Broadview
www.broadview.com.cn

Java技术大系

# Lightweight J2EE

# 轻量级J2EE企业应用实战

## ——Struts+Spring+Hibernate整合开发

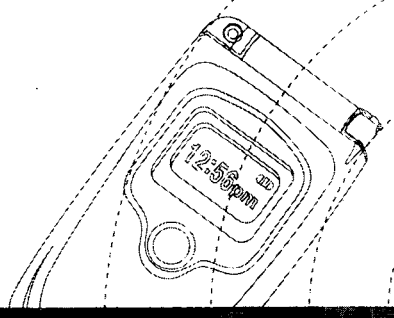
李刚 著

- 全面介绍J2EE的流行框架
- 详细介绍时下全部架构模式
- 包含多达7个实体关联的实用案例

光盘包含：
本书所有实例代码

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
HTTP://WWW.PHEI.COM.CN

Java技术大系



# 轻量级J2EE企业应用实战

## ——Struts+Spring+Hibernate整合开发

李刚 著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

## 内 容 简 介

本书所介绍的内容是作者多年 J2EE 开发经验的总结，内容涉及 Struts、Hibernate 和 Spring 三个开源框架，还介绍了 Tomcat 和 Jetty 两个开源 Web 服务器的详细用法，以及 J2EE 应用的几种常用架构。

本书不仅是一本 J2EE 入门图书，还详尽而细致地介绍了 JSP 各个方面，包括 JSP 2.0 的规范、Struts 的各种用法、Hibernate 的详细用法，以及 Spring 的基本用法。书中所介绍的轻量级 J2EE 应用，是目前最流行、最规范的 J2EE 架构，分层极为清晰，各层之间以松耦合的方法组织在一起。书的最后配备了两个实例，均采用了贫血模式的架构设计，以便于读者更快地进入 J2EE 应用开发。而第 8 章所介绍的其他架构模式则可作为读者对架构有更好把握后的提高部分。本书配套光盘包括各章内容所用的代码，以及整个应用所需要的开源类库等相关项目文件。

本书适用于有较好的 Java 编程基础，有初步的 J2EE 编程基础的读者。本书既可以作为 J2EE 初学者的入门书籍，也可作为 J2EE 应用开发者的提高指导。



**图书在版编目（CIP）数据**

轻量级 J2EE 企业应用实战：Struts＋Spring＋Hibernate 整合开发 / 李刚著.—北京：电子工业出版社，2007.4
（Java 技术大系）
ISBN 978-7-121-03998-0

I. 轻… II. 李… III. ①JAVA 语言－程序设计 ②软件工具－程序设计 IV. TP312 TP311.56

中国版本图书馆 CIP 数据核字（2007）第 033300 号

责任编辑：高洪霞 裴 杰
印 刷：北京天宇星印刷厂
装 订：三河市鹏成印业有限公司
出版发行：电子工业出版社
北京市海淀区万寿路 173 信箱 邮编 100036
开 本：787×1092 1/16 印张：34 字数：765 千字
印 次：2007 年 4 月第 1 次印刷
印 数：5000 册 定价：65.00 元（含光盘 1 张）

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系电话：（010）68279077；邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

# 前　言

目前，J2EE 应用确实很流行，从银行、证券系统，到企业信息化平台，甚至一些小公司，都争相使用 J2EE 应用。几年前，J2EE 应用是很“贵族”的产品，那时候使用 EJB 作为 J2EE 的核心，开发成本高，部署成本也高，开发者的学习曲线也陡峭。今天，轻量级 J2EE 应用的流行，让 J2EE 应用开始进入寻常百姓家。当然，轻量级 J2EE 应用是对经典 J2EE 应用的简化，在保留经典 J2EE 应用的架构、良好的可扩展性、可维护性的基础上，简化了 J2EE 应用的开发，降低了 J2EE 应用的部署成本。

轻量级 J2EE 应用让 J2EE 平台以更快的速度占领电子商务、电子政务等各种信息化平台市场。笔者从不否认对经典 J2EE 应用架构的喜爱，那种严谨的架构、全方位考虑的设计、优秀的分布式架构，无疑是一种编程的艺术。但它们太豪华了，以致于限制了它的市场占有率。可以这样说：经典 J2EE 应用是面向开发者的，而轻量级 J2EE 应用则面向用户。优秀的开发者会感慨并喜欢经典 J2EE 应用的设计，但市场则喜欢轻量级 J2EE 应用。轻量级 J2EE 应用模仿了经典 J2EE 应用的架构，保留了经典 J2EE 应用的各种优点，降低了学习难度和开发、部署成本，是一种更实际的信息化平台架构。

