DADP4.2微服务框架

开发手册

胡砥峰

2021年06月

# 目录

[目录 2](#_Toc70059758)

[文档修订记录 4](#_Toc70059759)

[一、 Eclipse开发工具调优 5](#_Toc70059760)

[1.1 Eclipse使用调优目的 5](#_Toc70059761)

[1.2 Eclipse版本选择建议 5](#_Toc70059762)

[1.3 Eclipse常用插件安装 6](#_Toc70059763)

[1.3.1 安装lombok插件【推荐】 6](#_Toc70059764)

[1.3.2 安装subeclipse插件(SVN) 【按需】 7](#_Toc70059765)

[1.3.3 安装Spring Tools插件【按需】 8](#_Toc70059766)

[1.4 Eclipse使用调优步骤 10](#_Toc70059767)

[1.4.1 修改eclipse.ini的启动参数 10](#_Toc70059768)

[1.4.2 增大Eclipse字体【可选】 10](#_Toc70059769)

[1.4.3 修改eclipse通用参数【可选】 11](#_Toc70059770)

[1.4.4 修改工程编码为UTF-8 12](#_Toc70059771)

[1.4.5 关闭文件校验 13](#_Toc70059772)

[1.4.6 修改properties文件为UTF-8编码 15](#_Toc70059773)

[1.4.7 修改java代码的格式化宽度 16](#_Toc70059774)

[1.4.8 修改javascript代码的格式化宽度 17](#_Toc70059775)

[1.4.9 修改xml代码的格式化宽度 19](#_Toc70059776)

[1.4.10 修改CSS和JSP代码的格式化宽度 20](#_Toc70059777)

[1.4.11 设置java和javascript的代码保存自动格式化 21](#_Toc70059778)

[1.4.12 设置Eclipse默认浏览器【可选】 22](#_Toc70059779)

[1.4.13 修改java代码package展示方式 【可选】 23](#_Toc70059780)

[1.4.14 新建jsp文件的默认编码改成utf-8 【可选】 23](#_Toc70059781)

[1.4.15 Crtl+H快捷搜索【可选】 24](#_Toc70059782)

[1.4.16 Crtl+F快捷搜索【可选】 25](#_Toc70059783)

[1.4.17 设定jdk编译版本【可选】 26](#_Toc70059784)

[1.4.18 增加eclipse的jvm内存参数 【可选】 27](#_Toc70059785)

[1.4.19 加大tomcat启动时间【可选】 27](#_Toc70059786)

[1.4.20 修改svn插件【可选】 28](#_Toc70059787)

[1.4.21 修改maven的仓库地址为阿里云(按需) 29](#_Toc70059788)

[1.4.22 修改eclipse的maven配置(按需) 31](#_Toc70059789)

[1.4.23 减少eclipse的启动项目【可选】 32](#_Toc70059790)

[1.4.24 忽略svn同步的某些目录【可选】 32](#_Toc70059791)

[1.4.25 优化完了之后,保存工程配置【可选】 33](#_Toc70059792)

# 文档修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| 版本编号 | 简要说明（变更内容和变更范围） | 日期 | 变更人 |
| v2021-04-23 | 重新拟稿 | 2021-04-23 | 胡砥峰 |
| v2021-06-17 | 增补内容 | 2021-06-26 | 胡砥峰 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 概述

## 研发目标

* 建设一个真正的**为程序员设计**的快速开发平台。
* 集成目前最主流的开发技术、为适应**初中级开发人员**的水平做综合取舍（舍难取易）。
* 定制统一开发规范，过去10多年历经几百个项目实践。
* 采用独特的Excel数据库设计模式和java代码生成模式。
* **轻松工作，摆脱加班。这个不是开玩笑，我们很多项目都做到过**。
* 提高用户满意度、降低开发成本、缩短开发周期、统一企业技术框架。

## 架构理念

* 选择用户数量最多的、开源的、轻量级组件。
* 选用大厂出品的、有前瞻性、趋势所向的组件。

## 编写目的

* 让DADP微服务开发平台的使用者尽快熟悉该工具的用途和使用方法。
* **规范！规范！规范！重要的事情说3遍！！！**
* 无规矩不成方圆，系统开发设计是一个团队工程。项目组必须遵循一致的开发规范，设计要求。
* 约束系统设计过程中命名规范，**这是代码自动生成的前提**，是提高开发效率的最佳体验，是快速交付项目的最佳法宝。

## 适用范围

* 开发人员
* 架构师
* 项目经理
* 其他需要了解开发平台的人员。

## 功能范围

* 系统框架核心代码的封装。
* 系统管理：含参数管理、代码类别、代码信息、菜单管理、功能管理、机构管理、角色管理(操作权限和数据权限)、用户管理、日志查询。
* 代码生成工具
* 代码API（swagger）
* 工作流

## 名词解释

* DADP：分布式应用开发平台（Distributed application development platform）。

## 您的建议

* 由于客观原因的限制，本文档可能考虑不够周详，我们会持续改进。假如您对本开发平台有任何意见，请按以下方式去联系：
* 邮件地址：difeng.hu@gientech.com
* 加微信：13666008179 胡砥峰，进入微信群。

# Eclipse开发工具调优

## Eclipse使用调优目的

* 工欲善其事必先利其器！
* 所有的代码都**必须格式化**！包括JAVA、xml、vue、js、CSS等

## Eclipse版本选择建议

* 本平台一般采用最新版的Eclipse For Java EE! 原因: 版本越新,功能更强。
* 当前采用版本：eclipse-jee-2020-12-R-win32-x86\_64.zip (版本编号：4.18.0；可通过打开Eclipse解压目录的 .eclipseproduct 文件看到)。
* 使用Idea开发工具的同事，一定要在代码格式化上和eclipse保持一致！我本人认为eclipse+STS的效果足够使用！而且eclipse更新更快。

