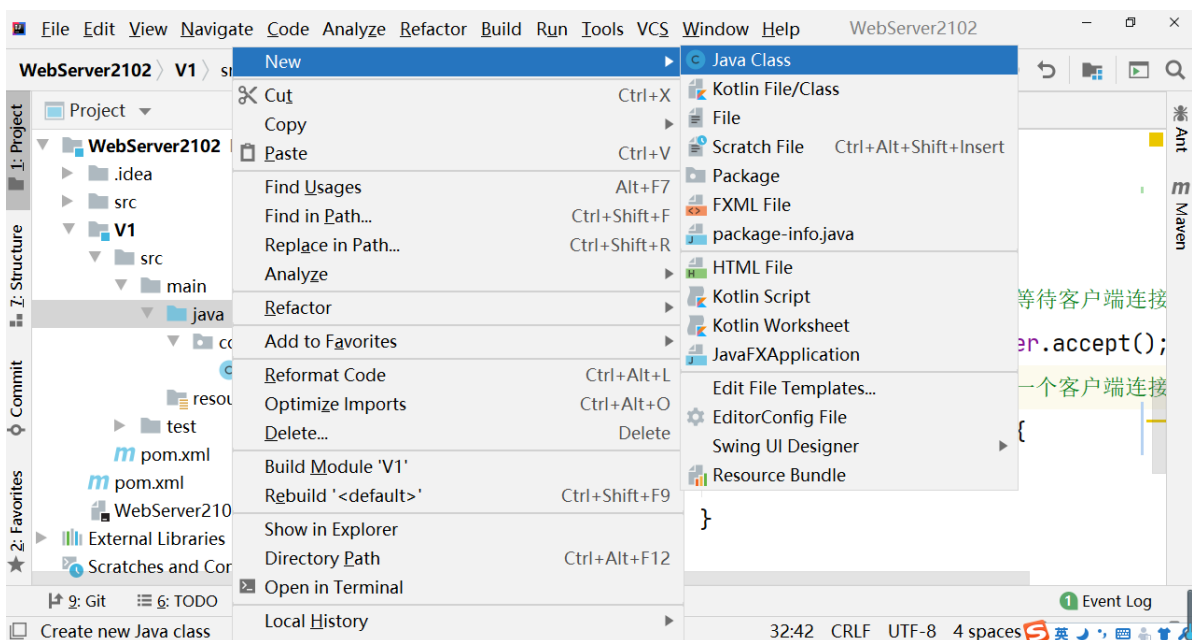


WebServer项目常见问题

1:项目创建后无法新建包或类

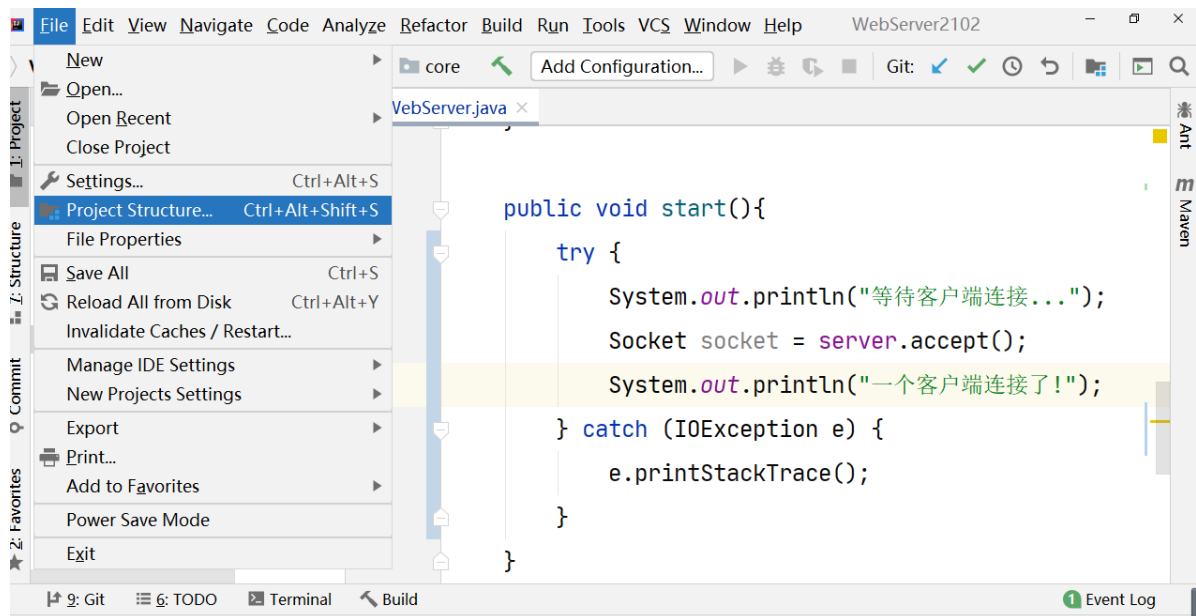
原因:项目下的src/main/java目录不是源码目录,将其设置为源码目录即可.

