



南昌大学实验报告

学生姓名：黄晨箬 学 号：6109119066 专业班级：计算机 193 班

实验类型：☐ 验证 ☐ 综合 ☐ 设计 ☐ 创新 实验日期：2021.3.31 实验成绩：

一、实验项目名称

Java 开发环境的熟悉

二、实验目的

- 1、掌握下载JDK 软件包、Eclipse软件的方法；
- 2、掌握设置Java 程序运行环境的方法；
- 3、掌握编写与运行Java 程序的方法；
- 4、了解Java 语言的概貌；
- 5、为学习Java 语言和进行Java 程序的实验做好准备工作。

三、实验基本原理

1、JDK 的下载与安装

下载能安装在相应的操作系统类型中的J2SE系统安装程序。为了建立基于SDK 的Java 运行环境，需要先下载Oracle公司的免费SDK软件包。SDK 包含了一整套开发工具，其中包含对编程最有用的是Java 编译器、Applet查看器和Java 解释器。

在浏览器中输入网址：

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>，可看到Oracle公司网站页面，然后在里面找到下载的连接进行下载就行。目前的版本最高是JDK1.15，但是在使用上更多的还是使用JDK1.8版本。

下载完后进行安装，和一般的软件安装过程差不多，自己选择目录进行jdk的安装。在本实验中假设JDK的安装路径是“D:\ProgramFiles\Java\jdk1.8.0_161”。

2、Java运行环境配置

JDK 中的工具都是命令行工具，需要从命令行即MS-DOS 提示符下运行它们。很多人可能会不习惯，但这是Oracle公司特意采取的策略，为的是把精力更多地投入到Java 语言本身而不是花在开发工具上。安装好JDK后，如果想在系统的任意目录下编译和运行编写好的Java程序，可手工配置Windows 的环境变量。

本实验我们将在 windows10 系统下环境变量中设置 JDK 运行路径。（实验中 JDK 的安装路径均使用安装程序的默认路径：D:\ProgramFiles\Java\jdk1.8.0_161）（如安装路径设置不同，则后续设置做相应路径变化）

第一步：鼠标右键点击“我的电脑”，选择“属性”，在弹出对话框中选择“高级属性设置”下的“环境变量”。添加系统环境变量 JAVA_HOME，让该环境变量的值是 JDK 目录结构的根目录，例如：D:\Program Files\Java\jdk1.8.0_161（如图 1 所示）

第二步：在“系统变量”中选择“Path”，点击“编辑”按钮，在弹出对话框中的“变量值”文本框中添加“%JAVA_HOME%\bin”（如果前一个变量值后没有“;”作为分隔符，则需添加“;”再写入路径信息），点击“确定”（如图 2 所示）。

第三步：在“系统变量”下点击“新建”按钮，在弹出对话框中的“变量名”后输入“classpath”，在“变量值”后输入“.;%JAVA_HOME%\lib”（注意此处的“.”必须要写），点击“确定”（如图 3 所示）。

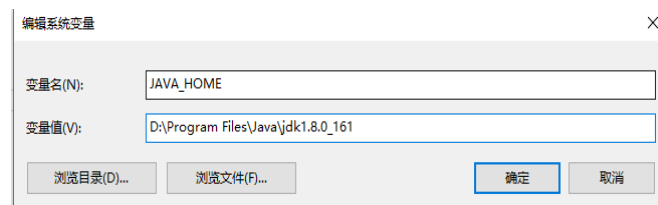


图 1 Windows 系统环境变量操作界面

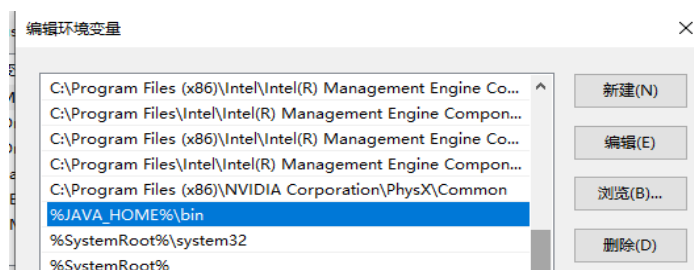


图 2 编辑系统变量 Path

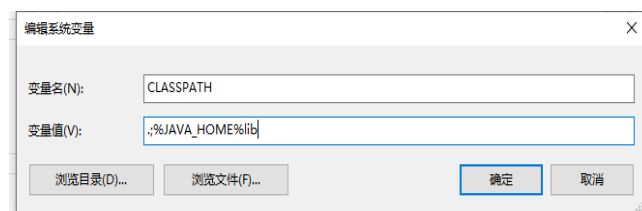


图 3 新建系统变量 classpath

环境变量设置完毕。以后需要通过 DOS 命令直接编译和执行 java 程序时，只需进入 java 程序所在的存放目录下，通过 javac 和 java 命令编译和执行相应程序。

