1.实例变量和局部变量

程序的基本功能是处理数据,程序中需要使用变量来接收并表示数据。

程序中的变量必须先定义之后才能使用。

定义变量包括定义变量的"类型"以及变量的"名字"。

例如: int a; Student s;

程序中的变量分为实例变量和局部变量。

1)实例变量(又称属性、全局变量、成员变量)

位置:

直接定义在类中,属于类中的一种成员(成员变量/成员方法)。

例如:

public class Student{

public String name;

}

作用范围:

是指变量定义之后,都在哪些地方可以被访问到。

由于实例变量是直接定义在类中的,所以在定义的时候可以使用修饰符进行修饰。

例如:这四种修饰符的作用之后会介绍到

public class Student{

public String name1;

protected String name2;

String name3;

private String name4;

}

以上是使用了四种访问权限控制修饰符来修饰了这四个变量。

但是不管是哪种修饰符情况,实例变量在当前类中的每一个方法都是可以被访问到的(static修饰的静态方法除外)。

public class Student{

public String name1;

protected String name2;

String name3;

private String name4;

//类中的方法里面可以直接访问实例变量

public void run(){

System.out.println(name1);

System.out.println(name2);

System.out.println(name3);

System.out.println(name4);

}

}

默认值:

类的中实例变量声明后,如果我们没有显式赋值的话,每种实例变量其实都是有自己的默认值的。

实例变量的赋值分为隐式赋值和显式赋值

隐式赋值就是JVM给它赋的值(就是说的默认值)

显式赋值就是我们主动给它赋的值 (=)

例如:

public class Student{

public int a;

public float f;

public char c;

public boolean b;

public String str;

public void println(){

System.out.println(a);

System.out.println(f);

System.out.println(c);

System.out.println(b);

System.out.println(str);

}

}

基本类型的实例变量:

整型 :默认值为 0

浮点型 :默认值为 0.0

布尔型 :默认值为 false

字符型 :默认值为 0 或者 '\u0000'

引用类型的实例变量:

默认值都是 null

生命周期:

实例变量是属于对象的,一个对象被创建出来的时候,这个对象中的实例变量就有了,直到这个对象被GC当做垃圾回收之后,这个实例变量也就没有了。

1)局部变量

位置:

定义在方法的代码块中或者方法的参数列表中

例如:a b c d四个都是局部变量

public void test(int a){

int b;

{

int c;

}

if(true){

int d;

}

}

作用范围:

1)如果是定义在方法的参数列表中,那么在当前方法的任何位置都可以访问该局部变量

例如:

public void test(int a){

System.out.println(a);

{

System.out.println(a);

}

if(true){

System.out.println(a);

}

}

2)如果是定义在方法中,那么就要看这个局部变量是"直接"位于哪一对大括号内

例如:

public void test(){

int a;

{

int b;

}

if(true){

int c;

}

}

观察局部变量a/b/c三个分别是"直接"位于哪一对大括号内,那么这个局部变量的作用范围就在那一对大括号内.

如果访问的地方出了这个范围,那么是访问不到这个局部变量的。