## 为什么写作本书

几年前，笔者主要从事实际的开发时，从未想过写一本书，忙是一个原因，更多原因是没有感触。如今，笔者已经在新东方 IT 培训中心担任 J2EE 培训讲师一年多，现已成为广州新东方软件工程师培训讲师的负责人。培训过程中，看到学生们求知若渴的眼睛，以及他们热切的需要：“老师，出一本关于某技术的书吧！”回想起 1999 年底时，笔者刚刚开发 J2EE 学习，完成一个简单的 EJB，居然花了将近一个月时间，其间苦痛只有程序员才懂。如今看到学生们的苦楚，想起更多 J2EE 学习者正备受煎熬，笔者愿意将多年的经验与大家一起分享，这些经验包含笔者多年废寝忘食后的恍然醒悟，也包含笔者跌落后艰难爬出的陷阱。希望这些经验能缩短读者朋友们的学习周期。

需要提醒读者朋友的是，所有的代码必须自己敲过，才是真正属于自己的代码。不要指望光看看本书，就可以成为一个编程高手，一定要踏踏实实地独立完成书中所有应用。学习编程是很辛苦却很有乐趣的事情，记得电影《阿甘正传》中有句话：“偶尔雨停了，可以看见星星。”这种场景很适合编程的境界，大部分时候都在埋头辛苦写代码，调试错误，只在当应用真正运行成功时，获得瞬间的快乐——这种快乐弥足珍贵，也是真正的快乐。

有时候我的学生会拿着他刚买的图书问我，这本书如何？很不幸，有时会发现名为 J2EE 的图书，居然在 JSP 页面中有 Hibernate 的 API。于是我无言以对，这样的图书到底想使读者成为怎样的开发者？阅读这样的图书不仅浪费时间，而且会造成错误的积累。

有感于此，笔者创作了本书，愿与各位读者共享多年的实践经验。即使今天，笔者依然与珠三角很多软件公司联系紧密，很多学生已在华为、立信、中企动力和京华网络等企业就业，有的学生已成长为技术经理，他们依然会就实际开发中的问题与笔者一起探讨，这些经验，都将出现在笔者的 J2EE 系列图书中。

## 本书有什么特点

与市面上已经存在的介绍 J2EE 应用的图书相比，本书有如下几个特色：

**1．经验丰富，针对性强**

笔者既担任过软件开发的技术经理，也担任过软件公司的培训导师，也从事过职业培训的专职讲师。这些经验影响了笔者写书的目的，不是一本学院派的理论读物，而是一本实际的开发指南。

**2．内容实际，实用性强**

本书所介绍的 J2EE 应用范例，规模可能并不大，但绝对是目前企业流行的开发架构，绝对严格遵守 J2EE 开发规范。而不是将各种技术杂乱地糅合在一起号称 J2EE。读者参考本书的架构，完全可以身临其境地感受企业实际开发。

**3．高屋建瓴，启发性强**

本书介绍的几种架构模式，几乎是时下最全面的 J2EE 架构模式。这些架构模式可以直接提升读者对系统架构设计的把握。

## 本书写给谁看

本书适用于有较好的 Java 编程基础，有初步的 J2EE 编程基础的读者。本书既可以作为 J2EE 初学者的入门书籍，也可作为 J2EE 应用开发者的提高指导。

李　刚

2007-1-12

# 目 录 CONTENTS

# 第1章

# J2EE 应用运行及开发环境的安装与配置

## 本章要点

- JDK 的下载和安装
- 环境变量的设置
- Tomcat 的安装和配置
- 在 Tomcat 中部署 Web 应用
- Jetty 的安装和配置
- 在 Jetty 中部署 Web 应用
- Eclipse 的安装
- Eclipse 插件的安装

J2EE 应用以其稳定的性能、良好的开放性及严格的安全性，深受企业应用开发者的青睐。实际上，对于信息化要求较高的行业，如银行、电信、证券及电子商务等行业，都会选择使用 J2EE 作为企业的信息平台。

