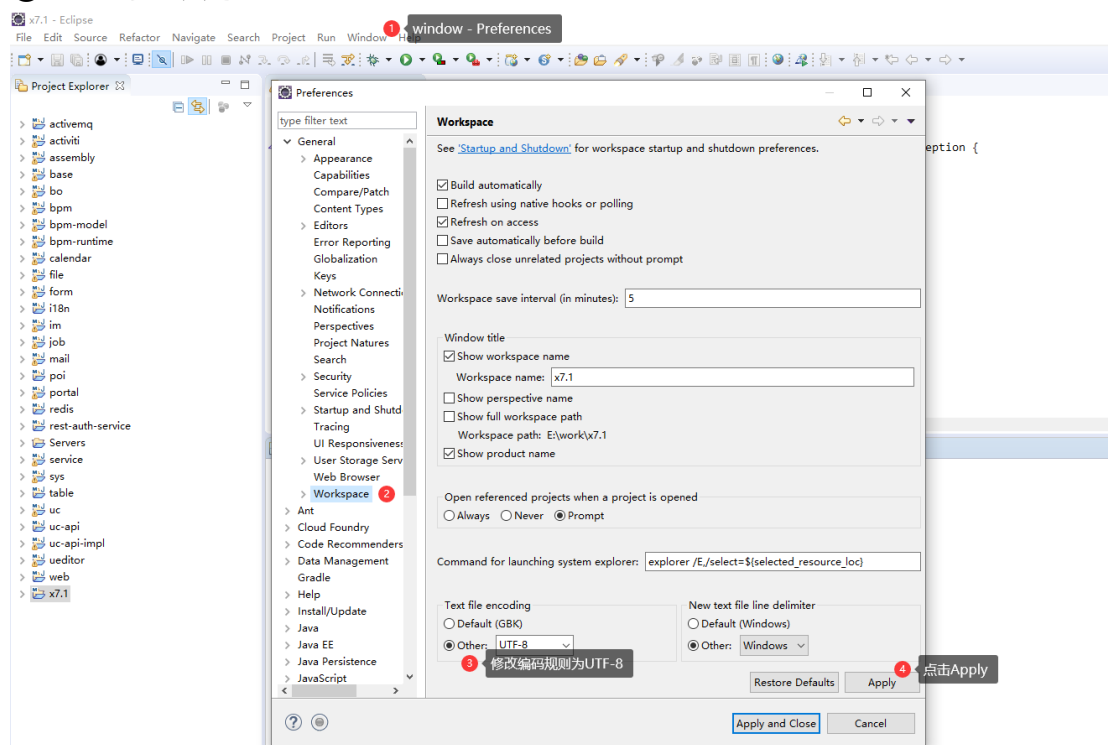


EIP7.2 开发环境分服务部署

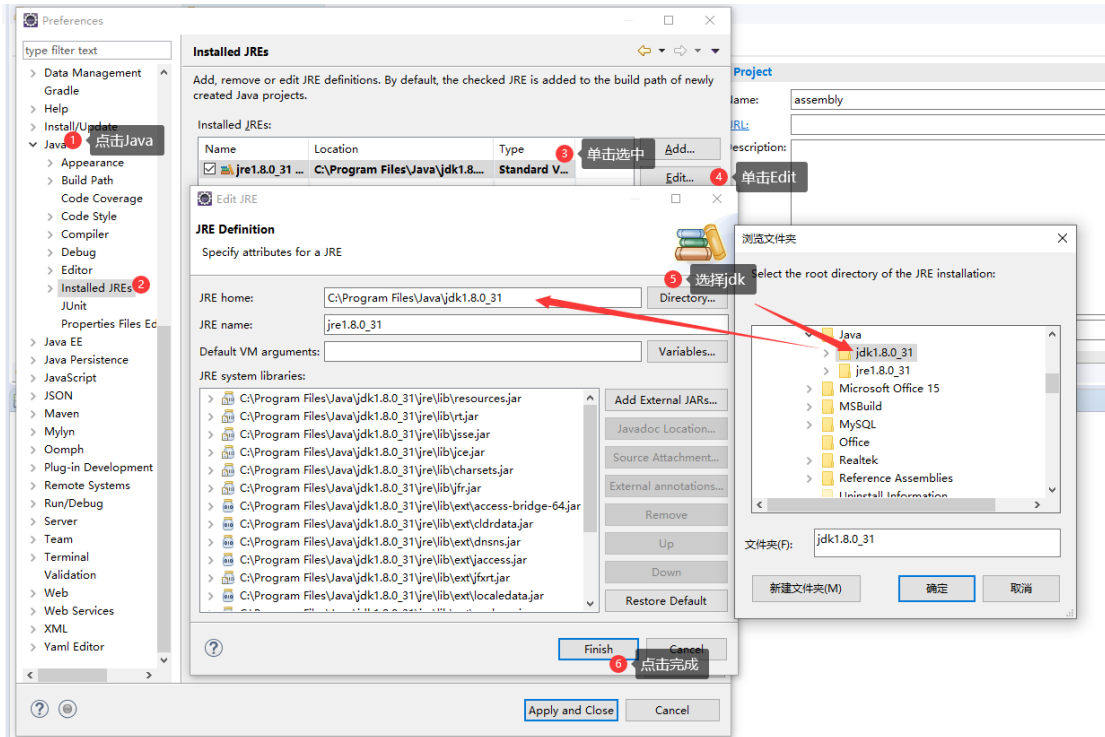
(Eclipse 版)

