《面向对象程序设计与训练》实验报告

信息学院 学院 计算机类 专业 2021 级

实验时间 2022 年 4 月 17 日

姓名 陈俊宏 学号 20211060245 分工

实验名称：接口与实现

实验成绩：

1. 实验目的

本实验的目的是让学生掌握类怎样实现接口，让学生掌握接口回调技术，让学生掌握面向接口编程思想。

二、实验仪器设备及软件

三、实验方案

1. 实验步骤

Test1:

import java.util.Scanner;

interface ComputerAverage {

void Average();

}

class Gymnastics implements ComputerAverage {

double score;

String name;

Gymnastics(String a) {

name = a;

get();

}

void get() {

System.out.println("姓名: " + name);

}

public void Average() {

Scanner reader = new Scanner(System.in);

int count;

double t, average, sum = 0, summ = 0;

System.out.print("统计跳体操次数：");

count = reader.nextInt();

double a[] = new double[count];

System.out.println("分别输出体操分数的五个值: ");

for (int i = 0; i < count; i++) {

a[i] = reader.nextDouble();

summ += a[i];

}

System.out.println("体操分数未去掉高分和低分之前和" + summ);

System.out.println("排序之后的数组:");

for (int k = 0; k < count; k++) {

for (int i = 1; i < count; i++) {

if (a[i - 1] > a[i]) {

t = a[i - 1];

a[i - 1] = a[i];

a[i] = t;

}

}

}

for (int l = 0; l < count; l++) {

System.out.print(+a[l] + " ");

}

for (int j = 1; j <= count - 2; j++) {

sum += a[j];

}

System.out.println();

System.out.println("体操分数未去掉高分和低分之后和" + sum);

if (count > 2) {

average = sum / (count - 2);

System.out.println("平均值: " + average);

} else {

System.out.println("该体操队员成绩无效！！！");

}

}

}

class School implements ComputerAverage {

double score;

String classname;

School(String a) {

classname = a;

get();

}

void get() {

System.out.println("姓名: " + classname);

}

public void Average() {

Scanner reader = new Scanner(System.in);

System.out.print("请输入要统计的科目名字:");

String name = reader.next();

System.out.print("请输入班上人数:");

int amount = reader.nextInt();

double[] a = new double[amount];

double summ = 0, average;

System.out.println("输入每位同学的成绩:");

for (int i = 0; i < amount; i++) {

a[i] = reader.nextDouble();

summ += a[i];

}

System.out.println(name + "科目" + "平均分:" + (summ / amount));

}

}

public class test1 {

public static void main(String args[]) {

Scanner reader = new Scanner(System.in);

System.out.println("1、体操 2、科目平均分");

System.out.print("请输入要查询的数字：");

int number = reader.nextInt();

switch (number) {

case 1: {

ComputerAverage C = new Gymnastics("小黄");

C.Average();

System.out.print("继续查询输入1，结束输入0");

int boo = reader.nextInt();

if (boo == 1)

main(args);

break;

}

case 2: {

School S = new School("计科一班");

S.Average();

System.out.println("继续查询输入1，结束输入0");

int boo = reader.nextInt();

if (boo == 1) {

main(args);

}

break;

}

default: {

System.out.println("数字不存在，请重新输入!!!!!!");

main(args);

}

}

}

}

=====================================================================

Test2:

**CalculateWeight.java**

package test2;

public interface CalculateWeight

{

int calweight();

}

**TV.java**

public class TV implements CalculateWeight

{

int count=0;

public int calweight()

{

count++;

return 10;

}

}

**PC.java**

public class PC implements CalculateWeight

{

int count=0;

public int calweight()

{

count++;

return 20;

}

}

**WashingMachine.java**

public class WashingMachine implements CalculateWeight

{

int count=0;

public int calweight()

{

count++;

return 30;

}

}

**Truck.java**

public class Truck {

public static void main(String[] args)

{

int N=10;

int weight [];

int sum=0;

TV tv=new TV();

PC pc=new PC();

WashingMachine wm=new WashingMachine();

for(int i=0;i<10;i++)

{

if((int)(Math.random()\*3+1)==1)

sum=sum+tv.calweight();

else if((int)(Math.random()\*3+1)==2)

sum=sum+pc.calweight();

else if((int)(Math.random()\*3+1)==3)

sum=sum+wm.calweight();

