProvider-1.0-SNAPSHOT.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-context\4.3.7.RELEASE\spring-context-4.3.7.RELEASE.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-aop\4.3.7.RELEASE\spring-aop-4.3.7.RELEASE.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-core\4.3.7.RELEASE\spring-core-4.3.7.RELEASE.jar;C:\Users\lianjia\.m2\repository\commons-logging\commons-logging\1.2\commons-logging-1.2.jar;C:\Users\lianjia\.m2\repository\org\springframework\spring-expression\4.3.7.RELEASE\spring-expression-4.3.7.RELEASE.jar;D:\ProgramFiles\IntelliJ IDEA 2018.1.2\lib\idea\_rt.jar" com.lianjia.DubboDemo.DubboDemoProvider.Consumer  
Connected to the target VM, address: '127.0.0.1:64932', transport: 'socket'  
五月 14, 2018 1:30:48 下午 org.springframework.context.support.ClassPathXmlApplicationContext prepareRefresh  
信息: Refreshing org.springframework.context.support.ClassPathXmlApplicationContext@90f6bfd: startup date [Mon May 14 13:30:48 CST 2018]; root of context hierarchy  
五月 14, 2018 1:30:48 下午 org.springframework.beans.factory.xml.XmlBeanDefinitionReader loadBeanDefinitions  
信息: Loading XML bean definitions from class path resource [dubbo-demo-consumer.xml]  
五月 14, 2018 1:30:49 下午 com.alibaba.dubbo.common.logger.LoggerFactory info  
信息: using logger: com.alibaba.dubbo.common.logger.jcl.JclLoggerAdapter  
五月 14, 2018 1:30:50 下午 com.alibaba.dubbo.remoting.transport.AbstractClient info  
信息: [DUBBO] Successed connect to server /10.33.73.254:20880 from NettyClient 10.33.73.254 using dubbo version 2.5.8, channel is NettyChannel [channel=[id: 0x2f62ea70, /10.33.73.254:64943 => /10.33.73.254:20880]], dubbo version: 2.5.8, current host: 10.33.73.254  
五月 14, 2018 1:30:50 下午 com.alibaba.dubbo.remoting.transport.AbstractClient info  
信息: [DUBBO] Start NettyClient PC-PC0UA6PD/10.33.73.254 connect to the server /10.33.73.254:20880, dubbo version: 2.5.8, current host: 10.33.73.254  
五月 14, 2018 1:30:50 下午 com.alibaba.dubbo.config.AbstractConfig info  
信息: [DUBBO] Refer dubbo service com.lianjia.DubboDemo.DubboDemoProvider.DemoService from url <dubbo://10.33.73.254:20880/com.lianjia.DubboDemo.DubboDemoProvider.DemoService?application=demo-consumer&dubbo=2.5.8&interface=com.lianjia.DubboDemo.DubboDemoProvider.DemoService&methods=HelloWorld&pid=8724&register.ip=10.33.73.254&revision=1.0-SNAPSHOT&side=consumer&timestamp=1526275849847>, dubbo version: 2.5.8, current host: 10.33.73.254  
HelloWorld  
五月 14, 2018 1:30:50 下午 com.alibaba.dubbo.config.AbstractConfig info  
信息: [DUBBO] Run shutdown hook now., dubbo version: 2.5.8, current host: 10.33.73.254  
Disconnected from the target VM, address: '127.0.0.1:64932', transport: 'socket'

Process finished with exit code 0

调用之后出现了HelloWorld

证明测试成功~~~