1. 配置 eclipse

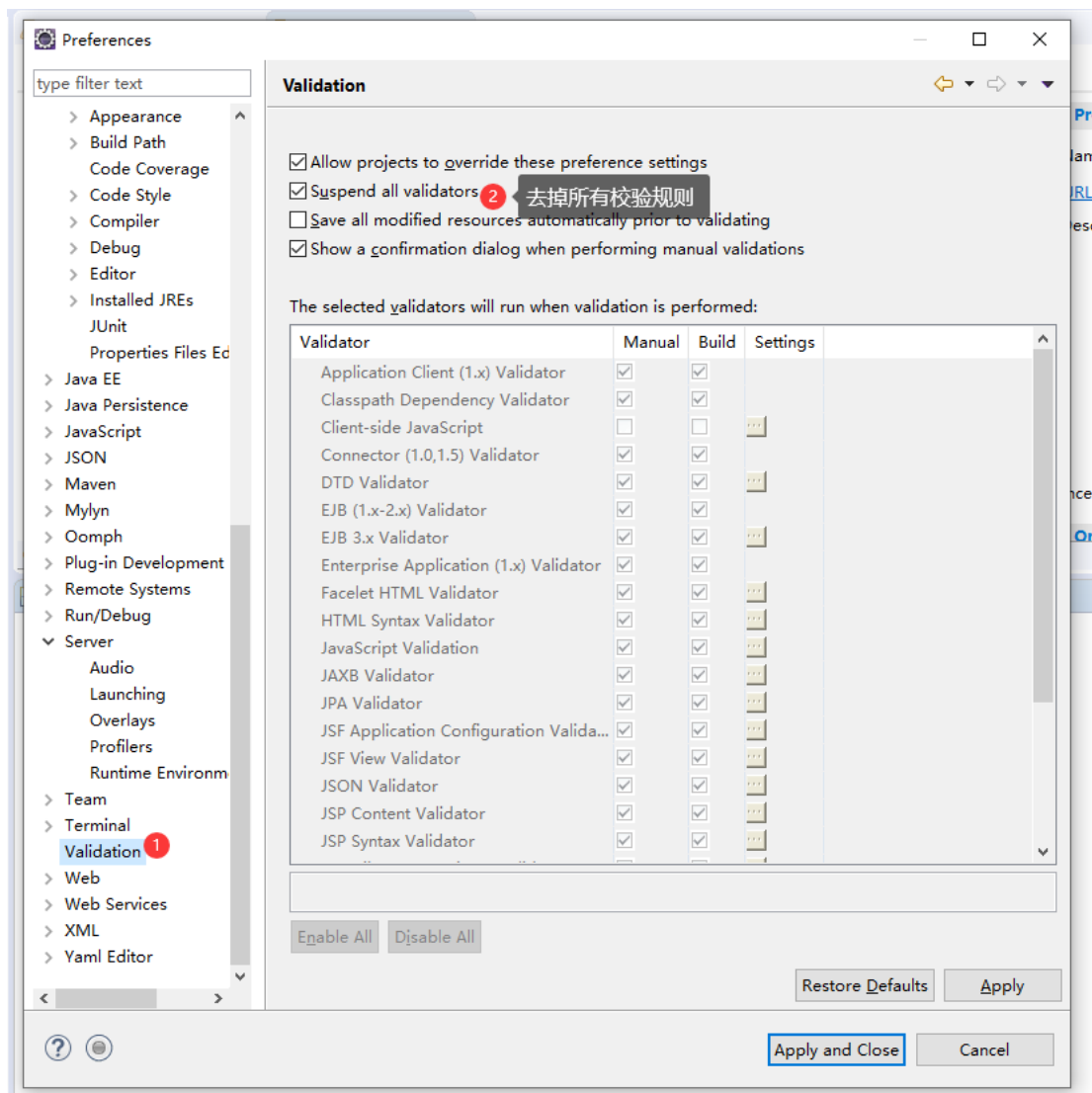
①配置编码规则



②配置 JAVA 环境

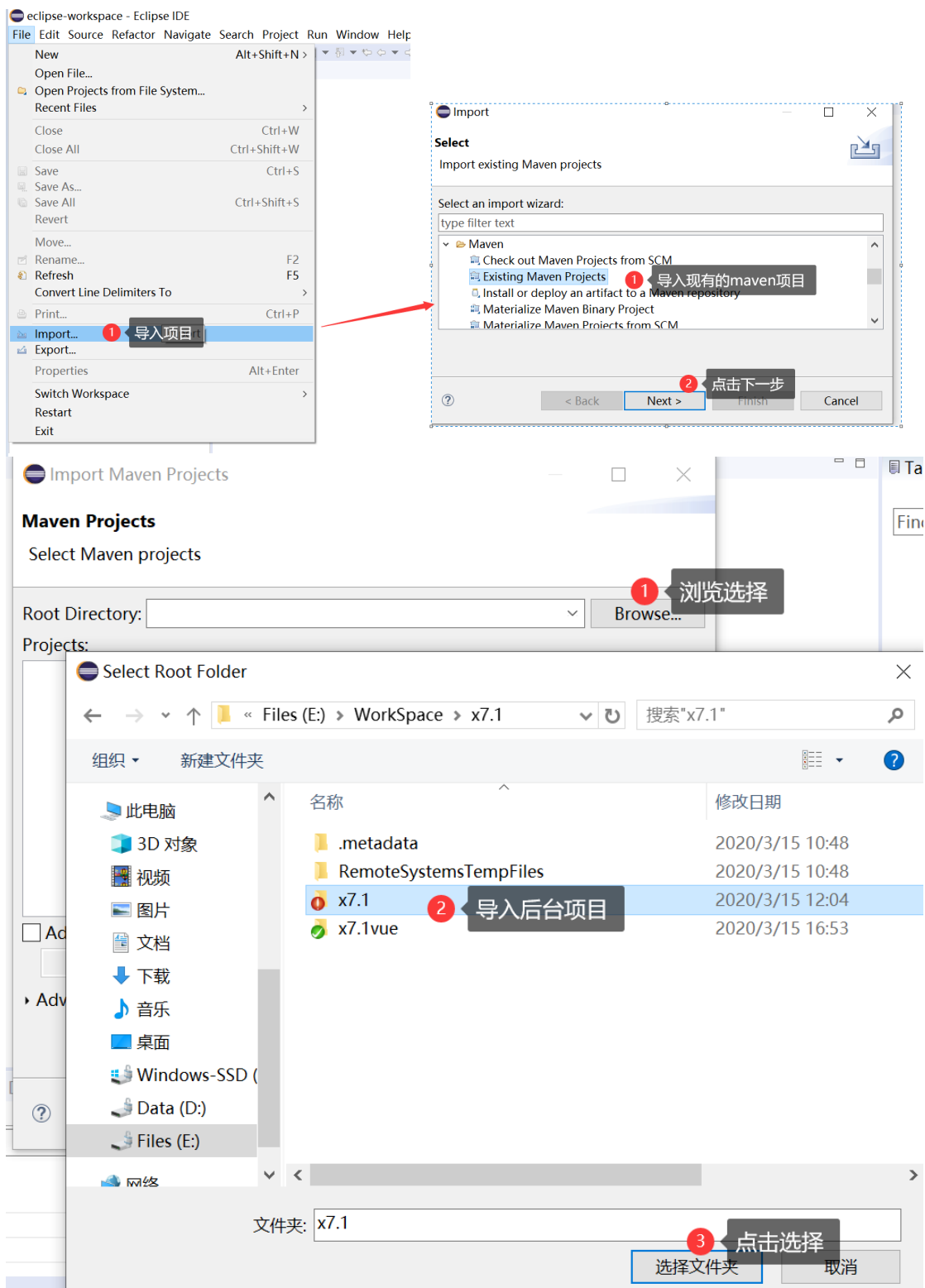