## Eclipse常用插件安装

### 安装lombok插件

**插件用途：**启动时候用lombok.jar 实体类不用写 get和set方法。方便。目前GitHub很多项目都用了lombok

* 步骤1：**复制下面的**lombok1.18. 16.jar 文件到Eclipse的根目录。

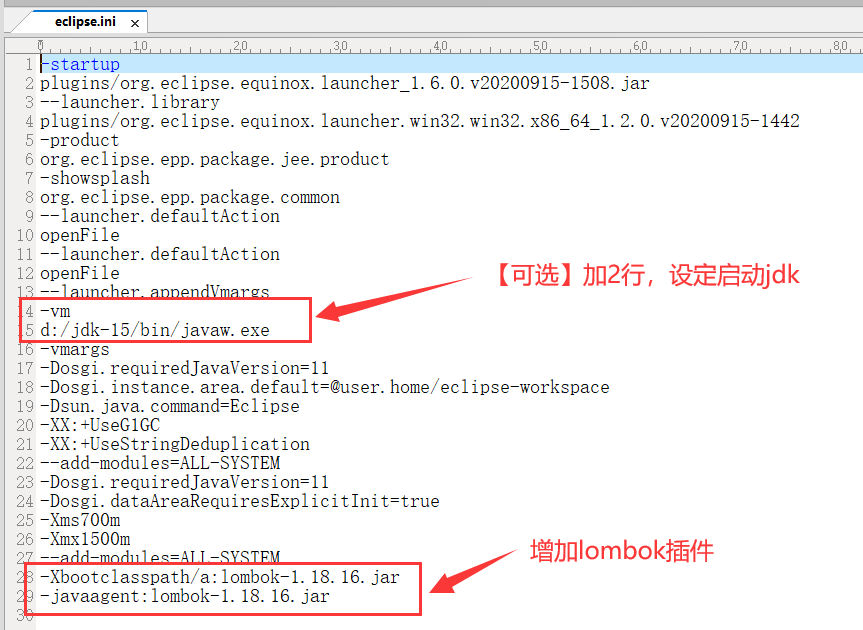




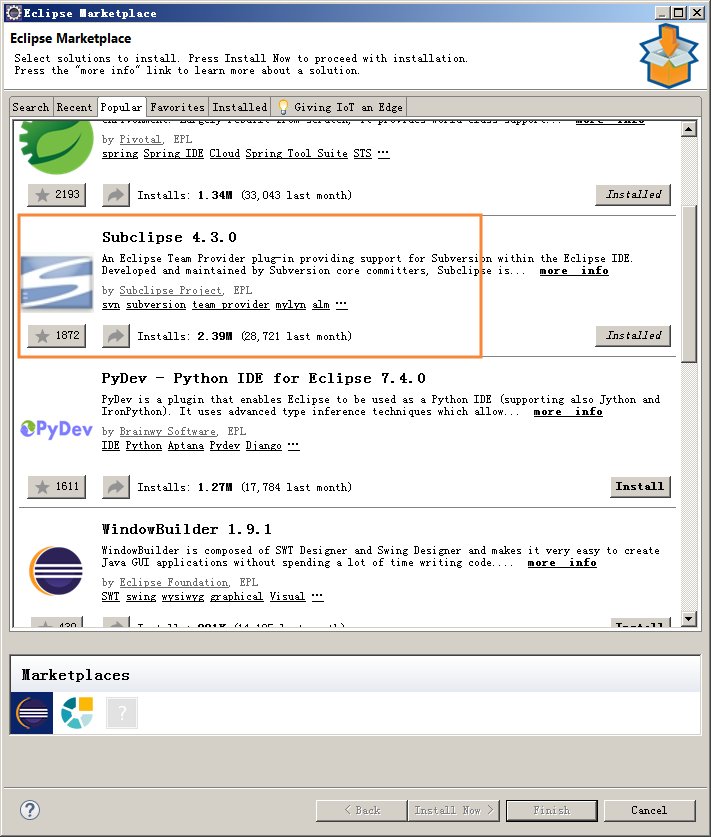
* 步骤2：在eclipse.ini最底下加上下面2行

-Xbootclasspath/a:lombok1.18.16.jar

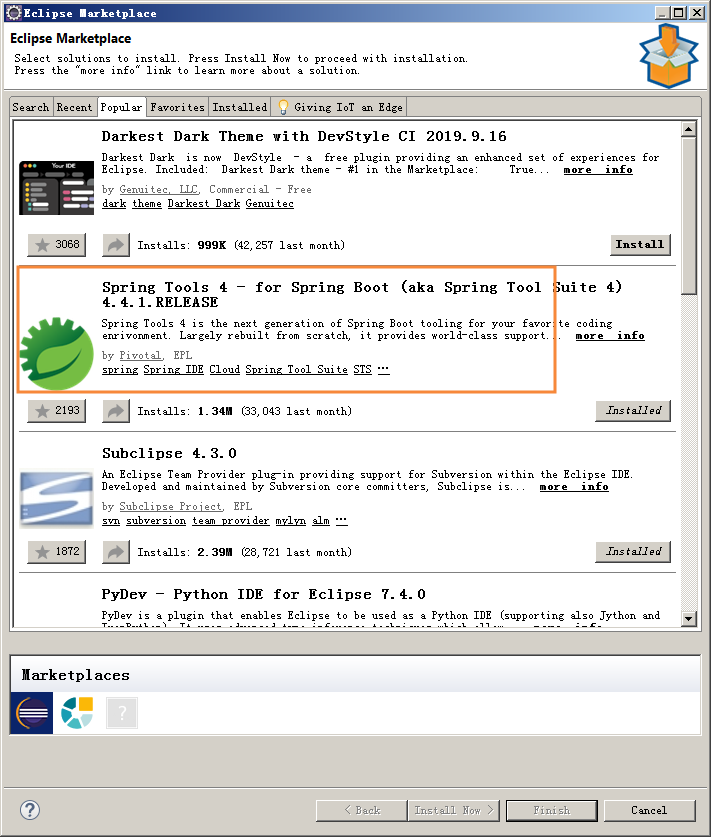
-javaagent:lombok1.18.16.jar



### 安装subeclipse插件(SVN)



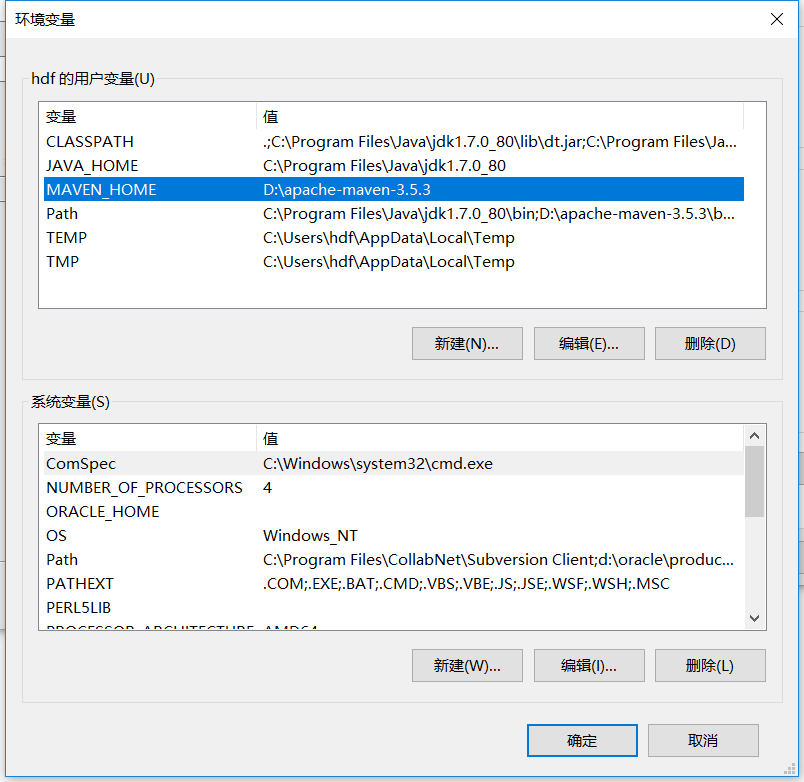
### 安装Spring Tools插件



## Maven配置和修改国内仓库

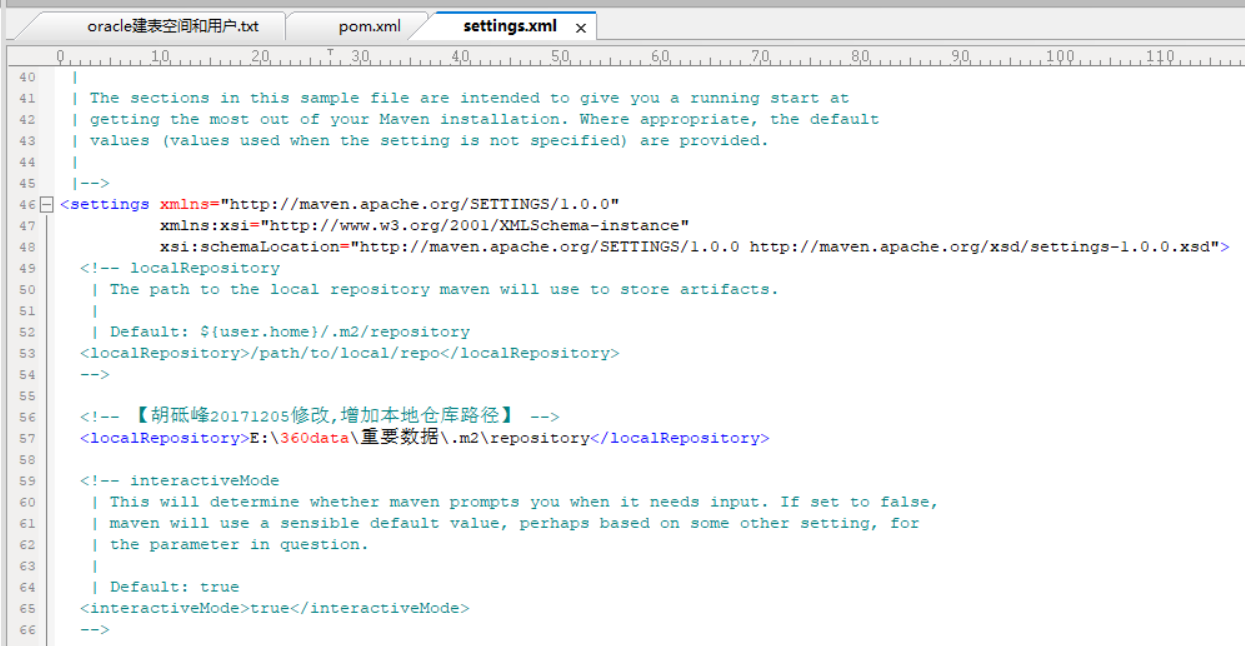
### 修改maven的仓库地址为阿里云

* 在环境变量中增加MAVEN\_HOME。



* 修改maven的配置settings.xml的3个地方。

1. 本地仓库路径。

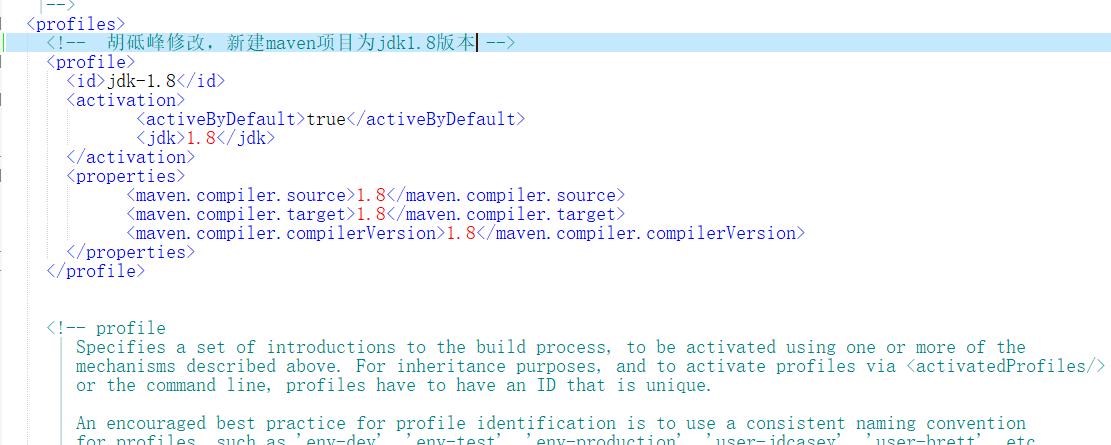


1. 改为阿里云远程仓库



|  |
| --- |
| <mirror>  <id>alimaven</id>  <name>aliyun maven</name>  <url>http://maven.aliyun.com/nexus/content/groups/public/</url>  <mirrorOf>central</mirrorOf>  </mirror> |

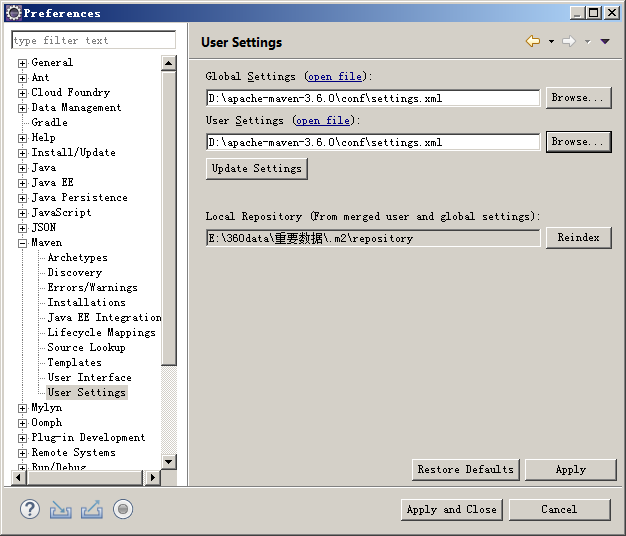
1. 新建项目默认jdk1.8



|  |
| --- |
| <profile>  <id>jdk-1.8</id>  <activation>  <activeByDefault>true</activeByDefault>  <jdk>1.8</jdk>  </activation>  <properties>  <maven.compiler.source>1.8</maven.compiler.source>  <maven.compiler.target>1.8</maven.compiler.target>  <maven.compiler.compilerVersion>1.8</maven.compiler.compilerVersion>  </properties>  </profile> |

