

python中变量的命名规则如下：

变量名必须是(大小写英文、数字和下划线_)的组合，
且不能以(数字)为开头。

使用 `()`函数将字符串转换为整数，语法如下：

`int(x)`

使用 `()`函数将数值转换为字符串，语法如下：

`str(x)`

输入函数

input()

Alt+P

以车牌中最后一位阿拉伯数字的奇、偶来决定单双号。

1) 输入: AD2367

2) 输入: F5678A

首先, 取车牌号的第6位, 判断是否数字, 否取第5位。

其次, 判断奇偶。

字符串

定义：它是一个有序的字符的集合，用于存储和表示基本的文本信息，' '或“ ”或“ ” ” ” 间包含的内容称之为字符串。

字符串可以用+号连接起来，用*号重复：

```
>>> word = 'F5678 ' + 'A'
```

```
>>> word
```

字符串可以被截取(检索)。

如果想取出部分字符，那么可以通过下标的方法，
(注意python中下标从 0 开始)

下标索引

所谓“下标”，就是编号，就好比超市中的存储柜的编号，通过这个编号就能找到相应的存储空间。

取1个字符

+	-	-	+	-	-	+	-	-	+	-	-	+	-	-	+	-	-	+
	F		5		6		7		8		A							
+	-	-	+	-	-	+	-	-	+	-	-	+	-	-	+	-	-	+
0	1		2		3		4		5									
-6	-5		-4		-3		-2		-1									

```
>>> word = ' F5678A '
```

```
>>> word[0]
```

```
'F'
```

```
>>> word[5]
```

```
'A'
```

```
>>> word[-1]
```

字符串切片是指对操作的对象**截取一部分**的操作。

切片的语法：**[起始:结束:步长]**

注意：从“起始”位开始，到“结束”位的**前一位**结束（不包含结束位本身）。

str[x:y]表示，
从索引**x**开始取，直到索引**y**为止，但不包括索引**y**。

```
>>> 'ABCDEFGH'[:3]
```

```
'ABC'
```

```
>>> 'ABCDEFGH'[-3:]
```

```
'EFG'
```

```
>>> 'ABCDEFGH'[:2]
```

```
'ACEG'
```

```
>>> 'ABCDEFGH'[::-1]
```

```
>>> word[1:5]
```


运算符

在Python程序设计语言中对常量或变量进行运算或处理的符号称为运算符，参与运算的对象称为操作数。操作数可以是常量、变量、函数等。

表达式

由运算符、操作数连接而成的式子称为表达式。

Python语言支持以下类型的运算符:

算术运算符

比较（关系）运算符

赋值运算符

逻辑运算符

位运算符

成员运算符

身份运算符

运算符优先级

算术运算符

a=10, b=20

运算符	描述	实例
+	加 - 两个对象相加	a + b 输出结果 30
-	减 - 得到负数或是一个数减去另一个数	a - b 输出结果 -10
*	乘 - 两个数相乘或是返回一个被重复若干次的字符串	a * b 输出结果 200
/	除 - x除以y	7 / 3 输出结果
%	取模 - 返回除法的 余数	7 % 3 输出结果
//	取整除 - 返回商的 整数 部分（向下取整）	7//3 输出结果
**	幂 - 返回x的y次 幂	a**b 为 10 的 20 次方， 输出结果 100000000000000000000

比较（关系）运算符

a=10, b=20

运算符	描述	实例
==	等于 - 比较对象是否相等	(a == b) 返回 False 。
!=	不等于 - 比较两个对象是否不相等	(a != b) 返回 true 。
>	大于 - 返回x是否大于y	(a > b) 返回 False 。
<	小于 - 返回x是否小于y。所有比较运算符返回1表示真，返回0表示假。这分别与特殊的变量 True 和 False 等价。	(a < b) 返回 true 。
>=	大于等于 - 返回x是否大于等于y。	(a >= b) 返回 False 。
<=	小于等于 - 返回x是否小于等于y。	(a <= b) 返回 true 。

逻辑运算符

True False a=10, b=20

1 0

2+True

运算符	逻辑表达式	描述	实例
and	x and y	布尔"与" - 如果 x 为 False ，x and y 返回 False ，否则它返回 y 的计算值。	(a and b) 返回 20。
or	x or y	布尔"或" - 如果 x 是非 0，它返回 x 的值，否则它返回 y 的计算值。	(a or b) 返回 10。
not	not x	布尔"非" - 如果 x 为 True ，返回 False 。如果 x 为 False ，它返回 True 。	not(a and b) 返回 False