对于一个企业而言，选择 J2EE 构建信息化平台，更体现了一种长远的规划：企业的信息化是不断整合的过程，在未来的日子里，经常会有不同平台、不同系统的异构系统需要整合。J2EE 应用提供的跨平台性、开放性及各种远程访问的技术，为异构系统的良好整合提供了保证。

本书介绍的不是基于 EJB 的 J2EE 应用的开发，因为 EJB 应用的开发周期过长，且必须运行在 J2EE 容器中。而本书介绍的轻量级 J2EE 应用，完全可以运行在 Web 容器中，无需 EJB 容器的支持，但其应用的稳定性及效果都可以得到保证。

# 1.1 JDK 的下载和安装

J2EE 应用的开发及运行都离不开 JDK 的支持。虽然 Java 程序是跨平台的，但 JDK 不是跨平台的。因此，在不同的平台上需要安装不同的 JDK。下面分别介绍在 Windows 和 Linux 下 JDK 的安装。

## 1.1.1 Windows 下 JDK 的下载和安装

目前，JDK 主要有如下三个版本。

J2SE：Java 标准版本，包括开发桌面应用的系列类库。

J2EE：包含 Java 标准版，还增加了企业应用开发所需的类库。

J2ME：Java 2 平台微型版，被使用在资源受限制的小型消费型电子设备上。

本书介绍的是 J2EE 应用的开发，推荐使用 J2EE 的 JDK。下载和安装请按如下步骤进行。

（1）登录 http://www.sun.com 站点，根据所使用的操作系统，选择 J2EE 的最新版本，笔者使用的是 j2eesdk-1_4_02_2005Q2，推荐读者也使用该版本的 J2EE。

下载完成后，得到一个名为 j2eesdk-1_4_02_2005Q2-windows-ml.exe 的可执行文件，该文件就是 J2EE 的安装文件。

（2）双击下载的可执行性文件，出现如图 1.1 所示的安装向导对话框，表明 JDK 开始安装。

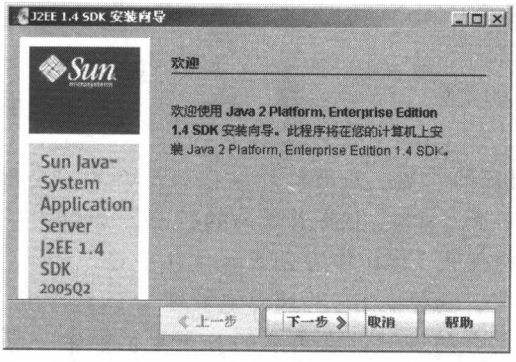



图 1.1 J2EE JDK 安装界面

（3）安装过程与安装其他 Windows 软件并没有太大的不同，同样是多次单击【下一步】按钮，需要选择安装路径等步骤。笔者建议不要修改 JDK 的安装路径，若要修改也仅仅修改其盘符，程序出现如图 1.2 所示的安装路径选择对话框。

笔者将安装路径选择在 D 盘（因为笔者将所有的工具软件都放在 D 盘），而建议不要修改后面的 Sun\AppSever 路径。直到出现如图 1.3 所示的对话框。