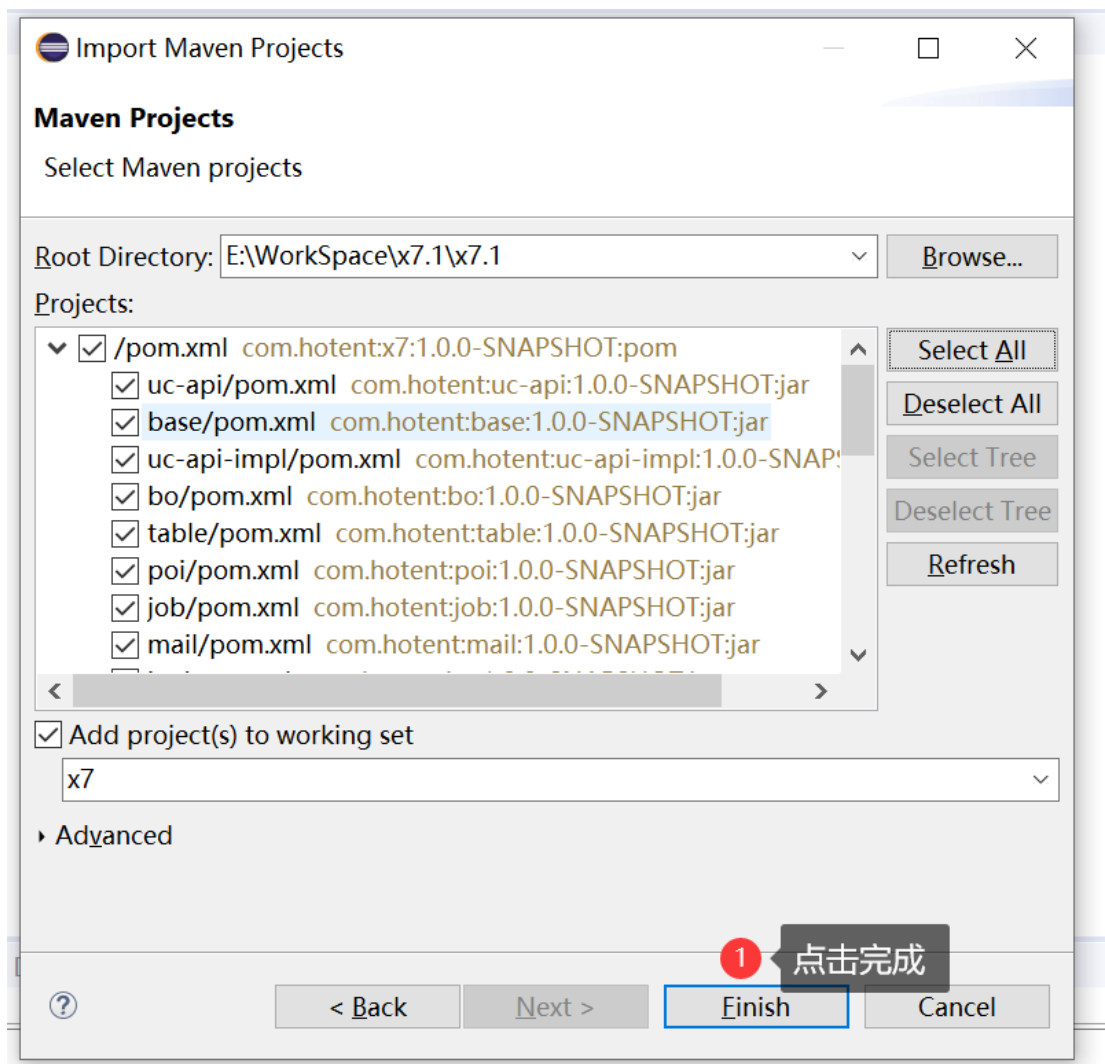
③去掉所有校验规则



2. 导入项目

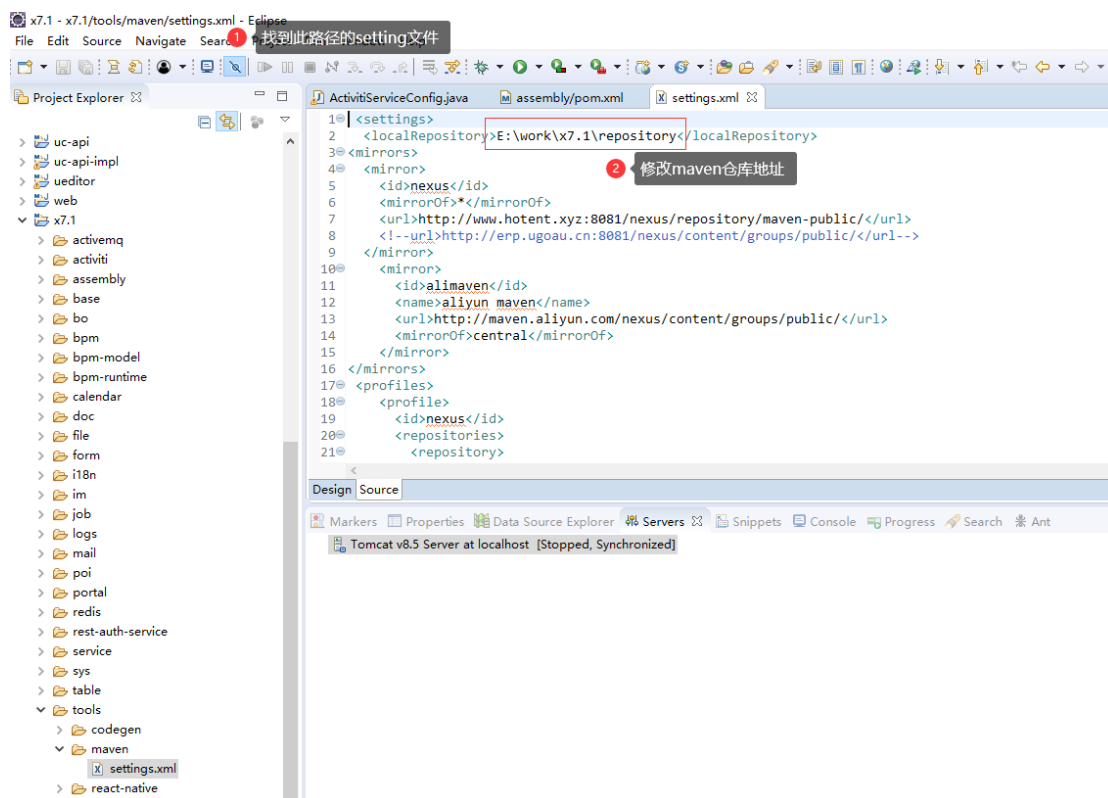
(注：如前端和后台代码分开，导入后台代码即可)





