

高等学校计算机基础教育规划教材

Java Web 应用开发基础教程

郭庆等编著

非
外
借

清华大学出版社



内容简介

本书主要介绍了基于JDK 1.7、Tomcat 7、MySQL 5、Eclipse 3.7.2、Eclipse JEE 部署 Java Web 应用开发中的常用技术。第1章介绍了 Java Web 应用开发的基本概念；第2~10章介绍了 Java Web 应用开发中的常用技术，包括 JDBC、Servlet、JSP、Javabeans、Spring、Struts、JSTL、过滤器和拦截器、Spring MVC、Spring 持久化、Spring MVC。

本书可作为高等院校计算机专业及相关专业的教材，也可作为从事 Java Web 应用开发的工程技术人员的学习参考书。

高等学校计算机基础教育规划教材

在服务器端编程方面，Java Web 技术以其可靠性高、安全性好等特点成为市场主流，因此深入学习和掌握 Java Web 应用开发的相关技术变得日益重要。

本书从事 Java、Java EE、Java Web 应用开发的教学和科研工作多年，积累了大量的 Java 技术的项目经验和教学经验，本书可作为高等院校计算机专业及相关专业的教材，也可作为从事 Java Web 应用开发的工程技术人员的学习参考书。

Java Web 应用开发基础教程

郭庆 田甜 王向辉 徐遵义 编



本书可作为高等院校计算机专业及相关专业的教材，也可作为从事 Java Web 应用开发的工程技术人员的学习参考书。

本书由清华大学出版社出版，ISBN 978-7-302-47773-0，定价 49.00 元。

清华大学出版社
北京

清华大学出版社
北京

内 容 简 介

本书分为13章:第1章主要介绍了基于JDK 1.7、Tomcat 7、MySQL 5、Eclipse JEE搭建Java Web集成开发环境、Eclipse集成Tomcat、Eclipse的基本使用;第2~10章介绍了Java Web应用开发中的常用官方技术标准,包括JDBC、Servlet、JSP、JavaBean、表达式语言、JSTL、过滤器和监听器、自定义标签、JPA;第11~13章介绍了目前软件开发企业使用较多的开源框架Spring,包括Spring框架核心基础、使用Spring持久化、Spring MVC。

本书在详细说明Java Web应用开发相关技术的理论和API基础上,结合第1章搭建的集成开发环境编写了大量工程性和系统性案例。掌握这些案例有助于深入理解Java Web相关技术的实际应用场景。

本书可作为高等学校Java Web应用开发课程的教材,也可供从事软件开发的工程技术人员自学使用。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。
版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Java Web应用开发基础教程/郭庆等编著. —北京:清华大学出版社,2018
(高等学校计算机基础教育规划教材)
ISBN 978-7-302-50736-9

I. ①J… II. ①郭… III. ①JAVA语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆CIP数据核字(2018)第172726号

责任编辑:袁勤勇 李 晔

封面设计:常雪影

责任校对:时翠兰

责任印制:宋 林

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦A座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:三河市国英印务有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:20.5

字 数:473千字

版 次:2018年10月第1版

印 次:2018年10月第1次印刷

定 价:49.00元

产品编号:075055-01

前言

第1章 集成开发环境搭建

在服务器端编程方面,Java Web 技术以其可靠性高、安全性好等特点成为市场主流,因此深入学习和掌握 Java Web 应用开发的相关技术变得日益重要。

笔者从事 Java、Java EE、Java Web 等相关课程的教学和培训 10 余年,主持和参与了多项基于 Java 技术的项目,积累了许多教学和实践经验。本书致力于为学习 Java Web 应用开发技术的读者提供一个坚实的理论基础和实践基础,在注重基础知识理解的同时强调实践能力的培养。本书具有如下特点:

(1) 内容简洁、完整,便于读者全面掌握 Java Web 应用开发技术的基础知识。

(2) 实践性强,以实例形式介绍多种 Java Web 应用开发技术。

(3) 系统性强,以实际常用的集成开发环境 Eclipse、Web 服务器 Tomcat、数据库 MySQL 等软件为主进行了整合,并针对 Java Web 应用开发中的常用技术编写了大量系统性案例。

