**Public CMS 2017**

二次开发手册

目录

[第一章 PublicCMS介绍 1](#_Toc477012697)

[1. 概述 1](#_Toc477012698)

[2. 获取PublicCMS 1](#_Toc477012699)

[3. 目录结构 1](#_Toc477012700)

[第二章 开发工具集成 3](#_Toc477012701)

[1. 工程导入 3](#_Toc477012702)

[2. 开发工具配置 5](#_Toc477012703)

[3. 工程调试 9](#_Toc477012704)

[4. 工程发布 14](#_Toc477012705)

[第三章 二次开发 19](#_Toc477012706)

[1. 工程代码简介 19](#_Toc477012707)

[2. Hibernate实体自动生成 20](#_Toc477012708)

[3. 逻辑代码及页面自动生成 25](#_Toc477012709)

[4. Mybatis代码自动生成 25](#_Toc477012710)

[5. 获取支持 25](#_Toc477012711)

# PublicCMS介绍

## 概述

PublicCMS是采用2017年最新主流技术开发的免费开源JAVACMS系统。架构科学。轻松支持上千万数据、千万PV；支持全站静态化，SSI，动态页面局部静态化等为您快速建站，建设大规模站点提供强大驱动，也是企业级项目产品原型的良好选择。

PublicCMS同时支持动态站点，静态站点的建设。

PublicCMS支持多站点建设，各个站点之间数据、模板完全分离。

PublicCMS开源版本采用最宽松的授权协议MIT。这意味着:

* 你可以自由使用，复制，修改，可以用于自己的项目。
* 可以免费商用。
* 唯一的限制是必须包含许可声明。

## 获取PublicCMS

稳定版PublicCMS代码托管

开源中国:<https://git.oschina.net/sanluan/PublicCMS>

GITHUB:<https://github.com/sanluan/PublicCMS>

CSDN:<https://code.csdn.net/zyyy358/publiccms>

预览版PublicCMS源码托管

开源中国:<https://git.oschina.net/sanluan/PublicCMS-preview>

GITHUB:<https://github.com/sanluan/PublicCMS-preview>

如果您没有使用过gradle或者maven，并且不打算使用它们，也可以在PublicCMS-war中war/WEB-INF/lib目录下复制PublicCMS依赖的库

## 目录结构

|  |
| --- |
| data/publiccms  publiccms  publiccms-gradle  publiccms-maven  LICENSE  README.md |

* data/publiccms目录

目录中包含以下子目录

|  |
| --- |
| indexes  task  template  web |

其中indexes目录为数据全文索引文件，task目录为任务计划脚本，template为模板文件目录，web未静态化文件和资源文件存放目录

* publiccms为PublicCMS源码工程
* publiccms-gradle为Gradle配置工程
* publiccms-maven为Maven配置工程
* LICENSE为PublicCMS授权协议
* README.md为markdown语言编写的简单介绍文档

# 开发工具集成

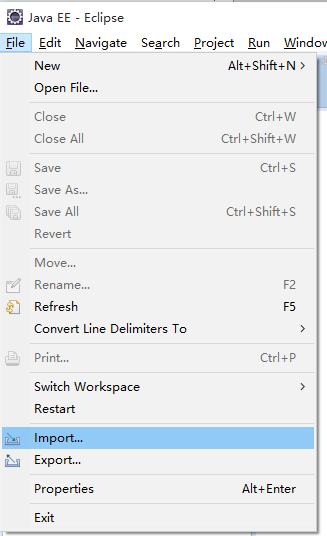
以下步骤Eclipse为例，下载地址 <https://www.eclipse.org/downloads/>

## 工程导入

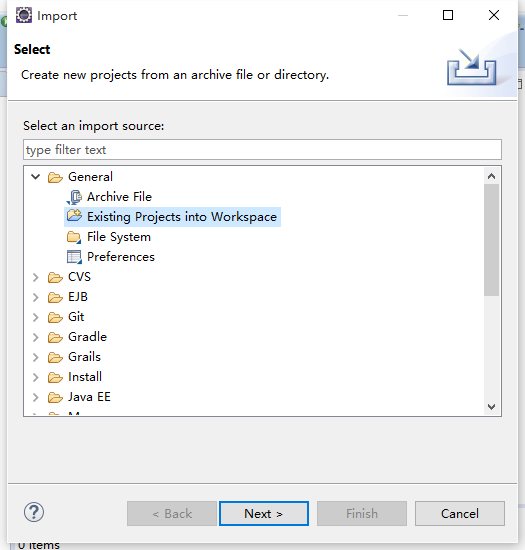
导入到开发工具中后，您可以自由的编译和修改PublicCMS源码，进行二次开发等。

### 操作步骤

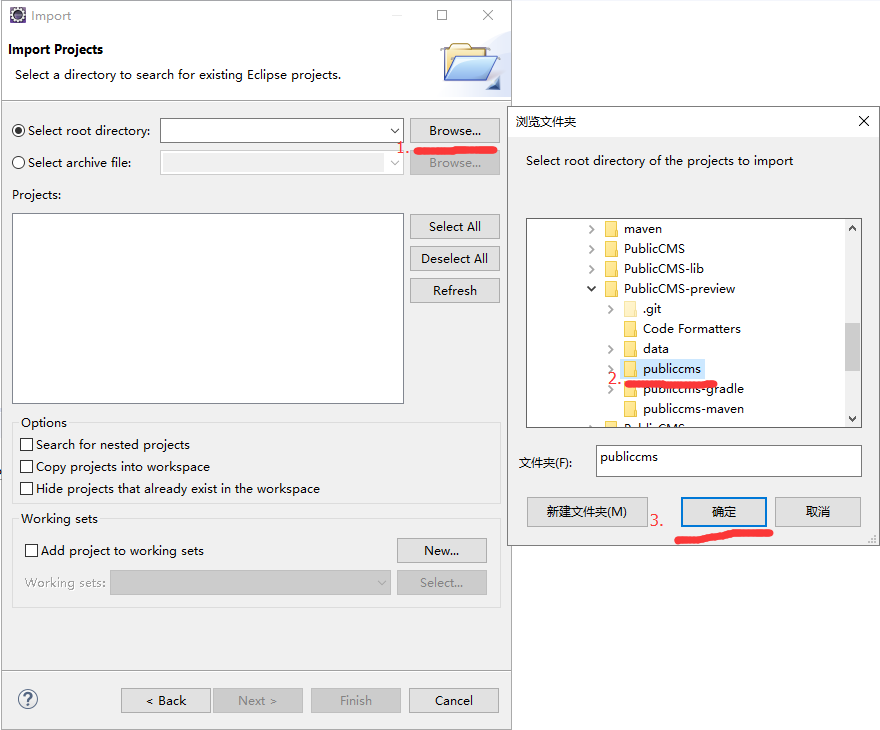
* 首先选择File菜单中的Import(导入)