如图:如果右键java目录选择new后没有右侧菜单中的Java Class就说明这个java目录不是源码目录了(图标不是蓝色)



步骤一:

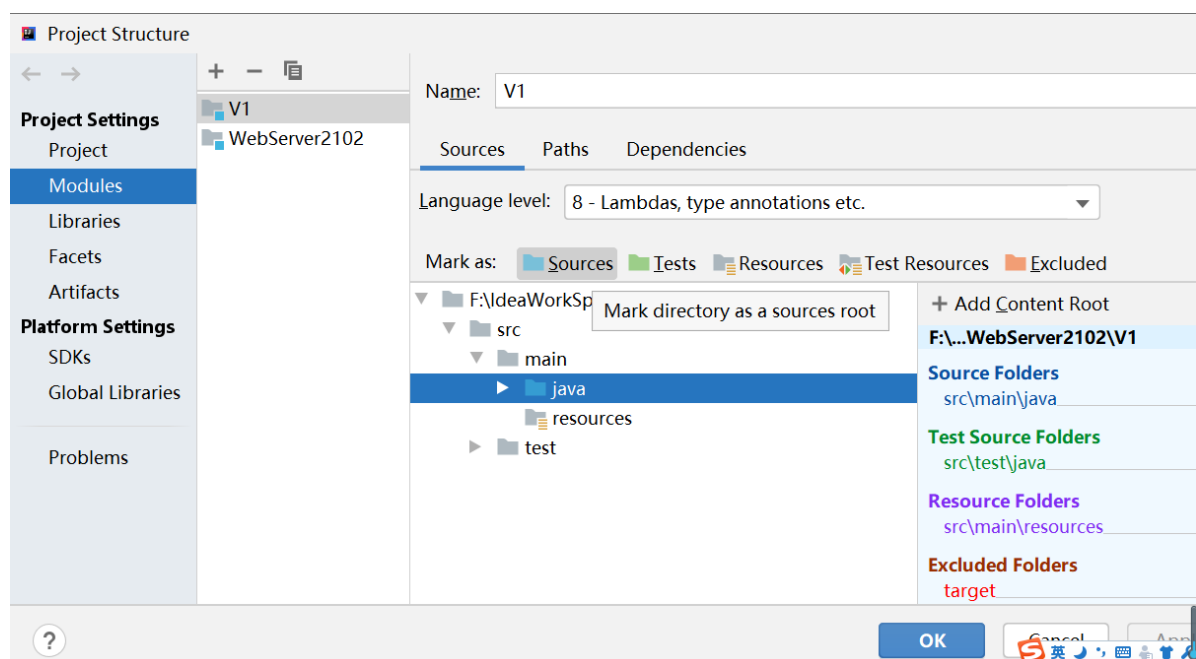
idea菜单顺序选择file->Project Struture



步骤二:

