



IT 兄弟连
ITXDL.CN

Java语言基础

【Day02】

Java 变量和注释

Variable note

什么是变量？

- 用于指代内存中的一块存储区域。
- 变量必须有特定的数据类型，不同的数据类型表示不同的数据存储结构。
- 每个变量都有自己的作用范围，叫作用域。

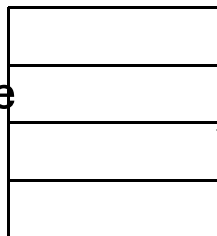
变量的声明和初始化

- 数据类型 变量名 = 初始值;
- 其中 =初始值 是可以省略的，但推荐写上。

数据类型

int age

变量名称



变量指内存中的存储区域，不同的变量类型决定不同的存储结构。

变量的注意事项一

- Java是强类型语言，变量在使用前必须声明，指明其数据类型。
- 编译器会根据变量的类型检测对变量的操作是否合法。

```
int a = 100;
```

```
System.out.println(a);
```

```
System.out.println(b);
```

变量没有被声明

```
a = 123.456;
```

对变量的赋值与变量的类型不匹配

变量的注意事项二

- Java变量在使用前必须初始化，也就是赋以确定的初值。
- 属性系统会自动初始化。

```
int b;
```

```
System.out.println(b)
```

 编译错误，未赋初值

```
int a;
```

```
a = 100;
```

```
System.out.println(a)
```

 在第一次使用前必须赋以确定值

```
int a = 100;
```

通常声明和赋初始值结合在一起

标识符的命名法则

- 变量名必须先标识符，标识符命名的一些规则：
 - ✓ 必须是字母、数字、下划线、\$等，其中数字不能开头。
 - ✓ 不能是Java关键字，比如：public static class ...
 - ✓ 大小写敏感，长度没有限制，但不宜过长。
 - ✓ 标识符尽量做到见名知意，可以是汉字，但不推荐使用。
- 标识符可以给类/变量/属性/方法/包 起名字。

Java语言中的关键字

- Java关键字全部都是小写。

关键字					
abstract	boolean	break	byte	case	catch
char	class	const	continue	default	do
double	else	extends	final	finally	float
for	goto	if	implements	import	instanceof
int	interface	long	native	new	package
private	protected	public	return	short	static
strictfp	super	switch	synchronized	this	throw
throws	transient	try	void	volatile	while
assert	enum				

项目案例

- 提示用户输入姓名和年龄信息，最后打印出来

Java API 的使用

- JDK中带有大量的API类，是有Java系统带来的工具库，这些工具来数以万计！是Java官方程序员的技术积累。
如：`java.lang.String`, `java.lang.System`
- 这里类存储在 `JAVA_HOME/jre/lib/rt.jar` 等文件中
如：`java.lang.String.class` `java.lang.System.class`
- 这些类可以大大简化编程，提高开发效率
使用API类要用import语句导入类
如：`import java.util.Scanner`
具体的API类功能，可以参阅Java的参考手册

Java的注释

- 注释用于解释代码，给程序员看但计算机会忽略注释，主要包括以下三种：
 - ✓ `//` 单行注释 ，从 `//` 开始，到本行结束，都是注释。
 - ✓ `/* */` 多行注释，从`/*` 开始，到`*/`结束，中间所有都是注释。
 - ✓ `/** */` 多行注释，从`/**`开始，到`*/`结束，是一种支持提取的注释。

Java 数据类型

data type

数据类型的分类

- Java分为基本数据类型和引用数据类型。
 - ✓ 其中基本类型有8种。
 - ✓ 其中引用数据类型包括：数组、类、接口、枚举和标注。
- 枚举可以看成特殊的类，标注可以看成特殊的接口。

进制转换

- 常见的进制有：二、八、十六进制 和 十进制。其中，十六进制用a到f代表10到15。八进制和十六进制其实都是二进制的简写。
- 正数的二进制转十进制只需要按照权重相加即可。
- 正数的十进制转二进制除2取余数再反向输出。
- 负数的需要补码：按位取反，再加1 。