}

System.out.println("共"+tv.count+"台电视，"+pc.count+"台电脑，"+wm.count+"台洗衣机。");

System.out.println("总重量为："+sum);

}

}

====================================================================

Test3:

interface DogState {

public void showState();

}

class SoftlyState implements DogState {

public void showState() {

System.out.println("听主人的命令！");

}

}

class MeetEnemyState implements DogState {

public void showState() {

System.out.println("上去咬一口！");

}

}

class MeetFriendState implements DogState {

public void showState() {

System.out.println("晃动尾巴，以示友好！");

}

}

class MeetAnotherDog implements DogState {

public void showState() {

System.out.println("嬉戏");

}

}

class Dog {

DogState state;

public void show() {

state.showState();

}

public void setState(DogState s) {

state = s;

}

}

public class test1 {

public static void main(String[] args) {

Dog yellowDog = new Dog();

System.out.println("狗在主人面前：");

yellowDog.setState(new SoftlyState());

yellowDog.show();

System.out.println("狗遇到敌人：");

yellowDog.setState(new MeetEnemyState());

yellowDog.show();

System.out.println("狗遇到朋友：");

yellowDog.setState(new MeetFriendState());

yellowDog.show();

System.out.println("狗遇到同伴：");

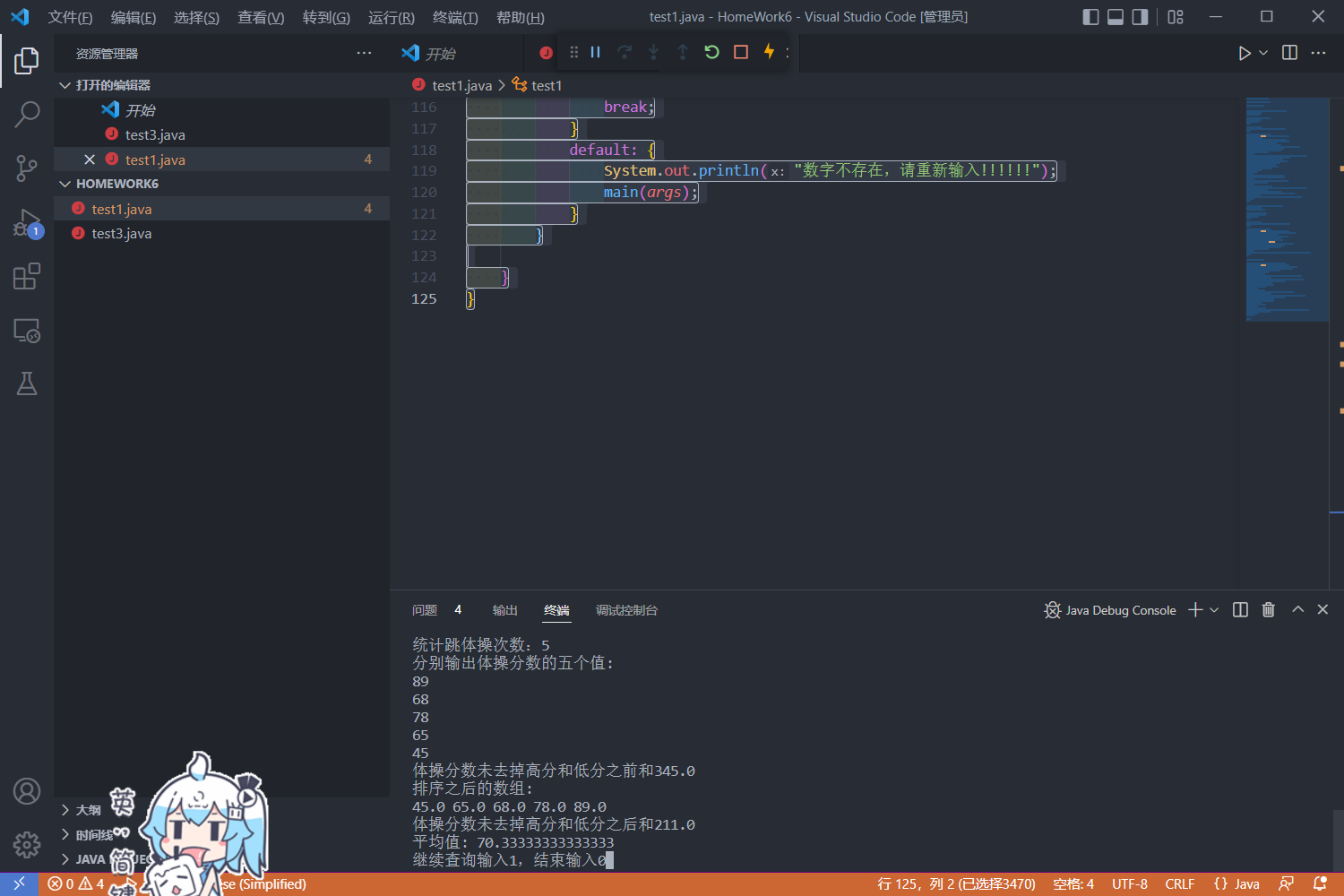
yellowDog.setState(new MeetAnotherDog());