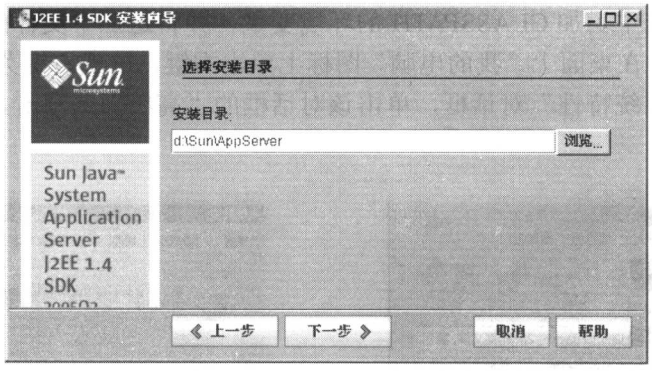



图 1.2 安装路径选择界面

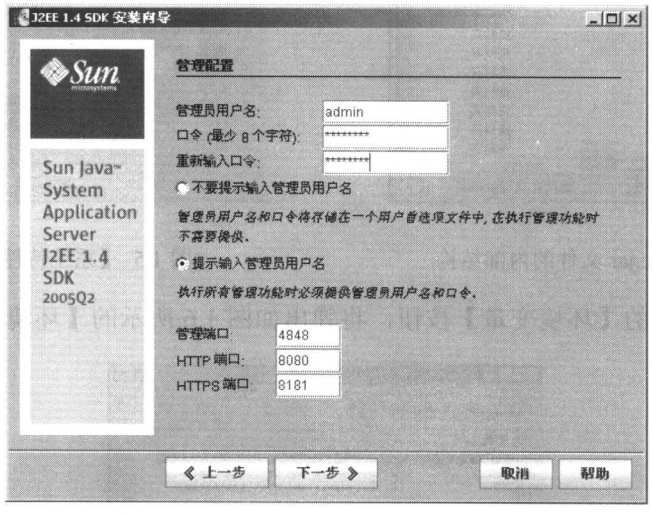



图 1.3 管理员口令窗口

（4）选择“提示输入管理员用户名”单选框，然后在口令和重新输入口令的密码框中输入两个系统的口令，口令的长度最少必须为 8 个字符。然后单击【下一步】按钮，在下一个对话框中再次单击【下一步】按钮，程序开始安装。

（5）安装成功后，增加编译和运行必需的环境变量。编译和运行 Java 程序必须增加 CLASSPATH 环境变量。在编译和运行 Java 程序时，需要 JDK 的系统类，如 java.lang.String 等，Java 程序会根据 CLASSPATH 环境变量指定的路径搜索这些类。因此该环境变量就是系列的搜索路径，编译和运行 J2EE 的应用主要需要如下三个 jar 文件：

- Sun\AppServer\jdk\lib\tools.jar
- Sun\AppServer\jdk\lib\dt.jar
- Sun\AppServer\lib\j2ee.jar

这些 jar 文件表面看起来是一个文件，其实是一系列的路径，选择 WinRAR 文件打开其中任意一个文件，看到如图 1.4 所示的窗口。

接着，在系统中增加 CLASSPATH 的环境变量，并将这三个文件的路径添加进去，添加的方法如下：在桌面上“我的电脑”图标上单击右键，出现右键菜单，单击“属性”菜单项，出现“系统特性”对话框，单击该对话框的“高级”选项卡，出现如图 1.5 所示的对话框。