### 修改eclipse的maven配置

* 选完本地settings.xml后，最好点击一下【update settings】和【Reindex】按钮。



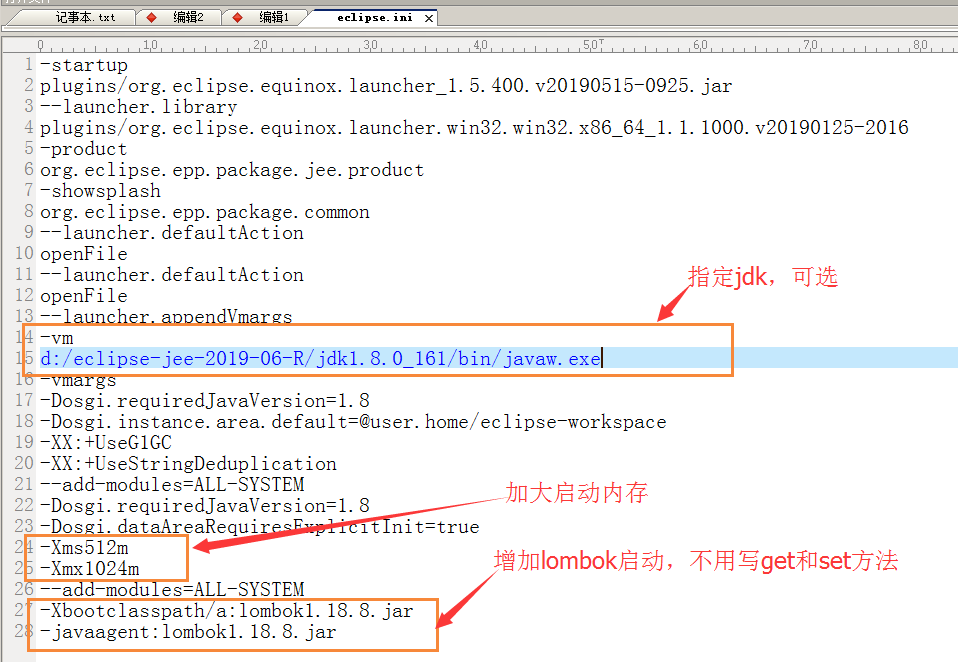
## Eclipse使用调优步骤

### 修改eclipse.ini的启动参数

修改eclipse.ini的启动参数,加大eclipse.ini中的内存大小,希望能运行更快一点。

指定启动JDK【可选，我本机有jdk1.8和jdk-15的工程】

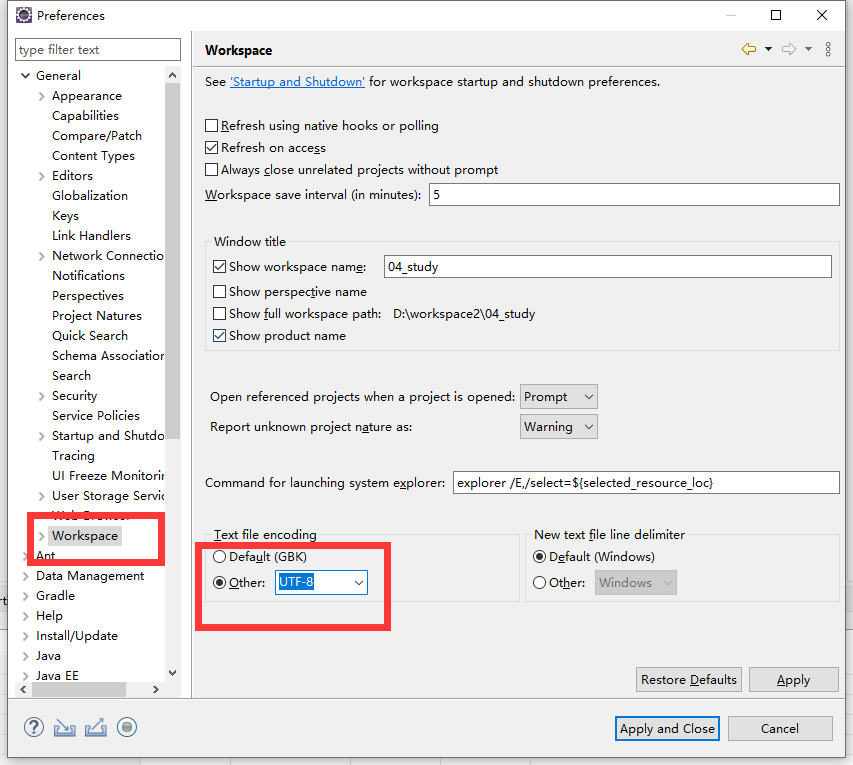
增加lombok启动jar包。



### 修改工程编码为UTF-8

考虑到未来工程会部署到不同的环境,项目工程中所有文件首选utf-8编码.

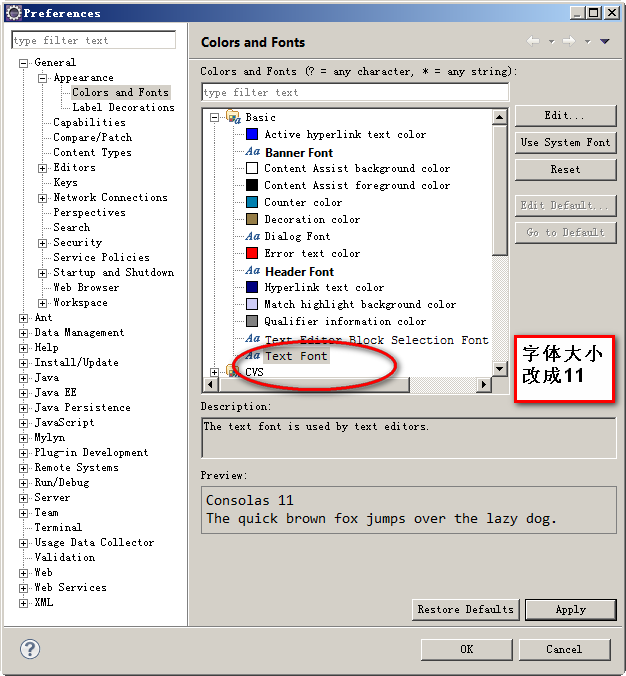
* 打开Windows > Preferences > General > Workspace :



### 增大Eclipse字体【可选】

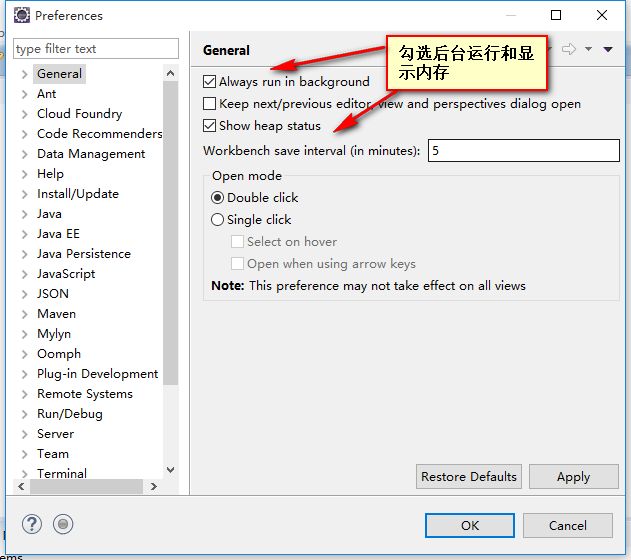
编辑器字体大小, 新版本的Eclipse显示字体有点小,可以按下图方式放大一号

* 打开Windows > Preferences > General > Appearance > Colors and Fonts > Basic > Text Font :



### 修改eclipse通用参数【可选】

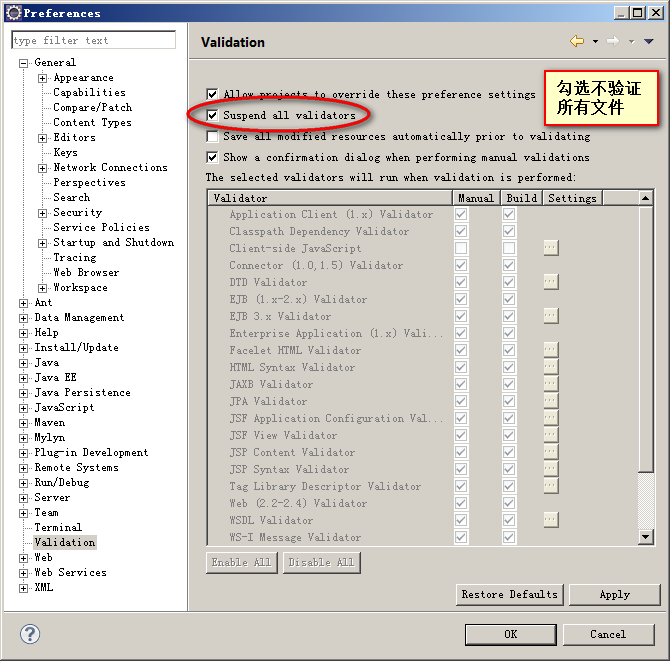
* 后台运行和右下角显示当前eclipse内存状况



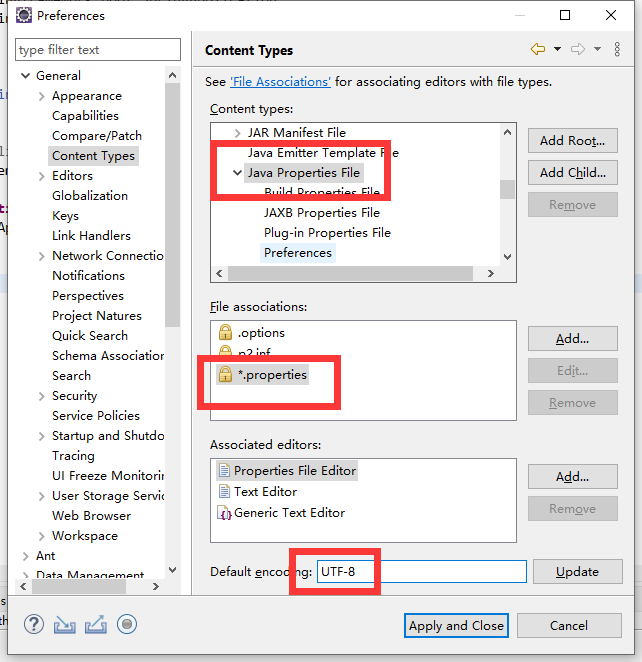
### 关闭文件校验

勾选关闭文件校验,要不然eclipse每次保存文件都会去自动校验,开发会非常慢.