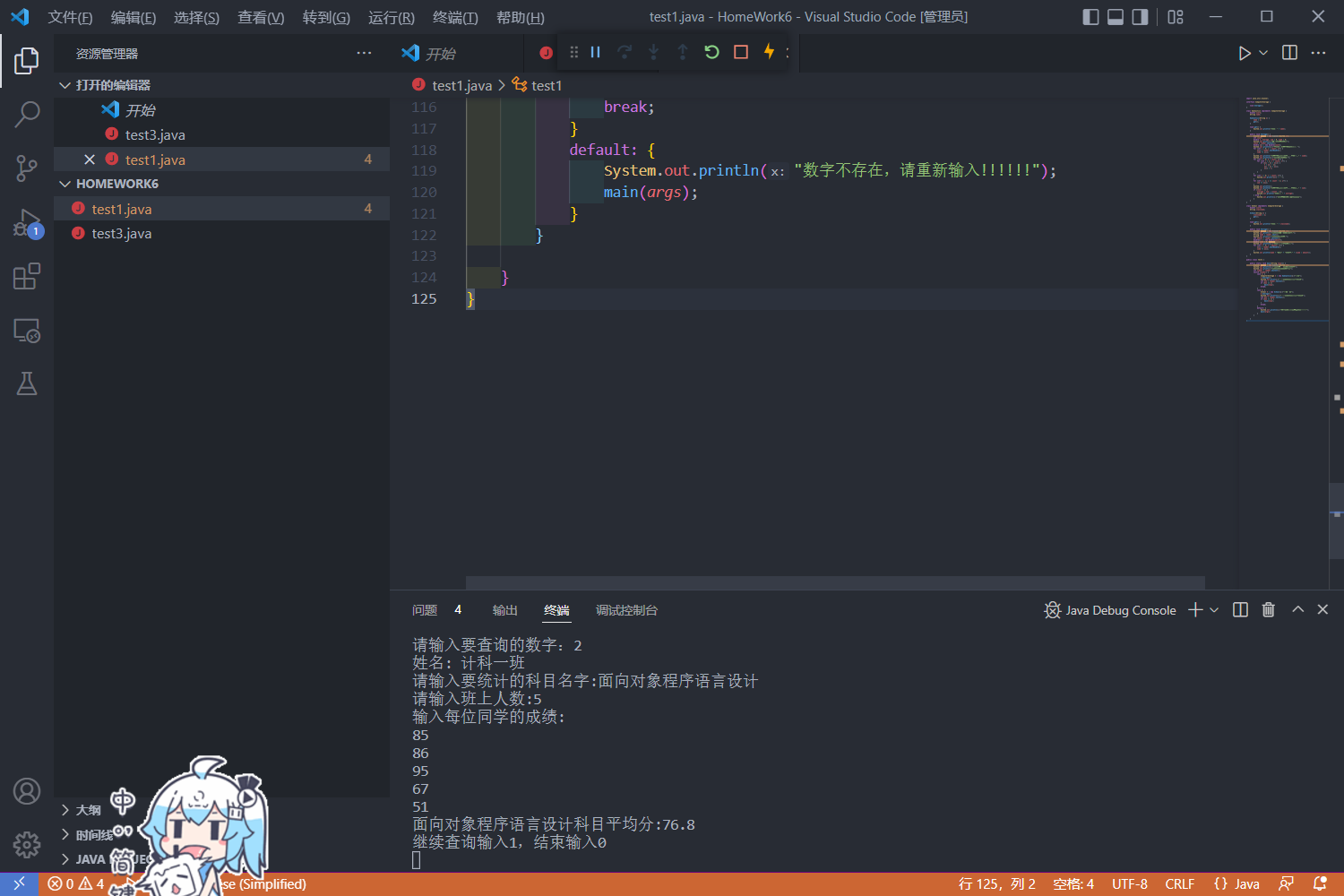
yellowDog.show();

