# 创建 Java 开发环境

## 实验目的和要求

**掌握JDK的下载、安装及配置的步骤及方法。**

**掌握Java程序创建、编译和运行的基本步骤。**

**掌握控制台输入输出基本方法。**

**掌握Java中的变量、表达式等基本语法。**

**掌握方法、数组的使用方法。**

**掌握数值类型的使用方法。**

**进一步熟悉顺序、选择和循环这三种基本结构。**

### 0.1.0 JDK简介

Java 开发工具箱即 Java Development Kit（JDK），是由 Sun 公司提供的免费软件包，其中 包括针对 Java 源代码的编译程序（javac）、执行编译程序的虚拟机器（virtual machine）、JDK 所提供类的说明文件，以及许多对应的实例程序等。JDK 是一个完整的程序开发平台，几乎所 有学习 Java 语言的程序员都知道这一套工具，因此掌握 JDK 是学好 Java 的第一步。 目前，主流的 JDK 是由 Sun 公司发布的，除了 Sun 公司之外，还有很多公司和组织都开发 了自己的 JDK，如 IBM 公司开发的 JDK，BEA 公司开发的 Jrocket，还有 GNU 组织开发的 JDK 等。

### 0.1.1 下载与安装 Java 开发工具箱

目前Sun公司官方网站http://www.sun.com/download/发布的较新版的JDK是 Java （TM） SE Development Kits，即 JDK 6。下载的是较新的 JDK 6，在下载页面 http://java.sun.com/ javase/downloads/index.jsp 找到较新的 JDK 6 Update 17 进行下载。关于 JDK 6 更加详细的信息 可以在 Sun 公司官方网页 http://java.sun.com/javase/6/webnotes/ReleaseNotes. html 查看，其中详 细介绍了 JDK 6 的新的变化及漏洞的修正等问题。可以选择 Windows 平台 的版本或者 Linux 平台的版本等。下载名为 jdk-6u17-windows-i586.exe 的安装 文件后，双击进入安装过程。 在【自定义安装】对话框中，读者可以自己选择安装目录，书中示例安装至默认文件夹 C:\Program Files\Java\jdk1.6.0\_17 中。选择好安装目录后，单击【下一步】按钮开始安装。 单击【完成】按钮，完成 JDK 的安装。 至此，已正确完成 JDK 6 的安装。下面将简单介绍安装 JDK 目录下的文件夹内容。

### 0.1.2 JDK 目录

打开 JDK 的安装目录，软件安装的根目录，包含版权、许可及 README 等文件。 src：组成 Java 核心 API 的所有类的 Java 编程语言源文件（即 java.\*、javax.\*和某些 org.\*包的源文件，但不包括 com.sun.\*包 的源文件） 。此源文件将帮助开发者学习和使用 Java 编程语言， 在 Eclipse 开发环境中可以加入这些源文件。这些文件不包含特 定于某个平台的实现代码，且不能用于重新生成类库。 bin 目录：包括 JDK 开发工具的可执行文件，如编译器、解释器等。该目录下主要的可 执行文件的作用如下所示。 javac：Java 编译器，将 Java 源代码编译成字节码文件（\*.class）。 java：Java 解释器，执行编译后的 Java 应用程序代码。 appletviewer（小程序浏览器） ：一种执行 HTML 文件中的 Java 小程序类的 Java 浏览器。

javadoc：根据 Java 源代码及其说明语句生成的 HTML 文档。 jdb：Java 调试器，可以逐行地执行程序、设置断点和检查变量。 javap：Java 反编译器，显示编译类文件中的可访问功能和数据，并显示字节代码含义。

javah：C/C++头文件生成器，产生可以调用 Java 程序的 C 过程，或创建能被 Java 程 序调用的 C 过程的头文件。 jar：多用途的存档及压缩工具，是一个 Java 应用程序，可将多个文件合并为单个.jar 归档文件。 lib 目录：开发工具使用的归档文件，包括 tools.jar（该包支持 JDK 的工具和实用程序的 非核心类） 、dt.jar 等。 demo 目录：主要包括 Java 平台的编程示例（包含源代码），如演示 Applet 和应用程序， 这些示例包括使用 Swing 和其他 Java 基础类以及 Java 平台调试器体系结构的示例。 include 目录：目录下是 Win32 子目录，都是本地方法文件，包括 C 的头文件，支持使 用 Java 本机界面、JVM 工具界面以及 Java 平台的其他功能进行本机代码编程的头文件。

sample 目录：其中是某些 Java API 的编程示例（包含源代码） 。 jre 目录：是运行时环境，包括 JDK 使用的 Java runtime environment（JRE）的实现。JRE 包括 Java 虚拟机（JVM）、类库以及其他支持执行以 Java 编程语言编写的程序文件。 可选安装的Java Runtime Environment是一个可单独下载的产品。通过JRE，程序员可以运 行用Java编程语言编写的应用程序。与JDK相似，JRE包含Java虚拟机（JVM）、组成Java平台 API的类及支持文件。与JDK不同的是，JRE不包含像编译器和调试器这样的开发工具。 在基本了解 JDK 目录中的内容后，接下来就可以配置开发环境了。