* 打开Windows > Preferences > Validation:



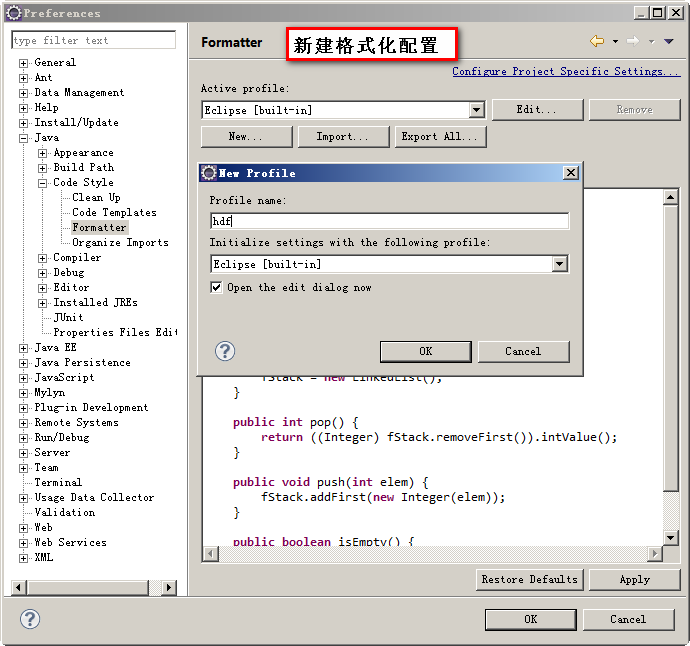
### 修改properties文件为UTF-8编码



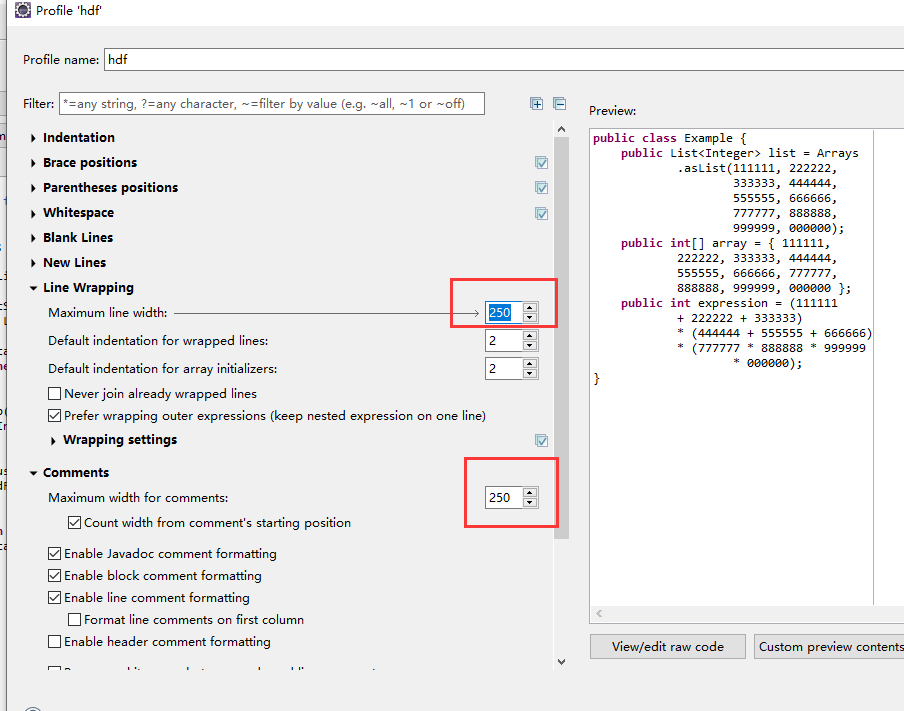
### 修改java代码的格式化宽度

为了代码更加美观，把java代码格式化的默认宽度80改为250.

* 打开Windows > Preferences > Java > Code Style > Formattor :



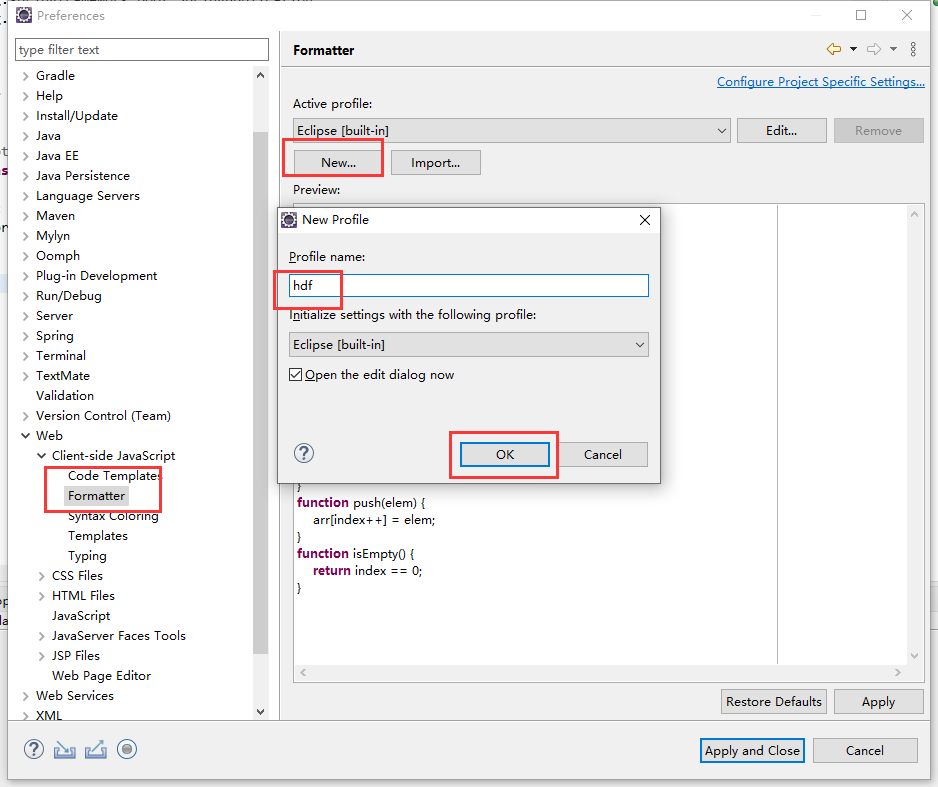
* Java代码宽度，改成每行250个字符：
* Java代码的注释宽度，改成每行250个字符：



### 修改javascript代码的格式化宽度

为了代码更加美观，把javascript代码格式化的默认宽度80改为250.

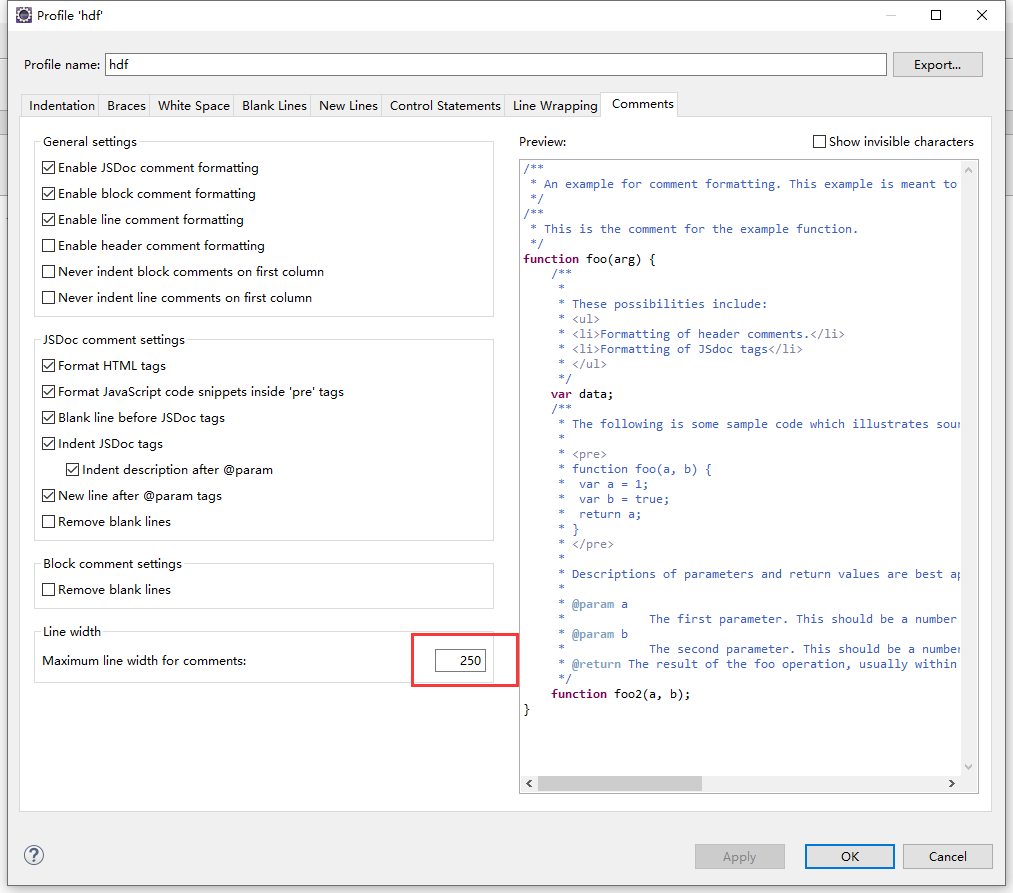
* 打开Windows > Preferences > Web> Client-side JavaScript > Formattor :



* Javascript代码宽度，改成每行250个字符：



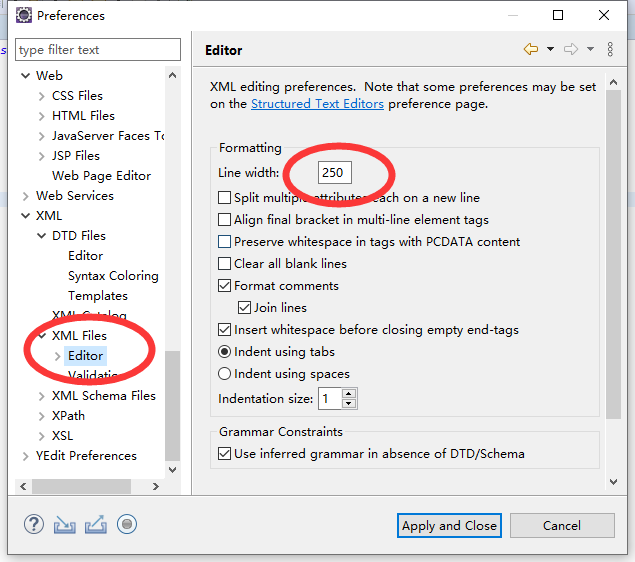
Javascript代码的注释宽度，改成每行250个字符：



### 修改xml代码的格式化宽度

为了代码更加美观，把xml代码格式化的默认宽度72改为250.

