学号 X41714008 专业 计算机科学与技术 姓名 李逢时

实验日期 **2019-2-18**  教师签字 成绩

实验报告

【实验名称】 第一章 初识JAVA

【实验目的】

1. 了解Java的起源、基本概念及相关术语
2. 了解Java的特性以及和c语言的区别
3. 熟悉Java 开发环境的构建
4. 熟悉Java 开发工具
5. 掌握Java 程序设计基本方法

【实验原理】

1. Java程序设计基本原理
2. 面向对象程序设计基本原理

【实验内容】

**一、填空题**

1. java源文件的扩展名是 .java ，java字节码文件的扩展名是 .class 。

2. java应用程序总是从 main 方法开始执行，对于Hello.java文件，进行编译和解释的完整命令分别是： javac Hello.java 和 java Hello 。

3. Java语言中，面向对象的三个特性是 继承性 、 多态性 和 封装性 。

**二、简答题**

1. 简述JAVA语言的平台无关性。

[解答] java语言经过编译后，生成可在不同平台下，不经修改直接运行的.class字节码文件。

2. 简述JDK、JRE、JVM的概念和关系。

[解答]

* **JDK：java development kit，开发者使用的java软件开发工具包**
* **JRE：java runtime environment，运行java程序所必需的环境集合。**
* **jvm：java virtual machine，用于为java字节码提供跨平台性的支持。**

**三、实验题**

**1. 实验一：使用命令行调试以下程序 (注:每一个步骤需要抓图)**

public class HelloWorldApp{

public static void main(String args[]){

System.out.println("Hello World!");

}

}

1、使用文本编辑器编写Java源程序，并命名为HelloWorldApp.java，可放在某盘根目录下。

2、点击“开始”－>“运行”，cmd，进入命令行状态，进入Java源程序所在目录

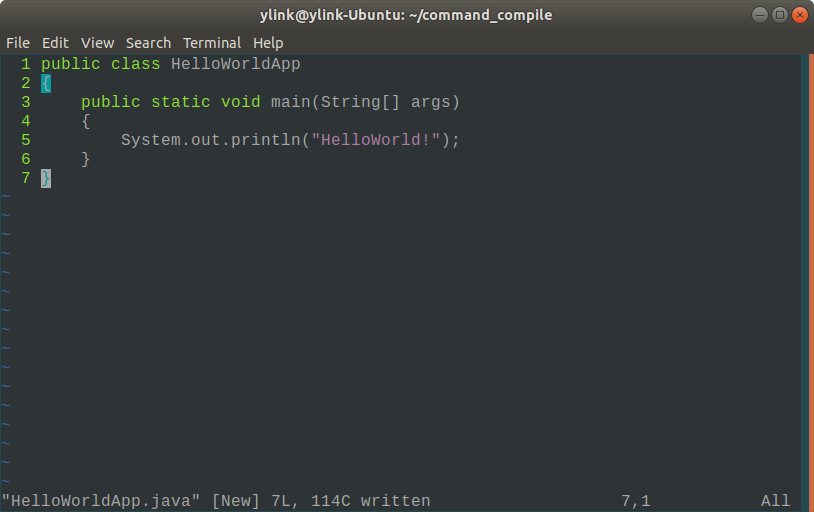
3、编译源程序javac HelloWorldApp.java

产生字节码文件(HelloWorldApp.class文件)