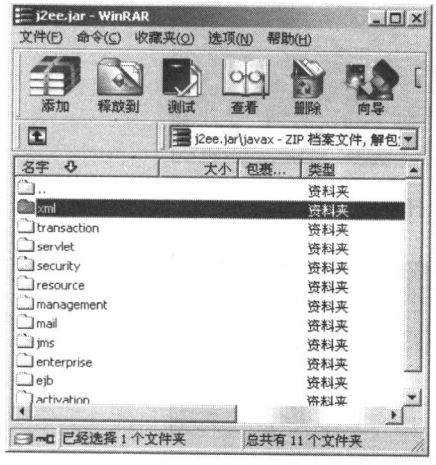



图 1.4 j2ee.jar 文件的内部结构

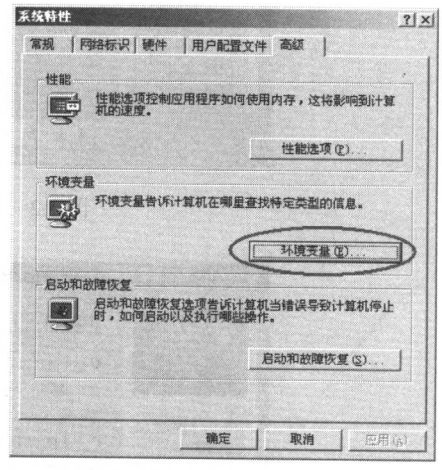



图 1.5 【系统特性】对话框

单击图 1.5 中的【环境变量】按钮，将弹出如图 1.6 所示的【环境变量】对话框。

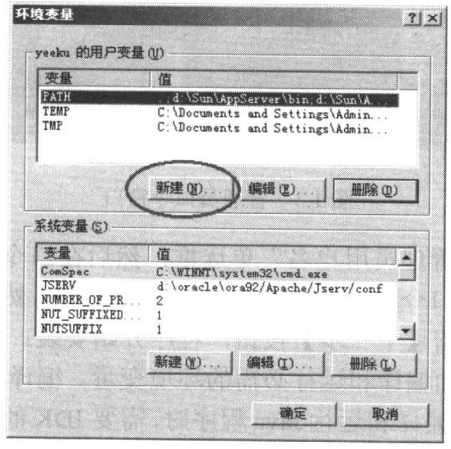



图 1.6 环境变量设置对话框

单击图 1.6 中的【新建】按钮，出现如图 1.7 所示的对话框，读者可以看到增加的 CLASSPATH 环境变量已将 dt.jar, tools.jar, j2ee.jar 三个文件设置到该环境变量中。因为 Windows 多个环境变量中的间隔符号是“;”，所以多个路径之间以英文分号隔开。增加后效果如图 1.7 所示。

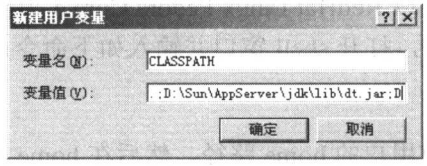



图 1.7 增加 CLASSPATH 环境变量后的效果

用户在编译和运行 Java 程序时，需要用到 java 和 javac 两个命令。由于 Windows 对于外部命令，都按 PATH 环境变量指定的路径搜索可执行性程序，因此为了可以执行 java 和 javac 等命令，应将 java 和 javac 所在的路径添加到 PATH 中。java 和 javac 的路径为 D:\Sun\AppServer\jdk\bin，通常系统已经有了 PATH 环境变量，因此只需将该路径添加到 PATH 变量中即可。

按上面的步骤修改 PATH 环境变量，修改后的效果如图 1.8 所示。

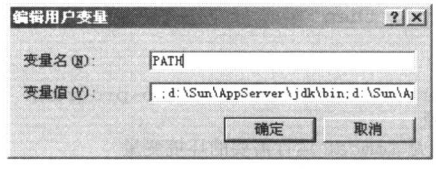



图 1.8 PATH 环境变量的设置

**注意**：在两个环境变量的设置中，都包含了一个“.”的路径，这个“.”代表系统的当前路径，如果没有增加该路径，可能导致运行 Java 程序时，class 文件已在当前路径，但在系统提供的文件中找不到该文件。

## 1.1.2 Linux 下 JDK 的下载和安装

Linux 下 JDK 的下载和安装与 Windows 下并没有太大的不同，只是对一些环境的设置稍有不同。下载和安装 Linux 下的 JDK 请按如下步骤进行。

（1）登录 http://www.sun.com 站点，同样选择 JDK 的 j2eesdk-1_4_02_2005Q2 的版本，区别只是在下载 Linux 版本成功后，得到的是 j2eesdk-1_4_02_2005Q2-linux-ml.bin 文件。

（2）进入 j2eesdk-1_4_02_2005Q2-linux-ml.bin 所在的路径，执行如下命令：

```
chmod  777  j2eesdk-1_4_02_2005Q2-linux-ml.bin
```

该命令修改 j2eesdk-1_4_02_2005Q2-linux-ml.bin 文件为可执行文件，然后输入如下命令：

```
./ j2eesdk-1_4_02_2005Q2-linux-ml.bin
```