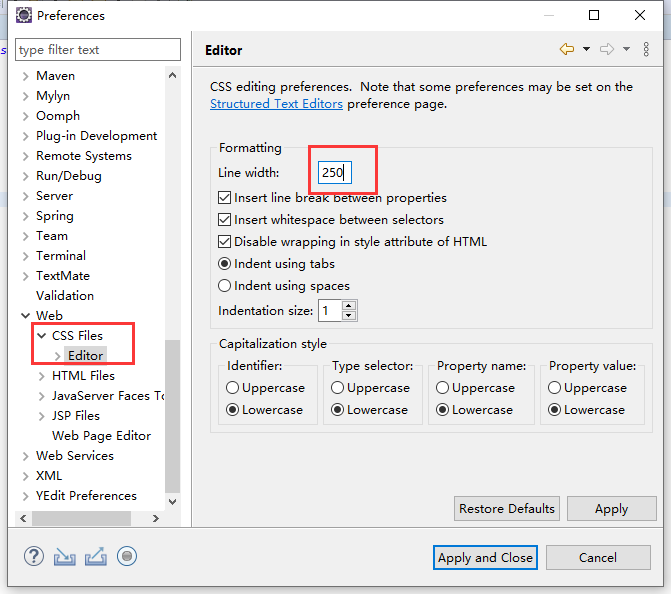
* 打开Windows > Preferences > Xml > Xml Files > Editor :



### 修改CSS、HTML代码的格式化宽度

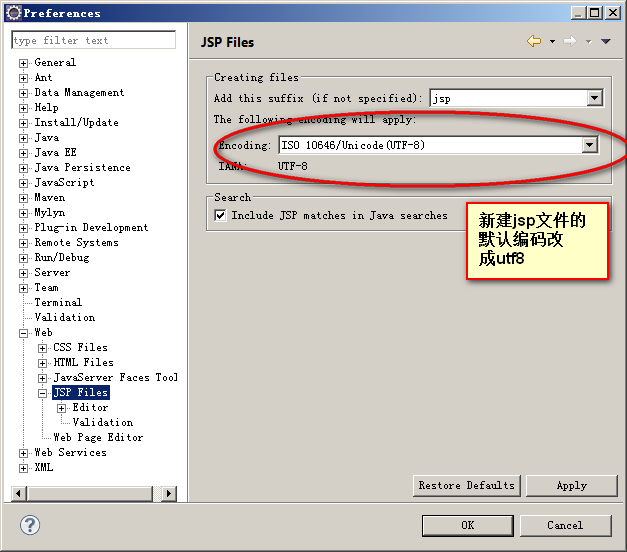
为了代码更加美观，把xml和html代码格式化的默认宽度72改为250（JSP也是依赖这2个格式化配置，所以只要改这2个）.

* 打开Windows > Preferences > Web > CSS Files > Editor :
* 打开Windows > Preferences > Web > HTML Files > Editor :



### 新建jsp文件的默认编码改成utf-8

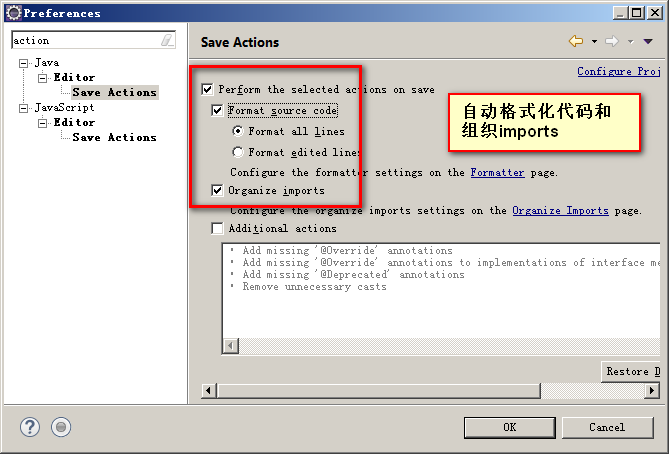
* 日常开发时，基本上都是copy原来的JSP页面来改，一般用不上.



### 设置java代码保存自动格式化

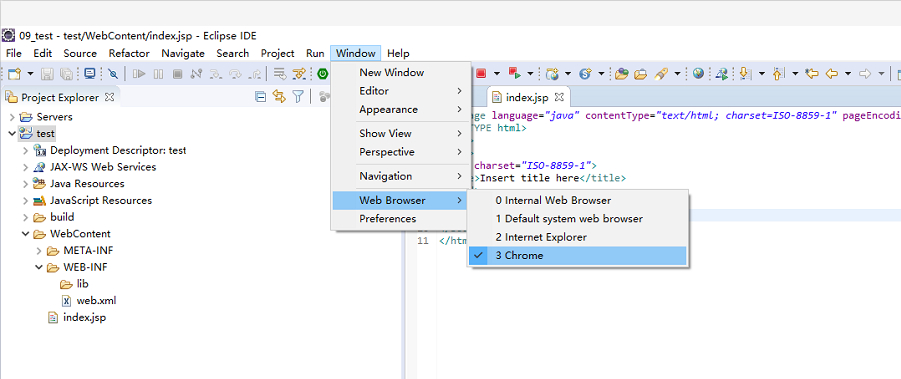
修改配置，使文件保存的时候， 自动把java代码格式化.

* 打开Windows > Preferences > Java > Editor > Save Actions :
* 打开Windows > Preferences > Javascript > Editor > Save Actions :



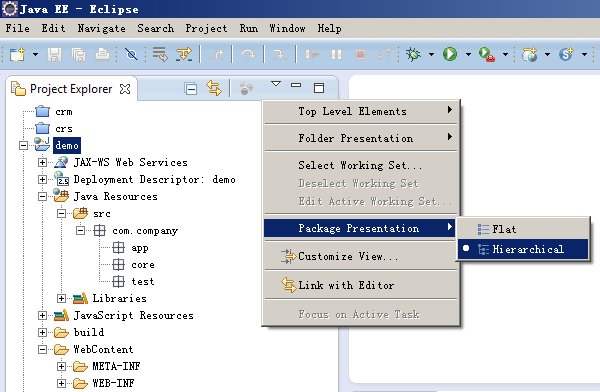
### 设置Eclipse默认浏览器【可选】

* 开发过程中，选择工程，右键run as之后,会自动调用浏览器打开首页。此步骤是个人习惯,可不改。



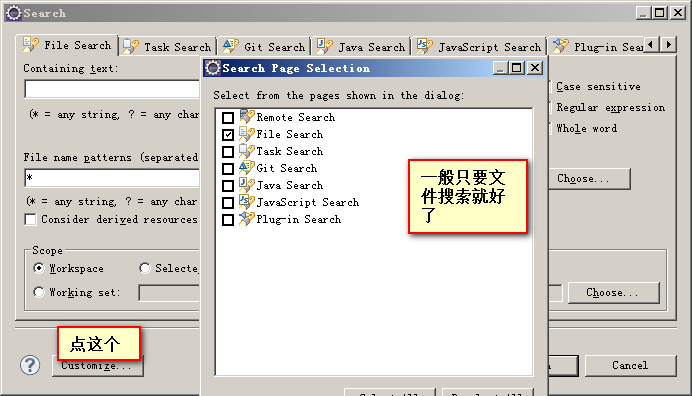
### 修改java代码package展示方式 【建议】

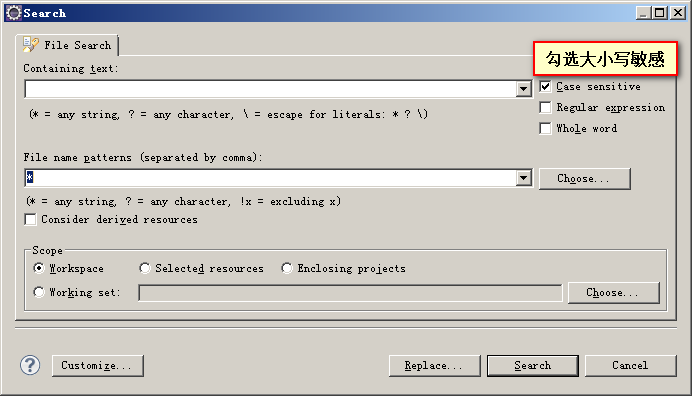
* 修改java代码package展示方式.此步骤是个人习惯,可不改.



### Crtl+H快捷搜索【可选】

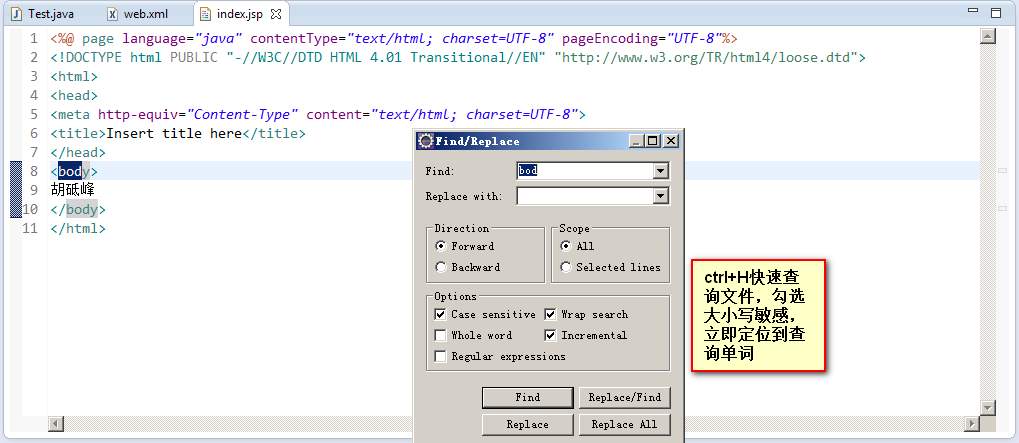
* Crtl+H快捷搜索,一般只需要文件搜索。这是一个很常用的动作。用于搜索整个工程。