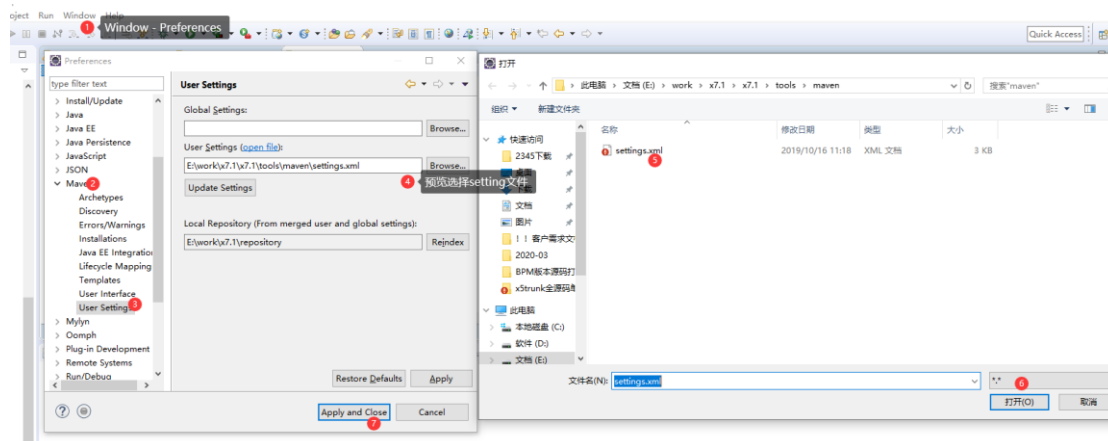
3. 修改 maven 仓库

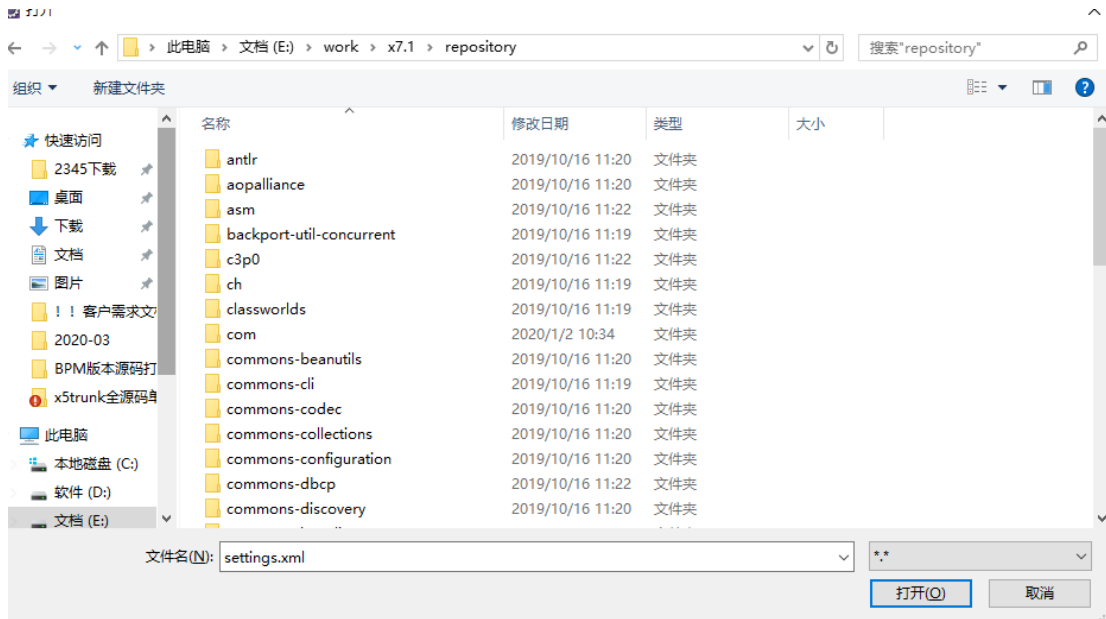
建议每个项目都新建一个文件夹作为 maven 仓库，以免不同项目的依赖会影响，导致后续问题产生。



4. 配置 maven 仓库

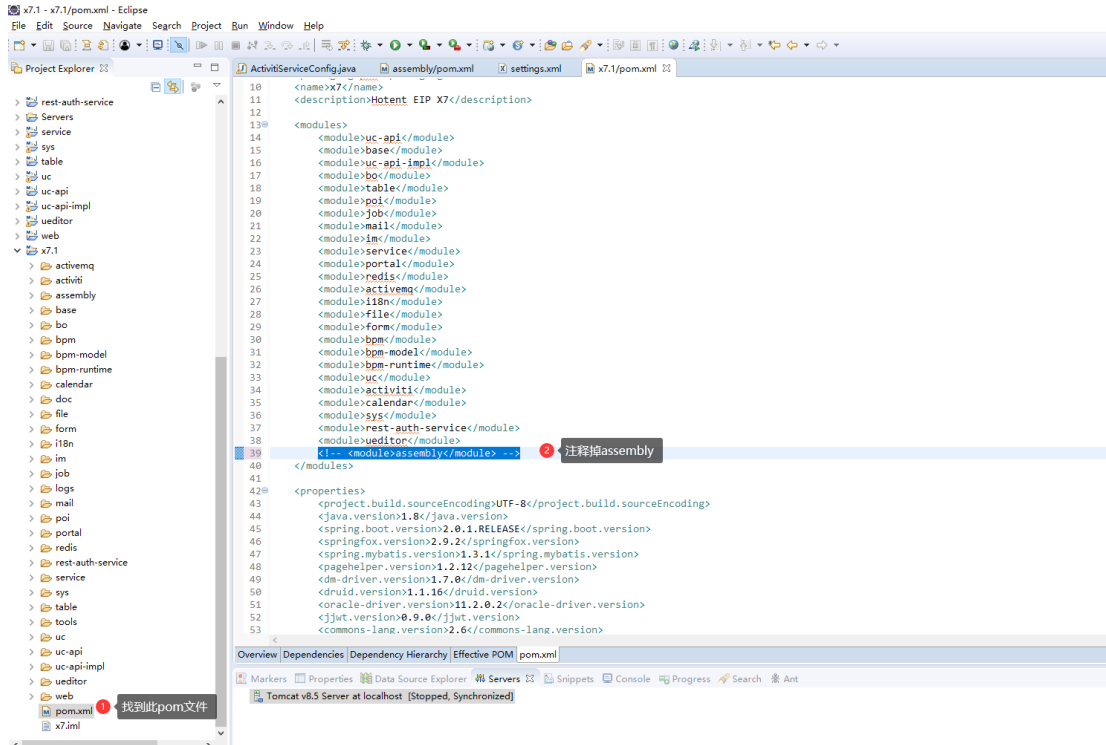
完成以下步骤，setting 文件所指向的 maven 仓库存放文件夹中会自动下载对应的依赖包