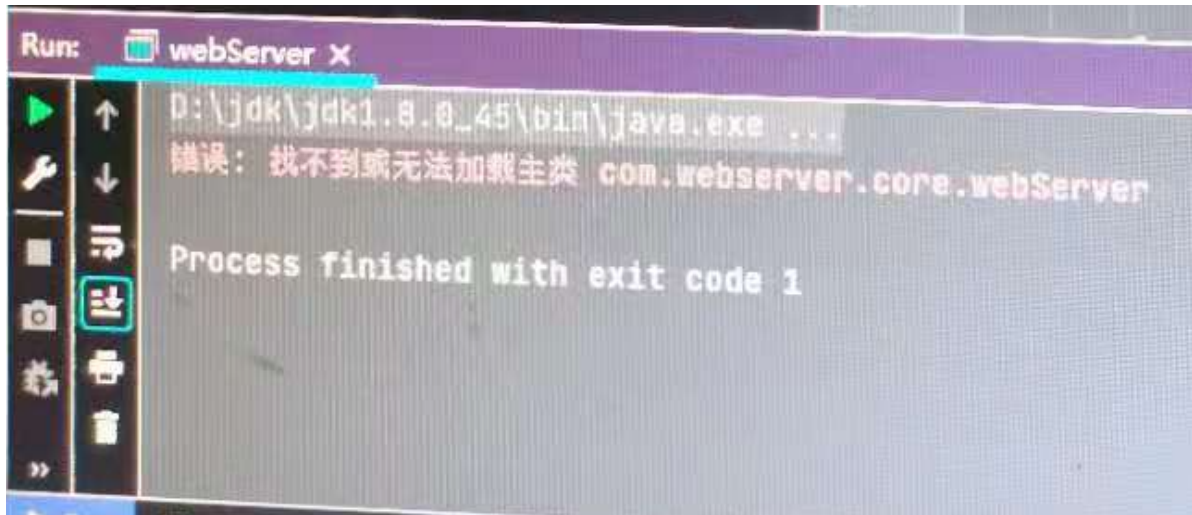
弹出框中依次选择

1. Modules
2. 中间一栏选择不能创建类的项目(图上的是V1)
3. 右侧选择Sources,然后下面找到对应的java目录,并选择Sources(蓝色图标)的源码目录,即可.



2:运行程序控制台显示无法加载主类

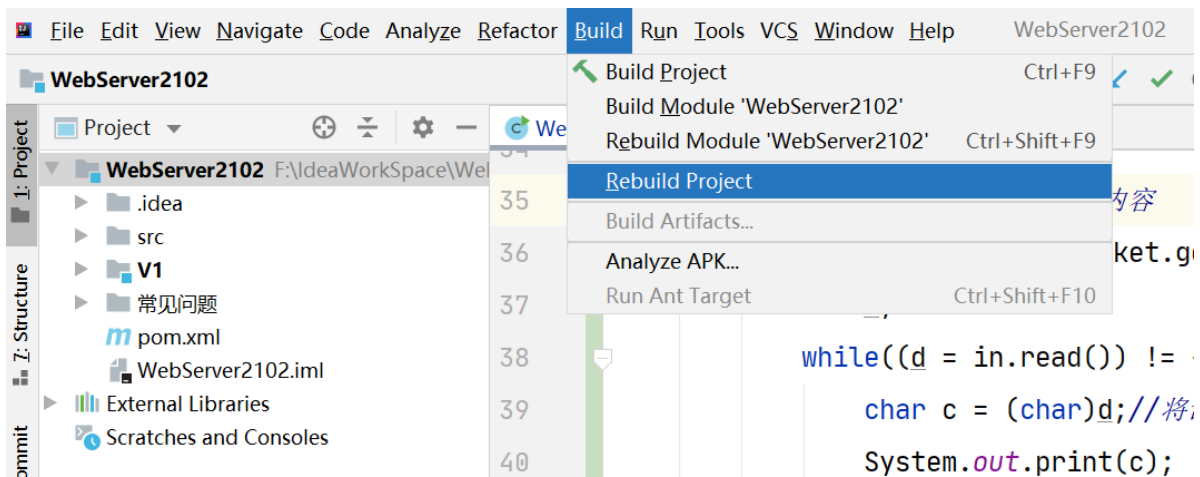
如图:



解决办法:

重构项目,通常就可以解决问题:选择菜单栏上的Build->Rebuild Project.

重构后在尝试运行.