(4) 侧重于服务器后端技术,详细介绍 Java Web 应用开发相关技术。

(5) 思考与习题基础性强,有助于提高读者兴趣,解决新手编程入门难的问题。

全书共分 13 章,内容包括:集成开发环境搭建与使用、JDBC、Servlet、JSP、JavaBean、表达式语言、JSTL、过滤器和监听器、自定义标签、JPA、Spring 框架核心基础、使用 Spring 持久化、Spring MVC。

本书既适合作为高等院校 Java Web 应用开发技术课程的教材,也可作为高职高专、成人高等教育、社会培训班的教材,还可作为 Java Web 应用开发技术自学者的教材或参考书。

本书由山东建筑大学的郭庆、田甜、王向辉、徐遵义编著。其中,徐遵义编写了第 1、9 章,郭庆编写了第 2~4 章和第 10~13 章,田甜编写了第 5、6 章,王向辉编写了第 7、8 章。

本书参考了大量的国内外文献,在此向这些文献的作者表示深深的敬意和衷心的感谢!

软件技术发展迅速,加之作者水平和时间有限,书中难免有不妥之处,恳请读者批评指正。

编者

2018 年 1 月

目录

第 1 章 集成开发环境搭建	1
1.1 集成开发环境搭建	1
1.1.1 安装 JDK	1
1.1.2 安装 Tomcat	3
1.1.3 安装 MySQL	4
1.1.4 安装 Eclipse	4
1.1.5 Eclipse 集成 Tomcat	5
1.2 Eclipse 基本使用	7
1.2.1 创建和运行 Java 项目	7
1.2.2 创建和运行 Java Web 项目	9
小结	11
思考与习题	11
第 2 章 JDBC	12
2.1 JDBC 简介	12
2.1.1 JDBC 的概念	12
2.1.2 JDBC 3.0 API 简介	12
2.1.3 JDBC 驱动程序类型	15
2.2 使用 JDBC	17
2.2.1 准备工作	17
2.2.2 使用 JDBC 一般步骤	18
2.3 JDBC 实现增、删、改、查	23
2.3.1 插入记录	23
2.3.2 查询记录	24
2.3.3 更新记录	25
2.3.4 删除记录	26
2.4 JDBC 进阶	27
2.4.1 数据库帮助类 DBConnection	27
2.4.2 使用 PreparedStatement	29

2.4.3	调用存储过程	30
2.5	JDBC 事务管理	33
2.5.1	事务的概念	33
2.5.2	JDBC 事务管理	33
2.6	JDBC 4. x	36
	小结	38
	思考与习题	38
第 3 章	Servlet	39
3.1	Servlet 简介	39
3.1.1	Servlet 的概念	39
3.1.2	Servlet 与 CGI 的区别	39
3.1.3	Servlet 的功能	40
3.1.4	Servlet 的优点	40
3.1.5	Servlet API 简介	41
3.1.6	Servlet 的生命周期	44
3.2	创建 Servlet	45
3.2.1	Java Web 应用的目录结构	45
3.2.2	创建和配置 Servlet	46
3.2.3	使用 Eclipse 创建和配置 Servlet	48
3.3	Servlet 常用功能	51
3.3.1	Servlet 接收请求参数	51
3.3.2	作用域与存取数据	53
3.3.3	Servlet 请求转发与重定向	55
3.3.4	获取 Servlet 初始化参数	56
3.3.5	配置 Servlet 加载顺序	57
	小结	58
	思考与习题	58
第 4 章	JSP	59
4.1	JSP 简介	59
4.1.1	JSP 的概念	59
4.1.2	JSP 的优点	59
4.1.3	JSP 执行过程和第一次访问	60
4.2	JSP 注释	61
4.3	JSP 指令元素	62
4.3.1	page 指令	62
4.3.2	include 指令	64

