**Java 基础语法**

**第一个Java程序**

下面看一个简单的 Java 程序，它将输出字符串 *Hello World*

**实例**

public class HelloWorld {

/\* 第一个Java程序 \* 它将输出字符串 Hello World \*/

public static void main(String[] args) {

System.out.println("Hello World");

// 输出 Hello World

}

}

日程表

描述已自动生成

**基本语法**

编写 Java 程序时，应注意以下几点：

* **大小写敏感**：Java 是大小写敏感的，这就意味着class Hello 与 hello 是不同的。
* **类名**：对于所有的类来说，类名的首字母应该大写。如果类名由若干单词组成，那么每个单词的首字母应该大写，例如 **MyFirstJavaClass** 。
* **方法名**：所有的方法名都应该以小写字母开头。如果方法名含有若干单词，则后面的每个单词首字母大写。
* **源文件名**：源文件名必须和类名相同。当保存文件的时候，你应该使用类名作为文件名保存（切记 Java 是大小写敏感的），文件名的后缀为 **.java**。（如果文件名和类名不相同则会导致编译错误）。
* **主方法入口**：所有的 Java 程序由 **public static void main(String[] args)** 方法开始执行。

**Java 变量**

Java 中主要有如下几种类型的变量

* 局部变量
* 类变量（静态变量）
* 成员变量（非静态变量）

内存中的一个存储区域

该区域有自己的名称（变量名）和类型（数据类型）

Java中每个变量必须先声明，后使用

该区域的数据可以在同一类型范围内不断变化

//定义变量的格式 ： 数据类型 变量名 = 初始化值;

如：int age = 10;

//可以通过变量的名字，访问变量的值

如：System.out.println(age);

//变量的值可以在同一类型范围内不断变化

如：age = 5;

变量的作用域(作用范围)

一对{ }之间有效

如：

{

int i = 10;

System.out.println(i);

}

以下不能访问

{

int i = 10;

}

System.out.println(i);

public class Variable{

static int allClicks=0; // 类变量

String str="hello world"; // 实例变量

public void method(){

int i =0; // 局部变量

}

}

package test;

public class Test{

public void pupAge(){

int age = 0; age = age + 7;

System.out.println("小狗的年龄是: " + age); }

public static void main(String[] args){

Test test = new Test();

test.pupAge();

}

}

**Java 标识符**

Java 所有的组成部分都需要名字。类名、变量名以及方法名都被称为标识符。

关于 Java 标识符，有以下几点需要注意：

* 所有的标识符都应该以字母（A-Z 或者 a-z）,美元符（$）、或者下划线（\_）开始
* 首字符之后可以是字母（A-Z 或者 a-z）,美元符（$）、下划线（\_）或数字的任何字符组合
* 关键字不能用作标识符
* 标识符是大小写敏感的
* 合法标识符举例：age、$salary、\_value、\_\_1\_value
* 非法标识符举例：123abc、-salary
* 命名规范：
* Java中的名称命名规范：
* 包名：多单词组成时所有字母都小写：xxxyyyzzz
* 类名接口名：多单词组成时，所有单词的首字母大写：XxxYyyZzz
* 变量名和方法名：多单词组成时，第一个单词首字母小写，第二个单词开始每个单词首字母大写：xxxYyyZzz
* 常量名：所有字母都大写。多单词时每个单词用下划线连接：XXX\_YYY\_ZZZ

**Java 数组**

数组是储存在堆上的对象，可以保存多个同类型变量。在后面的章节中，我们将会学到如何声明、构造以及初始化一个数组。

**Java 关键字**

1.关键字

就是被java语言赋予了特殊的含义，用做专门用途的字符串（单词）

关键字的特点都是小写的。如：class,public,void,static,if,for,int,byte....

下面列出了 Java 关键字。这些保留字不能用于常量、变量、和任何标识符的名称。

|  |  |  |
| --- | --- | --- |
| **类别** | **关键字** | **说明** |
| 访问控制 | private | 私有的 |
| protected | 受保护的 |
| public | 公共的 |
| default | 默认 |
| 类、方法和变量修饰符 | abstract | 声明抽象 |
| class | 类 |
| extends | 扩充,继承 |
| final | 最终值,不可改变的 |
| implements | 实现（接口） |
| interface | 接口 |
| native | 本地，原生方法（非 Java 实现） |
| new | 新,创建 |
| static | 静态 |
| strictfp | 严格,精准 |
| synchronized | 线程,同步 |
| transient | 短暂 |
| volatile | 易失 |
| 程序控制语句 | break | 跳出循环 |
| case | 定义一个值以供 switch 选择 |
| continue | 继续 |
| default | 默认 |
| do | 运行 |
| else | 否则 |
| for | 循环 |
| if | 如果 |
| instanceof | 实例 |
| return | 返回 |
| switch | 根据值选择执行 |
| while | 循环 |
| package | 包 |
| 基本类型 | boolean | 布尔型 |
| byte | 字节型 |
| char | 字符型 |
| double | 双精度浮点 |
| float | 单精度浮点 |
| int | 整型 |
| long | 长整型 |
| short | 短整型 |
| void | 无返回值 |
| 保留关键字 | goto | 是关键字，但不能使用 |
| const | 是关键字，但不能使用 |
| null | 空 |

**Java注释**

类似于 C/C++、Java 也支持单行以及多行注释。注释中的字符将被 Java 编译器忽略。

public class HelloWorld { /\* 这是第一个Java程序 \* 它将输出 Hello World \* 这是一个多行注释的示例 \*/ public static void main(String[] args){ // 这是单行注释的示例 /\* 这个也是单行注释的示例 \*/ System.out.println("Hello World"); } }

**Java 空行**

空白行或者有注释的行，Java 编译器都会忽略掉。