整数类型--int

- int (32位) 的最大表示范围为： $-2^{31} \sim 2^{31}-1$ ，即 -2147483648 ~ 2147483647。如果表示更大的整数可以使用long。
- 直接量(literal)，即直接写出的常量。整数的直接量的类型默认是int类型，整数直接量也经常写16进制的形式（以0X或0x开头）。

```
int a = 100000;  
int b = 0x186a0;  
int c = 0303240;
```

16进制以0X或0x开头，8进制以0开头

```
int d = 10000000000
```

直接量超过的整数的表达范围。

整数类型--long

- long (64位) 的最大表示范围为: $-2^{63} \sim 2^{63}-1$ (9223372036854775807)。
- 如果要表示long直接量, 需要以L或l结尾。

```
long timeMillis = System.currentTimeMillis();  
System.out.println(timeMillis);
```

`System.currentTimeMillis()`; 方法返回1970年1月1日零点到此时此刻所经历的毫秒数, 该方法经常用于计时操作。

```
long d1 = 10000000000;  
long d2 = 100000000000L;
```

直接写出的即为int类型, 但超过的整数的表达范围

浮点类型

- 浮点数就是小数，包括 float类型和 double类型
- double的精度要大于float，因此，一般只使用double计算浮点数。默认的浮点数字面量是double类型。
- 其中float占四个字节，表示范围：-3.403E38~3.403E38
- 其中double占八个字节，表示范围：-1.798E308~1.798E308
- 由于舍入误差的原因，浮点数不能精确运算。

```
double money = 3.0;
```

```
System.out.println(money - 2.9); 0.10000000000000009
```

注意舍入误差的问题，如果需要精确计算，可以使用BigDecimal

项目案例

- 提示用户输入下落的时间，根据下面的公式计算下落的位移并打印出来。

自由落体位移公式为： $0.5 * 9.8 * t * t$

字符类型

- 字符类型事实上是一个16位无符号整数，这个值是对应字符的编码
- Java字符类型采用Unicode字符集编码。Unicode是世界通用的定长字符集，所有的字符都是16位
- 字符直接量可以采用诸 ‘中’ 的形式，也可以采用16进制的表示形式，例如： ‘\u4e2d’

```
char c1 = '中';
```

```
char c2 = '\u4e2d';
```

‘4e2d’ 为 ‘中’ 所对应的16位Unicode编码的16进制形式。

字符类型

- 对于不方便输入的字符采用转义字符表示, 如: `'\n'` `'\t'` `'\\'` `'\b'` `'\r'` `'\''` `'\"'`
- 数字编码: `'0':48`, `'1':49...`
- 英文编码: `'A':65`, `'B':66...` `'a':97`
- 字符是整数, 可以参与运算

```
char c = 'A' + 1;    int n = '6' - '0' ;
```
- 只有本地字符集中存在的字符才能显示



ASCII表

(American Standard Code for Information Interchange 美国标准信息交换代码)