001	4.3.3	taglib 指令	65
101	4.4	脚本元素	65
101	4.5	动作元素	66
	4.5.1	<jsp:forward>	67
801	4.5.2	<jsp:include>	68
201	4.6	内建对象	69
801	4.6.1	out 对象	69
801	4.6.2	response 对象	70
401	4.6.3	request 对象	72
201	4.6.4	session 对象	73
901	4.6.5	application 对象	76
111	4.6.6	pageContext 对象	76
311	4.6.7	config 对象	77
811	4.6.8	exception 对象	79
911		小结	79
121		思考与习题	80
751			88
	第 5 章	JavaBean	81
761	5.1	JavaBean 规范	81
761	5.2	访问 JavaBean	82
861	5.2.1	使用脚本段代码访问 JavaBean	82
921	5.2.2	使用动作元素访问 JavaBean	83
1041	5.3	JSP+JavaBean 开发模式	87
1041	5.4	JSP+Servlet+JavaBean 开发模式	90
1141		小结	91
1241		思考与习题	91
			173
	第 6 章	表达式语言	93
831	6.1	表达式语言基础	93
831	6.1.1	表达式语言语法	93
141	6.1.2	. 运算符与[]运算符	94
641	6.1.3	获取变量时的搜索顺序	94
741	6.1.4	自动转型	96
841	6.1.5	保留字	96
921	6.1.6	内建对象	96
121	6.1.7	运算符	97
121	6.2	表达式语言函数	99
121	6.2.1	表达式语言定义函数	99

6.2.2	JSP 页面使用表达式语言调用函数	100
	小结	101
	思考与习题	101
第 7 章 JSTL		103
7.1	JSTL 简介	103
7.1.1	JSTL 构成	103
7.1.2	在 JSP 页面使用 JSTL	103
7.2	核心标签库	104
7.2.1	一般操作	105
7.2.2	流程控制操作	109
7.2.3	迭代操作	111
7.2.4	URL 操作	115
7.3	l18N 格式标签库	118
7.3.1	国际化标签	119
7.3.2	消息标签	121
7.3.3	数字、时间日期格式化	127
7.4	SQL 标签库	136
7.4.1	<sql:setDataSource>	137
7.4.2	<sql:query>	137
7.4.3	<sql:update>	138
7.4.4	<sql:param>	139
7.4.5	<sql:dateParam>	140
7.4.6	<sql:transaction>	140
	小结	141
	思考与习题	142
第 8 章 过滤器和监听器		143
8.1	过滤器	143
8.1.1	javax.servlet.Filter 接口	143
8.1.2	配置过滤器	144
8.1.3	过滤器解决中文乱码	146
8.2	监听器	147
8.2.1	监听器接口	148
8.2.2	配置监听器	150
8.2.3	监听器统计在线人数	151
	小结	151
	思考与习题	152

881	第 9 章 自定义标签	153
191	9.1 自定义标签简介	153
101	9.2 传统标签	154
281	9.2.1 传统标签 API	154
201	9.2.2 传统标签生命周期	155
501	9.2.3 实现 Tag 接口的传统标签	156
701	9.2.4 继承 TagSupport 类的传统标签	158
701	9.2.5 带属性和标签体的传统标签	159
801	9.2.6 修改内容的传统标签	161
101	9.3 简单标签	163
501	9.3.1 简单标签 API	163
601	9.3.2 简单标签生命周期	163
801	9.3.3 继承 SimpleTagSupport 的简单标签	164
701	9.3.4 输出标签体内容的简单标签	165
701	9.3.5 带属性的简单标签	166
801	9.3.6 修改标签体内容的简单标签	168
011	小结	169
111	思考与习题	169
811	第 10 章 JPA	171
111	10.1 JPA 简介	171
411	10.1.1 O/R 映射与 JPA	171
611	10.1.2 Eclipse 下搭建 JPA Java SE 环境	172
811	10.2 实体	174
811	10.2.1 实体类的编写规范	174
811	10.2.2 @Entity 注解	175
811	10.2.3 @Table 注解	176
911	10.2.4 @Id 注解	176
811	10.2.5 @Column 注解	177
811	10.2.6 @Transient 注解	178
	10.2.7 属性注解使用的位置	178
011	10.3 EntityManager	179
011	10.3.1 获取 EntityManager 实例	179
011	10.3.2 配置持久化单元	181
011	10.3.3 实体对象的状态与 EntityManager API	182
411	10.3.4 刷新操作	185
411	10.3.5 实体生命周期回调	186