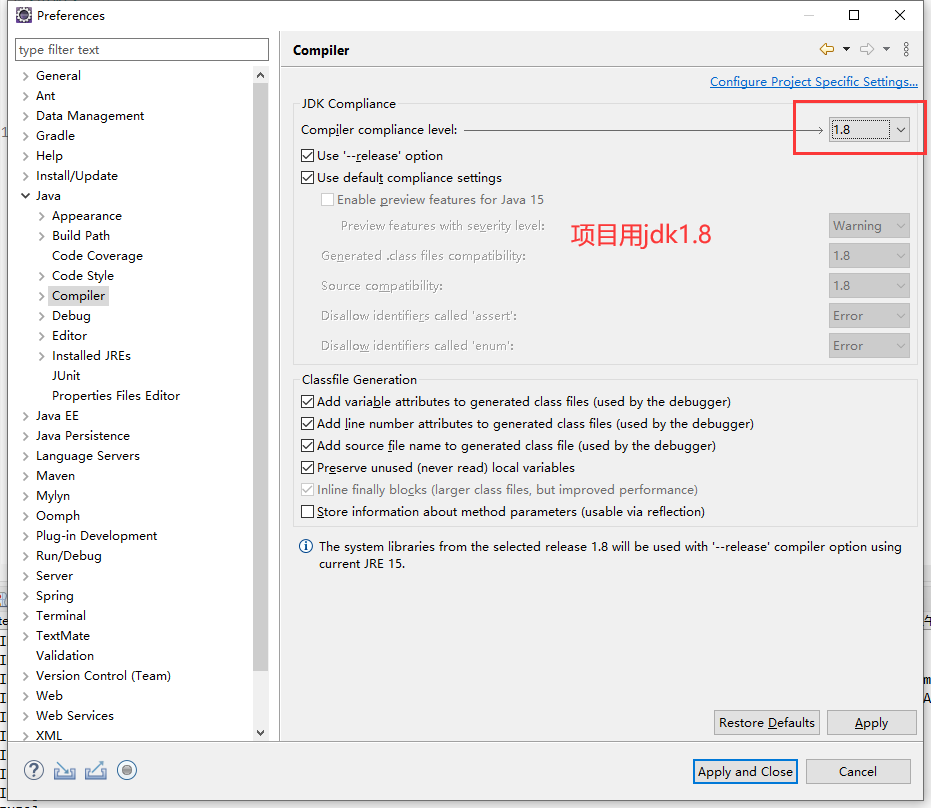


### Crtl+F快捷搜索【可选】

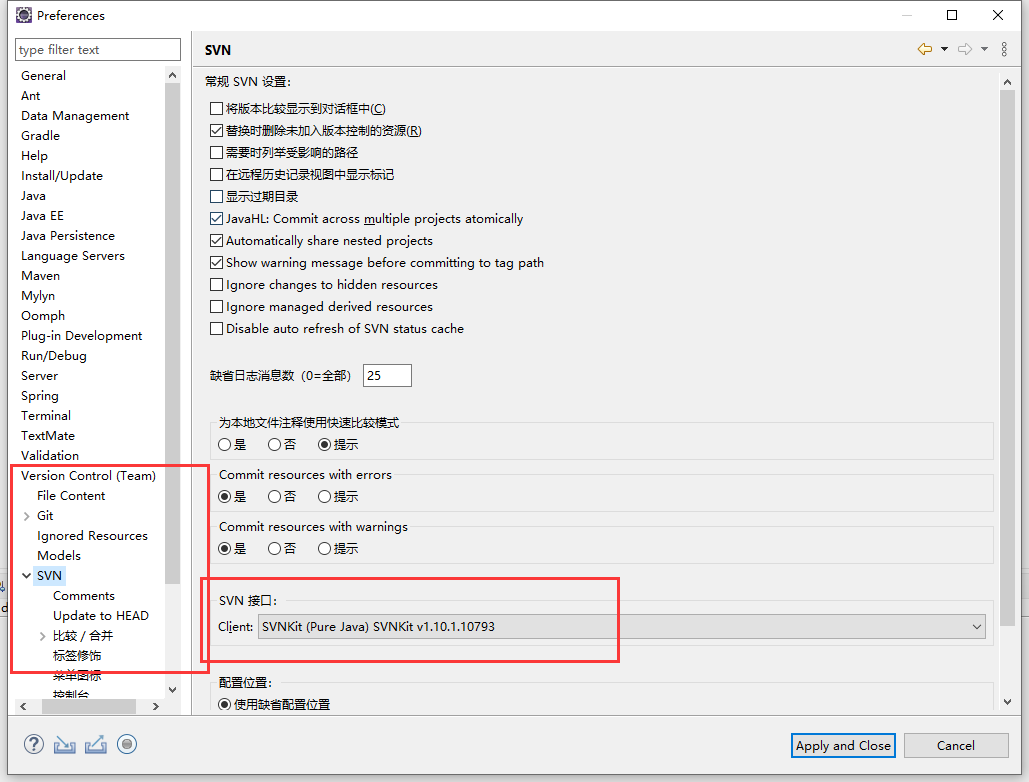
* Crtl+ F快速查询关键字,勾选大小写敏感,立即定位到查询单词。这是一个很常用的操作。用于搜索当前页面。
* 快捷键Crtl+K和Crtl+Shift+K操作也受这个大小写影响。



### 设定jdk编译版本【按需】



### 修改svn插件【可选】



### 忽略svn同步的某些目录【可选】

1. 打开菜单

Window>Preferences>Version Control(Team)>IgnoredResource

2、 点击Add Pattern…依次添加如下内容

\*/.classpath

\*/.project

\*/target

\*/.settings

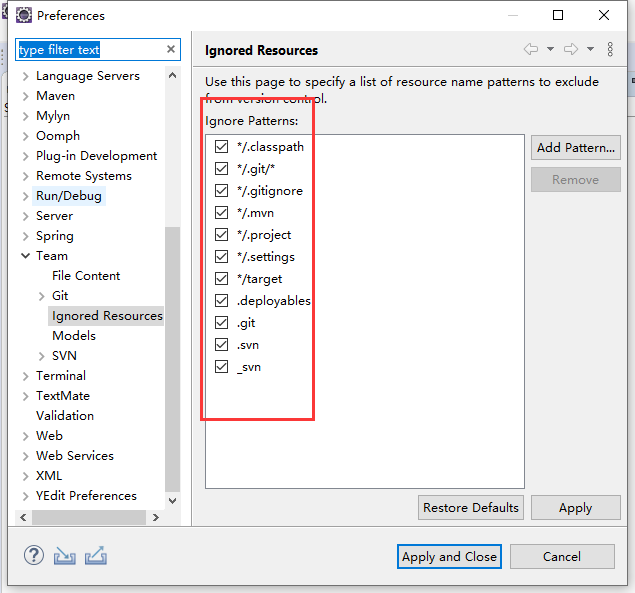
\*/.mvn

\*/.gitignore

\*/.svn

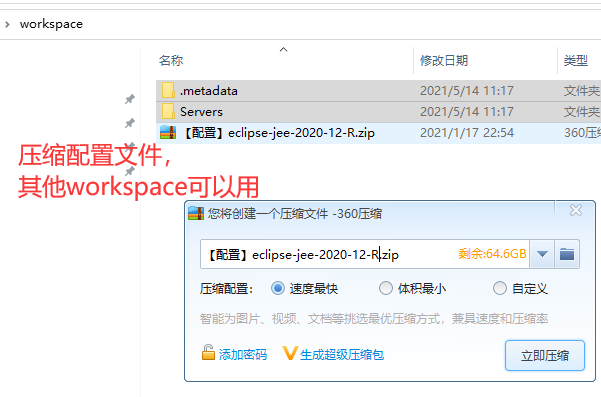
\*/node\_modules

3、 点击保存即可看到效果



### 优化完了之后,保存工程配置【可选】

找到workspace的.metadata目录，我们上面修改的所有配置，都保存在这个目录,压缩保存这个目录,方便以后新开workspace时使用(copy过去，不用重新设置一遍)。



# 设计开发规范

无规矩不成方圆，系统开发设计是一个团队工程。项目组遵循一致的开发规范，设计要求。 这些都是快速、高质量的交付的必要条件

## 总体命名约定

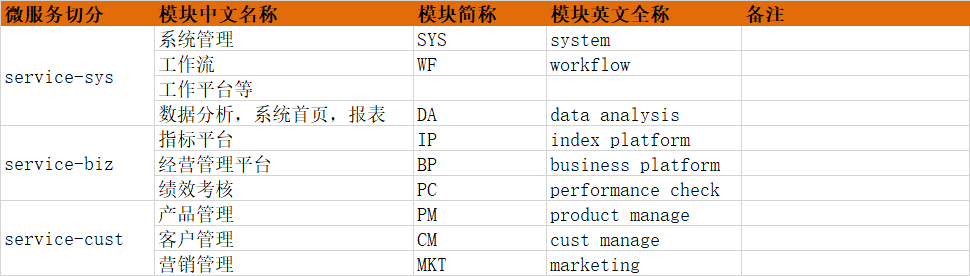
1. 数据对象命名统一用大写字母（包括：表、视图、函数、存储过程、索引、序列）。
2. 命名规范：**“前缀”+“\_”+“模块缩写”+“\_”+“业务对象含义缩写”**
3. 在定义数据库表、视图、存储过程等数据对象时，统一采取三段式命名方式，保证此数据对象命名在系统范围内唯一。

|  |  |  |  |
| --- | --- | --- | --- |
| **数据对象类型** | **前缀** | **举例** | **举例备注** |
| 表 | T | T\_SYS\_USER  T\_SYS\_ORG  T\_SYS\_CODE\_TYPE | 后台数据ETL要用2个字母T（X），X表示数据分层 |
| 视图 | V | V\_SYS\_USER |  |
| 存储过程 | P | P\_JX\_MAIN  P\_JX\_I1ToI2 | 绩效主过程  绩效接口1层加工到接口2层 |
| 函数 | F | F\_SYS\_GET\_NEXT\_DAY | 得到下一天 |
| 索引 | IDX | IDX\_SYS\_USER\_SEX（单字段）  IDX\_SYS\_USER\_1（多字段索引） | 1. IDX+表名+字段名 2. IDX+表名+编号 |
| 序列Sequence | SEQ | SEQ\_SYS\_USER\_ID  SEQ\_SYS\_LOG\_ID | 系统管理的用户表主键Sequence |