* 在弹出的导入对话框中选择Existing Projects into Workspace(导入已有的工程)



* 使用Select root directory(选择跟目录)的方式浏览选择PublicCMS目录或某个工程目录



* 在根据需要再导入publiccms-gradle工程或者publiccms-maven工程，建议勾选Copy projects into workspace(复制到工作空间)

### 注意事项

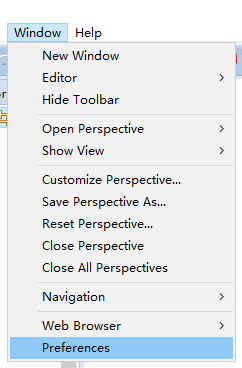
* 如果是gradle工程，按照publiccms-gradle中readme.txt进行操作配置工程
* 如果是maven工程，按照publiccms-maven中readme.txt进行操作配置工程
* 如果您没有使用gradle或者maven，请在工程src/main/webapp/WEB-INF目录下创建lib文件夹，并将PublicCMS-war/war/WEB-INF/lib目录下所有jar包复制到lib文件夹下。

## 开发工具配置

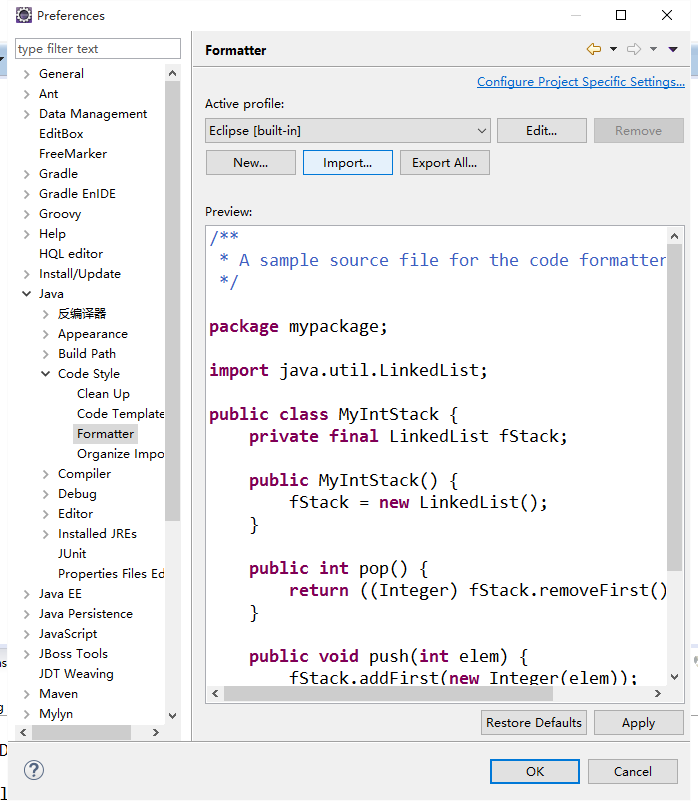
开发工具配置选项是为了您更好的对PublicCMS进行开发，并保持代码风格的一致性