注 意 关于 JDK 目录下 bin 目录中的可执行文件的用处，可以查阅详细资料，学习其具体的 使用方法。

### 0.1.3 配置环境变量

虽然已经安装了 Java 的运行环境，但如果要生成各个平台都可以运行的程序，还需要用 javac 编译器把 Java 源程序（\*.java）转换成与平台无关的字节码文件（\*.class），并使用 Java 解释器执行字节码文件。 javac 编译器、Java 解释器可以在 Windows 操作系统中的命令提示符窗口下运行。由于命 令提示符窗口一般只执行系统默认的 DOS 命令，如 cd、dir、clear 等，因此要执行 JDK 中的命 令，如 javac、java、javadoc、javap 等，则需要配置 Path、CLASSPATH 和 JAVA\_HOME 环境 变量。 这 3 个环境变量的含义分别如下所示。 Path：系统搜索可执行程序的路径。 CLASSPATH：该变量的含义是搜索 Java 类的路径，这个路径可以包含一些 JAR 归档文 件，也可以是一个目录。 JAVA\_HOME：JDK 的安装目录。 设置环境变量主要针对 Windows NT（包括 Windows 2000、Windows XP、Windows 2003） 进行说明，如果读者需要设置 Linux、Solaris 等操作平台的环境变量，可以查阅相关资料，其 设置内容与 Windows 一样，只是设置方法略有不同。

#### 1．操作平台 Windows NT（Windows 2000、Windows XP、Windows 2003） 具体的操作步骤如下所示。

（1）右击【我的电脑】图标，弹出快捷菜单，选择【属性】命 令，打开【系统属性】对话框。 （2）在【系统属性】对话框中切换到【高级】标签，单击【环 境变量】按钮，打开【环境变量】对话框。 （3）在【系统变量】选项区域的【变量】栏中选择变量名 Path， 并单击【编辑】按钮，打开【编辑系统变量】对 话框。

（4）打开【编辑系统变量】对话框后，保持【变量名】为 Path，在【变量值】文本框中添 加“ ;C:\Program Files\Java\jdk1.6.0\_17\bin”（注意不要修改文本框中原来的值）， 单击【确定】按钮。这样就完成了 Path 环境变量的设置。 这步完成后，就可以在 Windows 的命令提示符窗口执行 javac、java、javadoc 等命令了。 但是只设置 Path 环境变量还是不够的，因为 Java 运行时还需要调用 JDK 系统默认的 Java 类包， 否则 javac 编译器无法将源文件编译成类文件，因此还需要设置 CLASSPATH 环境变量。 （5）返回【环境变量】对话框中，单击【系统变量】选项区域下的【新建】 按钮，打开【新建系统变量】对话框。 （6）在【新建系统变量】对话框的【变量名】文本框中输入 CLASSPATH，在【变量值】文本 框中输入“.;C:\Program Files\Java\jdk1.6.0\_17\lib\dt.jar;C:\Program Files\Java\jdk1.6.0\_17\lib\ tools.jar”，然后单击【确定】按钮，这样就完成了 CLASSPATH 环境变量的设置。这里的“.” 非常重要，加上“.”后表示在任意目录下均可执行 JDK。 （7）JAVA\_HOME 变量的设置跟 CLASSPATH 相同这里的“C:\Program Files\Java\jdk1.6.0\_17”是作者安装 JDK 的目录，该文本框中的内容根据读者自己安装 JDK 目录 的不同而不同。

#### 2．测试设置是否正确

设置完成后，测试环境变量是否配置成功，选择【开始】|【运行】命令，在打开的对话框 中输入 cmd，即可打开 Windows 的命令提示符窗口（注意：为了更加清晰地展现操作，本书中 的命令提示符窗口背景颜色全部设置成了白色），输入 java -version，然后按【Enter】键，出现提示信息，显示读者目前安装的版本是 1.6.0\_17 等信息；输入 java、javac 等命令同样都出现提示信息，则表明读者安装 JDK 成功。否则就 要查看环境变量是否配置正确。