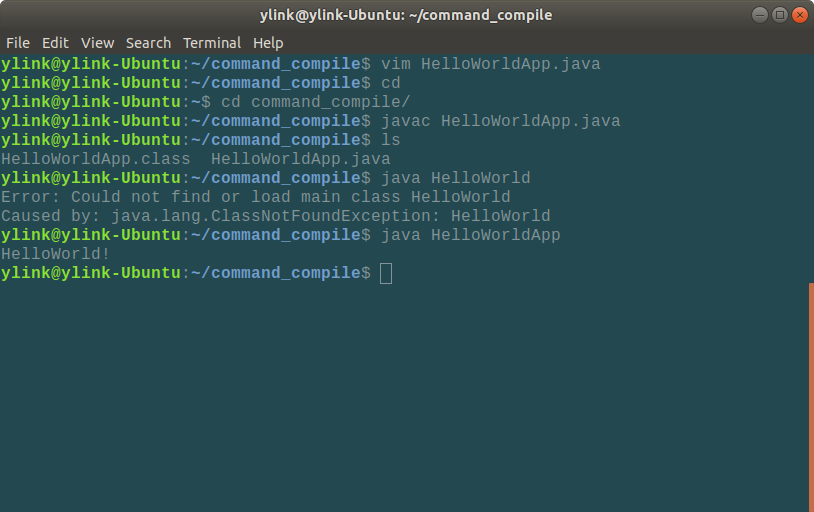
4、运行程序java HelloWorldApp

由java解释器解释执行

第一步：



第2、3、4步（图中的命令包括了这三步的操作）：



源代码：

public class Hhlloworldapp

{

public static void main(String[] args)

{

System.out.println("HelloWorld!");