### 导入格式化文件

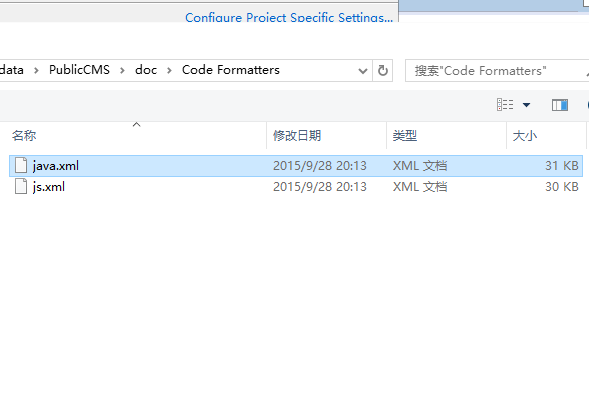
选择Window菜单下的Preferences菜单项



展开到Java->Code Style->Formatter选项卡点击Import按钮



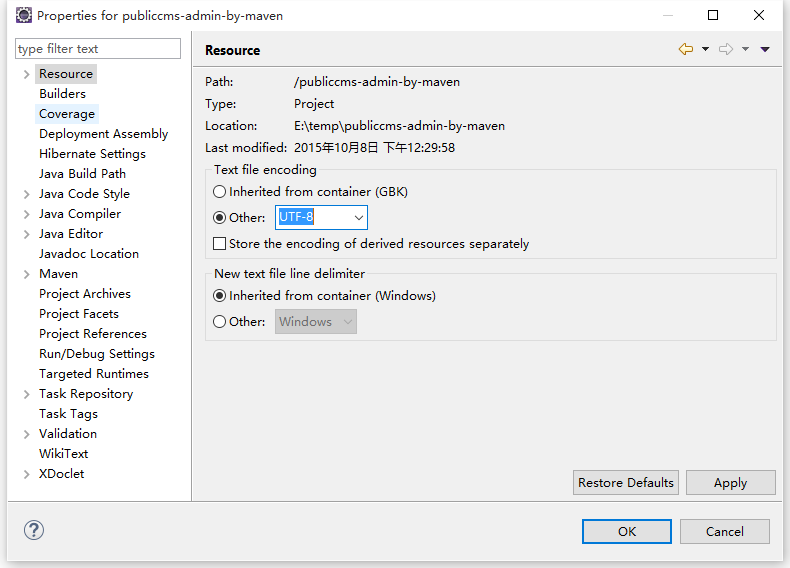
选择java.xml文件



使用相似的步骤导入js.xml到JavaScript->Code Style->Formatter中

### 字符编码设置

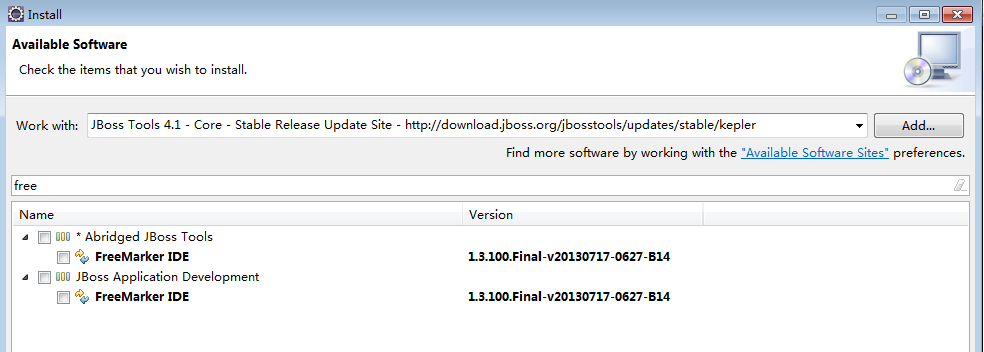
工程的编码都是UTF-8格式



### 开发工具插件

FreeMarker编辑器Eclipse插件被包含在jboss tools中

<http://download.jboss.org/jbosstools/updates/stable/> (需要选择IDE版本)



通过过滤选择FreeMarker IDE插件安装

需要注意的是该插件更新较慢，而且不是非常稳定，对于FreeMarker比较熟悉的人员来说，它并不能加快开发速度，因此只建议FreeMarker的初学者或对FreeMarker指令不熟悉的开发人员使用

## 工程配置

工程配置文件在src/main/resources目录下

cms.properties为CMS配置文件

|  |  |  |
| --- | --- | --- |
| 配置项 | 描述 | 备注 |
| cms.filePath | CMS数据目录的绝对路径 |  |
| cms.autoInstall | 是否进入安装引导界面 | 该项为true或者数据库配置错误时，访问CMS首页会进入引导安装界面 |
| cms.masterSiteIds | 管理站点ID | 多个ID以半角逗号分隔 |
| cms.defaultSiteId | 默认站点ID |  |
| cms.tokenizerFactory | 分词器名称 | keyword,letter,lowercase, whitespace,edgengram, ngram, pathhierarchy, pattern, classic,standard, ax29urlemail, thai, wikipedia, hmmchinese |
| cms.task.threadCount | 任务计划线程数 |  |
| cms.multipart.maxUploadSize | 上传文件文件大小最大值 |  |

config/database-default.properties为数据库链接配置文件

|  |  |  |
| --- | --- | --- |
| 配置项 | 描述 | 备注 |
| jdbc.url | 数据库链接 | 只需修改数据库地址和数据库名 |
| jdbc.username | 数据库用户名 |  |
| jdbc.password | 数据库密码 |  |

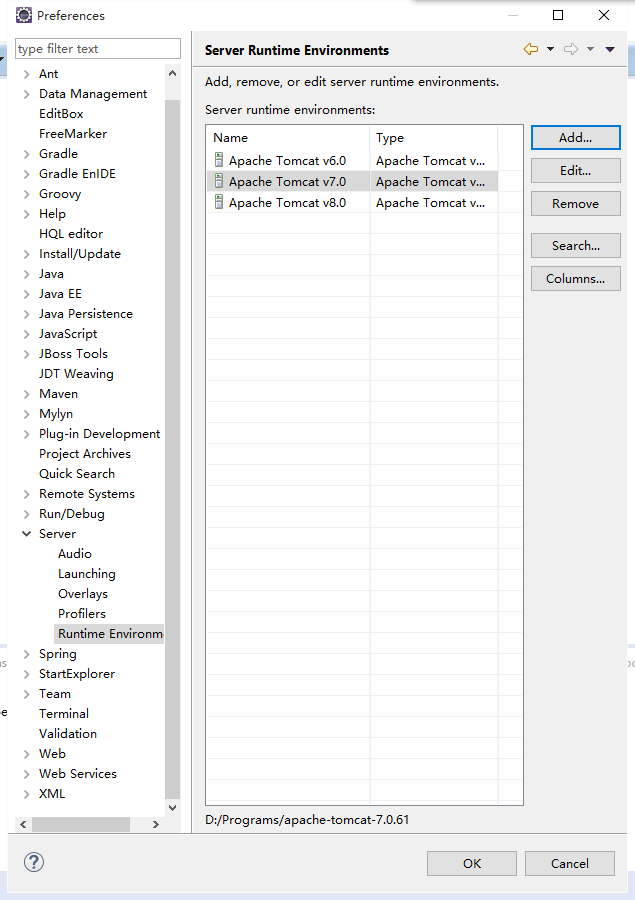
在使用maven或者gradle打包时，默认会进行自动测试，此时也需要保证数据库链接配置正确，并且数据库已经初始化才能打包成功，您也可以选择跳过自动测试。

数据库初始化脚本在src/main/resources/com/publiccms/common/initialization/mysql/目录下；数据库升级脚本在src/main/resources/com/publiccms/common/initialization/upgrade/mysql/目录下。