10.4	实体映射关系	188
10.4.1	单向一对一映射	188
10.4.2	双向一对一映射	191
10.4.3	单向一对多映射	191
10.4.4	双向一对多映射	193
10.4.5	单向多对一映射	195
10.4.6	单向多对多映射	195
10.4.7	双向多对多映射	197
10.5	实体映射继承与多态	197
10.5.1	整个类继承层次结构使用单个表	198
10.5.2	各子类使用单独的表	201
10.5.3	各个具体实体类使用单个表	203
10.5.4	实体继承总结	206
10.6	JPA 查询语言	206
10.6.1	查询单个实体	207
10.6.2	查询实体属性和关系属性(投影)	207
10.6.3	使用 IN 访问关系集合属性	208
10.6.4	连接实体	210
10.6.5	使用参数	211
10.6.6	分页功能	213
10.6.7	ORDER BY	213
10.6.8	DISTINCT	214
10.6.9	在查询中构建对象	214
10.6.10	批量更新和批量删除	215
10.6.11	使用 WHERE 子句	215
10.6.12	GROUP BY 和 HAVING	215
10.6.13	NativeQuery	216
10.6.14	命名查询	216
10.6.15	调用存储过程	217
	小结	218
	思考与习题	219
第 11 章	Spring 框架核心基础	220
11.1	Spring 框架简介	220
11.1.1	Spring 体系结构	220
11.1.2	Java SE 环境下使用 Spring	222
11.2	IOC 容器	224
11.2.1	BeanFactory 容器	224

11.2.2	ApplicationContext 容器	224
11.3	依赖注入	225
11.3.1	setter 注入	225
11.3.2	构造方法注入	226
11.4	注入参数详解	229
11.4.1	字面值注入	230
11.4.2	引用其他 Bean	230
11.4.3	嵌套 Bean 注入	231
11.4.4	null 值注入	231
11.4.5	级联属性注入	232
11.4.6	集合注入	232
11.5	简化配置	238
11.6	Bean 的作用域和生命周期	239
11.6.1	Bean 的作用域	239
11.6.2	Bean 的生命周期	240
11.7	使用 XML 的自动装配	240
11.8	使用 Java 配置	241
11.8.1	使用 Java 手动配置	241
11.8.2	使用 Java 自动装配	245
11.9	AOP	247
11.9.1	AOP 简介	247
11.9.2	AOP 的术语	249
11.9.3	Spring AOP 基础	249
11.9.4	使用注解实现 Spring AOP 前置和后置增强	251
11.9.5	使用注解实现 Spring AOP 环绕增强	254
11.9.6	使用 XML 配置 Spring AOP 实现前置和后置增强	255
	小结	258
	思考与习题	258
第 12 章	使用 Spring 持久化	259
12.1	使用 Spring JDBC	259
12.1.1	使用 JdbcTemplate	259
12.1.2	JdbcTemplate 调用存储过程	264
12.2	事务管理	266
12.2.1	Spring 事务管理简介	266
12.2.2	编程式事务	269
12.2.3	声明式事务	271
12.3	Spring 整合 JPA	275

12.3.1	配置 LocalEntityManagerFactoryBean	276
12.3.2	配置从 JNDI 获取 EntityManagerFactory	276
12.3.3	配置 LocalContainerEntityManagerFactoryBean	277
12.3.4	Spring 整合 JPA 时使用 Spring Data	280
12.3.5	Spring Data JPA 的自定义查询	283
12.3.6	自定义查询方法的使用顺序	286
	小结	287
	思考与习题	287
第 13 章 Spring MVC		288
13.1	Spring MVC 配置	288
13.1.1	使用 XML 配置 Spring MVC	288
13.1.2	使用 Java 配置 Spring MVC	292
13.2	编写控制器	294
13.2.1	第一个简单的控制器	294
13.2.2	处理请求参数	295
13.2.3	处理路径参数	296
13.2.4	处理表单参数	297
13.3	数据校验	299
13.4	视图解析	302
13.4.1	JSP 视图	303
13.4.2	Tile 视图	304
13.4.3	返回 Json	309
13.5	文件上传	310
	小结	313
	思考与习题	314



集成开发环境搭建

现在 Java Web 应用开发中常见的集成开发软件主要有开源软件 Eclipse jee(以下简称 Eclipse)和 NetBeans、商业软件 MyEclipse 和 IntelliJ 等。其中,Eclipse 使用者众多。

Java Web 应用开发集成环境搭建通常包括安装 JDK、Tomcat、Eclipse、数据库服务器 MySQL,并需要对安装的某些软件进行配置。另外还需要在 Eclipse 中集成 Tomcat。