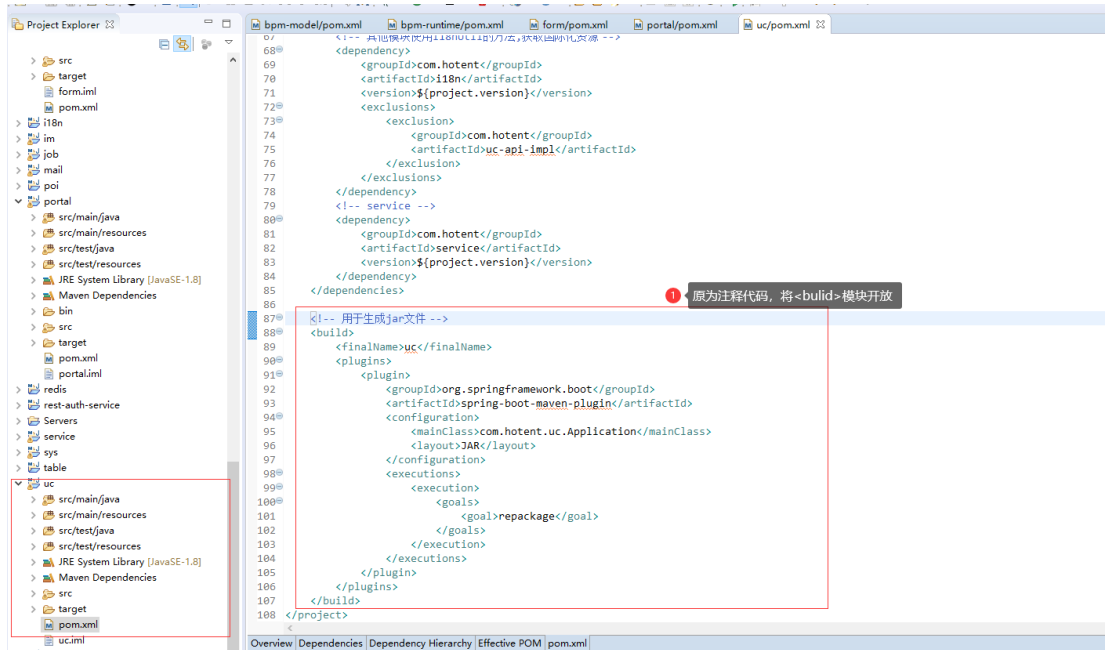
5. 分服务配置

将项目下...\pom.xml 这个文件中的 assembly 注释



6. 开放分服务

将每个服务 (bpm-model/bpm-runtime/uc/portal/form)下的 pom.xml 文件中的<build>模块开放



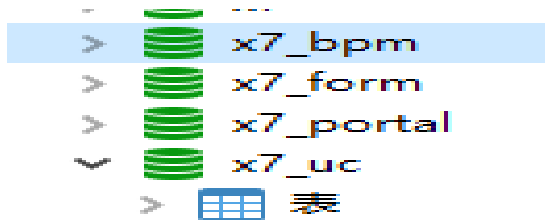
7. 初始化 SQL

需要创建四个数据库，分别对应 bpm,uc,form,portal 几大服务，

注：bpm_runtime 和 bpm_model 统一用一个数据库

具体步骤请查看 [初始化 sql 执行说明](#) 文档

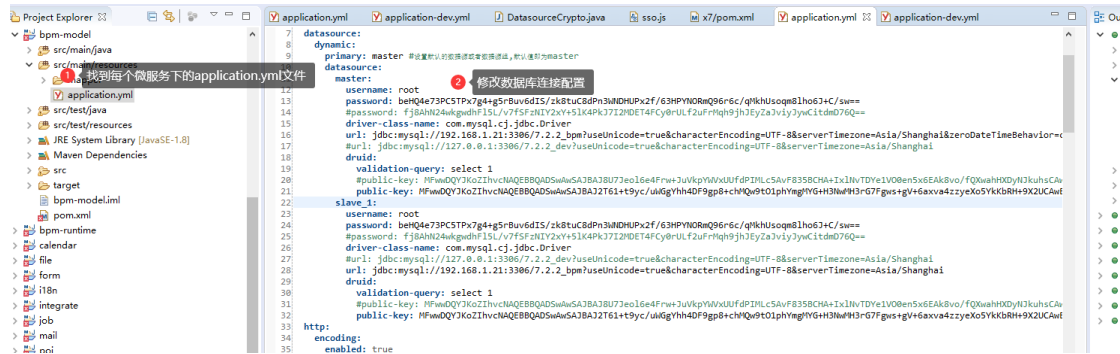
change	文件目录
mysql	文件目录
oracle	文件目录
seata	文件目录
sqlserver2012	文件目录
初始化数据	文件目录
索引index	文件目录
初始化sql执行说明.docx	150 KB Microsoft Word 文档



8. 修改服务配置文件

④修改每个服务(bpm_runtime,bpm_model,uc,portal,form)下的配置文件
application.yml

注：每个服务对应自己服务的数据库



数据源连接需要使用加密数据，通过 DataSourceCrypto.java 类生成数据源的
密码加密数据及公钥数据

```

1 package com.hotent.base;
2
3 import com.baomidou.dynamic.datasource.toolkit.CryptoUtils;
4
5 /**
6  * druid数据源密码加密生成器
7  *
8  * @company 广州宏天软件股份有限公司
9  * @author heyifan
10  * @email heyifan@jee-soft.cn
11  * @date 2020年4月6日
12  */
13 public class DatasourceCrypto {
14     public static void main(String[] args) throws Exception {
15         // 数据源密码
16         String password = "root"; 数据库密码
17
18         // 自定义publicKey
19         String[] arr = CryptoUtils.genKeyPair(512);
20         // 对称加密的私钥
21         System.out.println("privateKey:" + arr[0]);
22         // 对称加密的公钥(这个公钥配置到spring.datasource.dynamic.datasource.数据源组名.druid.public-key)
23         // 注意: 每一次生成生成的public-key与密码加密字符串都是成对的, 需要一起替换
24         System.out.println("publicKey:" + arr[1]);
25         // 数据库密码加密后的字符串(这个配置到spring.datasource.dynamic.datasource.数据源组名.password)
26         // 注意: 每一次生成的密码加密字符串与public-key都是成对的, 需要一起替换
27         System.out.println("password:" + CryptoUtils.encrypt(arr[0], password));
28     }
29 }
30