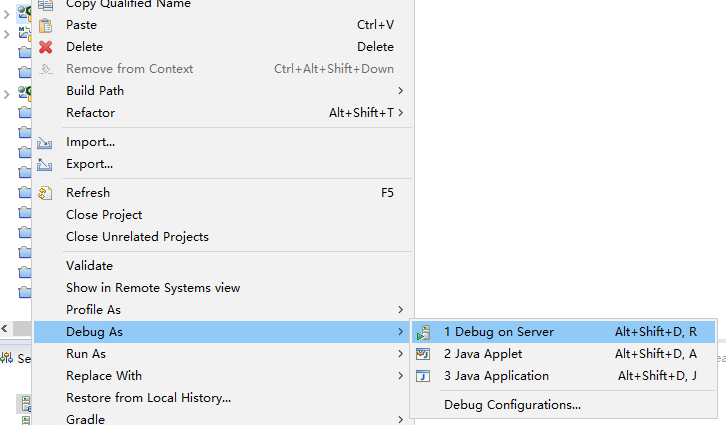
## 工程调试

在开发工具中调试时您可以选择编写测试代码进行单元测试，也可以直接在容器中运行工程，进行在线调试。模板的调试建议您选择在线进行。

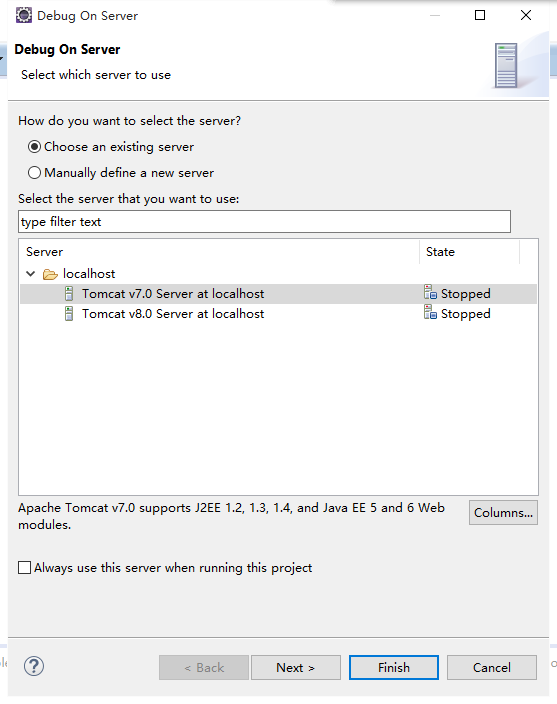
选择Window菜单下的Preferences菜单项, 展开到Server->Runtime Environments选项卡，新增一个Tomcat7.0.57以上版本的容器



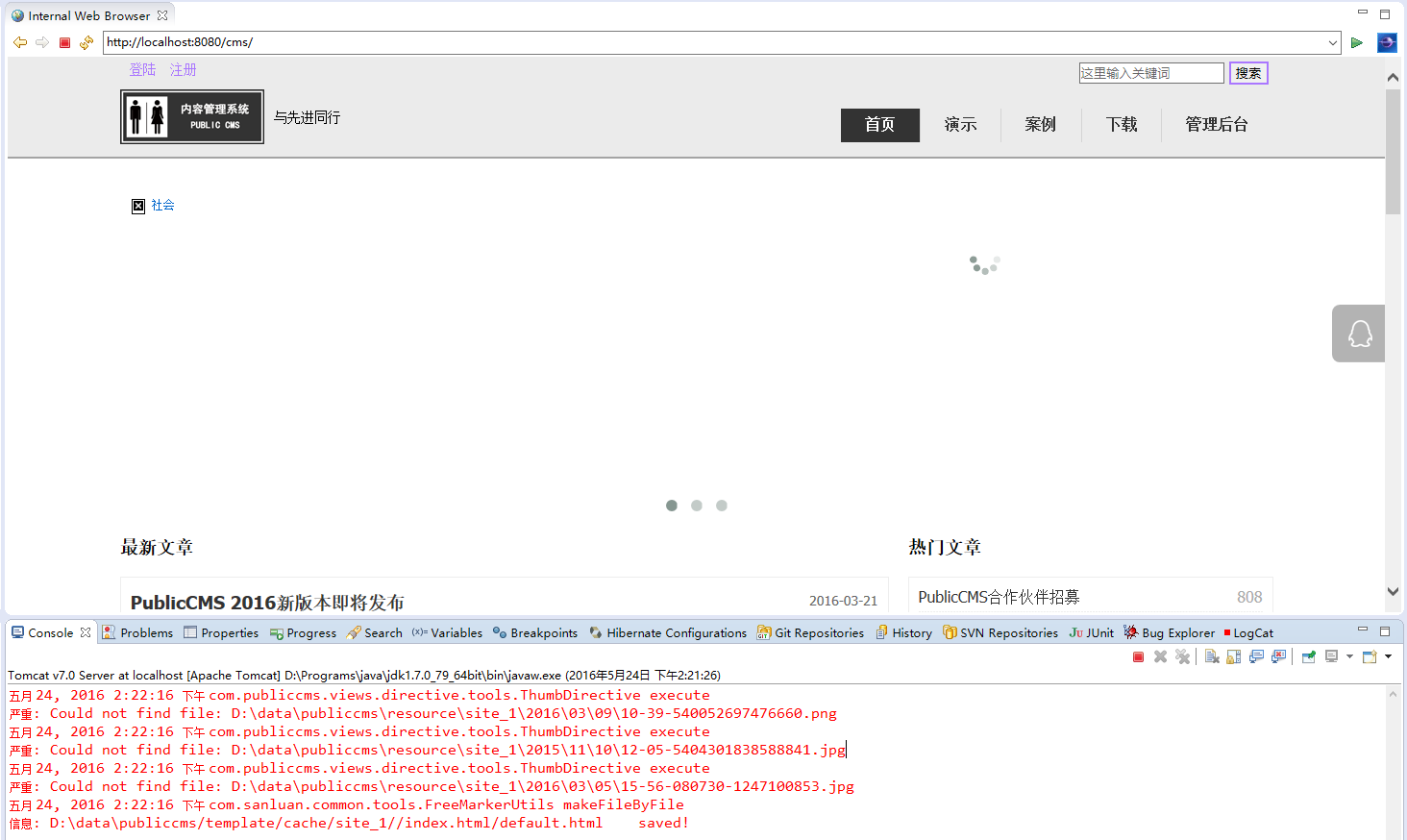
在工程上右键，选择Debug As菜单的Debug on Server子菜单项或者Run As菜单的Run on Server子菜单项