本书的 Java Web 集成开发环境基于 Windows 10 64 位操作系统,因此 JDK、Eclipse、Tomcat 均采用的是 64 位版本。对于 32 位 Windows,请到相关网站下载对应的 32 位版本的软件。

1.1 集成开发环境搭建

1.1.1 安装 JDK

1. 安装 JDK

双击 jdk-7u75-windows-x64.exe 文件进行安装。默认安装路径 C:\Program Files\Java\jdk1.7.0_75,也可以在安装时修改为其他路径。

2. 配置 JDK

(1) 新增系统变量 JAVA_HOME。在 Windows 10 下,右击“此电脑”→“属性”→“高级系统设置”,弹出“系统属性”对话框,如图 1-1 所示。

在图 1-1 中,选择“高级”→“环境变量”,在“系统变量”中单击“新建”按钮,弹出“新建系统变量”对话框,如图 1-2 所示。

在图 1-2 中,在“变量名”文本框中输入 JAVA_HOME,在“变量值”文本框中输入 JDK 安装路径,例如 C:\Program Files\Java\jdk1.7.0_75。

依次单击“新建系统变量”对话框、“环境变量”对话框、“系统属性”对话框中的“确定”按钮,完成新增系统变量 JAVA_HOME 的操作。

(2) 编辑系统变量 Path。在 Windows 10 下,单击“此电脑”→“属性”→“高级系统设

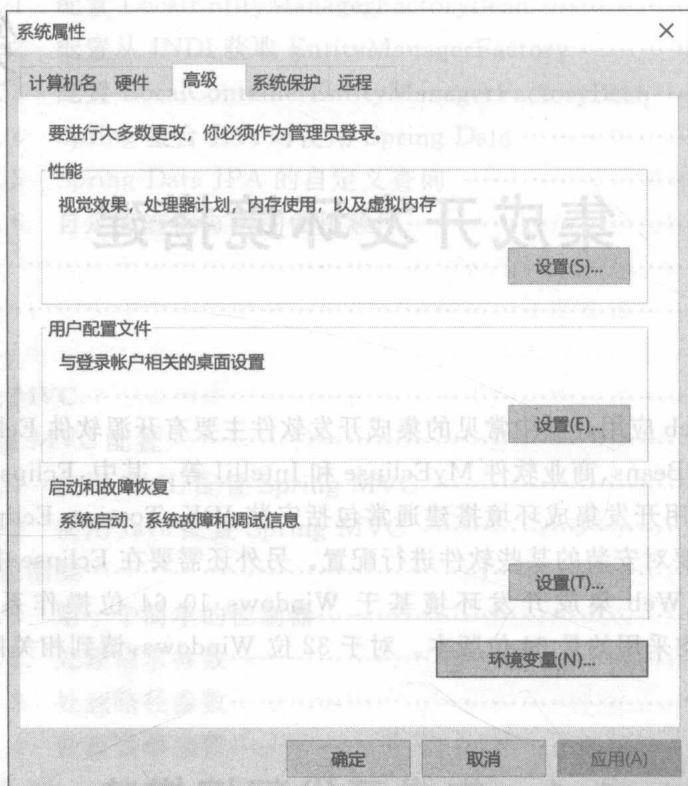


图 1-1 “系统属性”对话框

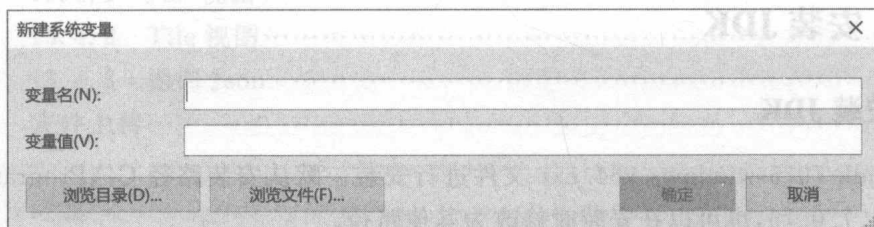


图 1-2 “新建系统变量”对话框

置”,选择“高级”→“环境变量”;选择“系统变量”下方窗口中的 Path 变量后,单击“编辑”按钮打开“编辑系统变量”对话框,单击“新建”按钮并输入%JAVA_HOME%\bin。