3、在 DOS 环境下编译运行 Java 应有程序

JDK 应用程序安装好后，可以通过下面两种方式编译、执行 Java 应用程序：

(i) 设置路径后再使用 JDK 命令编译、运行程序

在按照实验 1 的步骤设置好 JDK 的路径后，使用任意文本编辑器编写该程序（程序源代码见程序清单 1-1）。

程序清单 1-1

```
//HelloWorld.java  
class HelloWorld
```

```

{
    public static void main(String args[])
    {
        System.out.println("Hello World");
    }
}

```

编写好后，将文件保存成以 HelloWorld.java 为文件名的文件。使用 DOS 命令进入 HelloWorld.java 文件所存放的目录（如：D:\Example）下，使用 javac 命令编译该应用程序，通过编译该文件夹下会生成一个名为 HelloWorld.class 的 class 文件，该文件包含程序的字节码，java 字节码中包含的是 Java 解释程序将要执行的指令码，使用 java 命令执行该 class 文件，可在控制台下看到执行结果（如图 4 所示）。

命令格式如下所示：

D:\Example>javac HelloWorld.java

D:\Example>java HelloWorld

该程序会在控制台下输出如下内容：

Hello World

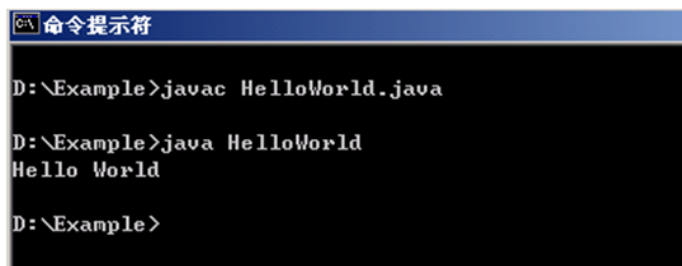


图 4 程序 HelloWorld.java 在控制台下的输出结果

编辑程序时，文件名称和程序段的公共类名称必须保持一致；编译程序时，使用 javac 命令编译相应 Java 程序要输入文件全名（包括文件名后缀“.java”），如：javac HelloWorld.java，但在使用 java 命令运行生成的字节码文件时，如：java HelloWorld，不要输入文件名的后缀。

（ii）直接使用 JDK 命令编译、运行程序

将编写好的 HelloWorld.java 程序放置到安装目录下的 bin 文件夹中。此处以安装路径为 C:\Program Files\Java\jdk1.8.0 为例，则将 HelloWorld.java 程序放入 C:\ProgramFiles\Java\jdk1.8.0\bin 中（如安装时指定其它安装路径，则做相应修改）。然后选择“开始”|“程序”|“附件”|“命令提示符”，打开 DOS 命令环境，进入 C:\Program Files\Java\jdk1.8.0\bin 目录下，分别使用 javac 和 java 命令编译和运行 HelloWorld.java 程序。

用上述方式直接编译和执行 Java 程序不需要事先设置系统的环境变量，可以在安装好 JDK 后直接使用。

4、安装并使用集成开发环境 Eclipse 编译运行 Java 程序

根据操作系统版本，下载对应 Eclipse 版本安装，使用方法及具体步骤详见课本 23-25 页 1.12 小节“使用 Eclipse 开发 Java 程序”，在此不再赘述。

四、实验步骤

- 1、先安装好JDK和Eclipse;
- 2、配置环境变量;
- 3、选择使用记事本或集成环境书写java程序;
- 4、编译运行程序，查看结果。
- 5、学习调试Java应用程序的方法。