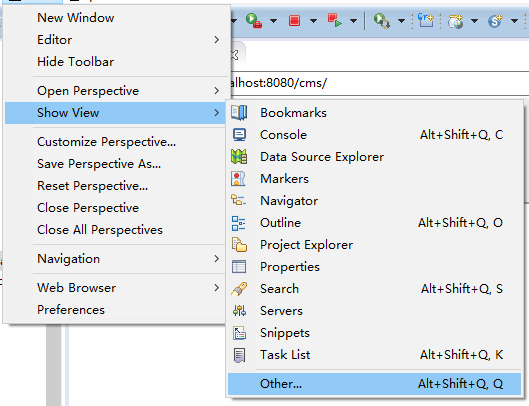
选择刚才添加的容器，然后点击Finish按钮



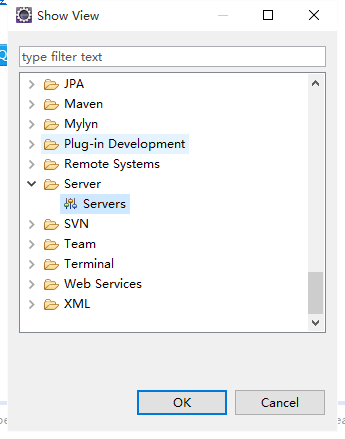
即可在工程中调试或运行工程



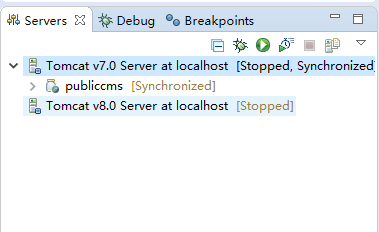
还可以在Window菜单中的Show View子菜单选择Other菜单项



打开Servers视图



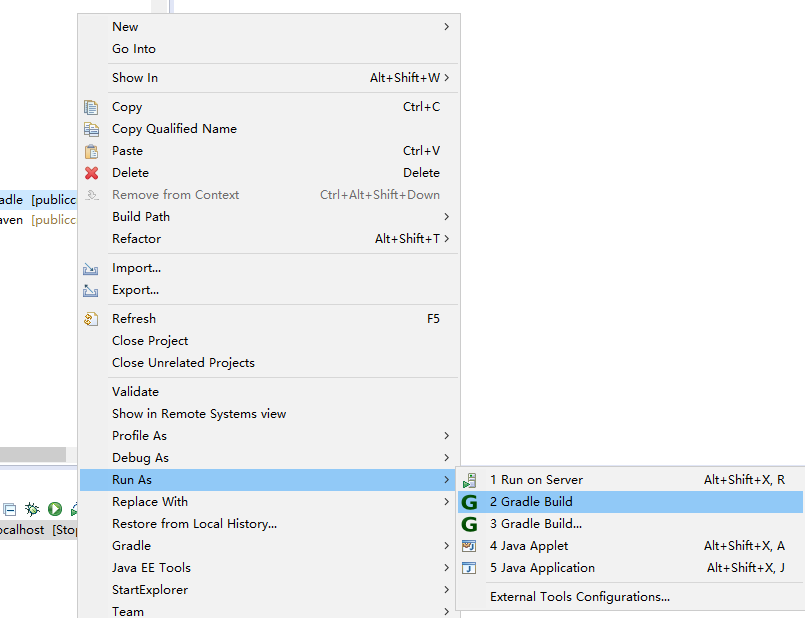
对所有容器和容器中运行或调试的工程进行管理



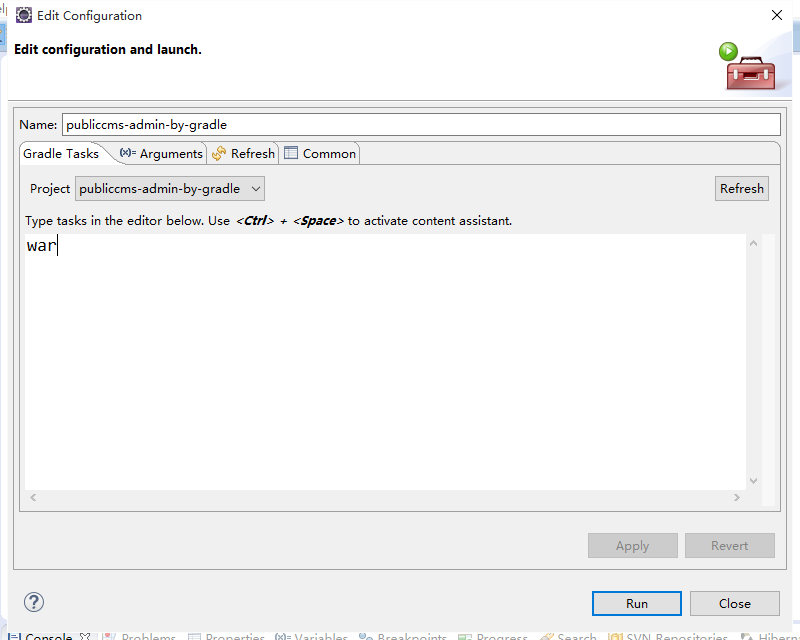
## 工程发布

### gradle工程发布

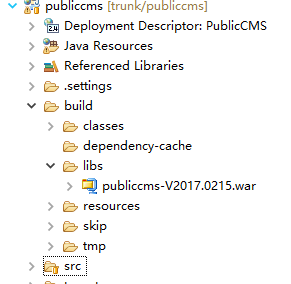
* 选择publiccms-gralde工程，右键弹出菜单中选择"Run As",在子菜单中选择"Gradle build"



* 在弹出的对话框中输入"war",点击按钮"Run"