}

}

Java程序的基本组成部分是类(class)，如本例中的HelloWorldApp类。最多只能有一个public类，且文件名需要此类相同。

1、修改类名：为HelloWorld，再编译，是否通过，为什么？

答：不通过，因为java语法规定，含有public的类的.java文件名需要与该public类的类名完全相同。

2、删除class前的public，编译，并看生成的字节码文件，是否通过？

答：可编译、解释通过。

3、添加public，修改类名为helloworldapp（注意是小写），是否通过，为什么？

答：不可编译通过。因java对类名、文件名的识别区别大小写。

**2. 实验二：用集成开发环境Jcreator或eclipse完成： (注:需要抓图)**

编写Java Application，类名为First，打印1~10。

public class First{

public static void main(String args[]){

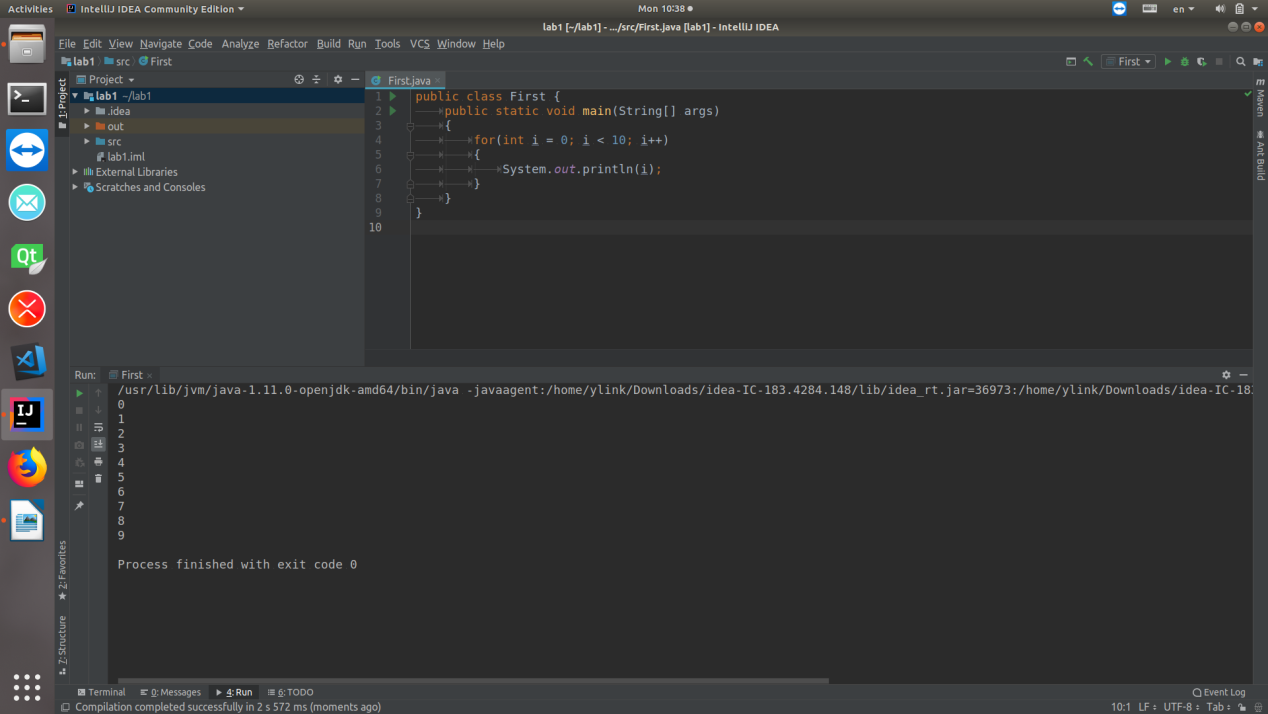
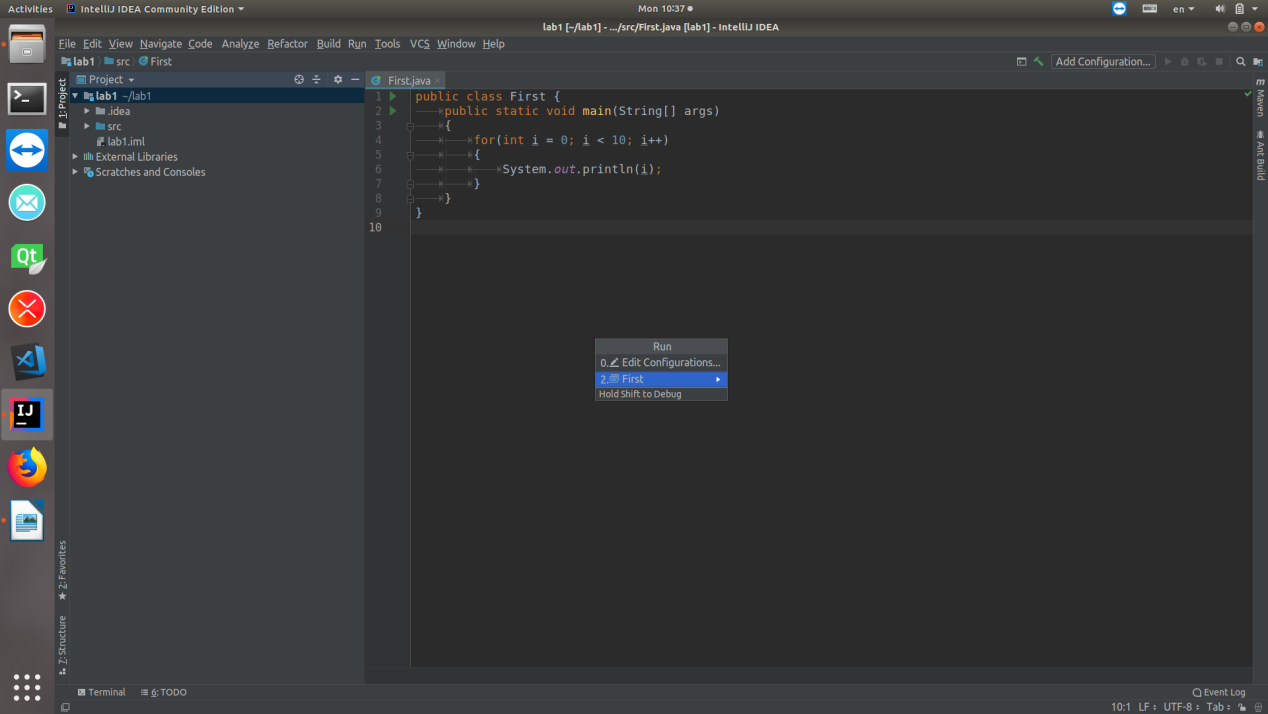
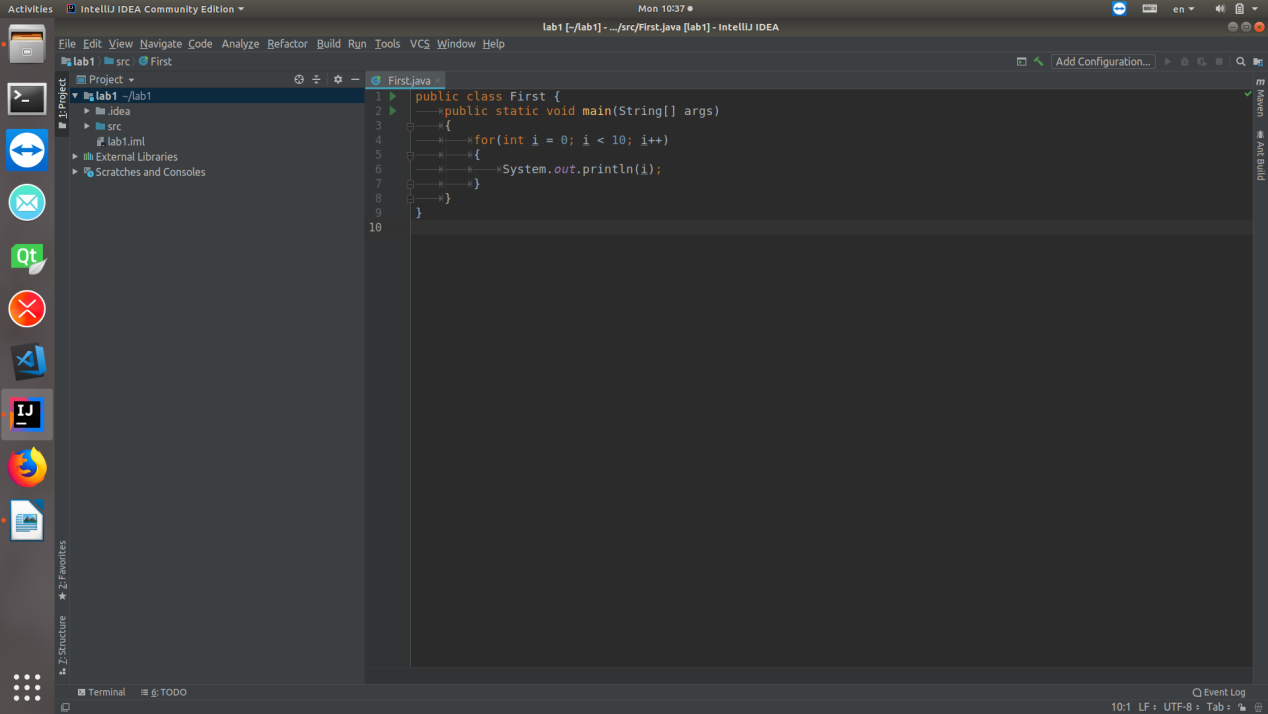
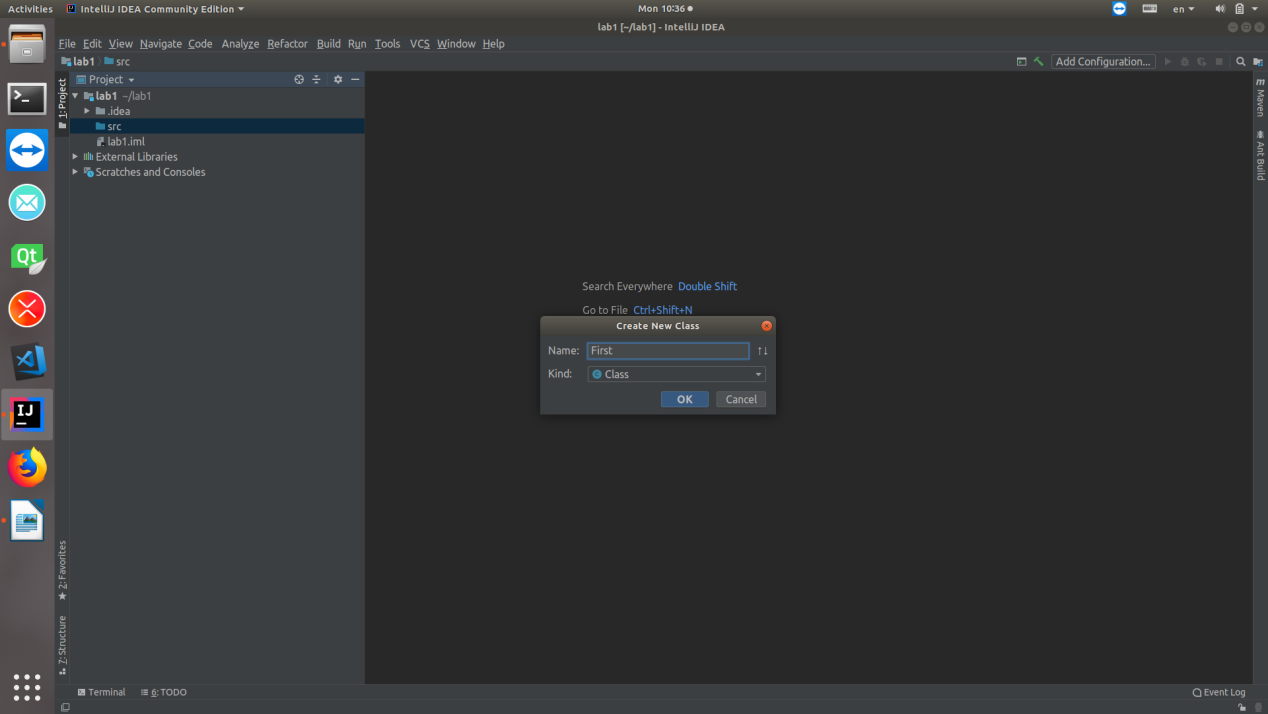
for(int i=1;i<=10;i++)

