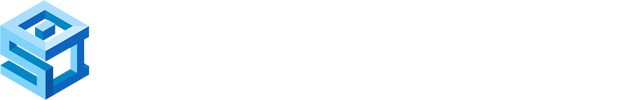
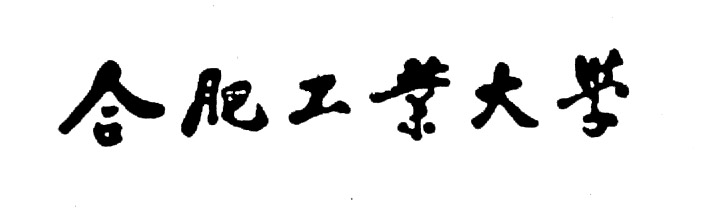
****



**《Java技术》实验报告**

|  |  |
| --- | --- |
| **学院名称：** | **计算机与信息学院（宣城校区）** |
| **专业班级：** | **物联网工程(2)班** |
| **姓 名：** | **党存远** |
| **学　 号：** | **2022217587** |
| **电子邮箱：** | **1207411532@qq.com** |
| **提交时间：** | **2023年 4 月 21 日** |

**实验报告格式说明**

1. 实验报告内容应包括：实验原理与方法、实验流程、实验操作过程及记录、实验结果、指导教师验收内容、心得体会等。
2. 实验报告应主要描述实际设计和完成的工作，篇幅不得超过40页，不得大量粘贴代码。
3. 实验源程序单独提交。
4. 实验报告作为一个word文档，每个实验单独一个表格。
5. 上交电子版本。
6. 标题：三号加粗黑体
7. 表中内容：
   1. 标题：5号黑体
   2. 正文：5号宋体，单倍行间距
   3. 纸张：A4(20cm×26.5cm)
8. 页边距：
   1. 上距：2.8cm
   2. 下距：2.2cm
   3. 左、右边距：2.8cm