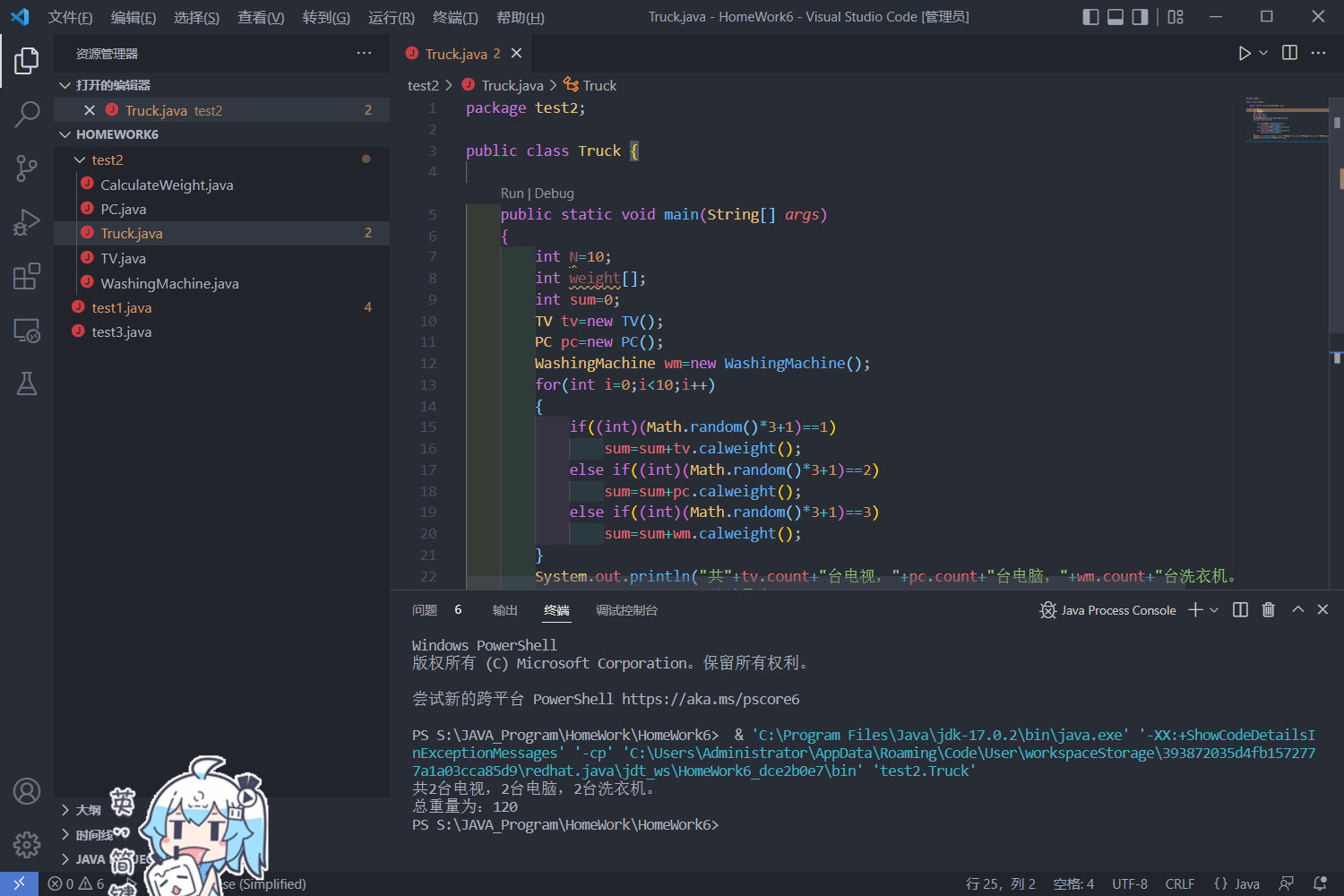
}