依次单击“编辑环境变量”对话框、“环境变量”对话框、“系统属性”对话框中的“确定”按钮,完成编辑系统变量 Path 的操作。

3. 测试 JDK 安装和配置是否成功

在 Windows 10 桌面左下方的搜索框输入 cmd,选择“命令提示符”。在“命令提示符”窗口输入命令 javac 后并按 Enter 键,如果显示如图 1-3 所示的 javac 使用帮助,说明 JDK 安装和配置成功了。

```

C:\Users\guoqing> javac
用法: javac <options> <source files>
其中, 可能的选项包括:
-g                生成所有调试信息
-g:none          不生成任何调试信息
-g: {lines, vars, source} 只生成某些调试信息
-nowarn         不生成任何警告

```

图 1-3 在“命令提示符”窗口输入命令 javac 后的显示效果

1.1.2 安装 Tomcat

Tomcat 官方网站提供了两种类型的文件供用户下载。exe 类型的文件是 Windows 下的可执行安装文件;. zip 类型的文件属于绿色软件,不用安装,解压即可使用。本书使用的是. zip 类型的文件。

1. 安装 Tomcat

将压缩文件 apache-tomcat-7.0.73.zip 解压到某个路径,例如 C:\JavaProgram\Tomcat\apache-tomcat-7.0.73\,即 Tomcat 的安装主目录为 C:\JavaProgram\Tomcat\apache-tomcat-7.0.73。在 Windows 10 资源管理器中,apache-tomcat-7.0.73 目录显示效果如图 1-4 所示。

bin	17/11/13 09:19
conf	17/11/13 09:19
lib	17/11/13 09:19
logs	18/3/6 11:55
temp	17/11/13 09:19
webapps	18/3/6 11:58
work	17/11/13 09:19
apache-tomcat-7.0.73-src	16/12/23 14:41
LICENSE	16/11/7 23:29
NOTICE	16/11/7 23:29
RELEASE-NOTES	16/11/7 23:29
RUNNING	16/11/7 23:29

图 1-4 apache-tomcat-7.0.73 目录显示效果

2. 测试 Tomcat 安装是否成功

首先到 apache-tomcat-7.0.73 的 bin 目录,双击 startup.bat,即可启动 Tomcat 服务器。启动 Tomcat 后,任务栏会显示 Tomcat DOS 窗口;请不要关闭该窗口。关闭该窗口将导致关闭 Tomcat。接着打开一个浏览器窗口,并在地址栏输入 http://localhost:8080 后按 Enter 键,如果浏览器显示 Tomcat 欢迎画面,如图 1-5 所示,说明 JDK、Tomcat 均已安装和配置成功。如果要关闭 Tomcat,请双击 bin 目录下的 shutdown.bat。

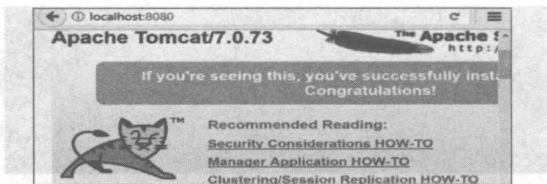


图 1-5 Tomcat 欢迎画面

1.1.3 安装 MySQL

解压文件 mysql-5.5.18-win32.rar 后,双击 MySQL 的安装文件 mysql-5.5.18-win32.msi 进行安装。默认安装路径为 C:\Program Files (x86)\MySQL\MySQL Server 5.5。为了与本书一致,注意在安装过程中将 MySQL 的登录账号 root 的密码设置为 root,字符编码改为 utf8(即 UTF-8)。

1.1.4 安装 Eclipse

1. 安装文件

解压文件 eclipse-jee-mars-2-win32-x86_64.zip 到 C:\JavaProgram\eclipse-jee-mars-2-win32-x86_64\即可完成 Eclipse 的安装。

2. 配置工作空间

Eclipse 安装结束后,在 C:\JavaProgram\eclipse-jee-mars-2-win32-x86_64\下创建文件夹 workspaces 作为工作空间。工作空间用于管理项目。在 Eclipse 平台中,要编写 Java 类需要首先创建项目。项目是 Eclipse 平台的基本工作单元。工作空间可以位于其他路径,如 F:\workspaces 或者 E:\MyApps。

