

//	整除	9 // 4 输出结果为2
----	----	---------------

**	指数	2 ** 4 输出结果为 16, 即 2 * 2 * 2 * 2
----	----	----------------------------------

```
Python Console x
C:\Users\黑马程序员\AppData\Local\Programs\Python\Python37\python3.exe "C:\Program Files\J
import sys; print('Python %s on %s' % (sys.version, sys.platform))
sys.path.extend(['C:\\Users\\黑马程序员\\Desktop\\code', 'C:/Users/黑马程序员/Desktop/code']

PyDev console: starting.

Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit (AMD64)]
>>> 1 + 1
2
>>> 1 + 1.1
2.1
>>> 1 - 1
0
>>> 1 - 0.5
0.5
>>> 2 * 3
6
>>> 2 * 0.5
1.0
>>> 4 / 2
2.0
>>>
```

在C中，除法是有一些特殊的。

如果是两个整数型的变量相除，如 `10/3`，其结果是 `3`，而不是 `3.3333333`；又如 `10/4`，其在C中的结果是 `2`，而不是 `2.5`。也就是说，整数型变量相除，如果其结果从数学角度上看是小数，无论是否可以四舍五入进位，C都会讲小数点后的部分抛弃。

只有将初始化变量为 **浮点数** 类型，其在C中相除的结果才会具有小数。

此外，如 `double a=5/4;`，其数学计算的结果是 `1.25`，但是在C中输出的结果是 `1`，即使是在语句中初始化变量 `a` 为浮点数。

这是因为初始化的变量是 `a`，而不是算子 `5` 或 `4`，这两个数的类型依旧是整数型，其结果依旧遵循整数型变量相除的结果，只是其结果 `1` 被赋值给了一个浮点数变量 `a`。

如果想对两个浮点数相除的结果取整，即只取其整数部分，可以像这样做：

```
1 | int centimeter = 35;
2 | int foot = centimeter / (100 * 0.3048);
```

利用 `int` 可以取整。