3:浏览器输入路径请求服务端后,服务端输出的内容是乱码

```
"C:\Program Files\Java\jdk1.8.0_201\bin\java.exe" ...
正在启动服务端...
服务端启动完毕!
等待客户端连接...
一个客户端连接了!
00000000`nİxaa0B?0aÃ00_000.şÜx0aMú0K-â&À,À+À0À/À$À#À(À'À
À À0À000=<5/
000000
000000000000000000000000#0ÿ00
```

原因:浏览器地址栏中的路径协议用的是https.

解决:浏览器地址栏输入地址时,前面以http开始即可.

4:启动WebServer后控制台报错

```
"C:\Program Files\Java\jdk1.8.0_191\bin\java.exe" ...
正在启动服务端...
等待客户端连接...
java.net.BindException: Address already in use: JVM_Bind
    at java.net.DualStackPlainSocketImpl.bind0(Native Method)
    at java.net.DualStackPlainSocketImpl.socketBind(DualStackPlainSocketImpl.java:106)
    at java.net.AbstractPlainSocketImpl.bind(AbstractPlainSocketImpl.java:387)
    at java.net.PlainSocketImpl.bind(PlainSocketImpl.java:190)
    at java.net.ServerSocket.bind(ServerSocket.java:375)
    at java.net.ServerSocket.<init>(ServerSocket.java:237)
    at java.net.ServerSocket.<init>(ServerSocket.java:128)
    at com.webserver.core.WebServer.<init>(WebServer.java:24)
    at com.webserver.core.WebServer.main(WebServer.java:50)
Exception in thread "main" java.lang.NullPointerException
    at com.webserver.core.WebServer.start(WebServer.java:37)
    at com.webserver.core.WebServer.main(WebServer.java:51)

Process finished with exit code 1
```

这个错误是端口被占用了.WebServer实例化时指定的端口(8088)

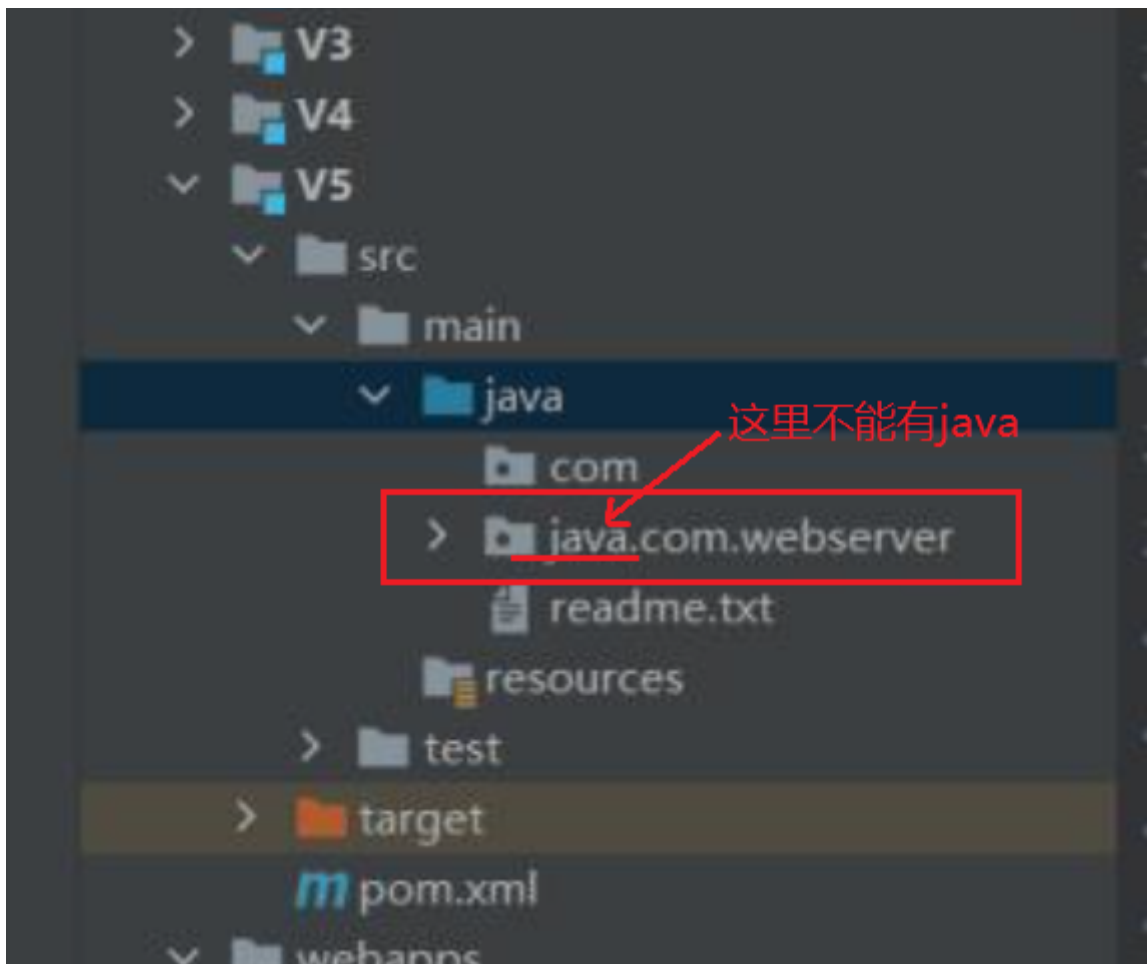
原因:通常是由于服务端启动了两次导致的.