## 实验目的

**掌握JDK的下载、安装及配置的步骤及方法。**

**掌握Java程序创建、编译和运行的基本步骤。**

**掌握控制台输入输出基本方法。**

**掌握Java中的变量、表达式等基本语法。**

**掌握方法、数组的使用方法。**

**掌握数值类型的使用方法。**

**进一步熟悉顺序、选择和循环这三种基本结构。**

## **实验内容与、结果及分析**

### T4 page 94 3.11

原题：（给出一个月的总天数）编写程序，提示用户输入月份和年份，然后显示这个月的天数。例如：如果用户输入的月份是2而年份是2012，那么程序应该显示“February 2012 has 29 days”（2012年2月有29天）。入如果用户输入的月份为3而年份为2015，那么程序就应该显示“March 2015 has 31 days”（2015年3月有31天）。

源代码：

**package** dengxiaoping\_1\_4;

**import** java.util.Scanner;

**public** **class** T4 {

**public** **static** **void** main(String[] args) {

//P94 3.11

Scanner in = **new** Scanner(System.***in***);

System.***out***.println("请输入年份：");

**int** y = in.nextInt();

System.***out***.println("请输入月份：");

**int** m = in.nextInt();

**switch**(m)

{

**case** 1:

**case** 3:

**case** 5:

**case** 7:

**case** 8:

**case** 10:

**case** 12:

System.***out***.printf("%d年的%d月有31天",y,m);

**break**;

**case** 2:

**if**((y%4==0&&y%100!=0)||y%400==0)

System.***out***.printf("%d年的%d月有29天",y,m);

**else**

System.***out***.printf("%d年的%d月有28天",y,m);

**break**;

**case** 4:

**case** 6:

**case** 9:

**case** 11:

System.***out***.printf("%d年的%d月有30天",y,m);

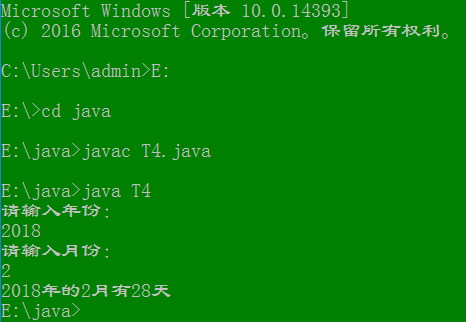
**break**;

}

}

}

运行结果及截图：



### T5 P123 4.6

原题：（圆上的随机点）编写一个程序，产生一个圆心在（0，0）、半径为40的圆上面的三个随机点，显示由这三个点组成的三角形的三个角的度数。（提示：产生0—2π之间的一个以弧度为单位的随机角度α，则这个角度所确定的点为（r\*cos（α），r\*sin（α）））。

源代码：

**package** dengxiaoping\_1\_5;

**import** java.util.Scanner;

**public** **class** T5 {

**final** **static** **int** ***r*** = 40;

**public** **static** **void** main(String[] args) {

//P128 4.6

Scanner in = **new** Scanner(System.***in***);

**double** x1,y1,x2,y2,x3,y3;

x1 = Math.*random*()\*80-39;

y1 = Math.*sqrt*(40\*40-x1\*x1);

x2 = Math.*random*()\*80-39;

y2 = Math.*sqrt*(40\*40-x2\*x2);

x3 = Math.*random*()\*80-39;

y3 = Math.*sqrt*(40\*40-x3\*x3);

**double** a = Math.*sqrt*((x2-x3)\*(x2-x3)+(y2-y3)\*(y2-y3));

**double** b = Math.*sqrt*((x1-x3)\*(x1-x3)+(y1-y3)\*(y1-y3));

**double** c = Math.*sqrt*((x1-x2)\*(x1-x2)+(y1-y2)\*(y1-y2));

**double** A = Math.*toDegrees*(Math.*acos*((a\*a-b\*b-c\*c)/(-2\*b\*c)));

**double** B = Math.*toDegrees*(Math.*acos*((b\*b-a\*a-c\*c)/(-2\*a\*c)));

**double** C = Math.*toDegrees*(Math.*acos*((c\*c-b\*b-a\*a)/(-2\*a\*b)));

System.***out***.printf("三个顶点分别为（%.2f，%.2f），（%.2f，%.2f），（%.2f，%.2f）\n",x1,y1,x2,y2,x3,y3);

System.***out***.println("the three angles are "+

Math.*round*(A\*100)/100.0+" "+

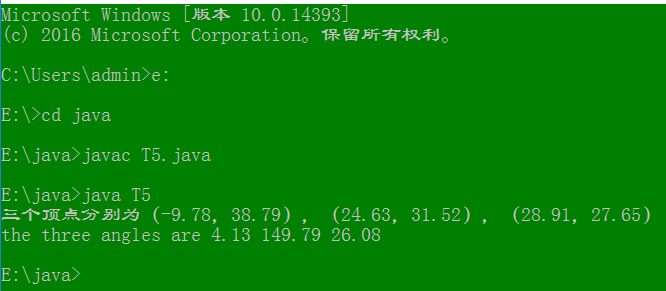
Math.*round*(B\*100)/100.0+" "+

Math.*round*(C\*100)/100.0);