System.out.println(i);

}

}

下面的截图中，IDE使用了intellij IDEA。

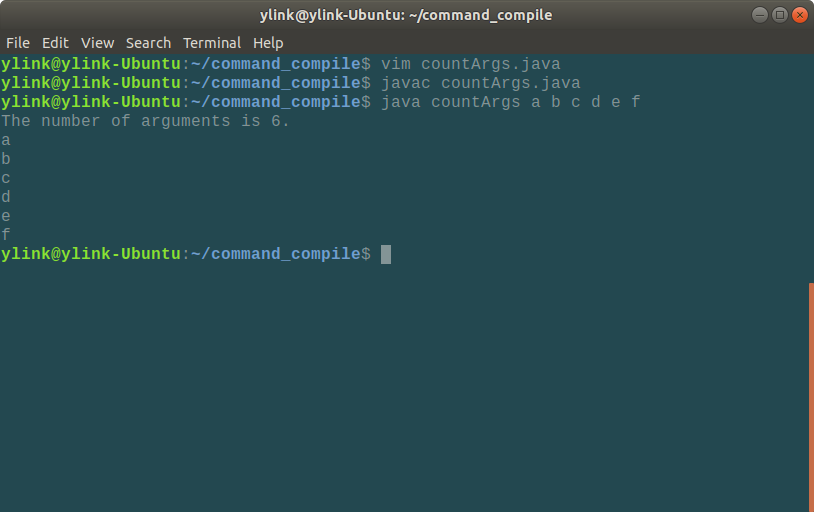
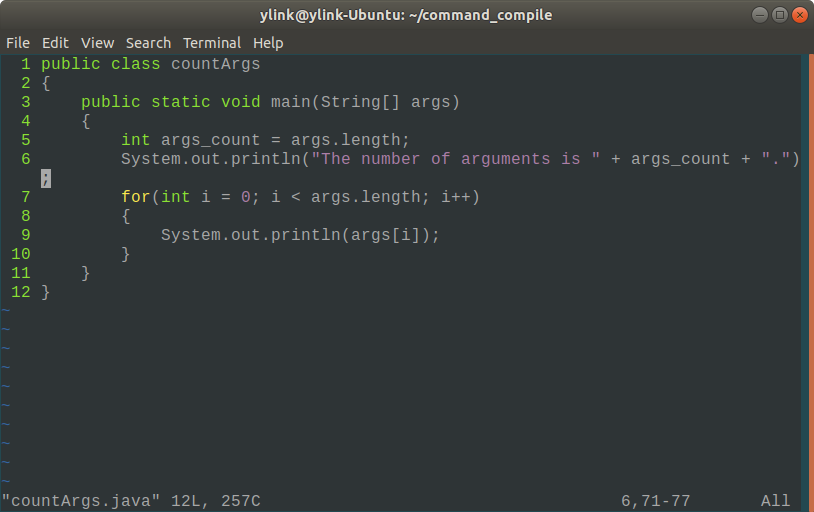


**3. 实验三：使用命令行调试以下程序 (注:每一个步骤需要抓图)**

public static void main(String args[])

args[]是main方法的参数，是一个字符串类型的数组，设计程序，统计main方法中输入参数的个数，并打印出每个参数的内容。

第一张截图包含实现功能的源代码。第二张截图包含了编译、解释命令及运行结果。



源代码：

public class countArgs

{

public static void main(String[] args)

{

int args\_count = args.length;

System.out.println("The number of arguments is " + args\_count + ".");

for(int i = 0; i < args.length; i++)

{

System.out.println(args[i]);

}

}

}

【小结或讨论】

通过本次试验，掌握了java运行及开发环境的配置步骤、java源代码在IDE、命令行的使用方式，尝试自行编写了简单的java命令行程序。对java使用打下足够基础。