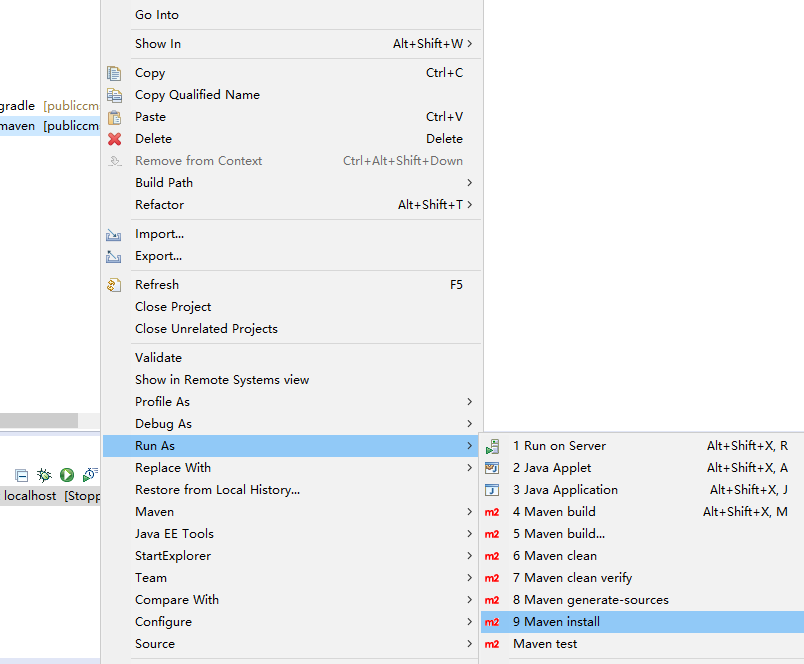


* 如果编译成功，您可以在publiccms工程下面 build/libs 目录下找到编译完成的 war包

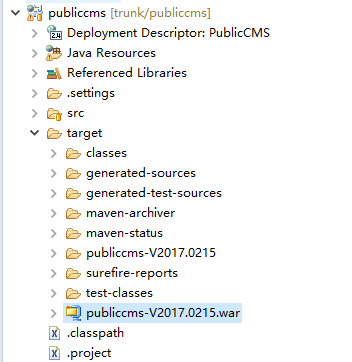


### maven工程发布

* 选择publiccms-maven工程，右键弹出菜单中选择"Run As",在子菜单中选择"Maven install"



* 如果编译成功，您可以在publiccms工程下面 target 目录下找到编译完成的 war包

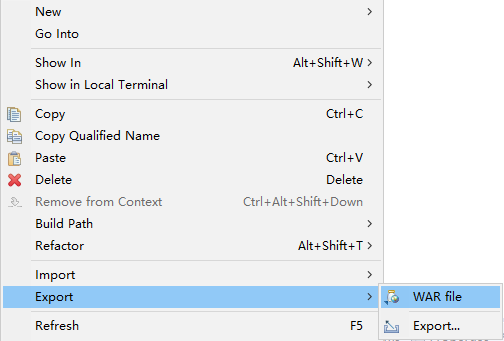


### Spring Boot工程发布

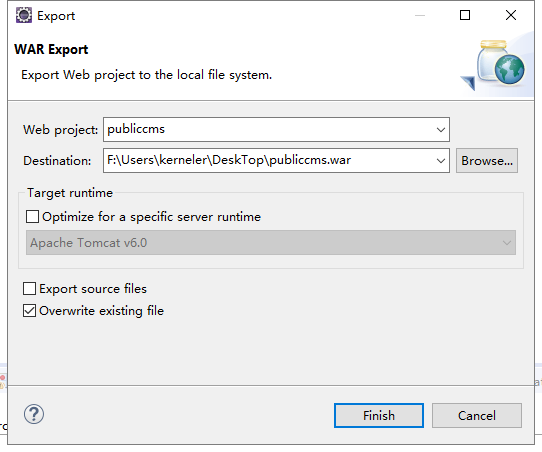
* 删除gradle或者maven配置文件中所有包含“delete this line for spring boot”的行
* 然后按照以上方式打包工程

### eclipse工程发布

* 选择publiccms工程，右键弹出菜单中选择"Export",在子菜单中选择"War file"



* 选择war包保存路径，然后点击Finash按钮



# 二次开发

## 工程代码简介

### 架构

1. 软件基础架构

软件使用了SpringMVC, Spring Boot, Hibernate, Hibernate Search,Mybatis, FreeMarker, Mysql jdbc, c3p0, jedis, jackson, quartz, Apache commons及其依赖的包,包依赖关系、工程编译等工作由Gradle或Maven处理。

1. 软件核心机制

软件使用IndexAdminController, IndexController做统一模板分发。数据读取操作交给FreeMarker指令扩展来处理，实现了视图层JAVA代码的最大化复用，表单的处理采用了传统的SpringMVC Controller接收并处理的方式。

### 命名规范

.common 工具类、解决方案类等同用类包

.base 基础类包

.database 数据源包

.directive 指令基础包

.constants 常量定义类包

.handler 分页处理，查询处理，FreeMarker异常等处理器包

.index 索引组件

.servlet Servlet包

.source 源码工具包

.tools 工具类包

.view 视图层基础包

.entities 实体类包

.logic 逻辑类包

.component 组件包

.dao 数据库操作类包

.service 事务与逻辑封装类包

.controller 控制器类包

.web 前台站点控制器类包

.admin 管理后台控制器类包

.api 接口控制器包

.views 视图类包

.directive freemaker自定义指令类包

.method freemaker自定义方法类包

.pojo 表单实体包

config 配置文件

language 语言文件

config.spring Spring配置类

config.boot SpringBoot入口类

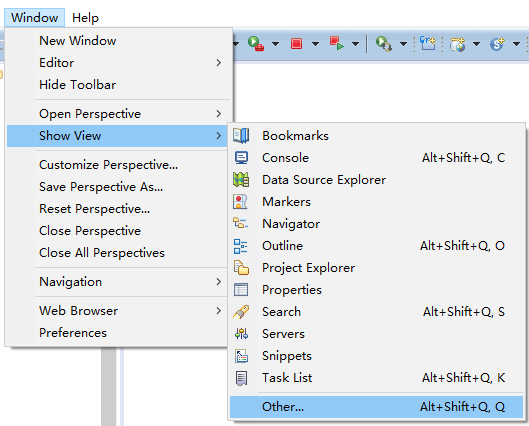
config.initializer Servlet加载器

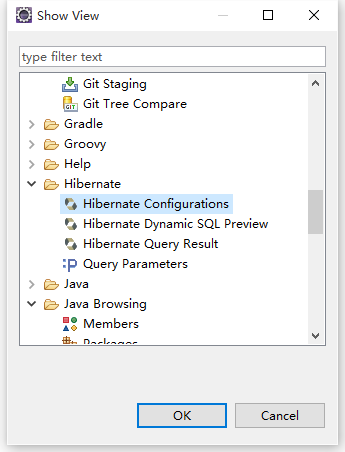
## Hibernate实体自动生成

1. 在开发工具中使用hibernate工具生成实体类，修改数据库配置文件

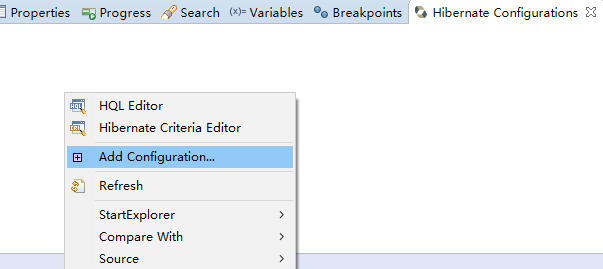
在Eclipse

将database/hibernate目录复制到工程中，也可以在后续操作中自己新建其中的配置文件，打开开发工具的Hibernate Configuration视图