五、实验内容

使用记事本或 Eclipse 集成环境来建立以下 Java 程序，编译并运行，然后记录运行结果。建议在硬盘上建立一个文件夹（如 E:\myprogram），将 Java 程序存储在其中，实验完毕后，再将源程序复制到 u 盘。**注意：在输入以下程序时，可故意漏打或打错几个字符，然后进行编译，仔细分析编译信息窗口，识别所显示的错误，逐个修改，直到不出现错误。**

习题一：使用命令行模式开发一个简单的 Java 应用程序。

```
//HelloWorld.java
import java.util.Date;
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello
World!");
        System.out.println("现在是： " + new Date().toString());
    }
}
```

- (1) 编辑、编译并运行该程序，得到实验结果。
- (2) 请添加适当的 Javadoc 注释，并练习相关的 javac、java、javadoc 命令中的可选参数。
- (3) 如要在运行时将该程序的输出结果重定向输出到当前目录下的“output.txt”文件中，应该如何敲入命令？

习题二：使用集成开发环境开发一个 Java 应用程序,输入您姓名的全拼，如：

“Zhang San”，程序输出“Hello, Zhang San! ”。

```
//HelloTest.java
import java.util.Scanner;
```

```

public class HelloTest {
    public static void main(String[] args) {
        System.out.println( "Please input your name:" );
        Scanner input = new Scanner (System.in);
        String yourName = input.next();
                                System.out.println("Hello,"
+yourName +" !" );
    }
}

```

(1) 编辑、编译并运行该程序，得到实验结果。

(2) 程序能得到预期结果吗？如不能，应如何修改源程序？尝试使用集成开发环境中的调试方法。

习题三：学习联合编译方法。

ClassRoom. java

```

public class ClassRoom {
    public static void main (String args[ ]) {
        【代码 1】//命令行窗口输出“教学活动从教室开始”
        Teacher zhang = new Teacher();
        Student jiang = new Student();
        zhang.introduceSelf();
        jiang.introduceSelf();
    }
}

```

Teacher. java

```

public class Teacher {
    void introduceSelf() {
        【代码 2】    //命令行窗口输出“我是张老师”
    }
}

```

Student. java

```

public class Student {
    void introduceSelf() {
        【代码 3】//命令行窗口输出“我是学生,名字是:***” ***请用您的名称代替
    }
}

```

(1) 请按模板中注释要求，将 3 个 java 源文件中的【代码】替换为 Java 程序代码，保存修改后，将 3 个 java 源文件存入相同的目录中，比如：“E:\myProgram\code3”；

(2) 使用 dos 命令仅编译 ClassRoom. java，观察目录中字节码文件的变化。

(3) 运行该程序，获得实验结果。

(4) 修改该程序中 Student. java 中的【代码 3】，仅单独编译 Student. java，不再编译其他文件（如 ClassRoom. java），运行该程序，获得实验结果。

- (5) 如果需要编译 code3 目录下的所有 Java 源文件，可以敲入什么 dos 命令？
(6) 尝试使用集成开发环境建立含有多个源程序的工程，获得该题运行结果。

习题四：编写源程序，读入圆柱体的半径和高，计算该圆柱体的底面积和体积。（运行示例参见课本 61 页 2.2）

习题五、（选做）尝试基于 JavaFX 图形窗口界面开发 Java 应用程序，在弹出的信息窗口输出信息 “Hello World! ”

六、主要仪器设备及耗材

仪器设备：PC 机一台 操作系统： windows10 编译软件及版本：Eclipse 2020 或 IntelliJIDEA

七、实验数据及处理结果 写出各个习题的算法（用流程图表示）、程序、结果并能对结果的正确性及产生的问题进行适当分析。

习题一：

[源程序]

```
// 输出 helloworld 和时间
import java.util.Date;
public class HelloJava {
    public static void main(String[] args) {
        System.out.println("Hello World!");
        System.out.println("现在是: " + new Date().toString());
    }
}
```

[设计思路或算法]

省略

[结果及截图]

```
Hello World!
现在是: Wed Mar 31 09:47:13 CST 2021
```

[分析]

在运行时使用 `java HelloWorld > output.txt` 即可将输出结果重定向到 txt 文件

习题二：

[源程序]

```
// 输出姓名全拼
import java.util.Scanner;
```

```

public class HelloJava {
    public static void main(String[] args) {
        System.out.println("Please input your name:");
        Scanner input = new Scanner (System.in);
        String yourName = input.next();
        System.out.println("Hello," + yourName + "!");
        input.close();
    }
}

```

[设计思路或算法]

省略

[结果及截图]

```

sairan@wangchuandemacbook-Air: javalearn % java HelloJava
Please input your name:
Huangchenruo
Hello,Huangchenruo!