**《Java技术》实验报告**

**实验一： 2023年3 月 8 日**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **学院** | 计算机与信息学院 | | **专业班级** | 物联网工程（2）班 | **姓名** | **党存远** | **成绩** |  |
| **课程**  **名称** | Java技术 | **实验项目**  **名 称** | | 实验一 Java开发环境的安装与配置 | | | **指导 教师** | 张延孔 |
| **一、实验目的**   * 1. 掌握JDK的安装步骤（有条件的话可以在Linux系统下安装）。   2. 理解环境变量PATH, CLASSPATH的作用，以及它们的设置方法(Win98和Win2000/XP、Win7、Win8下的设置方法有所区别)。   3. 熟悉记事本（或JCreator）编辑环境，编写简单的Application程序，并编译和解释执行。   4. 记录编译和执行Java程序当中的系统错误信息提示，并给出解决办法。   5. 复习、理解OOP的三大特征——封装、继承和多态。   6. 学会使用Java编写简单的类。理解类的属性和方法。   7. 掌握关键字private, protected, public的作用，掌握静态类，内部类的概念。   8. 掌握多态的概念。   **二、实验原理**  1．例如，掌握安装Jdk1.8的过程。  2．掌握eclipse的使用方法  3．学会创建一个项目并且让这个项目运行起来  **三、使用硬件、软件环境**  例如，PC 计算机一台，配置为CPU为1.5G，内存为512M，硬盘为40G，安装WIN10操作系统。另外，使用JCreator，JDK8等软件  使用Eclipse软件进行程序的编写与运行。  **四、实验过程、步骤及原始记录(算法、原程序、测试结果，分析等)**  **1.** 下载最新版本的JDK，并安装。  通过百度搜索JDK下载JDK安装包实现此步骤。  **2.** 设置环境变量PATH, CLASSPATH, 使得Java程序能正确编译和执行。  双击打开我的电脑，之后右键点击选择我的属性，点击高级系统设置，选择高级，点击环境变量，实现个各种环境变量的配置，当打开cmd输入java出现以下图标时证明java程序可以正常运行。  **图一**  **3.**在记事本（或JCreator）环境下编写一个HelloWorld.java程序，在DOS提示符下编译并执行这个程序。  **源代码：**  **public class HelloWorld**  **{**  **public static void main(String args[])**  **{**  **System.out.print("HelloWorld");**  **}**  **}**    **4.** 编写一个类A，它位于包a中，保存到A.java中，再编写一个类B，它位于包b中，保存到B.java中。在类B中实例化类A的一个对象。分别编译类A和类B。  **源代码：**  **package** a;  **public** **class** A {  **public** **static** **void** main(String[] args) {  // **TODO** 自动生成的方法存根    }  }  **package** b;  **import** a.\*;  **public** **class** B **extends** A{  **public** **static** **void** main(String[] args) {  // **TODO** 自动生成的方法存根  A a=**new** A();  }  }  **如图所示**      **5.** 分别编写两个类Point2D，Point3D来表示二维空间和三维空间的点，使之满足下列要求：  1) Point2D有两个整型成员变量x, y (分别为二维空间的X,Y方向坐标)，Point2D的构造方法要实现对其成员变量x, y的初始化。  2) Point2D有一个void型成员方法offset(int a, int b)，它可以实现Point2D的平移。  3) Point3D是Point2D的直接子类，它有有三个整型成员变量x,y,z (分别为三维空间的X,Y,Z方向坐标)，Point3D有两个构造方法：Point3D(int x, int y, int z)和Point3D(Point2D p, int z)，两者均可实现对Point3D的成员变量x, y, z的初始化。  4) Point3D有一个void型成员方法offset(int a, int b, int c)，该方法可以实现Point3D的平移。  5) 在Point3D中的主函数main()中实例化两个Point2D的对象p2d1，p2d2，打印出它们之间的距离，再实例化两个Point3D的对象p3d1，p3d2，打印出他们之间的距离。  **源代码：**  **package** firstDemo1;  **public** **class** Demo {      **public** **static** **class** Point2D  {  **private** **int** x,y;  **public** Point2D()//默认构造实现对x,y的初始化  {  x=0;  y=0;  }  **public** Point2D(**int** m\_x,**int** m\_y)//Point2D的有参构造  {  **this**.x=m\_x;  **this**.y=m\_y;    }  **public** **void** offset(**int** a,**int** b)  {  **this**.x=**this**.x+a;  **this**.y=**this**.y+b;  }  }  **public** **static** **class** Point3D **extends** Point2D  {  **private** **int** x,y,z;  **public** Point3D(**int** M\_x,**int** M\_y,**int** M\_z)  {  **super**(M\_x,M\_y);  **this**.x=M\_x;  **this**.y=M\_y;  **this**.z=M\_z;  }  **public** Point3D(Point2D p,**int** M\_z)  {  **super**();  **this**.x=p.x;  **this**.y=p.y;  **this**.z=M\_z;  }  **public** **void** offset(**int** a,**int** b,**int** c)  {  **this**.x+=a;  **this**.y+=b;  **this**.z+=c;  }  }    **public** **static** **void** main(String[] args) {  // **TODO** 自动生成的方法存根  Point2D p2d1=**new** Point2D(5,4);  Point2D p2d2=**new** Point2D(10,9);    Point3D p3d1=**new** Point3D(0,5,8);  Point3D p3d2=**new** Point3D(p2d2,4);      **double** d1=java.lang.Math.*sqrt*((p2d1.x-p2d2.x)\*(p2d1.x-p2d2.x)+(p2d1.y-p2d2.y)\*(p2d1.y-p2d2.y));  **double** d2=java.lang.Math.*sqrt*((p3d1.x-p3d2.x)\*(p3d1.x-p3d2.x)+(p3d1.y-p3d2.y)\*(p3d1.y-p3d2.y)+(p3d1.z-p3d2.z)\*(p3d1.z-p3d2.z));  System.***out***.println("Point2D中两点间的距离为： "+d1);  System.***out***.println();  System.***out***.println("Point3D中两点间的距离为："+d2);  }    }  **运行结果如下：**  **思考题：**  **1.** 什么是JVM，它的作用是什么？  答： （1）jvm是一种用于计算设备的规范，它是一个虚构出来的机器，是通过在实际的计算机上仿真模拟各种功能实现的。 （2）jvm包含一字节码指令集，一组寄存器，一个栈，一个垃圾回收堆和一个存储方法域。 （3）JVM屏蔽了与具体操作系统平台相关的信息，使Java程序只需生成在Java虚拟机上运行的目标代码（字节码）,就可以在多种平台上不加修改地运行。JVM在执行字节码时，实际上最终还是把字节码解释成具体平台上的机器指令执行。  **2.** 如何理解PATH和CLASSPATH,，如何设置它们。  **答：**  1.JAVA\_HOME指向的是JDK的安装路径，如D:\JDK\_1.4.2，在这路径下应该能够找到bin、lib等目录。 2.classpath环境变量告诉类装载器到哪里去寻找第三方提供的类和用户定义的类。JVM和其他JDK工具通过依次搜索平台库，扩展库,和类路径来查找类。  3.path环境变量指定了JDK命令搜索路径,设置path的作用是让操作系统可以找到JDK命令。path环境变量原来Windows里面就有，只需修改一下，使他指向JDK的bin目录，这样在控制台下面编译、执行程序时就不需要再键入一大串路径了。设置方法是保留原来的path的内容，并在其后加上%JAVA\_HOME%\bin  **3.** Java中子类对象的创建过程的详细步骤是什么？  **答：**   1. 首先写出这个子类，并且继承父类对象 2. 之后在子类中写出子类自己的构造函数及自己本身所包含的对象 3. 若需要调用父类中的对象需要加上super关键字来使用父类中和子类同名的函数   **4.**什么是面向对象的多态性，在程序中如何体现的？  **答：**  面对对象的多态性就是一个父类下的不同子类在同一个抽象函数下所体现的方法不同。  多态性是一种允许使用一个界面来访问一类动作的特性，特定的动作可以由不同的具体情况而定（传入不同的参数）。多态性是发送消息给某个对象，让该对象自行决定响应何种行为。通过将子类对象引用赋值给超类对象引用变量来实现动态方法调用。  在使用多肽时通过写抽象类来对子类进行实现或者采用接口的方法来实现多态。  **五、实验结论、分析、思考题与心得体会**  例如，通过本次实验，我有以下几方面的收获、认识和体会：  1．学会了在三种环境下编写程序及运行程序的方法；  2．加深了Java中相关语法的认识：  3.加深了我对于多态的运用，同时也加深了我对如何用java来编写一个简单的程序的使用。  4.通过这次实验，我学会了如何在不同的包下来书写函数，以及如何在另一个包中实现函数（加入import关键字来包含另一个包中的所有函数）。  5.学会了如何自己配置环境变量来让java程序运行起来。  6.学会了如何利用cmd来运行一个java程序，通过javac HelloWorld.java进行程序的运行  7.通过这次实验掌握了书写一个简单的类，并且进行封装等简单操作使代码的封装性完好。 | | | | | | | | |