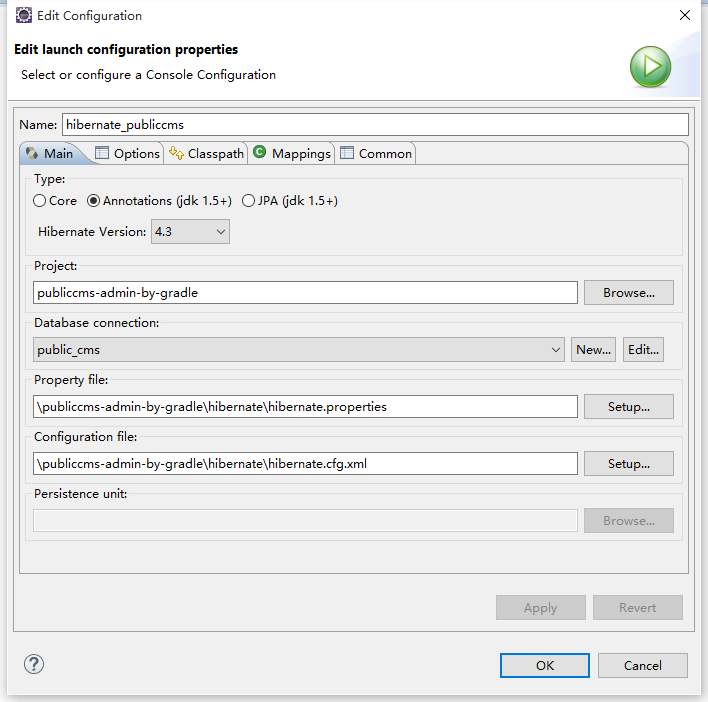




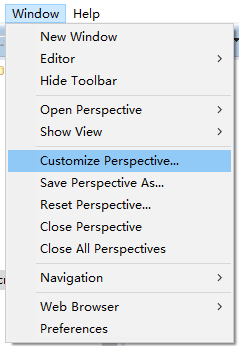
增加Hibernate配置



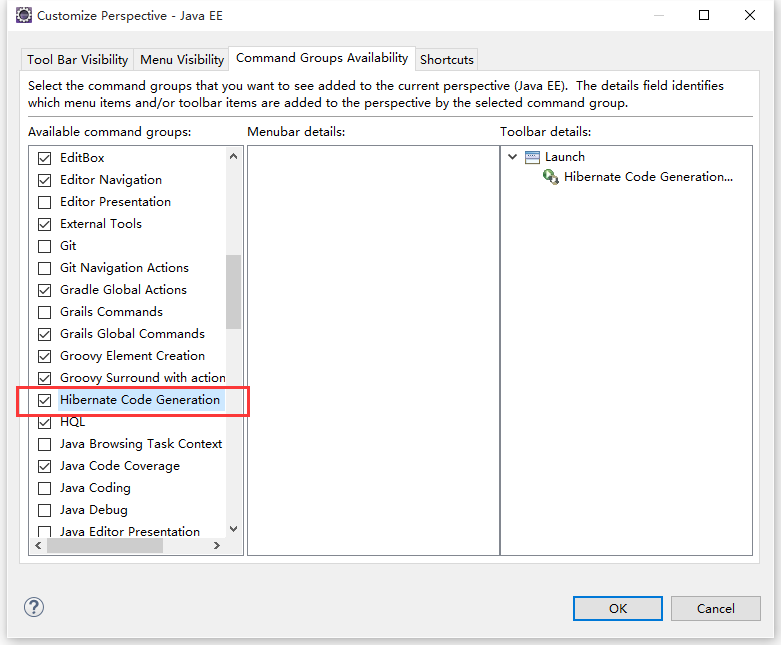
选择Annotations注解类型的实体，新建或者选择已有的数据库，选择hibernate目录下的properties和cfg.xml文件或者自己新建这两个文件，点击ok保存



