

# 第二章 变量和数据类型

## 课前回顾

### 1. 什么是程序？

程序就是为了解决某个问题或者实现某个目标而编写的一系列有序指令的集合

### 2. Java程序是如何执行的？

Java是一门高级语言，计算机不能够直接识别我们编写的Java程序。因此，Java 提供了 `JVM` (Java Virtual Machine, Java虚拟机)，`JVM` 能够将class文件转换为计算机能够识别的指令，而我们编写的源程序是 `java` 文件，所以还需要使用编译器将我们编写的源程序编译为class文件

`javac` `java`文件的位置 => 类名.class

`java` 类名 => 告诉 `JVM` 应该加载指定类名的类进入 `JVM`，然后执行该类的main方法，程序开始执行

### 3. Java程序是如何实现跨平台的？

首先，要执行Java程序，必须要具备Java运行时环境，也就是 `JRE`，而 `JRE` 中就包含有 `JVM`，而 `JVM` 能够将class文件翻译为当前平台能够识别的指令。因此，不管什么平台，只要安装 `JRE`，都会具备 `JVM`，只要具备 `JVM`，就能将class文件翻译为当前平台能够识别的指令。

windows => 安装windows版的 `JRE`

linux => 安装 linux 版 `JRE`

### 4. 所有Java程序执行的入口是什么？

```
1 public static void main(String[] args){
2
3 }
```

## 主要内容

- 使用 `IntelliJ Idea` 创建工程 熟悉
- Java 中的数据类型 重点
- 变量的定义与使用 重点
- 运算符 熟悉
- 数据类型转换 重点
- `Scanner` 的基本使用 熟悉

## 课程目标

- 熟悉使用 `IntelliJ Idea` 创建工程
- 掌握 Java 中的8种基本数据类型
- 掌握变量的本质及运用
- 掌握数据类型之间的相互转换
- 熟练使用 `Scanner`

## 第一节 使用 `IntelliJ Idea` 创建工程

演示

## 第二节 Java 中的数据类型

### 1.数据的概念

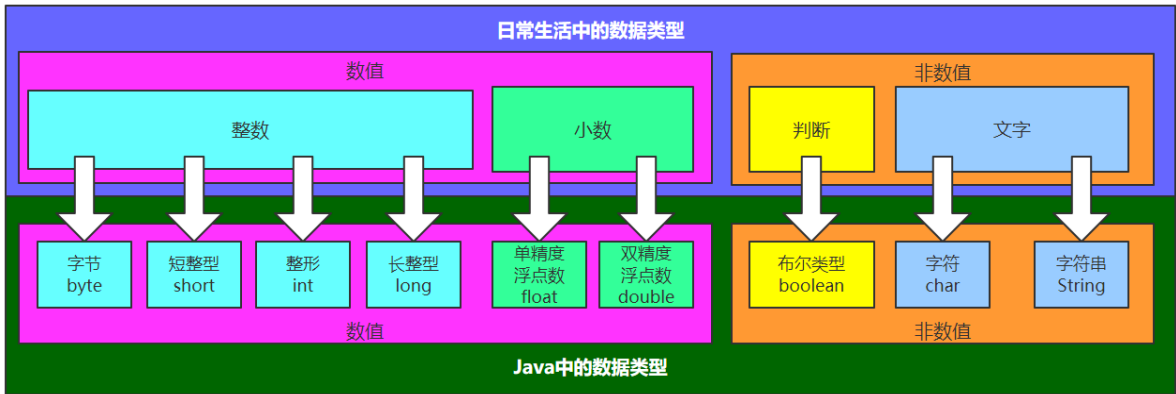
数据就是信息的符号表示。

华为手机 荣耀 999.9元 2年

刘德华 帅 20 185 富豪

### 2.数据类型

生活中我们常见的数据类型



a abcde 他长得很帅

**Java中的数据类型分为 基本数据类型 和 引用数据类型 两大类**

Java 中的基本数据类型

大类型	类型名称	英文	占用空间	取值范围
整型	字节	byte	1字节	$-2^7 \sim 2^7-1$
	短整型	short	2字节	$-2^{15} \sim 2^{15}-1$
	整型	int	4字节	$-2^{31} \sim 2^{31}-1$
	长整型	long	8字节	$-2^{63} \sim 2^{63}-1$
小数	单精度浮点数	float	4字节	$-2^{127} \sim 2^{127}-1$
	双精度浮点数	double	8字节	$-2^{1023} \sim 2^{1023}-1$
判断	布尔类型	boolean	1字节	true或者false
文字	字符	char	2字节	键盘能敲出来的单个文字或符号

a 男 1

其中，字节是计算机存储数据的基本计量单位。计算机中最小单位是位，英文（Bit），位与字节的单位换算关系为：1 B = 8 bit

常用计量单位换算

类型名称	英文名称	缩写	单位换算
字节	Byte	B	-
千字节	KiloByte	KB	1KB = 1024B
兆字节	MegaByte	MB	1MB = 1024KB
吉字节	GigaByte	GB	1GB = 1024MB
太字节	TeraByte	TB	1TB = 1024GB
拍字节	PetaByte	PB	1PB = 1024TB