```

[分析]

示例中的代码无法得到预期的结果，首先，代码中包含有中文字符。

同时，会给出警告 Resource leak: 'input' is never closed

解决方案是在 main 函数的末尾添加一个 `input.close()`;

问题解决，正常运行

习题三：

[源程序]

```

//ClassRoom.java
package exp_code3;

public class ClassRoom {
    public static void main (String args[ ]) {
        System.out.println("教学活动从教室开始");
        Teacher zhang = new Teacher();
        Student huang = new Student();
        zhang.introduceSelf();
        huang.introduceSelf();
    }
}

//Student.java
package exp_code3;

```

```

public class Student {
    void introduceSelf() {
        System.out.println("我是学生，名字是：黄晨箬");
    }
}

//Teacher.java
package exp_code3;

public class Teacher {
    void introduceSelf() {
        System.out.println("我是张老师");
    }
}

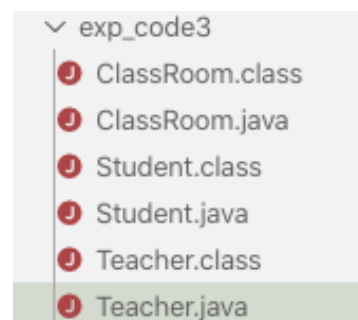
```

[设计思路或算法]

省略

[结果及截图]

创建处在同一目录下的三个文件



运行

```

sairen@wangchuandeMacBook-Air javalearn % java exp_code3.ClassRoom
教学活动从教室开始
我是张老师
我是学生，名字是：黄晨箬

```

当单独编译 Student. java 时：

```

sairen@wangchuandeMacBook-Air javalearn % java Student
我是学生，名字是：黄晨箬

```

[分析]

编译目录下的所有源文件，可以使用 `javac *.java` 指令

习题四：

[源程序]

```

import java.util.Scanner;

```



```

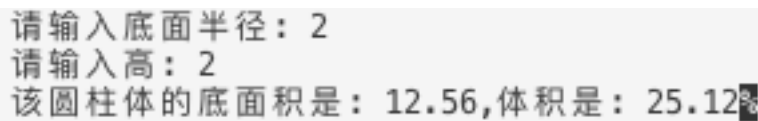
public class Cylinder {
    public static void main(String[] args) {
        float r,h,s,v;
        final float PI = 3.14f;
        Scanner reader = new Scanner(System.in);
        System.out.print("请输入底面半径: ");
        r = reader.nextFloat();
        System.out.print("请输入高: ");
        h = reader.nextFloat();
        s = PI * r * r;
        v = PI * r * r * h;
        System.out.print("该圆柱体的底面积是: " + s + ",体积是: " + v);
        reader.close();
    }
}

```

[设计思路或算法]

使用 scanner 读取所需要的底面半径和高，然后经过计算之后 print 所需结果

[结果及截图]



```

请输入底面半径: 2
请输入高: 2
该圆柱体的底面积是: 12.56,体积是: 25.12

```

[分析]

读取+计算+输出，无阻碍问题。

八、思考题

1、熟悉一个用于java程序开发的工具 Eclipse或IDEA。如何使用IntelliJ IDEA来开发JAVA程序？与Eclipse有何不同？

打开IDEA，新建一个工程项目，选择JDK以及需要添加的插件，然后创建一个空项目，编辑项目名称，创建包和源文件然后编写代码，然后运行即可。

比起Eclipse而言，IDEA具有强大的快捷键，以及自动补全等自动提示，给编写带来了很多便利。De

2、JDK和JRE的区别？

JRE主要包含java类库的class文件和虚拟机，而JDK主要包含java类库的class文件且自带一个JRE。

九、实验总结及体会

通过这次实验，我学习熟悉了 java 的开发环境，能够使用编译软件编写并运行一些简单的 java 程序。实际操作了联合编译的情况，对其有了更深的了解。