赋值运算符

赋值号左边只能是变量，不能是常量或表达式。

运算符	描述	实例
=	简单的赋值运算符	c = a + b 将 a + b 的运算结果赋值为 c
+=	加法赋值运算符	c += a 等效于 c = c + a
-=	减法赋值运算符	c -= a 等效于 c = c - a
*=	乘法赋值运算符	c *= a 等效于 c = c * a
/=	除法赋值运算符	c /= a 等效于 c = c / a
%=	取模赋值运算符	c %= a 等效于 c = c % a
**=	幂赋值运算符	c **= a 等效于 c = c ** a
//=	取整除赋值运算符	c //= a 等效于 c = c // a

成员运算符

2+True

运算符	描述	实例
in	x in y 如果 x 是序列 y 的成员， 则计算结果为 True ， 否则返回 False 。	3 in [1,2,3,4]
not in	x not in y 如果 x 不是序列 y 的成员， 则计算结果为 True ， 否则返回 False 。	3 not in [1,2,3,4]

运算符优先级

运算符	描述
**	指数 (最高优先级)
~ + -	按位翻转, 一元加号和减号 (最后两个的方法名为 +@ 和 -@)
* / % //	乘, 除, 取模和取整除
+ -	加法减法
>> <<	右移, 左移运算符
&	位 'AND'
^ 	位运算符
> >= < <=	比较运算符
== !=	等于运算符
= %= /= //= -= += *= **=	赋值运算符
is is not	身份运算符
in not in	成员运算符
not and or	逻辑运算符

字符串切片

[起始:结束:步长]

运算符

幂运算>>

正负号>>

*, /, %>>+, ->>

关系运算符>>

not、and、or

$n \% 3 \neq 0$ 表示的意思为 ()

A n能被3整除

☒ B n不能被3整除

C 百分数n精确到小数点后3位

下列表达式中，值为字符串类型的是 ()。

(1) str(10) (2) chr(65) (3) "123"+45 (4) input() (5) [1,3,4,'A']

A (1)(3)(5)

B (2)(3)(5)

C (2)(4)(5)

☒ D (1)(2)(4)

给定一个四位数的年份，判断它是否为闰年：四年一闰、百年不闰；四百年又闰（年份能被4整除但不能被100整除是闰年，或者年份能被400整除也是闰年。例如2004年和2000年都是闰年，2100年不是闰年）。下列判断年份。下列判断年份y是闰年的表达式正确的是（ ）。

A $y \% 4 = 0$ or $y \% 100 \neq 0$ and $y \% 400 = 0$

B $y \% 4 = 0$

☒ C $y \% 4 = 0$ and $y \% 100 \neq 0$ or $y \% 400 = 0$

D $y \% 400 = 0$

将数学表达式转化为Python表达式，下列选项正确的是（ ）

$$\frac{1}{2}gt^2$$

图 1

A $1/2g*t^2$

B $1/2gt^2$

C $1/2g*t^2$

D $1/2*g*t^2$



表示“x 属于区间[a,b)”的正确表达式是（ ）。

- A $a \leq x$ and $x < b$
- B $a \leq x$ or $x < b$
- C $a \leq x$ and $x < b$
- D $a \leq x < b$

以车牌中最后一位阿拉伯数字的奇、偶来决定单双号。

1) 输入: AD2367

2) 输入: F5678A

首先, 取车牌号的第6位, 判断是否数字, 否取第5位。

str.isdigit()

```
x=input()
```

```
F5678A
```

```
x[5].isdigit()
```

```
x[4].isdigit()
```

其次, 判断奇偶。

```
y=int(x[4])
```

```
y%2
```

偶数

拓 展

字符串里面的%是一个普通字符
怎么办？这个时候就需要转义，
用%%来表示一个%：

```
>>> 'growth rate: %d %%' % 7
```

```
'growth rate: 7 %'
```

len()函数计算的是**str**的字符数，
如果换成**bytes**，**len()**函数就计算字节数：

```
>>> len('中文'.encode('utf-8'))
```

6