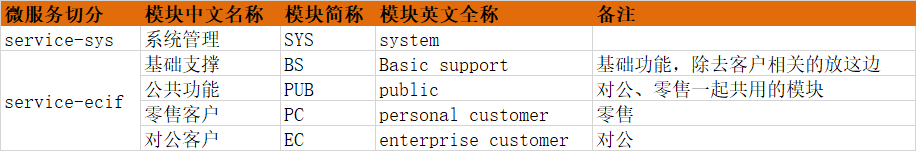
1. **模块缩写：**一般采用2~5个字母(一般是**2-3**个)，尽量用英文缩写，次选用拼音。
2. 常用模块的编号如下(模块编号为项目内部自行约定区分，同项目开发人员一看就明白属于什么模块，不要盲目追求命名优雅)：

|  |  |  |
| --- | --- | --- |
| **模块中文名(曾经用过的)** | **模块缩写** | **备注说明** |
| 系统管理 | SYS | System的缩写 |
| 工作流 | WF | Work FLow的缩写 |
| 代码示例 | DEMO | 代码示例 |
| 对公客户管理 | CCM | 对公客户管理 |
| 零售客户营销 | PCM | 个人客户管理 |
| …… | …… | …… |

* 参考1



* 参考2



1. **形容词后置：**不管是表名，还是字段名，设计的时候，建议形容词后置。

例如1：AMT\_USD美元余额、AMT\_RMB人民币余额

例如2：T\_EC\_REL\_PER 对公客户和个人客户关系表

T\_EC\_REL\_ENT 对公客户和对公客户关系表

## 字段命名约定

1. 所有字段命名为大写字母
2. 字段命名格式：单个单词简写为1段，2个单词必须用下划线分开。比如用户ID字段必须用USER\_ID，不能用USERID。
3. 不能使用ID作为表的主键，必须带上业务含义，在系统模块出现时，方便理解且尽量保证唯一性。例如：T\_SYS\_LOG表，主键必须为LOG\_ID；T\_SYS\_USER表的主键必须为USER\_ID.
4. 所有的主键和外键都必须以“\_ID”结尾，不能使用CUST\_NO(客户号)作为主键。
5. 不可以建组合主键！通过组合唯一索引进行约束。
6. 所有的外键命名必须和它作为主键的命名一致(**可以加前缀**)。例如：ORG\_ID（机构ID）在T\_SYS\_USER表中会出现，在机构表和用户表的该字段都必须是ORG\_ID。还有在业务表中会有以下常用字段：

|  |  |  |
| --- | --- | --- |
| **字段英文名** | **字段中文名** | **备注说明** |
| **IS\_DEL varchar(1)** | **是否删除** | **强制要求！！逻辑删除标志。** |
| **VER int(8)** | **版本号** | **需要编辑的表，一定要加这个！乐观锁** |
| CREATE\_USER\_ID | 创建人 | 32位String |
| CREATE\_ORG\_ID | 创建机构 | 32位String |
| CREATE\_TIME | 创建时间 | DateTime |
| UPDATE\_USER\_ID | 修改人 | 业务有要求才加该字段。32位String |
| UPDATE \_ORG\_ID | 修改机构 | 业务有要求才加该字段。32位String |
| UPDATE \_TIME | 修改时间 | 业务有要求才加该字段。DateTime |

1. 字段命名尽量用常用的英文单次或缩写(CET4以内)，我们的意图是让设计更容易理解。实在想不出英文或英文单词太长，可以用拼音。
   * 例如：岗位、职位： 可以用JOB\_TITLE(工作头衔)
2. 字段要进行排序（一般按重要性），按便于理解的顺序放置。

## 系统设计要求

1. 数据库设计采用Excel进行(采用2007以上版本)，保留每次变更记录。不使用PowerDesigner和ERWin的原因是：Excel管理更加方便，可以描述的内容更多，代码自动生成工具将解析这个Excel设计。**注意：每个模块一个Excel。**
   * **EXCEL设计路径：/dadp-codeCreate/数据库设计/数据库设计【SYS\_系统管理】.xlsx**

效果图如下：





1. 按数据出现的顺序设计表。关联表，主从表尽量放在一起，看上去层次清楚。
   * 例如：现有菜单表，才会有按钮表。
   * 例如：用户表要选机构和角色，所以机构表，角色表要放前面。
   * 例如：用户登录后才记录日志，日志表要在用户表后面。
2. 在数据库中不创建外键（国内开发通用）。影响性能，不方便开发测试。基于字段命名来识别外键。
3. 为了保证数据的严谨性，加强业务字段的唯一索引和组合字段唯一索引的设计。
   * 例如：手机商品条码GOODS\_ID在商品表是唯一的，在销售表中GOODS\_ID是外键，但我们系统并不创建外键，所以为了保证销售数据的唯一性，我们要为该字段设计唯一索引（IDX\_XXX\_SALE\_01）。
4. 对于字符串，使用固定长度（一般不用不定长度），如果字节>4000的，要使用CLOB字段。
5. 同一个字段在多个表出现时，命名**必须一致！必须一致！必须一致！**

## 代码包路径和文件命名规范

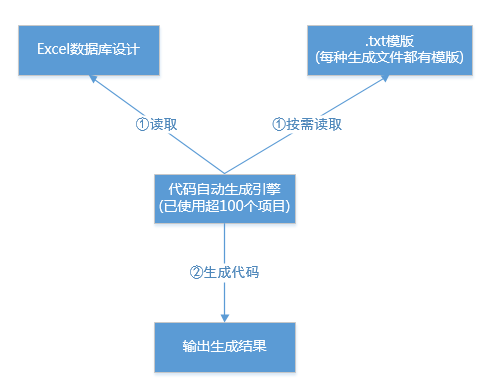
1. 代码自动生成工具，根据数据库表名生成相关代码。其代码命名规范也由表名倒推而得。
2. 以T\_SYS\_CODE\_TYPE表为例（代码类别表，就是下拉框数据字典类别）：

|  |  |  |
| --- | --- | --- |
| **代码分类** | **文件名和放置位置** | **备注说明** |
| 数据库实体类 | src/com/gientech/sys/codeType/ SysCodeType.java |  |
| 查询结果类VO | src/com/gientech/sys/codeType/ SysCodeTypeVO.java | 用于查询条件 |
| 各种DTO | src/com/gientech/sys/codeType/ SysCodeTypeDTO4List.java  SysCodeTypeDTO4Save.java  SysCodeTypeDTO4Update.java  SysCodeTypeDTO4Delete.java |  |
| Rest控制器 | src/com/gientech/sys/codeType/ SysCodeTypeController.java |  |
| 业务Service类 | src/com/gientech/sys/codeType/ SysCodeTypeService.java |  |
| MybatisPlus的mapper | src/com/gientech/sys/codeType/ SysCodeTypeMapper.java |  |
| Mybatis的mapper | src/com/gientech/sys/codeType/ SysCodeTypeMapper.xml |  |