按回车键后，出现与图 1.1 类似的【安装向导】对话框。

（3）等待安装结束后，设置环境变量。环境变量的设置与在 Windows 下完全相同，

只是设置的方式稍有区别（以 RedHat Linux Fedora Core 4 为例）。

（4）登录 Linux 系统后，打开 shell 窗口并输入如下命令：

```
cd
```

（5）该命令将进入当前用户的 home 路径，然后在 home 路径下输入如下命令：

```
ls -a
```

（6）该命令将列出当前路径下所有的文件（包括隐藏文件），环境变量的设置是通过.bash_profile 文件设置的。

（7）使用无格式编辑器编辑该文件来增加 CLASSPATH 环境变量，并修改 PATH 环境变量，修改后的.bash_profile 文件如下（文件中以#开头的行是注释）：

```
# Get the aliases and functions
# 如果.bashrc 文件存在，执行该文件的内容
# 因此，也可以在.bashrc 文件中设置变量
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi
# User specific environment and startup programs
#下面用于设置环境变量
#JAVA_HOME 环境变量是 Tomcat 运行需要的环境变量
JAVA_HOME=/home/yeeku/sun/jdk
#其中$JAVA_HOME，表示引用宏变量，第二条路径实际就是
#  /home/yeeku/sun/jdk/bin
PATH=.:$PATH:$HOME/bin:$JAVA_HOME/bin
# 设置 CLASSPATH 环境变量
CLASSPATH=.:$JAVA_HOME/lib/dt.jar:JAVA_HOME/lib/tools.jar
# 导出环境变量
export ANT_HOME
export JAVA_HOME
export CLASSPATH
export PATH
unset USERNAME
export LD_LIBRARY_PATH=/usr/local/lib:$LD_LIBRARY_PATH
```

（8）在上面中的环境变量中，路径之间以“:”分隔，因为 Linux 以“:”作为路径分隔符。

（9）重新登录，或者执行如下命令：

```
source .bash_profile
```

两种方式都是为了运行该文件，使文件中关于环境变量的设置起作用。

## 1.2 Tomcat 的下载和安装

Tomcat 是 Java 领域最著名的开源 Web 容器，简单、易用且稳定性极好。既可以作为个人学习之用，也可以作为商业产品发布。

Tomcat 不仅提供了 Web 容器的基本功能，还支持 JAAS 和 JNDI 绑定等。

### 1.2.1 Tomcat 的下载和安装

因为 Tomcat 完全以 Java 编写，因此与平台无关，既可以运行在 Windows 平台上，也可以运行在 Linux 平台上。两个平台上的安装和配置也基本相同，只是环境变量的设置稍有差别而已，关于 Linux 下环境变量的设置请参考 1.1.2 节。本节将以 Windows 平台为例。

下载和安装 Tomcat 按如下步骤进行。

（1）登录 http://tomcat.apache.org 站点，下载 Tomcat 合适的版本，如果使用 JDK1.4，则建议使用 Tomcat 5.0.x 系列，而不是使用 Tomcat 5.5.x 系列。

Tomcat 5.0.x 目前最新的稳定版本是 5.0.28，建议下载该版本。在 Windows 平台下载 zip 包，Linux 下载 tar 包。建议不要下载其安装文件。

（2）解压缩刚下载到的压缩包，解压缩后应有如下文件结构。

- bin：存放启动和关闭 Tomcat 的命令的路径。
- common：存放所有的 Web 应用都需要的类库等。
- conf：存放 Tomcat 的配置，所有的 Tomcat 的配置都在该路径下设置。
- log：这是一个空路径，该路径用于保存 Tomcat 每次运行后产生的日志。
- server：存放 Tomcat 运行所需要的基础类库，该路径是 Tomcat 运行的基础。该路径下还包含一个 webapps 路径，并存放 Tomcat 两个控制台。
- shared：该路径也是一个空路径，用于系统共享的类库，该路径下包括 classes 和 lib 两个路径，其中 classes 用于存放 class 文件，而 lib 用于存放 Jar 文件。
- temp：保存 Web 应用运行过程中生成的临时文件。
- webapps：该路径用于部署 Web 应用，将 Web 应用复制在该路径下，Tomcat 会将该应用自动部署在容器中。
- work：保存 Web 应用运行过程中编译生成的 class 文件。该文件夹可以删除，但每次应用启动时将自动建立该路径。
- LICENSE 等相关文档。

（3）将解压缩后的文件夹放在到任意路径下。

（4）Tomcat 的运行需要一个环境变量：JAVA_HOME。不管是 Windows 还是 Linux，只需要增加该环境变量即可，该环境变量的值指向 JDK 安装路径。

（5）启动 Tomcat，对于 Windows 平台，只需要双击 Tomcat 安装路径下 bin 路径中的 startup.bat 文件即可。对于 Linux，进入 Tomcat 安装路径的 bin 路径下，然后运行如下命令：

```
chmod 777 startup.sh
```

该命令将 startup.sh 文件变成可执行性文件，接着用如下命令运行 startup.sh 即可：

```
./startup.sh
```

启动 Tomcat 之后，打开浏览器，在地址栏输入 http://localhost:8080，然后回车，浏