每年的10月24日都是程序员节日

## 第三节 变量的定义与使用

### 1. 变量的概念

从字面意思看，变量就是会变化的量。

从计算机专业角度看，变量就是一个内存空间地址的表示。

**在Java中，内存分为栈内存和堆内存两大块**

### 2. 变量的声明

#### 语法

```
1 数据类型 变量名;  
2 变量名 = 变量值;  
3  
4 数据类型 变量名 = 变量值;
```

#### 示例

```
1  /**  
2   * 变量的定义  
3   * @author wu  
4   */  
5  public class Example1 {  
6  
7      public static void main(String[] args) {  
8          //数据类型 变量名;  
9          //变量名 = 变量值;  
10         byte b; //告诉计算机在栈上开辟一块1个字节大小的空间，并将这块空间命名为b  
11         b = 10; //告诉计算机将整数10放入开辟好的空间b中  
12         //数据类型 变量名 = 变量值;  
13         //告诉计算机在栈上开辟一块2个字节大小的空间，并将这块空间命名为s，  
14         //然后将整数5放入这块空间中  
15         short s = 5;  
16         //告诉计算机在栈上开辟一块4个字节大小的空间，并将这块空间命名为age  
17         //然后将整数20放入这块空间中  
18         int age = 20;  
19         //将整数25放入名为age的内存空间中  
20         age = 25;  
21         long time = 20210312;  
22         //在Java中，浮点数默认是双精度浮点数。如果要给一个单精度浮点数变量赋值，需要在变  
23         量值  
24         //后面加上一个f或者F，表示是单精度浮点数  
25         float f = 10.0F;  
26         double score = 65.5;  
27         //在Java的布尔类型变量，编译完成后就不存在了。因为Java在C++基础上开发的，而  
28         C++中使用  
29         //0 与 非0表示假和真。而JVM是执行Java程序的场所，在JVM规范中有明确说明，单个  
30         boolean  
31         //类型的变量，在编译的过程中会被翻译为int类型的变量，此时占用4个字节的空间。对于  
32         boolean  
33         //类型的数组，在翻译的过程中会被翻译为byte类型的数组，此时每个变量占用1个字节的  
34         空间。  
35         boolean isNice = true;
```

```

31         //char类型变量能否存在汉字?
32         //能。Java采用的是Unicode编码，而Unicode编码中收录的汉字占用的是2个字节，我们
    的char类型
33         //变量也是占用2个字节的空间，因此，在Unicode编码中收录的汉字可以使用char类型变
    量存储。而
34         //在Unicode编码中未收录的汉字不一定能够使用char类型变量存储。
35         char sex = '男';
36     }
37 }

```

**在Java中变量名必须保证唯一**

### 3. 变量的命名规则

**变量名只能以字母、下划线和\$开始，其余部分由任意多的字母、数字、下划线和\$组成。**

**变量名不能是Java中的保留字（关键字）。**

Java语言是严格区分大小写的

```

1 //Java语言是严格区分大小写的
2 int number = 5;
3 int NUMBER = 5;

```

### 4. 变量的命名规范

在Java中，变量名采用的是驼峰命名法。变量名组成的第一个单词的首字母小写，其余单词首字母均大写。

## 第四节 运算符

### 1. 常规运算符 (+ - \* /)

示例

```

1 /**
2  * 运算符操作 + - * /
3  * @author wu
4  */
5 public class Example2 {
6
7     public static void main(String[] args) {
8         //变量在使用之前必须完成初始化操作。换言之就是给变量赋值。
9         //    int a, b;
10        //    a = 5;
11        //    b = 10;
12        int a = 5, b = 10; //可以在一行代码中同时声明多个变量并完成赋值
13        int c = a + b;
14        System.out.println("a + b = " + c);
15        int d = b - a;
16        System.out.println("b - a = " + d);
17        int e = a * b; //注意：在Java中乘法使用的是*
18        System.out.println("a * b = " + e);
19        int f = b / a; //注意：在Java中除法使用的是/
20        System.out.println("b / a = " + f);
21    }
22 }

```

## 2.取模 (%)

### 示例

```
1 int a = 10 % 3; //求10除以3的余数
2 System.out.println("10 % 3 = " + a);
```

### 案例

给定一个数，如5678，求组成这个数的每一位数字

### 代码实现

```
1 //21 % 10 = 1
2 //5678 / 10 = 567
3 //5678 / 100 = 56
4 //5678 / 1000 = 5
5 int number = 5678;
6 int ge = number % 10;
7 //在Java中，两个整数相除，得到的结果一定是整数
8 int shi = number / 10 % 10;
9 int bai = number / 100 % 10;
10 int qian = number / 1000;
11 System.out.println("个位: " + ge); //ctrl + d 复制一行代码
12 System.out.println("十位: " + shi);
13 System.out.println("百位: " + bai);
14 System.out.println("千位: " + qian);
```