# 代码自动生成

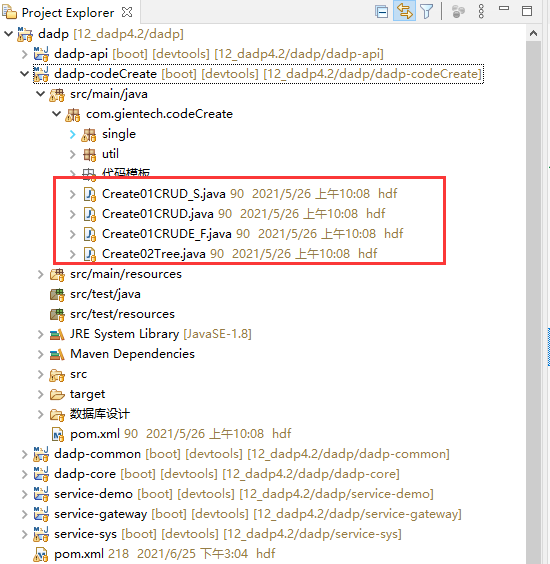
## 实现原理

基于上文的Excel数据库设计，系统定制模版，由解析引擎负责生成代码到指定目录。**在本项目中生成模版和输出结果会有一定的变化，目前只是举例说明。**

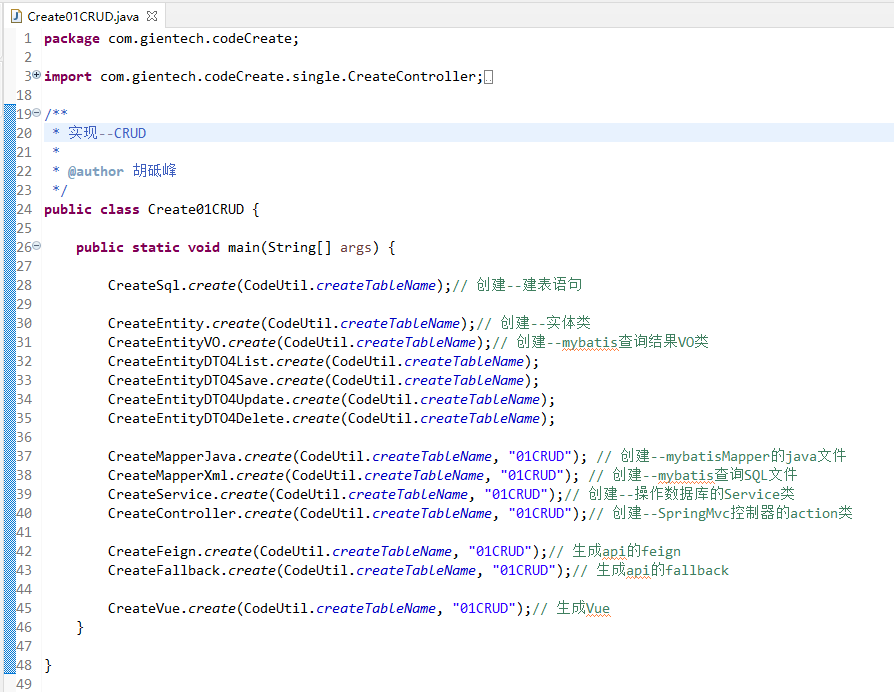


## 实现功能说明

**代码生成引擎结构：**



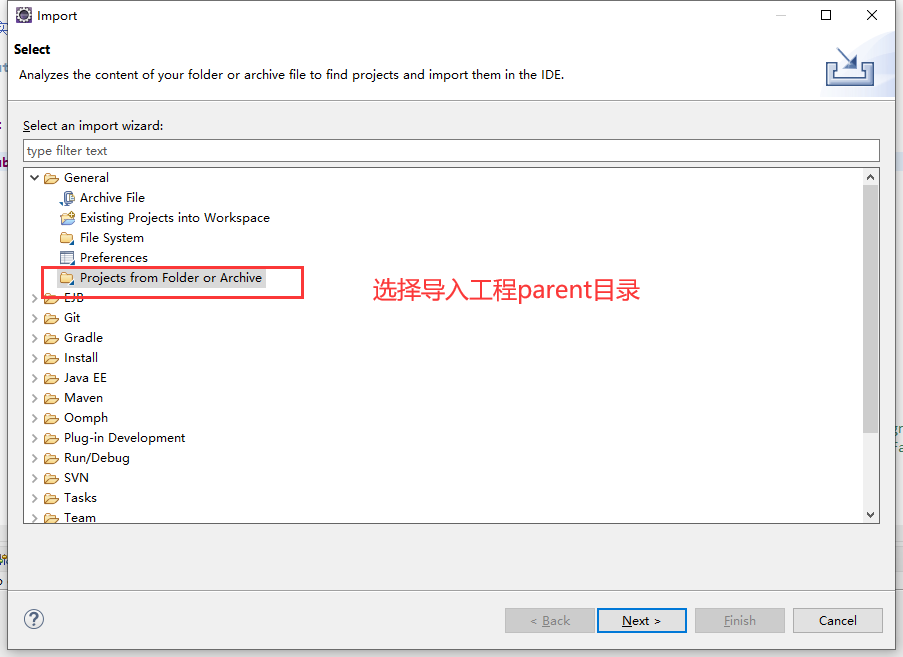
1. **主要实现功能有(参考上图)：**
   * Create01CRUD， 生成增删改查代码。
   * Create01CRUD\_S， 生成增删改查代码(带排序)。
   * Create01CRUDE\_F，生成增删改查导出，有数据权限、能排序。
2. **以**Create01CRUD**.java为例(参考下图)**



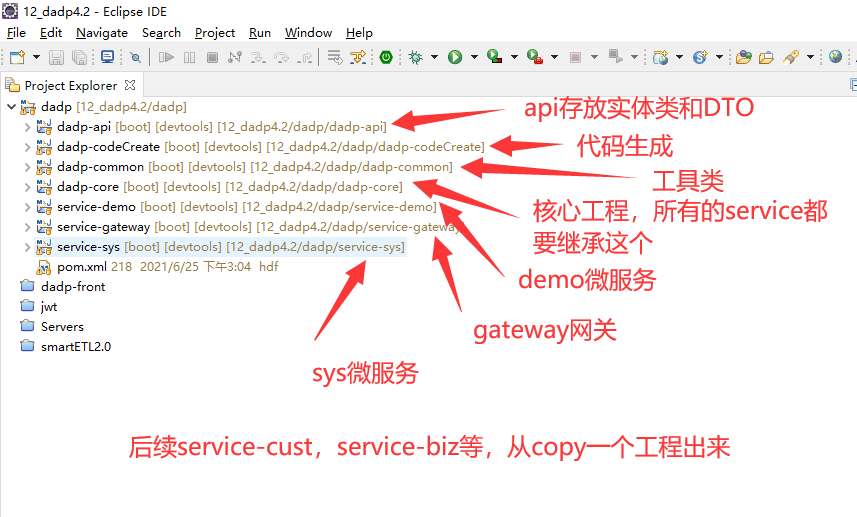
* + 生成建表语句(根据不同数据库)
  + 生成数据库实体类、DTO类
  + 生成Rest的控制器文件
  + 生成数据操作层Service
  + 生成mybatis查询mapper文件。
  + 生成mybatis-plus的java mapper文件
  + 以上生成的代码，一般直接就能运行。**为了美观，请格式化代码。**

# 快速开始

## 导入工程



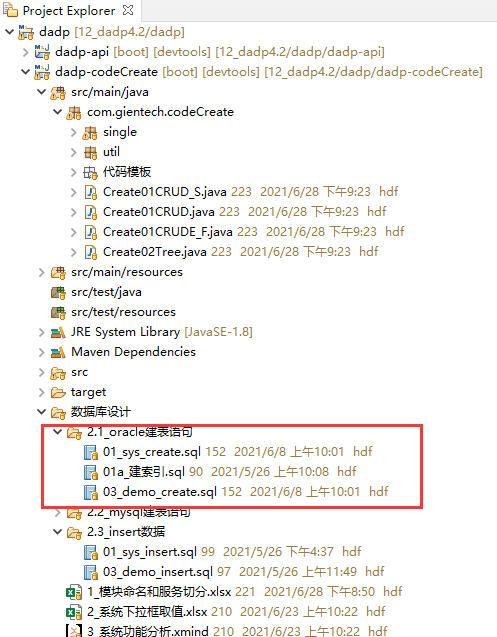
## 工程结构



## 启动nacos2.02和redis

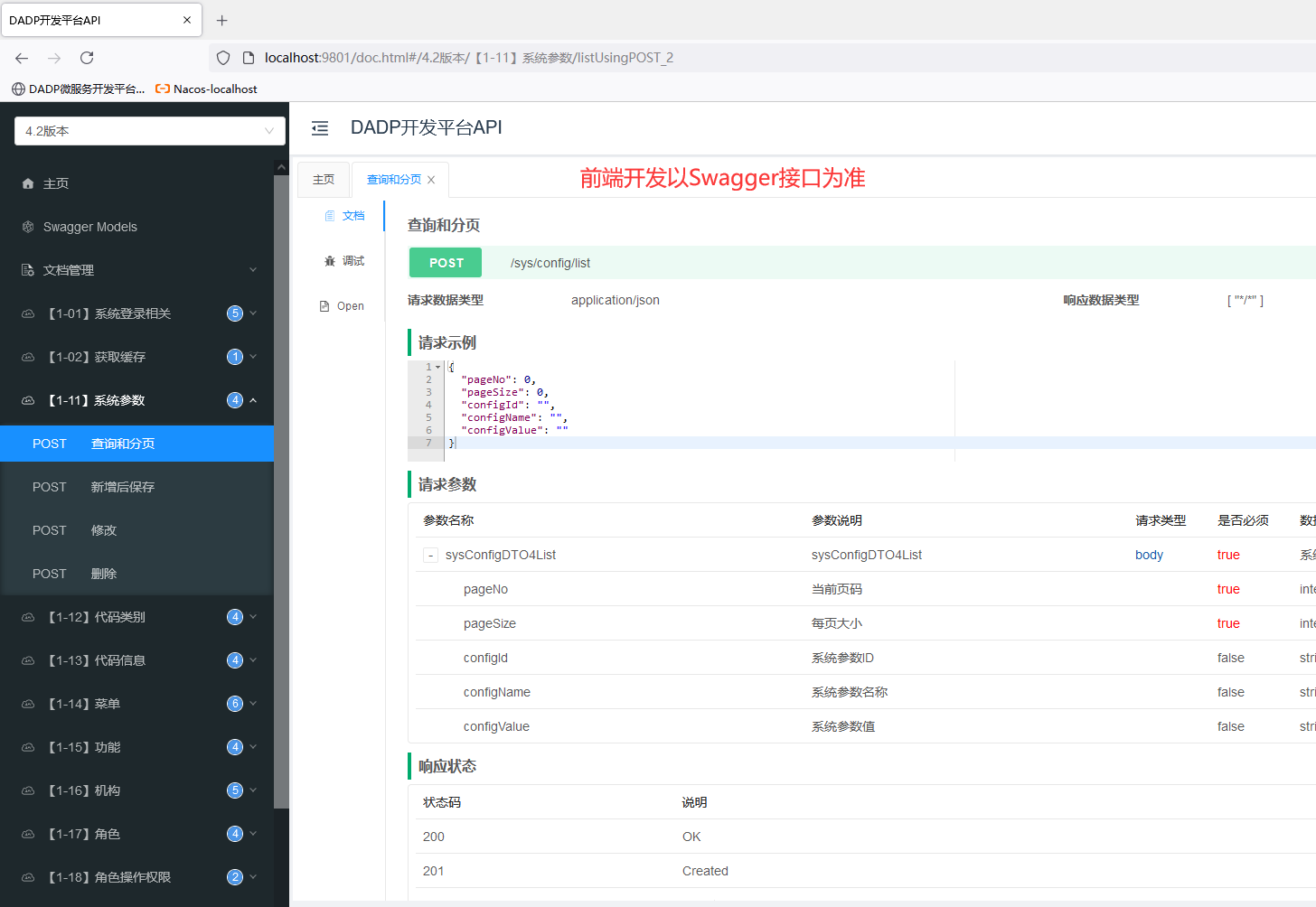
## 导入工程后，消除编译错误。运行mvn install

## 新建数据库，脚本在 /dadp-codeCreate/数据库设计



## 启动gateway、service-sys、service-demo

* 访问<http://localhost:9801/doc.html>



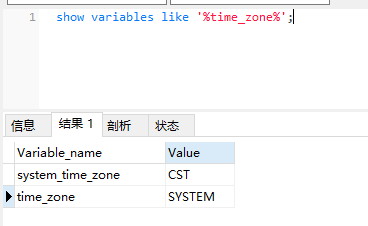
## 前端vue开发

* 参考《前端开发手册.md》，用Typora软件打开。

# 时差8小时的问题

## Mysql数据库

show variables like '%time\_zone%';



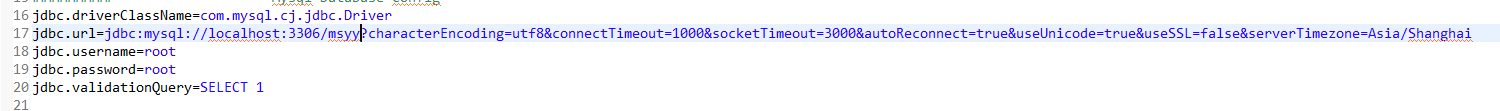
* Mysql默认安装就是这2个值。
* Central Standard Time (CST)，中央标准时间
* Time\_zone 的值为 SYSTEM，表示操作系统所在的时区
* **代码里面对日期型，和时间型分别加上,GMT+8。**



* **Jdbc的连接串，指定**serverTimezone=Asia/Shanghai

jdbc.driverClassName=com.mysql.cj.jdbc.Driver

jdbc.url=jdbc:mysql://localhost:3306/uadp6?characterEncoding=utf8&connectTimeout=1000&socketTimeout=3000&autoReconnect=true&useUnicode=true&useSSL=false&serverTimezone=Asia/Shanghai



## Oracle数据库

在Oracle客户端操作时区

-- 查询系统时区和session时区

SELECT DBTIMEZONE, SESSIONTIMEZONE FROM DUAL;。