解决:将之前的服务端关闭后再启动.经常是因为例如:V1的启动没有关闭,然后创建新版本V2后再启动V2的WebServer时报错.此时要先将原V1的WebServer关闭

5:程序启动后出现下图错误

```
java.lang.SecurityException Create breakpoint : Prohibited package name: java.com.webserver.core
    at java.lang.ClassLoader.preDefineClass(ClassLoader.java:655)
    at java.lang.ClassLoader.defineClass(ClassLoader.java:754)
    at java.security.SecureClassLoader.defineClass(SecureClassLoader.java:142)
    at java.net.URLClassLoader.defineClass(URLClassLoader.java:468)
    at java.net.URLClassLoader.access$100(URLClassLoader.java:74)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:369)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:363) <1 internal call>
    at java.net.URLClassLoader.findClass(URLClassLoader.java:362)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:418)
    at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:355)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:351)
    at sun.launcher.LauncherHelper.checkAndLoadMain(LauncherHelper.java:495)
Error: A JNI error has occurred, please check your installation and try again
Exception in thread "main"
进程已结束，退出代码为 1
```

原因:复制上一个版本的java目录时，粘贴到当前版本的java目录中了.



解决:将上一个版本的java目录粘贴到当前版本的main目录中覆盖java目录

6:浏览器请求服务端后，服务端出现下面错误

等待客户端连接...

一个客户端连接了...

```
java.lang.ArrayIndexOutOfBoundsException: 1
    at com.webserver.http.HttpRequest.parseRequestLine(HttpRequest.java:54)
    at com.webserver.http.HttpRequest.<init>(HttpRequest.java:35)
    at com.webserver.core.ClientHandler.run(ClientHandler.java:28)
    at java.lang.Thread.run(Thread.java:748)
```


对应的报错代码为:

```
String line = readLine();
System.out.println(line);
String[] data = line.split(regex: "\\s");//按照空格拆分
method = data[0];
uri = data[1];
protocol = data[2];
//测试路径如果是:http://localhost:8088/myweb/index.html
```

原因:由于浏览器发送了空请求

解决:关闭浏览器重新启动并访问,或换一个浏览器访问,或等待后期版本服务端忽略空请求来解决

7:浏览器请求后出现下面错误

A screenshot of a terminal window showing a Java exception. The text is as follows:

```
headers: {Cookie=1dea-61ae1b72-810d0b7c-0431-410d-8bda-0b0b2110027d, Accept=text/html,
java.net.SocketException: Software caused connection abort: socket write error
    at java.net.SocketOutputStream.socketWrite0(Native Method)
    at java.net.SocketOutputStream.socketWrite(SocketOutputStream.java:109)
    at java.net.SocketOutputStream.write(SocketOutputStream.java:153)
    at com.webserver.core.ClientHandler.run(ClientHandler.java:64)
    at java.lang.Thread.run(Thread.java:745)

Process finished with exit code 0
|
```