**在Java中，两个整数相除，得到的结果一定是整数**

## 3. ++ 和 --

### 示例

```
1 //++运算符分为前置++和后置++
2 int count = 0;
3 count++; //表示自加1 => count = 1;
4 ++count; //表示自加1 => count = 2;
5 //如果一行代码中，只有++运算符，没有其他运算符，那么不论++在前，还是在后执行结果都是一样的
6 //因为++运算符在变量的后面，因此，先将变量count的值赋值给变量number1，然后变量count自加1
7 int number1 = count++; //number1 = 2 count = 3;
8 //因为++运算符在变量的前面，因此，变量count先自加1，然后将count变量赋值给变量number2
9 int number2 = ++count; //count = 4 number2 = 4;
10 System.out.println(number1);
11 System.out.println(number2);
```

面试题：

分析下面代码的执行过程

```

1  int c = 10;
2  //在java中运算都是从左往右依次执行，因此c++先执行，++c后执行。
3  //c++ => c = 10; => c = 11;
4  //++c => c = 12
5  int d = c++ + ++c;
6  System.out.println(d); //22 11+11? 10+12?

```

#### 4. +=、-=、\*=、/=、%=

##### 示例

```

1  /**
2   * 运算符操作 += -= *= /= %=
3   */
4  public class Example4 {
5
6      public static void main(String[] args) {
7          int number1 = 10;
8          //将number1 + 10 的结果赋值给number1
9          number1 = number1 + 10;
10         number1 += 10; //与上一行代码的作用是一样的
11
12
13         double d1 = 20.0;
14         d1 = d1 / 2;
15         d1 /= 2; //与上一行代码的作用是一样的
16
17         int result1 = 10; //整数
18         double add = 2.0; //浮点数
19         // result1 = result1 + add;
20         result1 += add; //说明 += 还具有其他的功能
21     }
22 }

```

## 第五节 数据类型转换

### 1. 自动类型转换

相互兼容的多种数据类型在运算时，小转大即为自动类型转换。如：10 + 0.5 => 10.0 + 0.5

### 2. 强制类型转换

相互兼容的多种数据类型在运算时，大专小即为自动类型转换。如：68.5 + 1.5 => (int)70.0 => 70

强制类型转换语法规则：

```

1  (目标数据类型)变量名

```

**在Java中，数值类型互相兼容，在进行运算时，必须保证表达式中的参与运算的数据的所有数据类型保持一致**

### 3. 案例

小明考试的时候考了59分，小明感觉很憋屈，于是奋发图强，然后下一次考试比上一次考试多考了1.5分，请问小明本次考试考了多少分？

### 4. 代码实现

```

1 public class Example5 {
2
3     public static void main(String[] args) {
4         int score = 59;
5         //自动类型转换
6         double nextScore = score + 1.5; // 59 + 1.5 => 59.0 + 1.5 => 60.5
7         System.out.println(nextScore);
8         //强制类型转换
9         int nScore = (int)(score + 1.5); // 59 + 1.5 => 59.0 + 1.5 => 60.5
10        System.out.println(nScore);
11    }
12 }

```

## 第六节 Scanner的基本使用

### 1. Scanner的作用

提供一种与用户交互的方式，用户可以在控制台输入一些数据，程序来获取这些数据

### 2. 常用方法

方法名	解释说明
<code>nextDouble()</code>	获取用户从控制台输入的浮点数，如果输入的不是数字，就会出错
<code>nextFloat()</code>	获取用户从控制台输入的浮点数，如果输入的不是数字，就会出错
<code>nextInt()</code>	获取用户从控制台输入的整数，如果输入的不是整数，就会出错
<code>nextByte()</code>	获取用户从控制台输入的整数，如果输入的不是整数，就会出错
<code>nextShort()</code>	获取用户从控制台输入的整数，如果输入的不是整数，就会出错
<code>nextLong()</code>	获取用户从控制台输入的整数，如果输入的不是整数，就会出错
<code>nextBoolean()</code>	获取用户从控制台输入的boolean值，只能输入true或者false，否则就会出错
<code>next()</code>	获取用户从控制台输入的字符串

### 3. 案例

从控制台输入3位学生的成绩，然后计算平均分

### 4. 代码实现

```
1 public class Example7 {
2
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         System.out.println("请输入3位学生成绩（每一位学生成绩可以使用空格隔开）：");
6         int score1 = sc.nextInt();
7         int score2 = sc.nextInt();
8         int score3 = sc.nextInt();
9         int avg = (score1 + score2 + score3) / 3; //在Java中，两个整数相除，得到
            的结果一定是整数
10        System.out.println(avg);
11    }
12 }
```