**实验一**

题目一：

P19 实验4 （5分）

请你编写一个java程序输出自己的姓名、年龄、身高和体重，分别通过控制台和Eclipse环境运行。

1. 题目分析：

本题要求实现java的键盘输入和输出，引入java中的scanner类帮助实现，它在java的util工具包下，主要用于用户从控制台输入的文本。

1. 代码实现

package shiyan1;

import java.util.Scanner; //导入键盘输入输出类

public class test01 {

public static void main(String[] args) {

var sc = new Scanner(System.in); // 创建Scanner类对象

System.out.println("请输入你的姓名：");

String name = sc.nextLine(); //调用方法接收各种类型的数据

System.out.println("请输入你的年龄：");

int age = sc.nextInt(); //输出int值

System.out.println("请输入你的身高：");

int height = sc.nextInt();

System.out.println("请输入你的体重：");

int weight = sc.nextInt();

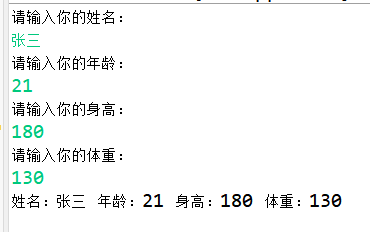
System.out.println("姓名："+name+" "+"年龄：" + age +" "+"身高："+height+" "+"体重："+weight);

}

sc.close(); //关闭Scanner类

}

1. 运行结果



题目二：

P41 实验 （5分）

打印出100到1000之间所有的"水仙花数"，所谓"水仙花数"是指一个三位数，其各位数字立方和等于该数本身。例如：153是一个"水仙花数"，因为153=1的三次方＋5的三次方＋3的三次方。

1. 题目分析

本题主要考察java中的循环结构以及运算符的使用，根据题目描述，应采用for循环结构。同时使用Math类中的方法pow进行运算。

1. 代码实现

package shiyan01\_1;

public class Narcissistic\_number {

public static void main(String[] args) {

for(int i=100;i<=1000;i++) {

int j= i%10; //取得个位数字

int k= (i/10)%10; //取得十位数字

int m= i/100; //取得百位数字

if((Math.pow(j, 3)+Math.pow(k, 3)+Math.pow(m, 3))==(i)) {

//注意判断时i的位置加（）

System.out.println(i);

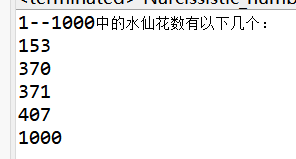
}

}

}

}

1. 运行结果



题目三：

创建一个类，此类中包含多个静态成员变量和普通成员变量（变量的类型可以自定义），使用不同的方法进行初始化，同时需要展示出变量初始化的顺序。 （5分）

1. 题目分析

本题考察静态成员变量与普通成员变量在初始化时对不同方法的次序差异，建立三个类，一个类checkout用于测试，类Fruit作为一个基本类，其中含有初始化函数Fruit（）和function（）方法，在类variate中创建Fruit对象调用Fruit函数，variate函数中同时引入Fruit类中的function函数；测试类中同时创建Fruit与variate对象，比较static变量和普通变量初始化的顺序。

1. 代码实现

Fruit类：  
package shiyan01\_2;

public class Fruit {

Fruit(int number){

System.out.println("Fruit("+number+")");

}

void function(int number) {

System.out.println("function("+number+")");

}

}

variate类：

package shiyan01\_2;

public class variate {

static Fruit b = new Fruit(2);//创建Fruit类对象b，静态变量

static Fruit c = new Fruit(0);

variate(int number){

System.out.println("variate()");

c.function(0); //调用Fruit中function函数

}

}

checkout类：

package shiyan01\_2;

public class checkout {

public static void main(String[] args) {

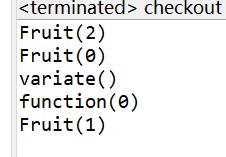
Fruit f = new Fruit(1); //普通类对象

}

static variate s = new variate(5);

}

3.运行结果



可以看到先初始化静态对象b，之后是c，然后是测试类中的s对象，同时c引用function函数，最后初始化普通成员变量f，显示出Fruit（1）

题目四：

创建一个包含多个类的继承结构，编写一个使用这些类的方法，展示动态绑定和静态绑定的区别。（5分）

1. 题目分析

本题考察类之间的继承关系与运行时动态绑定和编写时静态绑定的区别，所以应建立三个类。类extendss作为超类，类kids作为子类展示继承关系，类check\_out作为测试类测试继承关系与运行时绑定的状态。

1. 代码实现

类extendss：

package shiyan\_04;

public class extendss {

int k;

extendss(){} //初始化

void f1() {

System.out.println("超类的方法1");

System.out.println("Father-A");

}

}

类kids：

package shiyan\_04;

public class kids extends extendss{ //使用extends表示继承超类extendss

void f1() { //超类方法的复用、重写

System.out.println("子类中的方法");

System.out.println("Kids-BB");

}

void kid2() {

System.out.println("子类正在使用父类中的变量->"+super.k);

}

}

类check\_out：

package shiyan\_04;

public class check\_out {

public static void main(String[] args) {

extendss f = new kids();//实际对象是kids类

f.f1(); //运行过程中，系统在程序中找到对象f对应类中的函数f1，运行针对对象而绑定

kids d = new kids();

d.kid2();

}

}

1. 运行结果