高四位 低四位		ASCII控制字符												ASCII打印字符														
		0000						0001						0010		0011		0100		0101		0110		0111				
		0						1						2		3		4		5		6		7				
		十进制	字符	Ctrl	代码	转义 字符	字符解释	十进制	字符	Ctrl	代码	转义 字符	字符解释	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	Ctrl
0000	0	0		^@	NUL	\0	空字符	16	▶	^P	DLE		数据链路转义	32		48	0	64	@	80	P	96	`	112	p			
0001	1	1	☺	^A	SOH		标题开始	17	◀	^Q	DC1		设备控制 1	33	!	49	1	65	A	81	Q	97	a	113	q			
0010	2	2	☹	^B	STX		正文开始	18	↕	^R	DC2		设备控制 2	34	"	50	2	66	B	82	R	98	b	114	r			
0011	3	3	♥	^C	ETX		正文结束	19	!!	^S	DC3		设备控制 3	35	#	51	3	67	C	83	S	99	c	115	s			
0100	4	4	♦	^D	EOT		传输结束	20	¶	^T	DC4		设备控制 4	36	\$	52	4	68	D	84	T	100	d	116	t			
0101	5	5	♣	^E	ENQ		查询	21	§	^U	NAK		否定应答	37	%	53	5	69	E	85	U	101	e	117	u			
0110	6	6	♠	^F	ACK		肯定应答	22	—	^V	SYN		同步空闲	38	&	54	6	70	F	86	V	102	f	118	v			
0111	7	7	•	^G	BEL	\a	响铃	23	↕	^W	ETB		传输块结束	39	'	55	7	71	G	87	W	103	g	119	w			
1000	8	8	▣	^H	BS	\b	退格	24	↑	^X	CAN		取消	40	(56	8	72	H	88	X	104	h	120	x			
1001	9	9	○	^I	HT	\t	横向制表	25	↓	^Y	EM		介质结束	41)	57	9	73	I	89	Y	105	i	121	y			
1010	A	10	◼	^J	LF	\n	换行	26	→	^Z	SUB		替代	42	*	58	:	74	J	90	Z	106	j	122	z			
1011	B	11	♂	^K	VT	\v	纵向制表	27	←	^[ESC	\e	溢出	43	+	59	;	75	K	91	[107	k	123	{			
1100	C	12	♀	^L	FF	\f	换页	28	└	^_	FS		文件分隔符	44	,	60	<	76	L	92	\	108	l	124				
1101	D	13	♪	^M	CR	\r	回车	29	↔	^]	GS		组分分隔符	45	-	61	=	77	M	93]	109	m	125	}			
1110	E	14	🎵	^N	SO		移出	30	▲	^^	RS		记录分隔符	46	.	62	>	78	N	94	^	110	n	126	~			
1111	F	15	🎵	^O	SI		移入	31	▼	^-	US		单元分隔符	47	/	63	?	79	O	95	_	111	o	127	␣	^Backspace 代码：DEL		

注：表中的ASCII字符可以用“Alt + 小键盘上的数字键”方法输入。

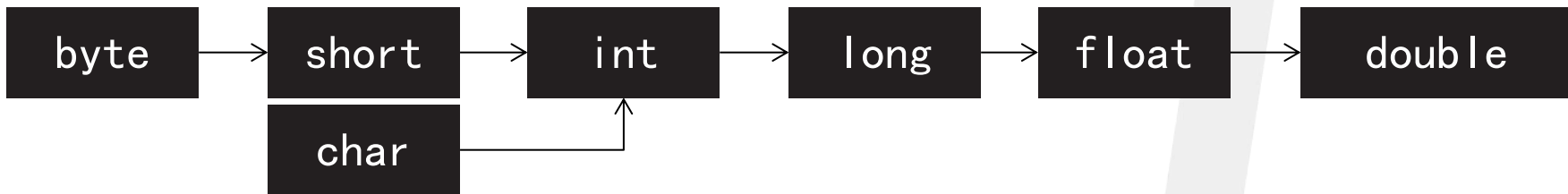
2013/08/08

8种基本数据类型

类型名称	字节空间	类型说明
byte	1字节（8位）	存储1个字节数据
short	2字节（16位）	兼容性考虑，一般不用
int	4字节（32位）	存储整数（常用）
long	8字节（64位）	存储长整数（常用）
float	4字节（32位）	存储浮点数
double	8字节（64位）	存储双精度浮点数（常用）
char	2字节（16位）	存储一个字符
boolean	1字节（8位）	存储逻辑变量（true、false）

基本数据类型之间的转换

- 自动类型转换：从小类型到大类型可以自动完成：



- 强制类型转换：从大类型到小类型需要强制转换，会造成精度损失或者溢出

```
long l = 1024L * 1024 * 1024 * 4;  
int i = (int) l;  
double pi = 3.1415926535897932384;  
float f = (float) pi;
```

总结与答疑

技术资料





IT兄弟连
ITXDL.CN

变态严管 让学习成为一种习惯