```

私钥
公钥
密码

```

<terminated> DatasourceCrypto [Java Application] C:\Development\Java\jdk1.8.0_111\bin\javaw.exe (2020年5月14日 上午11:09:34)
privateKey:MIIBVQIBADANBgkqhkiG9w0BAQEFAASCAT8wggE7AgEAAkAmPFZ3sN0MGOhg8+VpG6bqZHuN6JAdK8J08sfoS0xugh8q/1av9p0lqwiR0
publicKey:MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAAJjxwSbDdDBj0YPP1aRm6mR7jeiQHSvCdPLH6EtMboB/Kv5w1fadJa1iKdKo1EBtWJ180UmKSD
password:Y0yis6b4uf8vPbU2po1LPCHuPeQ+mEvIZ8ff+PTJj/QwT0E2h7ws+HvX6CDukGwML6Bisn2B+HnyK3Q69x+7g==

```

将公钥及密码替换至配置文件

```

7 datasource:
8   dynamic:
9     primary: master #设置默认的数据源或者数据源组, 默认该组为master
10    datasources:
11      master:
12        username: root
13        password: beHQ4e73PCSTP7g4+g5rBuv6dIS/zk8tuc8dPn3WHDHUPx2f/63HPYNORmQ96r6c/qMkhUsoqm8lho6J+C/sw==
14        driver-class-name: com.mysql.cj.jdbc.Driver
15        url: jdbc:mysql://192.168.1.21:3306/7.2.2_bpm?useUnicode=true&characterEncoding=UTF-8&serverTimezone=Asia/Shanghai&zeroDateTimeBehavior=convertToNull
16        #url: jdbc:mysql://127.0.0.1:3306/7.2.2_dev?useUnicode=true&characterEncoding=UTF-8&serverTimezone=Asia/Shanghai
17      druid:
18        validation-query: select 1
19        #public-key: MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAAJ38U7je016e4Fw+JuVkpYwXUuFdpIMLC5AvF835BCHA+Ix1NvTDYe1V00en5x6Eak8vo/fQXwahXdyNjkuhsCAw
20        public-key: MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAAJ2T61+t9yc/uIGYhh4DF9gp8+chhQw9t01phYmgMYG+H3NmH3rG7Fgws+gV+6axva4zzyeXo5YkKbRH+9X2UCAwf
21      slave-1:
22        username: root
23        password: beHQ4e73PCSTP7g4+g5rBuv6dIS/zk8tuc8dPn3WHDHUPx2f/63HPYNORmQ96r6c/qMkhUsoqm8lho6J+C/sw==
24        #password: fJ8AHn24wkgwdhF15L/v7F5FzNIY2xY+51K4Pk37I2HDEt4FCy8rULf2uFrMqh9jhjEyZa3viy3ywCitdmD76Q==
25        driver-class-name: com.mysql.cj.jdbc.Driver
26        url: jdbc:mysql://127.0.0.1:3306/7.2.2_dev?useUnicode=true&characterEncoding=UTF-8&serverTimezone=Asia/Shanghai
27        #url: jdbc:mysql://192.168.1.21:3306/7.2.2_bpm?useUnicode=true&characterEncoding=UTF-8&serverTimezone=Asia/Shanghai
28      druid:
29        validation-query: select 1
30        #public-key: MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAAJ38U7je016e4Fw+JuVkpYwXUuFdpIMLC5AvF835BCHA+Ix1NvTDYe1V00en5x6Eak8vo/fQXwahXdyNjkuhsCAw
31        public-key: MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAAJ2T61+t9yc/uIGYhh4DF9gp8+chhQw9t01phYmgMYG+H3NmH3rG7Fgws+gV+6axva4zzyeXo5YkKbRH+9X2UCAwf
32      http:
33        encoding:
34          enabled: true
35

```

```

49 mime-types: 'text/html,text/xml,text/plain,text/css,text/javascript,application/json'
50 min-response-size: 1024
51
52 system:
53   id:
54     machineName: bpm-model
55
56 eureka:
57   client:
58     healthcheck:
59       enabled: true
60     service-url:
61       defaultZone: http://192.168.1.222:8761/eureka/
62     instance:
63       lease-expiration-duration-in-seconds: 30
64       lease-renewal-interval-in-seconds: 10
65
66 # feign配置
67 feign:

```

需要启动eureka, 所以要将此改为eureka所在环境的ip,默认端口8761

②修改 base 下的 application-dev.yml, 修改 Redis/JMS 的相关配置

```

16: redis:
17:   database: 2
18:   host: 192.168.1.100
19:   port: 6379
20:   password:
21:   timeout: 2000
22:   #cluster:
23:   #nodes:
24:   #max-redirects: 6
25:   #sentinel:
26:   #master:
27:   #nodes:
28:   lettuce:
29:     pool:
30:       max-active: 8
31:       max-idle: 8
32:       min-idle: 0

35: # tcp://localhost:61616
36: jms:
37:   pub-sub-domain: true
38:   cache:
39:     enabled: false
40:   activemq:
41:     broker-url: tcp://192.168.1.100:61616
42:     in-memory: false
43:     pool:
44:       enabled: false
45:     packages:
46:       trust-all: true
47:       trusted: com.hotent.activemq.model,com.hotent.base.jms,java.util,java.lang #信任所有的包
48:     user:
49:     password:
50:
51: jms:
52:   queue:
53:     name: eipQueue
54:   topic:
55:     name: eipTopic
56:

