

العمليات الحسابية عندنا في البايثون يعني مش محتاجه كلام عنها كثير عندنا العمليات العادية، يمكن اللي محتاجين نعرفه ان الأس بيتعمل عبارته عن علامتين ضرب (**).

ال (`floor division(//)`) هي عملية بتجيب الناتج بتاع اقرب رقم اقل من الرقم بتاعي اللي بيقبل القسمة على المقسوم عليه بحيث تطلعلي رقم صحيح، كلام ملغبط بس بمعنى اصح لو انا بقسم 119 على 20 باستخدام ال `floor division` بيأخد ال 119 ويقعد يقلل فيها لحد م يجيب رقم بيقبل القسمة على 20 وفي الحالة دي اللي هي 100 وبديني الناتج بتاع قسمة 100 على 20 والكود مبيكدش.

```
# -----
# -- Arithmetic Operators --
# -----
# [+] Addition
# [-] Subtraction
# [*] Multiplication
# [/] Division
# [%] Modulus
# [**] Exponent
# [//] Floor Division
# -----

# Addition

print(10 + 30) # 40
print(-10 + 20) # 10
print(1 + 2.66) # 3.66
print(1.2 + 1.2) # 2.4

# Subtraction

print(60 - 30) # 30
print(-30 - 20) # -50
print(-30 - -20) # -10
print(5.66 - 3.44) # 2.22

# Multiplication

print(10 * 3) # 30
print(5 + 10 * 100) # 1005
print((5 + 10) * 100) # 1500

# Division

print(100 / 20) # 5.0
print(int(100 / 20)) # 5

# Modulus

print(8 % 2) # 0
print(9 % 2) # 1
```

```
print(20 % 5)  # 0
print(22 % 5)  # 2

# Exponent

print(2 ** 5)  # 32
print(2 * 2 * 2 * 2 * 2)  # 32
print(5 ** 4)  # 625
print(5 * 5 * 5 * 5)  # 625

# Floor Division

print(100 // 20)  # 5
print(119 // 20)  # 5
print(120 // 20)  # 6
print(140 // 20)  # 7
print(142 // 20)  # 7
```

CODE

```
PS C:\Users\Muhammad> & python "c:/Users/Muhammad/Documents/Python Course/first.py"
40
10
3.66
2.4
30
-50
-10
2.22
30
1005
1500
5.0
5
0
1
0
2
32
32
625
625
5
5
6
7
7
PS C:\Users\Muhammad> █
```

OUTPUT