**《Java技术》实验报告**

**实验二： 货物进销管理系统 2023年 3月 23 日**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **学院** | 计算机与信息学院 | | **专业班级** | 物联网工程2班 | **姓名** | **党存远** | **成绩** |  |
| **课程**  **名称** | Java技术 | **实验项目**  **名 称** | | 实验二 货物进销管理系统 | | | **指导 教师** | 张延孔 |
| **一、实验目的**  1、掌握抽象类的使用等。  **二、实验原理**  1．利用Java中文件IO流完成数据的读写操作  2．掌握Java中提供的实用类(Vector,ArrayList)来存放数据  3．使用字符串类对象来完成数据的读入操作(String，StringBuffer)  4．通过工厂模式来进行Java编程设计操作  5．构建不同类采用面向对象的方式实现  **三、使用硬件、软件环境**  例如，PC 计算机一台，配置为CPU为1.5G，内存为512M，硬盘为40G，安装WIN10操作系统。另外，使用JCreator，JDK8等软件  使用Eclipse软件进行程序的编写与运行。  **四、实验过程、步骤及原始记录(算法、原程序、测试结果，分析等)**  **package Inventory;**  **import java.io.\*;**  **import java.util.Vector;**  **//仓库信息**  **class Inventory{**  **String Itemnumber = null;**  **int Quantity =0;**  **String Supplier = null;**  **String Description = null;**  **Inventory(){**  **}**  **}**  **//发货信息**  **class Oitem{**  **String Itemnumber = null;**  **int Quantity = 0;**  **String Supplier = null;**  **Oitem(){**  **}**  **}**  **//到货信息**  **class Ritem{**  **String Itemnumber = null;**  **int Quantity = 0;**  **Ritem(){**  **}**  **}**  **//货物添加信息**  **class Aitem{**  **String Itemnumber = null;**  **String Supplier = null;**  **String Description = null;**  **Aitem(){**  **}**  **}**  **//货物删除信息**  **class Ditem{**  **String Itemnumber = null;**  **Ditem(){**  **}**  **}**  **//主类**  **public class Item {**  **public static void main (String[] args) {**  **File f = null;**  **BufferedReader in=null;**  **PrintWriter out=null;**  **int len=0;//inv的长度**  **Vector<Inventory> inv = new Vector<Inventory>();//仓库**  **Vector<Oitem> O=new Vector<Oitem>();//发货信息**  **Vector<Ritem> R=new Vector<Ritem>();//收货信息**  **Vector<Ditem> D=new Vector<Ditem>();//删除信息**  **Vector<Aitem> A=new Vector<Aitem>();//添加信息**  **//信息读取**  **try{**  **//读取货物信息**  **f=new File("D:\\Java\\.vscode\\Inventory\\Inventory\\Inventory.txt");**  **in = new BufferedReader(new FileReader(f));**  **String line=in.readLine();**  **while(line!=null){**  **String [] infor=line.split(" ");**  **Inventory a = new Inventory();**  **a.Itemnumber=infor[0];**  **a.Quantity=Integer.valueOf(infor[1]);**  **a.Supplier=infor[2];**  **a.Description=infor[3];**  **inv.addElement(a);**  **line=in.readLine();**  **}**  **in.close();**  **//读取操作信息**  **f=new File("D:\\Java\\.vscode\\Inventory\\Inventory\\Transactions.txt");**  **in=new BufferedReader(new FileReader(f));**  **line=in.readLine();**  **while(line!