

1、已知如下数据：

姓名	成绩
张三	90
李四	80
小明	85
王五	60
何六	70

编写程序实现：

（1）编制一个 Student 类描述上述数据

```
private String name;  
private int score;  
  
public String getName() {  
    return name;  
}  
  
public int getScore() {  
    return score;  
}  
  
public Student(String name, int score) {  
    this.name = name;  
    this.score = score;  
}  
  
public Student() {  
  
}
```

（2）使用 Vector 存放每条记录，  
主函数中使用

```
Vector<Student> students = new Vector<>();  
students.add(new Student("张三", 90));  
students.add(new Student("李四", 80));  
students.add(new Student("小明", 85));  
students.add(new Student("王五", 60));  
students.add(new Student("何六", 70));
```

（3）查找成绩为 60 的学生姓名，

```
public static Student sortSelect(Vector<Student> students,int compareScore){  
    Student stu = null;  
    for(int i=0;i<=students.size();i++){  
        stu = students.get(i);
```

```

        if(stu.getScore() == compareScore){
            break;
        }
    }
    return stu;
}
@Override
public String toString() {
    return "Student [name=" + name + ", score=" + score + "]";
}

```

主函数：

```

Student stu = sortSelect(students,60);
System.out.println(stu.toString());

```

(4) 对上述记录按照成绩降序排列（使用两种方法：（1）直接对成绩排序；（2）使用 Arrays, Collections 的 sort 方法）。

直接排序：

```

for(int i = 0; i < students.size(); i++)
    for(int j = i; j < students.size()-1; j++) {
        if(students.get(j).getScore() < students.get(j+1).getScore())
        {
            Collections.swap(students,j,j+1);
        }
    }

```

使用 Arrays, Collections 的 sort 方法

```

@Override
public int compareTo(Student arg0) {
    if(this.score < arg0.getScore()){
        return 1;
    } else if(this.score == arg0.getScore()){
        return 0;
    } else{
        return -1;
    }
}

```

主函数：

```

Collections.sort(students);

```

**2、Java 中的异常分成哪两大类？其分类标准是什么？Java 对于这两种异常的处理有什么区别？**

(1) 运行时异常：runtimeexception 的子类，间接子类，出现此种情况往往是程序的问题（可能是程序员编写问题）可以交给 JVM 处理

(2) 非运行时异常：语法上交给程序元处理，否则无法通过编译（checked 异常，必查异常）分类标准：是否必须被处理才能编译运行。处理：本地捕获、处理（try-catch）或向方法的上级传递、报告

- 3、编写两个函数，分别使用字节流、字符流实现文件的拷贝（在注释段中间编程，两个函数名根据主函数中调用的两个子函数来确定）

```
Public class FileCopy{
    public static void main(String[] args){
        copy1("c:\\src1.txt","dst1.txt");
        copy2("c:\\src2.txt","dst2.txt");
    }
    /*****Program*****/

    /****End*****/
}
package time1;

import java.io.*;

public class Filecopy{
    public static void main(String[] args){
        copy1("d:\\src1.txt","dst1.txt");
        copy2("d:\\src2.txt","dst2.txt");
    }

    //字节流 copy1
    public static void copy1(String sou,String aim) {
        FileInputStream in;
        try {
            in = new FileInputStream(sou);
            FileOutputStream out = new FileOutputStream(aim);
            byte[] buf = new byte[1024];
            int len = 0;
            while((len=in.read(buf)) != -1){
                out.write(buf,0,len);
            }
            out.flush();
            in.close();
            out.close();
        } catch (FileNotFoundException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

```

    }

}

public static void copy2(String sou,String aim) {
    BufferedReader bf;
    try {
        bf = new BufferedReader(new FileReader(sou));
        BufferedWriter bw = new BufferedWriter(new FileWriter(aim));
        String str;
        while((str=bf.readLine()) != null) {
            bw.write(str);
            bw.newLine();
        }
        bf.close();
        bw.close();
    } catch (FileNotFoundException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
}
}

```

#### 4、重写（覆盖）Thread 的 run()方法的时候，如果 run()方法中的语句有可能抛出非运行时异常，能否在 run()方法头部声明抛出(throws)? 为什么?

不能，根据重写的原则，Thread 没有设置抛出异常，重写时候不能抛出异常因为子类不能比父类抛出新的异常，如果可以抛出新的异常的话则不遵守原则。