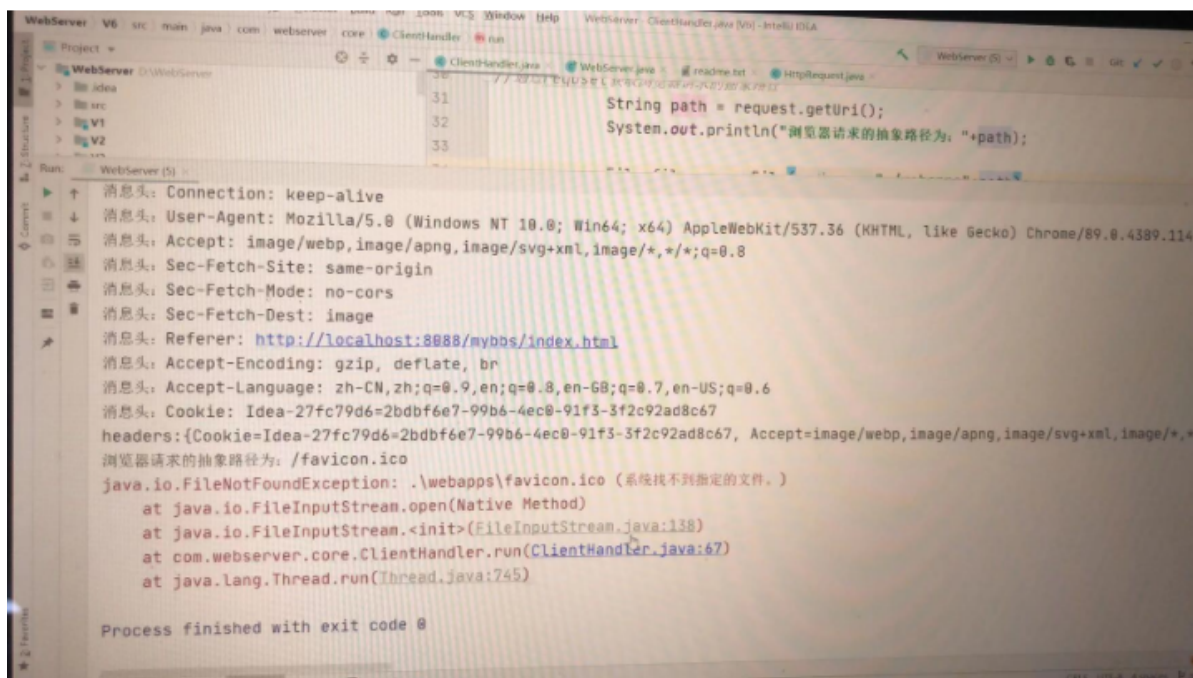
原因:

可能的原因有两种

1:HttpRequest解析请求时,没有将请求所有内容读取完毕(请求行和消息头),要检查代码

2:发送响应时Content-Type的值与实际发送的响应正文长度不一致。检查这两部分的发送工作

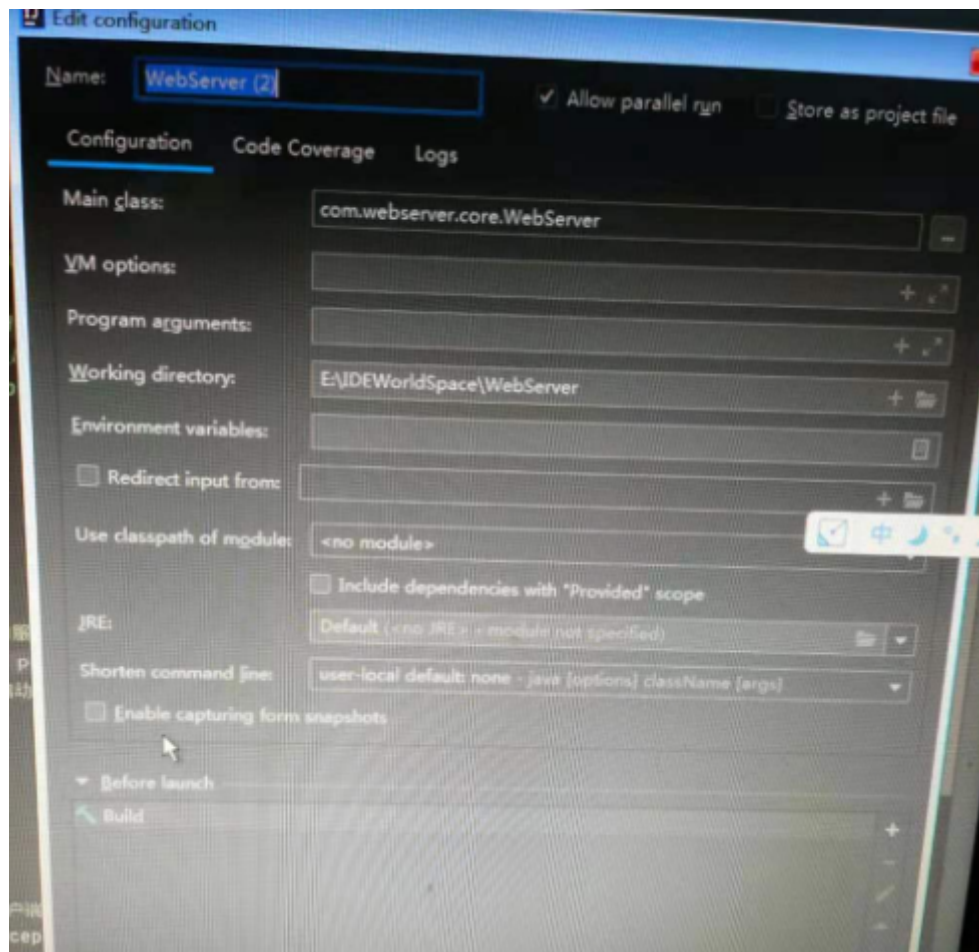
8:浏览器输入一个路径发起请求后，服务端出现如下错误



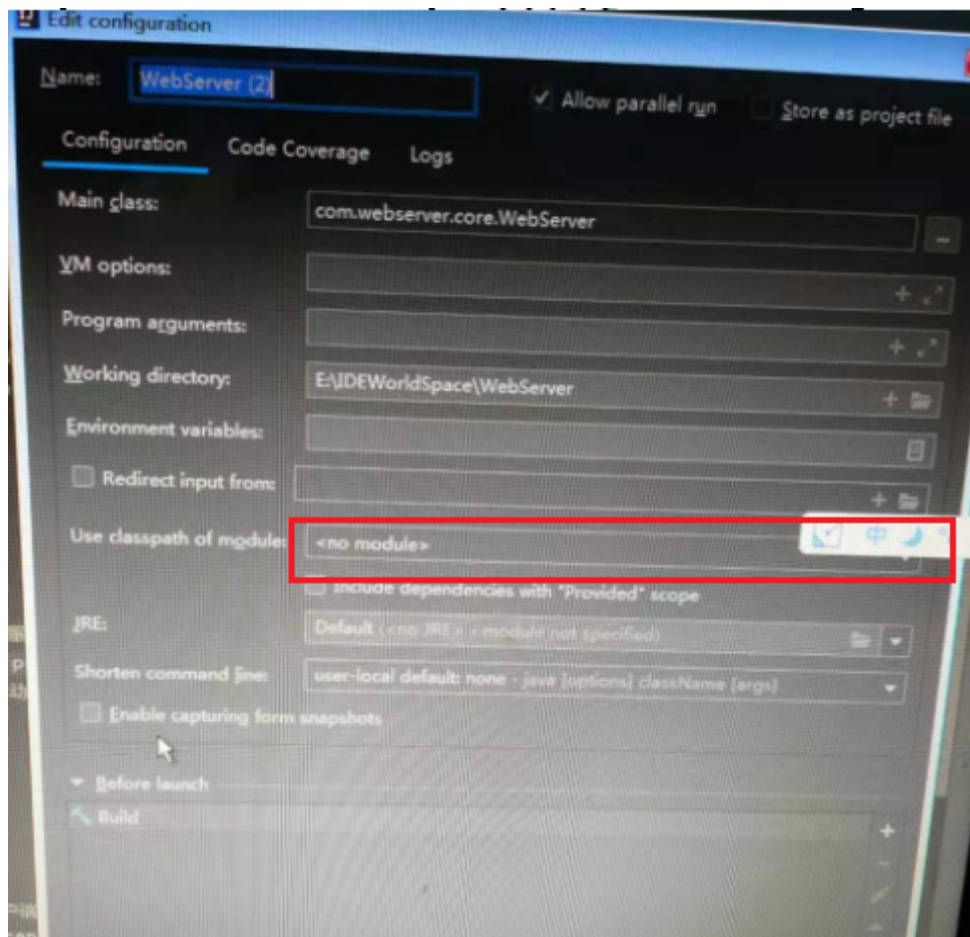
原因:系统找不到指定的文件是一个名为favicon.ico的文件。这是浏览器的自发行为，请求服务端的一个图标。