```

9. 修改前端配置文件

①找到 web 中 manage 文件夹下的 sso.js 文件，进入编辑修改访问地址及服务访问地址及端口

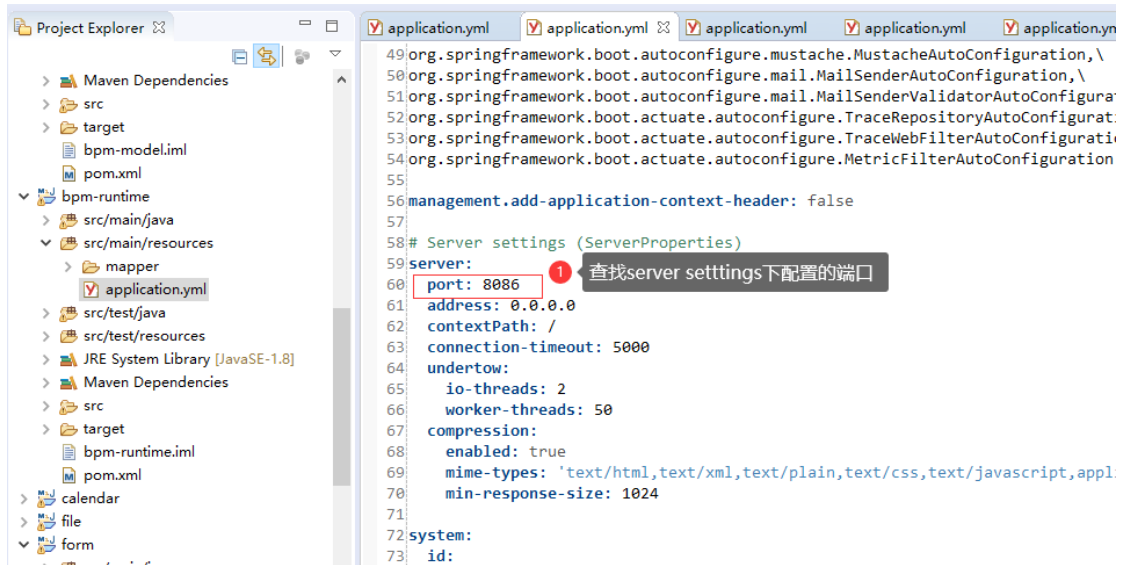


```

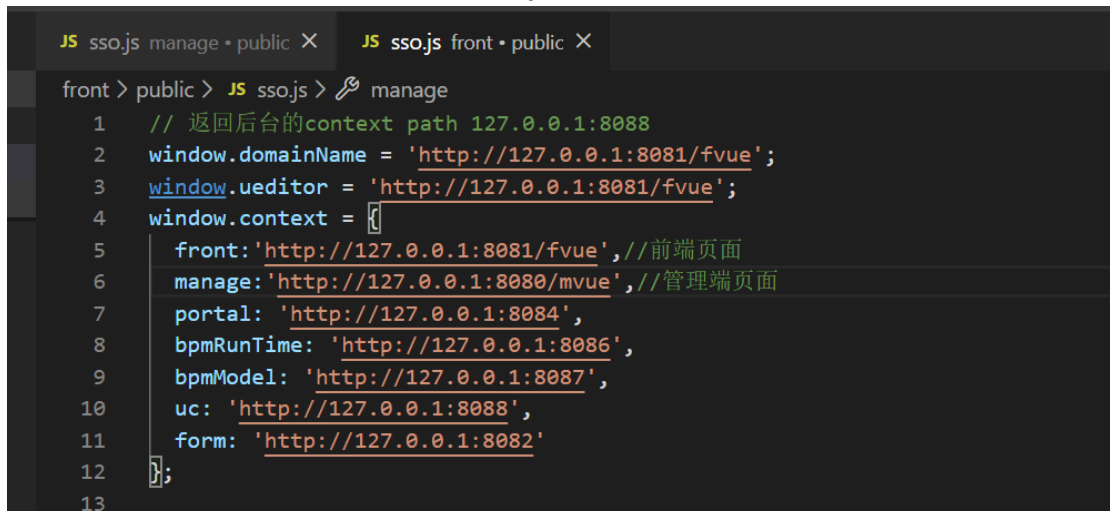
JS sso.js manage • public X JS sso.js front • public
manage > public > JS sso.js > ...
1 // 返回后台的context path localhost:8088
2 window.domainName = 'http://127.0.0.1:8080/mvue';
3 window.ueditor = "/mvue"
4 window.context = {
5   front: 'http://127.0.0.1:8081/fvue', //前端页面
6   mobile: 'http://127.0.0.1:8082/mobilevue', //手机端页面
7   manage: '/mvue', //管理端页面
8   portal: 'http://127.0.0.1:8084',
9   bpmRuntime: 'http://127.0.0.1:8086',
10  bpmModel: 'http://127.0.0.1:8087',
11  uc: 'http://127.0.0.1:8088',
12  form: 'http://127.0.0.1:8082'
13 };
14

```

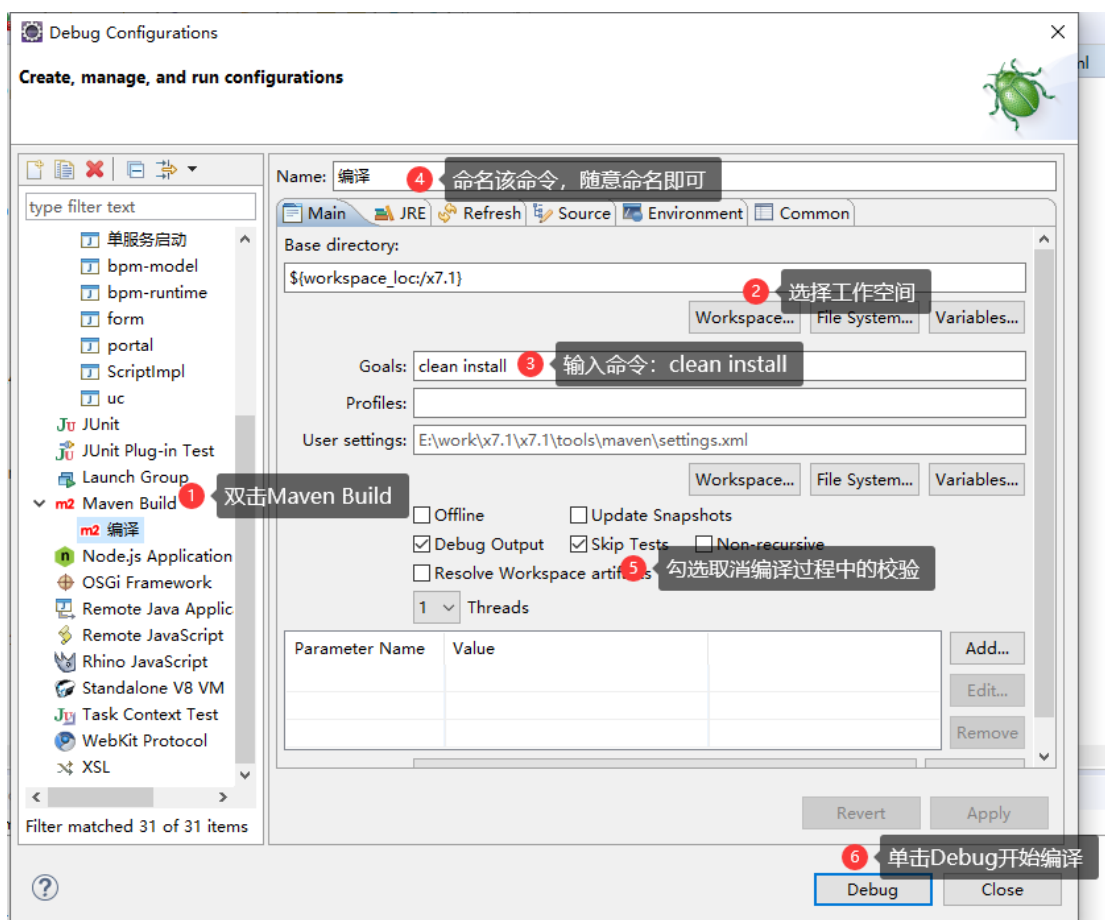
每个服务端口都参考服务下的 application.yml 配置文件，以下以 bpm_runtime 为例