=null){**  **String [] infor=line.split(" ");**  **if(line.charAt(0)=='O'){**  **Oitem a=new Oitem();**  **a.Itemnumber=infor[1];**  **a.Quantity=Integer.valueOf(infor[2]);**  **a.Supplier=infor[3];**  **O.addElement(a);**  **}//发货信息**  **else if(line.charAt(0)=='R'){**  **Ritem a=new Ritem();**  **a.Itemnumber=infor[1];**  **a.Quantity=Integer.valueOf(infor[2]);**  **R.addElement(a);**  **}//到货信息**  **else if(line.charAt(0)=='A'){**  **Aitem a=new Aitem();**  **a.Itemnumber=infor[1];**  **a.Supplier=infor[2];**  **a.Description=infor[3];**  **A.addElement(a);**  **}//货物添加信息**  **else if(line.charAt(0)=='D'){**  **Ditem a=new Ditem();**  **a.Itemnumber=infor[1];**  **D.addElement(a);**  **}//货物删除信息**  **line=in.readLine();**  **}**  **in.close();**  **//货物操作进行操作**  **//输出环境的搭建**  **File fs=new File("D:\\Java\\.vscode\\Inventory\\Inventory\\Shipping.txt");**  **File fe=new File("D:\\Java\\.vscode\\Inventory\\Inventory\\Error.txt");**  **File fn=new File("D:\\Java\\.vscode\\Inventory\\Inventory\\NewInventory.txt");**  **//创建文件**  **if(!fs.exists()){**  **fs.createNewFile();**  **}**  **if(!fe.exists()){**  **fe.createNewFile();**  **}**  **if(!fn.exists()){**  **fn.createNewFile();**  **}**  **//清空文件内容**  **out=new PrintWriter(new FileWriter(fs));**  **out.write("");**  **out.close();**  **out=new PrintWriter(new FileWriter(fe));**  **out.write("");**  **out.close();**  **out=new PrintWriter(new FileWriter(fn));**  **out.write("");**  **out.close();**  **out=null;**  **//正常打开文件**  **PrintWriter outs =new PrintWriter(new FileWriter(fs,true));**  **PrintWriter oute =new PrintWriter(new FileWriter(fe,true));**  **PrintWriter outn =new PrintWriter(new FileWriter(fn,true));**  **//货物添加**  **int alen=A.size();**  **for(int i=0;i<alen;i++){**  **Aitem a=(Aitem)A.elementAt(i);**  **Inventory item=new Inventory();**  **item.Itemnumber=a.Itemnumber;**  **item.Quantity=0;**  **item.Supplier=a.Supplier;**  **item.Description=a.Description;**  **inv.addElement(item);**  **}**  **//到货操作**  **int rlen=R.size();**  **for(int i=0;i<rlen;i++){**  **Ritem r=(Ritem)R.elementAt(i);**  **len=inv.size();**  **for(int j=0;j<len;j++){**  **Inventory item=(Inventory)inv.elementAt(j);**  **if(r.Itemnumber.equals(item.Itemnumber)){**  **item.Quantity+=r.Quantity;**  **inv.set(j,item);**  **break;**  **}**  **}**  **}**  **//出货操作**  **//冒泡排序**  **len=O.size();**  **Oitem o1=null,o2=null,o=null;**  **for(int i=0;i<len;i++){**  **for(int j=0;j<len-1;j++){**  **o1=(Oitem)O.elementAt(j);**  **o2=(Oitem)O.elementAt(j+1);**  **if(o1.Quantity>o2.Quantity){**  **O.set(j,o2);**  **O.set(j+1,o1);**  **}//交换**  **}**  **}**  **//正常出货**  **int olen=O.size();**  **len=inv.size();**  **for(int i=0;i<olen;i++){**  **o=O.elementAt(i);**  **for(int j=0;j<len;j++){**  **Inventory item=(Inventory)inv.elementAt(j);**  **if(o.Itemnumber.equals(item.Itemnumber)){**  **if(o.Quantity<=item.Quantity){**  **item.Quantity-=o.Quantity;**  **inv.set(j,item);**  **line=o.Supplier+' '+o.Itemnumber+' '+Integer.toString(o.Quantity)+'\n';**  **outs.write(line);**  **break;**  **}//正常出货**  **else{**  **line=o.Supplier+' '+o.Itemnumber+' '+Integer.toString(o.Quantity)+'\n';**  **oute.write(line);**  **break;**  **}//出货异常**  **}**  **}**  **}**  **//删除操作**  **int dlen=D.size();**  **Ditem d=null;**  **for(int i=0;i<dlen;i++){**  **d=D.elementAt(i);**  **len=inv.size();**  **for(int j=0;j<len;j++){**  **Inventory item=inv.elementAt(j);**  **if(d.Itemnumber.equals(item.Itemnumber)){**  **if(item.Quantity==0){**  **inv.removeElementAt(j);**  **break;**  **}//正常删除**  **else{**  **line='0'+' '+item.Itemnumber+' '+Integer.toString(item.Quantity)+'\n';**  **oute.write(line);**  **break;**  **}//删除异常**  **}**  **}**  **}**  **//更新仓库**  **len=inv.size();**  **Inventory item=null;**  **for(int i=0;i<len;i++){**  **item=inv.elementAt(i);**  **line=item.Itemnumber+' '+Integer.toString(item.Quantity)+' '+item.Supplier+' '+item.Description+'\n';**  **outn.write(line);**  **}**  **//关闭资源**  **outs.close();**  **oute.close();**  **outn.close();**  **}catch(IOException e){**  **e.printStackTrace();**  **}**  **}**  **}**  **代码如图：**  **图一：Inventor****y.txt中的数据，每个数据之间利用Tab键隔开位置**  **图二：Transaction.txt中的数据**  **图三：根据题目要求****创建5个类来完成系统的构建**  **图四：主类的实现**  **1、在主类的实现中首先创建Vector类的对象分别存储自己定义的五个数据类型。**  **2、根据题目要求首先读取Inventory.txt中的文本信息，读完并关闭。**  **图五：主类的实现**   1. **将读完的数据分别进行判断，判断所读取道德内容是需要进行何种操作** 2. **将读取到的数据分类完成之后，调用不同的方法进行系统的构建。**   **图六：****主类的实现**   1. **根据系统所需创建三个文件对象完成对不同数据的输入操作。** 2. **首先判断文件是否存在，若不存在则重新创建文件** 3. **由于文件中可能原来有部分数据因此将文件中的内容进行清空操作** 4. **在将三个文件中的内容全部清空之后，重新打开文件，并且设置为向文件内部输入数据。**   **Ps:PrintWriter继承Writer类实现向文件内部输入数据**  **图七：主类的实现**   1. **根据要求实现货物添加操作** 2. **根据要求实现到货记录操作，统计数量使用循环操作。**   **图八：主类的实现**   1. **出货操作时，根据系统要求需要将货物的序号按照升序存储，因此采用冒泡排序的算法实现** 2. **在出货过程中要注意可能存在有错误情况，需要统计并且将错误情况记录在错误的文档中**   **图九：主类的实现**  **1、根据系统需求对不需要的货物进行删除操作**  **2、更新仓库中剩余的货物信息**  **3、关闭资源，防止内存泄漏。**    **图十：更新后的货物信息**    **图十一：发货信息**    **图十二：错误的数据信息**  **五、实验结论、分析、思考题与心得体会**  例如，通过本次实验，我有以下几方面的收获、认识和体会：  1．学会了在三种环境下编写程序及运行程序的方法；  2．加深了Java中相关语法的认识：  3．对于文件的输入输出有了更深的理解  4．明白了在编程过程中对于不同对象的不同处理方式，更深一步理解了面向对象编程的好处  5．在文件的输入与输出过程中通过Tab键可以更好的读取数据，更加直观，在文件中一般不使用空格键。  6．通过本次实验，更加熟悉了对于类的书写，对于不同的操作对象要通过不同的类来实现不同的操作。  7.在读取数据时使用到了BufferReader类，通过该类可以读取字符文件并且读取的速度较快，同时调用了readline方法每一次读取一行数据。  8.在读取需要进行何种操作的文件时，通过判断每行文件中第一个标识来对操作进行分类，之后根据不同的分类将数据存放到不同的所属空间。  9.当将不同的操作进行分类之后，创建三种不同文件实现不同的操作，使用了vector类中的set方法将编号为j位置的元素赋值为item。  10.在所有操作结束之后重新更新仓库中的数据并且关闭所有文件。 | | | | | | | | |