第一次启动 Eclipse,会弹出设置工作空间的窗口。将工作空间设置为刚刚创建的 workspaces 的路径 C:\JavaProgram\eclipse-jee-mars-2-win32-x86_64\workspaces。

3. 设置 Eclipse 的默认 JRE

通常需要将 Eclipse 的默认 JRE 设置为前面安装的 JDK。

在 Eclipse 中,选择 Window→Preference,单击左侧的 Java→Installed JREs 节点,单击右方窗口中的 Add 按钮,在弹出的 JRE Type 窗口选择 Standard VM,单击 Next 按钮后弹出 JRE Definition 窗口,如图 1-6 所示。

在 JRE Definition 窗口中,单击 JRE home 右边 Directory 按钮浏览 JDK 的主目录 C:\Program Files\Java\jdk1.7.0_75,单击 OK 按钮后,在 JRE Definition 窗口单击 Finish 按钮,最后单击 Installed JREs 窗口下方的 OK 按钮,完成 Eclipse 默认 JRE 的设置。

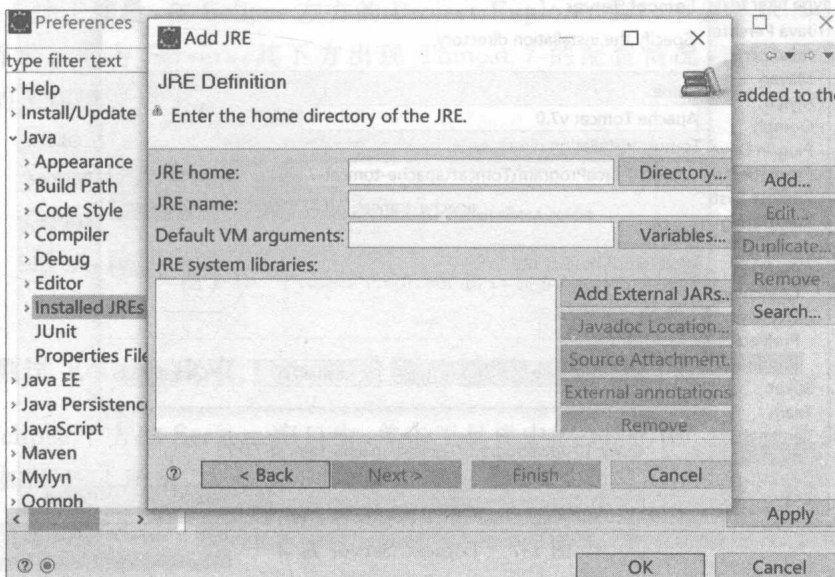


图 1-6 JRE Definition 窗口

1.1.5 Eclipse 集成 Tomcat

前面已经成功安装了 Tomcat 和 Eclipse。实际开发中如果总是到 %Tomcat%\bin\ 双击 startup.bat 和 shutdown.bat 来启动和关闭 Tomcat, 显得太麻烦且不能同步调试代码。Eclipse 提供了集成 Tomcat 的插件, 可以实现在 Eclipse 中启动和关闭 Tomcat, 进行 Web 应用程序的调试。

1. Eclipse 集成 Tomcat

Eclipse 集成 Tomcat 需要进行以下两步操作。

(1) 配置服务器运行环境。依次选择 Window → Preference → Server → Runtime Environment, 在 Server Runtime Environments 右上方窗口单击 Add 按钮; 在 New Server Runtime Environment 窗口选择 Apache Tomcat 7.0, 单击 Next 按钮, 弹出 Tomcat Server 窗口, 如图 1-7 所示。

在 Tomcat Server 窗口中, 单击 Browse 按钮浏览 Tomcat 7.0.73 的安装主目录 C:\Win10-JavaProgram\Tomcat\apache-tomcat-7.0.73 并选中, 单击下方的 Finish 按钮。

在 Server Runtime Environments 窗口中单击 OK 按钮完成服务器运行环境 Tomcat 7.0.73 配置。

(2) 新建服务器。在 Eclipse 下方的 Servers 窗口中, 如图 1-8 所示, 右击并选择 New → Server 命令, 打开 New Server 向导。

在 New Server 向导打开的 Define a New Server 窗口 (如图 1-9 所示) 中, 选择 Tomcat v7.0 Server, 其 Server runtime environment 为前面配置的 Apache Tomcat v7.0, 单击下方的 Finish 按钮。