②找到 web 中 front 文件夹下的 sso.js 文件，编辑修改如下地址及端口



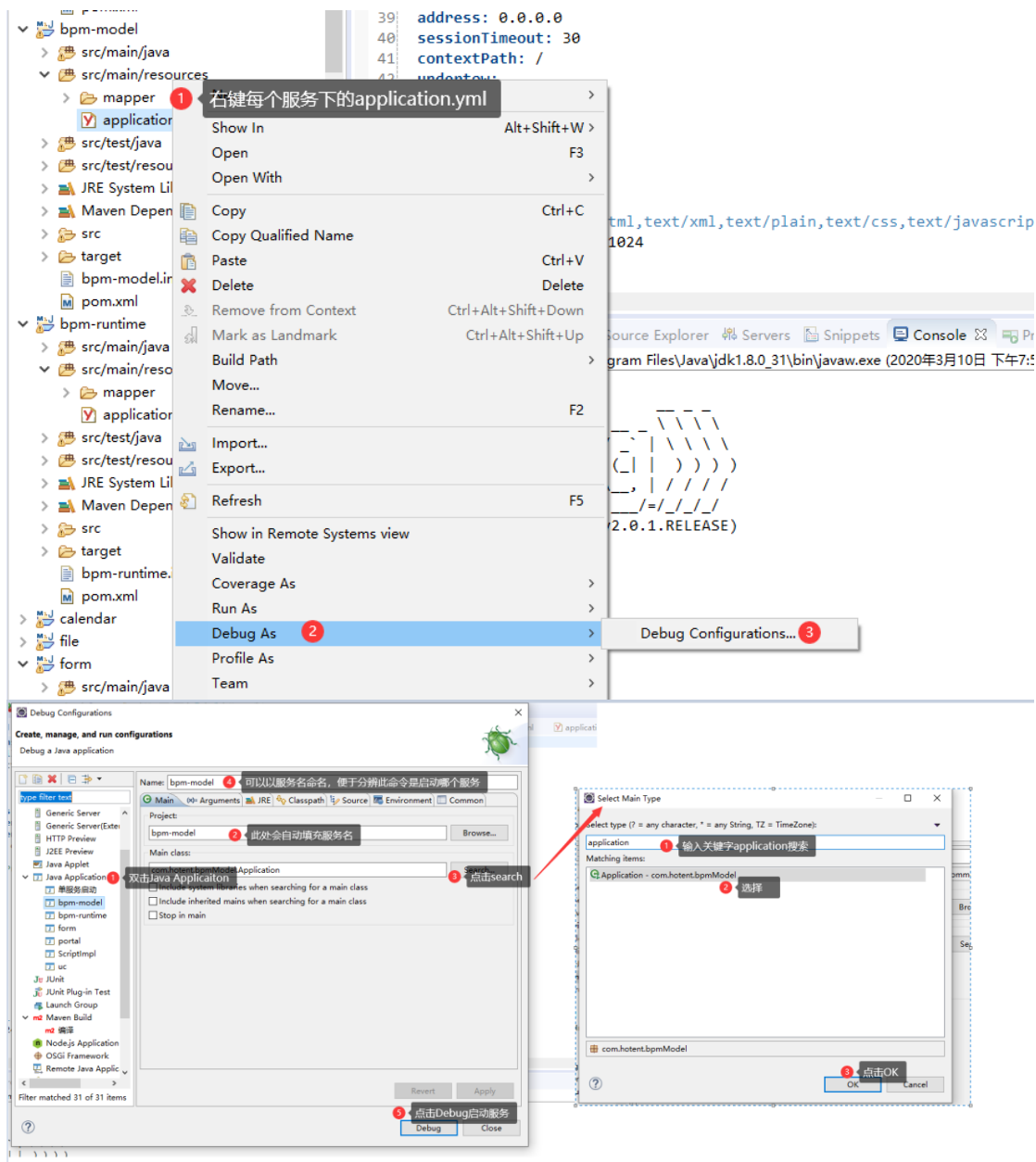
10. 配置 maven 编译命令



11. 配置服务启动命令

每个服务都要进行配置，服务包括：bpm-model,bpm-runtime,form,uc,portal

以下以 bpm-model 的服务启动配置为例：



注：每个服务启动命令配置都参考以上步骤

12. 启动项目

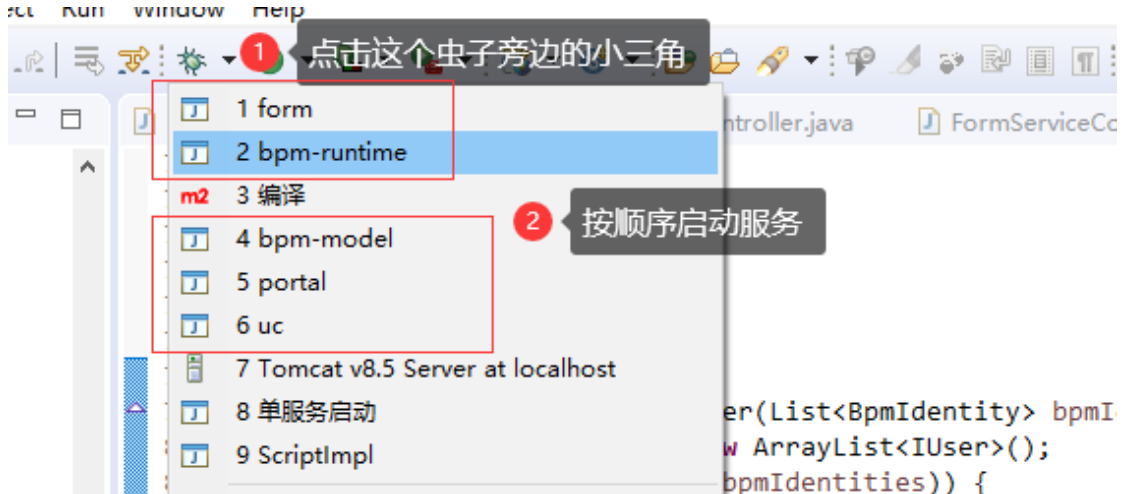
以上配置完成后，依次启动后台服务和前端即可

(注：需保证使用 eurekaServer 是启动状态)

①启动 eurekaServer



②后台服务启动



③前端启动

注：提前安装 node.js，网上搜索 node.js 进入官网，通常下载稳定版，下载安装即可，安装成功后继续以下操作



使用 VS Code 前端启动，依次输入命令

npm install

npm run build

```
npm run serve
```

到达这一步，你就部署成功啦，现在试试浏览器访问看看。

注意事项：需要中间件 ActiveMQ\Redis\Eureka