

读输入

读输入

```
Scanner in = new Scanner(System.in);  
System.out.println(in.nextLine());
```

- 将第一行抄在笔记本上，任何需要读用户输入的程序都需要这行
- 输入Scan之后，按下键盘上的Alt键和“/”键，让Eclipse自动为你补全必要的代码

今后课件上的程序都只有main的{}里面的部分

运行

- 输入也在终端窗口中
- 红色的方块表示程序正在运行
- 输入是以行为单位进行的，行的结束标志就是你按下了回车键。在你按下回车之前，你的程序不会读到任何东西

读入一行文字

- `System.out.println(in.nextLine());`
- 让in这个对象做读入下一行的动作，结果交给System.out这个对象去打印一行

字符串的+

- 用来连接两个字符串形成更长的

```
System.out.println("Hello" + " world.");  
System.out.println("Hello" + 2);  
System.out.println("Hello" + 2 + 3);  
System.out.println(2 + 3 + "Hello");
```

try

- 在输出的时候加上文字表明是回答而非单纯的重复

变量

找零计算

- 输入一个表示应该付多少元的数字，比如12，它告诉你88，意思是如果你付100

```
Scanner in = new Scanner(System.in);  
int price = 0;  
price = in.nextInt();  
int change = 100 - price;  
System.out.println(change);
```


变量

- `int price = 0;`
- 这一行，定义了一个变量。变量的名字是 `price`，类型是 `int`，初始值是 `0`。
- 变量是一个保存数据的地方，当我们需要在程序里保存数据时，比如上面的例子中要记录用户输入的价格，就需要一个变量来保存它。用一个变量保存了数据，它才能参加到后面的计算中，比如计算找零。

变量定义

- 变量定义的一般形式就是：
 - <类型名称> <变量名称>;
- `int price;`
- `int amount;`
- `int price, amount;`

变量的名字

- 变量需要一个名字，变量的名字是一种“标识符”，意思是它是用来识别这个和那个的不同的名字。
- 标识符有标识符的构造规则。基本的原则是：标识符只能由字母、数字和下划线组成，数字不可以出现在第一个位置上，Java的关键字（有的地方叫它们保留字）不可以用做标识符。

Java保留字

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

quiz

- 以下哪些标识符是符合规则的?
 - ☐ currency ☐ lastName ☐ fireplace
 - ☐ _last_name ☐ xingming ☐ class
 - ☐ goto ☐ 4ever ☐ time-machine
 - ☐ Int ☐ \$12 ☐ 我是变量

变量赋值

赋值和初始化

- `int price = 0;`
- 这一行，定义了一个变量。变量的名字是 `price`，类型是 `int`，初始值是 `0`。
- `price=0` 是一个式子，这里的“=”是一个赋值运算符，表示将“=”右边的值赋给左边的变量。

赋值

- 和数学不同， $a=b$ 在数学中表示关系，即a和b的值一样；而在Java中， $a=b$ 表示要求计算机做一个动作：将b的值赋给a。关系是静态的，而动作是动态的。在数学中， $a=b$ 和 $b=a$ 是等价的，而在Java中，两者的意思完全相反

初始化

- 当赋值发生在定义变量的时候，就像程序中的第7行那样，就是变量的初始化。虽然Java语言并没有强制要求所有的变量都在定义的地方做初始化，但是要求所有的变量在第一次被使用（出现在赋值运算符的右边）之前被赋值

变量初始化

- `<类型名称> <变量名称> = <初始值>;`
 - `int price = 0;`
 - `int amount = 100;`
- 组合变量定义的时候，也可以在这个定义中单独给单个变量赋初值，如：
 - `int price = 0, amount = 100;`

插入

- 写出以下变量的初始值：
 - `int price, amount = 100, age;`

读整数

- `price = in.nextInt();`

如果用户输入的不是整数会怎样?

- 要求in这个对象做读入下一个整数的动作，读到的结果赋值给变量price

表达式

- “=”是赋值运算符，有运算符的式子就叫做表达式。
- `price=0;`
- `change=100-price;`

try

- 给程序增加一个int类型的变量：bill，初始化为100。在计算change时，用变量bill代替100。
- 这个变量在哪里定义合适呢？

变量类型

- `int price = 0;`
- 这一行，定义了一个变量。变量的名字是 `price`，类型是 `int`，初始值是 `0`。
- Java是一种强类型语言，所有的变量在使用之前必须定义或声明，所有的变量必须具有确定的数据类型。数据类型表示在变量中可以存放什么样的数据，变量中只能存放指定类型的数据，程序运行过程中也

常量

- `int change = 100 - price;`
- 固定不变的数，是常数。直接写在程序里，我们称作直接量（literal）。
- 更好的方式，是定义一个常量：
 - `final int AMOUNT = 100;`

final

- final是一个修饰符，加在int的前面，用来给这个变量加上一个final（最终的）的属性。这个final的属性表示这个变量的值一旦初始化，就不能再修改了。
- `int change = AMOUNT - price;`
- 如果你试图对常量做修改，把它放在赋值运算符的左边，就会被Java发现，指出为

try

- 让用户输入变量bill的值，而不是使用固定的初始值

tips

- 程序要求读入多个数字时，可以在一行输入，中间用空格分开，也可以在多行输入
- 每次召唤`in.nextInt()`，它就等待用户输入一个整数