**《Java技术》实验报告**

**实验三： 基于GUI的网络通信程序设计 2023年4月 10 日**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **学院** | 计算机与信息学院 | | **专业班级** | 物联网工程(2) | **姓名** | **党存远** | **成绩** |  |
| **课程**  **名称** | Java技术 | **实验项目**  **名 称** | | 实验三 基于GUI的网络通信程序设计 | | | **指导 教师** | 张延孔 |
| 教师评语 | 教师签名：  年 月 日 | | | | | | | |
| **一、实验目的**  1.掌握Java中GUI程序的编写，包括事件监听机制。  2．掌握Java的网络通信编程，ServerSocket，Socket类的使用。  3．掌握Java中多线程的编程，Thread类，Runnable接口的使用。  4．掌握用面向对象的方法分析和解决复杂问题。  **二、实验原理**  1．利用多线程，GUI以及网络编程技术  **三、使用硬件、软件环境**  例如，PC 计算机一台，配置为CPU为1.5G，内存为512M，硬盘为40G，安装WIN10操作系统。另外，使用JCreator，JDK8等软件  **四、实验过程、步骤及原始记录(算法、原程序、测试结果，分析等)**  **客户端：**  **图一：在ClientFrame中继承JFrame设置三个JTextField对象，用来分别表示ip地址输入，端口地址输入，以及消息输入，创建文本显示区域。创建两个Button对象分别表示连接和发送。在ClientFrame的构造方法中设置Gui窗口的一些属性。**  **图二：设置客户端的组件以及窗口布局，设置箱式布局等进行客户端的设置。**    **图三：设置监听器，使得按下按钮后窗口会出现不同的反应**  **图四：创建一个读取数据的线程完成多线程任务**  **服务器端：**  **图五：与客户端相同，设置窗口的布局与属性**  **图六：创建客户端的窗口布局与客户端大致相同**  **图七：创建监听器，使得按下不同的按钮进行不同的操作**  **图八：****鼠标点击不同的按钮产生不同的事件**  **图九：实验的实现**  **五、实验结论、分析、思考题与心得体会**  例如，通过本次实验，我有以下几方面的收获、认识和体会：   1. 学会了GUI编程的具体实现，掌握了如何利用GUI来创建一个可视化窗口来实现不同操作 2. 在使用GUI进行窗口创建时，加深了对于JTextField类，JTextArea类，JButton类，以及自己创建的监听器类的使用。 3. 创建客户端窗口时，对于Box类也有了一些初步的了解，对于一个窗口的搭建需要由大到小，先放置大盒子，之后将需要的一点点的添加进入小盒子中，同时这些Box类都需要添加在JFrame类的底层容器中。 4. 在对每一个盒子中加入数据时，需要加入Label类创建不同的文本。 5. 在窗口创建中遇到了一些不常见的像一些滚动窗口，在编写这些时先通过API帮助文档查找方法，但是没有看懂参数以及返回类型，于是借助网洛了解了部分类的方法实现。 6. 在创建自己的监听器时，调用了例如InetAddress类来完成输入一个ip地址来创建连接，同时使用socket类来完成端口的绑定与端口与端口之间的连接。 7. 在传输数据时创建了输入流和输出流类用来传输数据，由于数据的传输需要双向进行，因此在客户端和服务端中都需要实例化输入流和输出流的对象。 8. 在发送数据时，遇到了一些问曾使用过的方法，通过网络csdn等了解了一些方法的作用。 9. 由于客户端与服务器端的数据传输与传入使两个同时继续进行的线程，因此使用的Thread来创建两个线程实现数据的双向传递。 10. 通过这一次实验加深了对多线程以及网络编程的实现   更深一步体会了Java的强大与简便之处 | | | | | | | | |

**《Java技术》实验报告**

**实验四： 2023年4月 日**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **学院** | 计算机与信息学院 | | **专业班级** |  | **姓名** |  | **成绩** |  |
| **课程**  **名称** | Java技术 | **实验项目**  **名 称** | | 实验四 Java Web编程（自学完成） | | | **指导 教师** |  |
| **一、实验目的**  1. XXXX。  **二、实验原理**  1．例如，掌握安装Jdk1.8的过程。  2．掌握….  **三、使用硬件、软件环境**  例如，PC 计算机一台，配置为CPU为1.5G，内存为512M，硬盘为40G，安装WIN10操作系统。另外，使用JCreator，JDK8等软件  **四、实验过程、步骤及原始记录(算法、原程序、测试结果，分析等)**  **五、实验结论、分析、思考题与心得体会**  例如，通过本次实验，我有以下几方面的收获、认识和体会：  1．学会了在三种环境下编写程序及运行程序的方法；  2．加深了Java中相关语法的认识： | | | | | | | | |

**（在学生的实验报告电子版的最后）**

**表. 实验成绩评定表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 评价内容 | | 权重 | 得分 |
| **验收** | 实验原理是否理解；程序能否运行；实验结果是否正确；任务是否全部完成。 | | 0.5 |  |
| **实验报告** | 1 | 报告格式是否规范，语言使用是否规范，行文是否流畅，是否图文并茂； | 0.2 |  |
| 2 | 实验原理、实验步骤描述是否正确、详实；  程序流程图是否规范，代码实现是否正确；  实验数据记录是否完整，实验结果是否正确；  实验结果的分析、对比是否充分； | 0.2 |  |
| 3 | 实验体会是否正确，是否提出了自己独到见解。 | 0.1 |  |
| 合计 |  | | | |
| 指导教师（签章）： 2023 年 05 月 日 | | | | |