解决:要么响应404，要么准备一个.ico的图标文件放在webapps目录下即可。

9:启动WebServer时idea弹出提示框



解决:将Use classpath of modules选项中选择当前项目版本即可



10:浏览器请求服务端后,出现如下错误



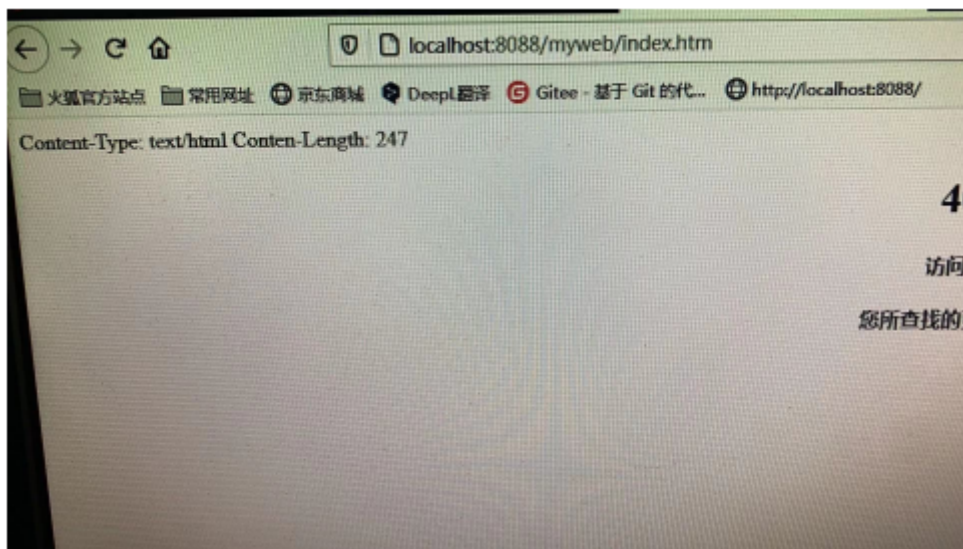
原因:通过request获取的抽象路径定位到了webapps目录,

这由于浏览器地址栏输入的:<http://localhost:8088/>
(这里没有输入内容)

而目录是无法读取并响应给客户端的

解决:在ClientHandler处理请求的环节,通过抽象路径实例化File时要判断表示的不是一个目录才响应

11:浏览器请求服务端后,展现的内容如下



原因:发送的响应未按照HTTP协议的要求发送.

解决:检查发送响应代码,看看是否有丢失或多发了CRLF这样的操作

并且图上还有一个错误,Content-Length的响应头名字拼写错误,这可能还导致页面展示的是源代码

12:ClientHandler出现下面错误

```
String path = request.getUri();
System.out.println("浏览器请求的抽象路径为:"+path);
File file = new File( pathname: "./webapps"+path);
//判断该文件是否存在并且要求是一个文件才能响应
if(file.exists() && file.isFile()){
    System.out.println("资源已找到!");
    HttpResponse response = new HttpResponse(socket);
    //1先将浏览器请求的资源文件设置到response中
    response.setEntity(file);
    //2发送响应
    response.flush();
} else { //如果资源不存在!
    System.out.println("资源不存在!");
    File notFound = new File( pathname: "./webapps/root/404.html");
    HttpResponse response = new HttpResponse(socket);
    response.setEntity(file);
    response.flush();
}
```

原因:

1:导包错误,导入的不是com.webserver.http包下自己定义的HttpResponse

2:HttpResponse没有提供构造方法

以上哪种要看idea提示的是哪种错误.