**Java变量类型**

在 Java 语言中，所有的变量在使用前必须声明。

声明变量的基本格式如下：

type identifier [ = value][, identifier [= value] ...] ;

格式说明：

type -- 数据类型。

identifier -- 是变量名，可以使用逗号 , 隔开来声明多个同类型变量。

以下列出了一些变量的声明实例。注意有些包含了初始化过程。

int a, b, c; // 声明三个int型整数：a、 b、c

int d = 3, e = 4, f = 5; // 声明三个整数并赋予初值

byte z = 22; // 声明并初始化 z

String s = "runoob"; // 声明并初始化字符串 s

double pi = 3.14159; // 声明了双精度浮点型变量 pi

char x = 'x'; // 声明变量 x 的值是字符 'x'。

Java 语言支持的变量类型有：

局部变量（Local Variables）：局部变量是在方法、构造函数或块内部声明的变量，它们在声明的方法、构造函数或块执行结束后被销毁，局部变量在声明时需要初始化，否则会导致编译错误。

public void exampleMethod() {

int localVar = 10; // 局部变量

// ...

}

实例变量（Instance Variables）：实例变量是在类中声明，但在方法、构造函数或块之外，它们属于类的实例，每个类的实例都有自己的副本，如果不明确初始化，实例变量会被赋予默认值（数值类型为0，boolean类型为false，对象引用类型为null）。

public class ExampleClass {

int instanceVar; // 实例变量

}

静态变量或类变量（Class Variables）：类变量是在类中用 static 关键字声明的变量，它们属于类而不是实例，所有该类的实例共享同一个类变量的值，类变量在类加载时被初始化，而且只初始化一次。

public class ExampleClass {

static int classVar; // 类变量

}

参数变量（Parameters）：参数是方法或构造函数声明中的变量，用于接收调用该方法或构造函数时传递的值，参数变量的作用域只限于方法内部。

public void exampleMethod(int parameterVar) {

// 参数变量

// ...

}

以下实例中定义了一个 RunoobTest 类，其中包含了一个成员变量 instanceVar 和一个静态变量 staticVar。

method() 方法中定义了一个参数变量 paramVar 和一个局部变量 localVar。在方法内部，我们将局部变量的值赋给成员变量，将参数变量的值赋给静态变量，然后打印出这些变量的值。

在 main() 方法中，我们创建了一个 RunoobTest 对象，并调用了它的 method() 方法。

实例

public class RunoobTest {

// 成员变量

private int instanceVar;

// 静态变量

private static int staticVar;

public void method(int paramVar) {

// 局部变量

int localVar = 10;

// 使用变量

instanceVar = localVar;

staticVar = paramVar;

System.out.println("成员变量: " + instanceVar);

System.out.println("静态变量: " + staticVar);

System.out.println("参数变量: " + paramVar);

System.out.println("局部变量: " + localVar);

}

public static void main(String[] args) {

RunoobTest v = new RunoobTest();

v.method(20);

}

}

运行以上代码，输出如下：

成员变量: 10

静态变量: 20

参数变量: 20

局部变量: 10