}

}

运行结果及截图：



### T6 P131 4.21

原题：（检查SSN）编写一个程序，提示用户输入一个社保号码，它的格式是DDD-DD-DDDD，其中D是一个数字。你的程序应该判断输入是否合法。下面是一个运行示例：

Enter a SSN: 232-23-5435

232-23-5435 is a valid social security number

Enter a SSN: 23-23-5435

23-23-5435 is an invalid social security number

源代码：

**package** dengxiaoping\_1\_6;

**import** java.util.Scanner;

**public** **class** T6 {

**public** **static** **void** main(String[] args) {

// P131 4.21

**int** i = 0;

**int** j = 0;

**int** k = 0;

Scanner in = **new** Scanner(System.***in***);

System.***out***.println("Enter a SSN:");

String s = in.nextLine();

**while**(**true**)

{

**if**((i==3||i==6)&&s.charAt(i)=='-')

k++;

**if**(Character.*isDigit*(s.charAt(i)))

j++;

i++;

**if**(k==2&&j==8)

{

System.***out***.printf("%s is a valid social security number",s);

**break**;

}

**else** **if**(i==10)

{

System.***out***.printf("%s is an invalid social security number",s);

**break**;

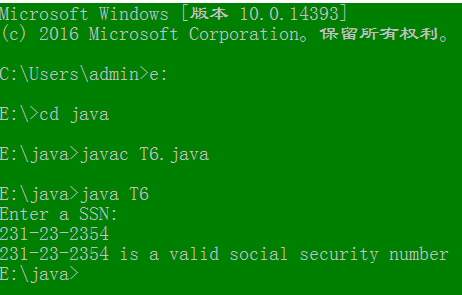
}

}

}

}

运行结果及截图：



### T7 P170 5.51

原题：（最长的共同前缀）编写一个程序，提示用户输入两个字符串，显示两个字符串最长的共同前缀。下面是运行示例：

Enter the first string:Welcome to C++

Enter the second string:Welcome to programming

The common prefix is Welcome to

Enter the first string:Atlanta

Enter the second string:Macon

Atland and Macon have no common prefix

源代码：

**package** dengxiaoping\_1\_7;

**import** java.util.Scanner;

**public** **abstract** **class** T7 {

**public** **static** **void** main(String[] args) {

// P170 5.51

**int** i = 0;

Scanner in = **new** Scanner(System.***in***);

System.***out***.println("Enter the first string:");

String s1 = in.nextLine();

System.***out***.println("Enter the second string:");

String s2 = in.nextLine();

**while**(**true**)

{

**if**(s1.charAt(i)==s2.charAt(i))

{

i++;

}

**else**

**break**;

}

System.***out***.println("The common prefix is "+s1.substring(0, i));

**if**(i==0)

{

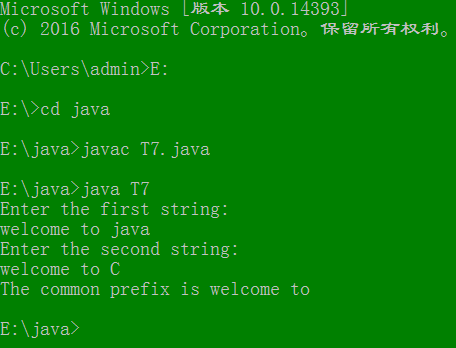
System.***out***.printf("%s and %s hava no common prefix",s1,s2);

}

}

}

运行结果及截图：



### T8 P60 2.6

原题：（求一个整数各位数的和）编写程序，读取一个在0—1000之间的整数，并将该整数的各位数字相加。例如：整数是932，各位数字之和为14.

提示：利用操作符%分解数字，然后使用操作符/去掉分解出来的数字。例如932%10=2，932/10=93。下面是一个运行示例：

Enter a number between 0 and 1000: 999

The sum of the digits is 17

源代码：

**package** dengxiaoping\_1\_8;

**import** java.util.Scanner;

**public** **class** T8 {

**public** **static** **void** main(String[] args) {

// P60 2.6

Scanner in = **new** Scanner(System.***in***);

System.***out***.println("Enter the number between 0 and 1000:");

**int** a = in.nextInt();

System.***out***.println("The sum of the digits is:"+(a/100+a%100/10+a%100%10));

}

}

运行结果及截图：