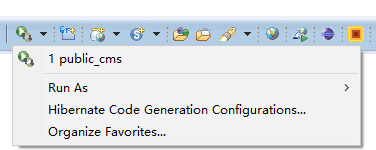
打开自定义视图



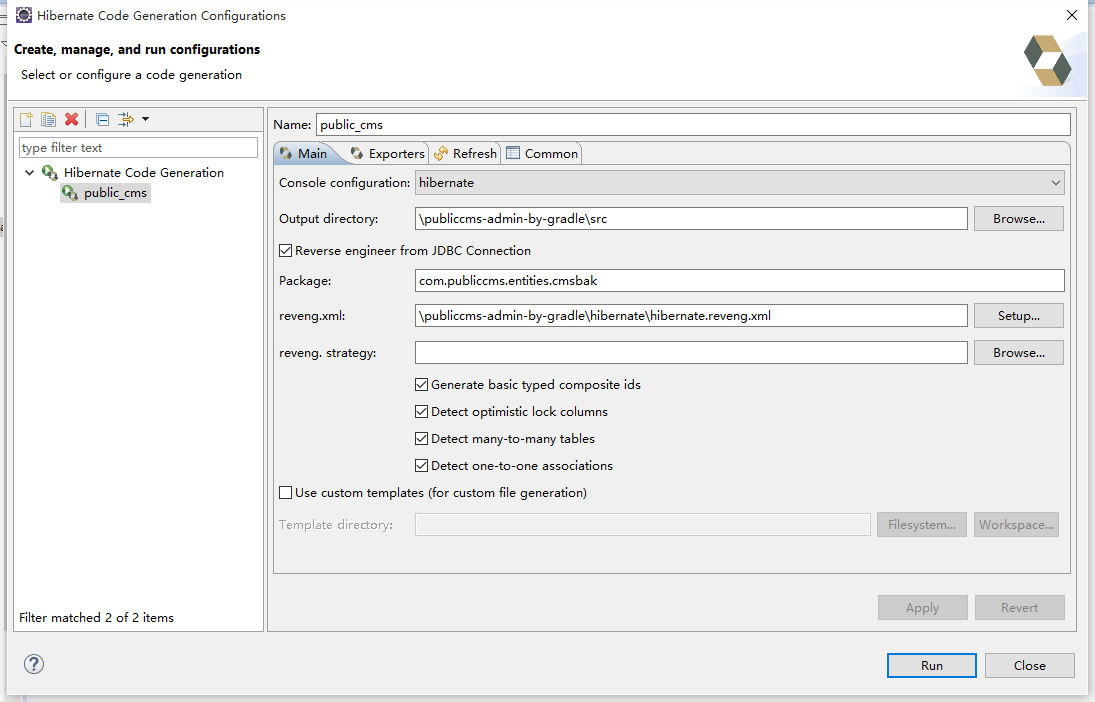
切换到Command Groups Availability选项卡，找到并勾选Hibernate Code Generation命令



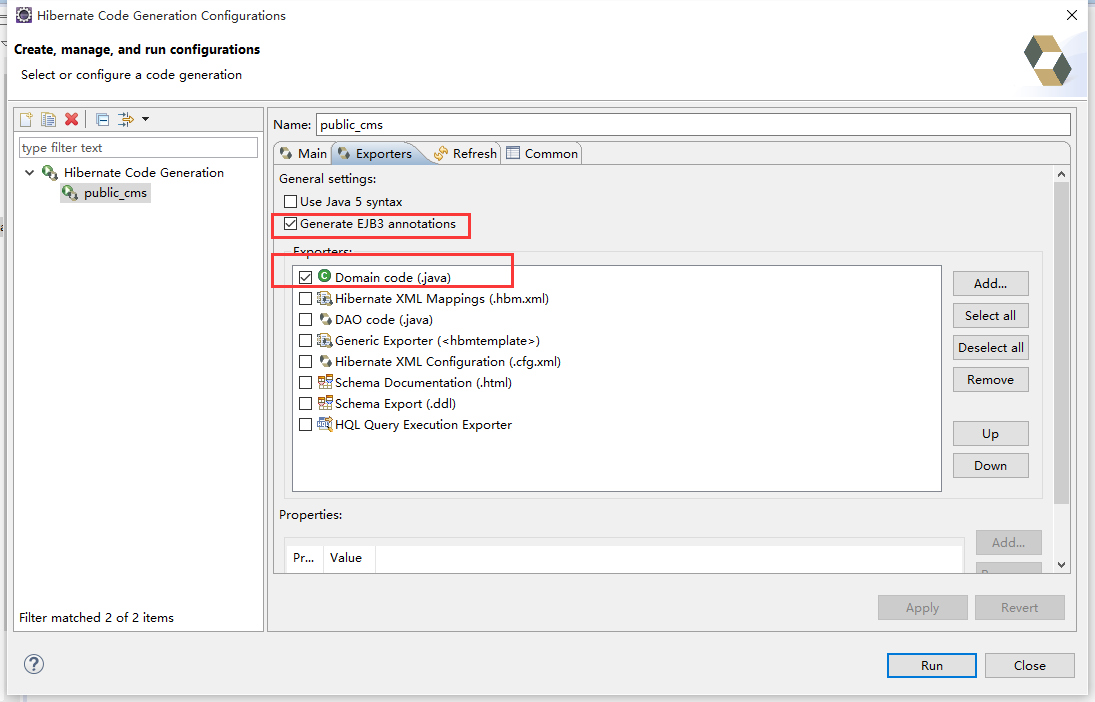
在eclipse工具条中即可找到Hibernate代码生成工具。点击Hibernate Code Generation Configurations新建代码生成配置



选择hibernate目录下的hibernate.reveng.xml.xml文件或者自己新建这个文件



切换到Exporters选项卡，勾选Generate EJB3 annotations和Domain code(.java)



然后点击run就可以生成带注解的实体类了。

## 逻辑代码及页面自动生成

该生成方法对hibernate实体进行标注后，根据标准信息生成类 生成自定义指令，Controller，Service，Dao，html，是CMS开发阶段采用的生成方式

* 1. 对实体类属性使用com.sanluan.common.source.entity.MyColumn注解类标记字段类型【condition:是否用于查询条件，order:是否用于排序，like:查询条件是否为like匹配，or:查询条件是否为or逻辑，name:字段名，当or为true时，相同name 的字段为同一组or条件，title:字段真实含义】
  2. 使用com.sanluan.common.generator.SourceGenerator代码生成代码文件

## Mybatis代码自动生成

* 1. 修改src/test/resources/mybatis/generatorConfig.xml的数据库相关属性，确保数据库是正确且可连接。
  2. 打开src/test/java/com/sanluan/common/generator/MybatisGenerator.java的文件
  3. 运行main函数，将会生成Mybatis代码文件

# 服务支持

## 免费支持

PublicCMS官网：<http://www.publiccms.com/>

QQ交流群：[191381542](http://shang.qq.com/wpa/qunwpa?idkey=8a633f84fb2475068182d3c447319977faca6a14dc3acf8017a160d65962a175)

Freemarker文档：<http://www.kerneler.com/freemarker2.3.23/>

在线调试FreeMarker：<https://www.sanluan.com/freemarker_test.html>

PublicCMS文档库：

<https://git.oschina.net/sanluan/PublicCMS-lib>

<https://github.com/sanluan/PublicCMS-lib>

## 商业化支持

目前只接受以下培训类项目：

产品使用、模板制作、环境部署、二次开发、系统架构、产品相关框架

联系方式：

QQ：[315415433](https://wpa.qq.com/msgrd?v=3&uin=315415433&site=qq&menu=yes)

邮箱：[zyyy358@126.com](mailto:zyyy358@126.com)