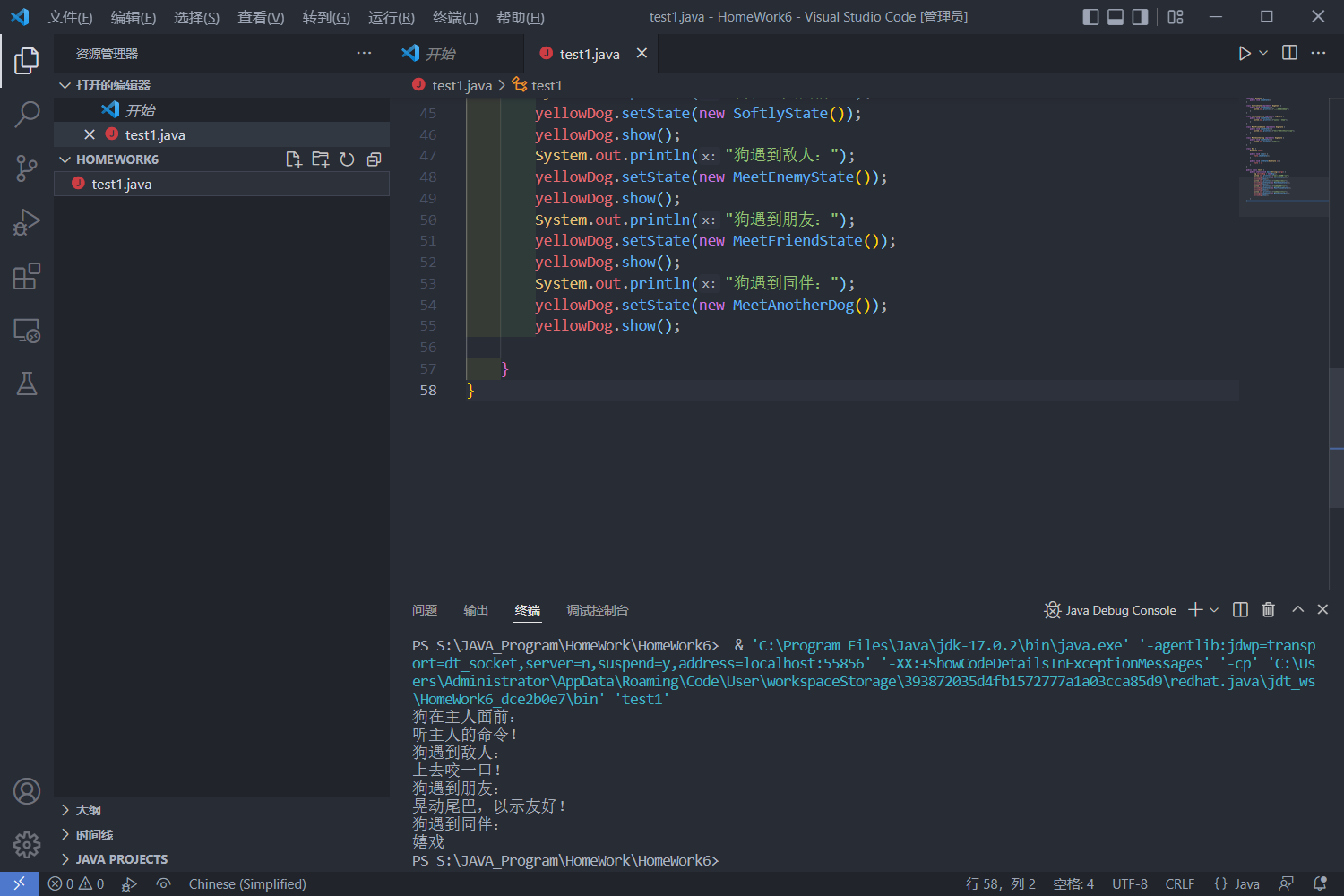
}

1. 实验结果及分析









六、实验总结及体会