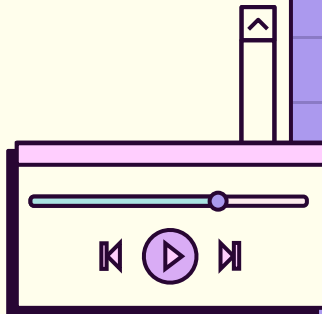




Python

if/else, loops, functions.

Leen Mohammed Alsaleh



0. Positive anything is better than negative nothing

يعطيك البرنامج رقم عشوائي موجب او سالب او صفر في كل مره
هنا معطينا كود جاهز ونكملة ومحدد لنا الرقم :

```
1  #!/usr/bin/python3
2  import random
3  number = random.randint(-10, 10)
4  # YOUR CODE HERE
```

What is the output based on
this question?

-10 / -1 is negative

0 is zero

1 / 10 is positive

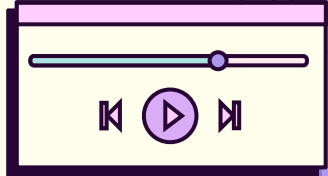


X is positive ← : لو كان موجب :
X is Negative ← : لو كان سالب :
X is Zero ← : لو كان صفر :

```
#!/usr/bin/python3
import random

number = random.randint(-10, 10)

if number > 0:
    print(f"{number} is positive")
elif number == 0:
    print(f"{number} is zero")
else:
    print(f"{number} is negative")
```



1. The last digit

فكرة السؤال يعطينا كود جاهز نكملة يحتوي على ارقام عشوائيه في كل مره نشغله بيغانا نطبع اخر رقم :

```
1  #!/usr/bin/python3
2  import random
3  number = random.randint(-10000, 10000)
4  # YOUR CODE HERE
```

لو كان الرقم اكبر من 5 : ← Is greater than 5

لو كان يساوي صفر : ← Is 0

لو كان اقل من 6 وماهو صفر : ← less than 6 and not 0

لو طلع لي الرقم :

number = -626

راح نحوله لموجب بعددين نرجعه سالب

بهذي الطريقة :

$-(626 \% 10) = -6$

Last digit of -626 is -6

and is less than 6 and 0

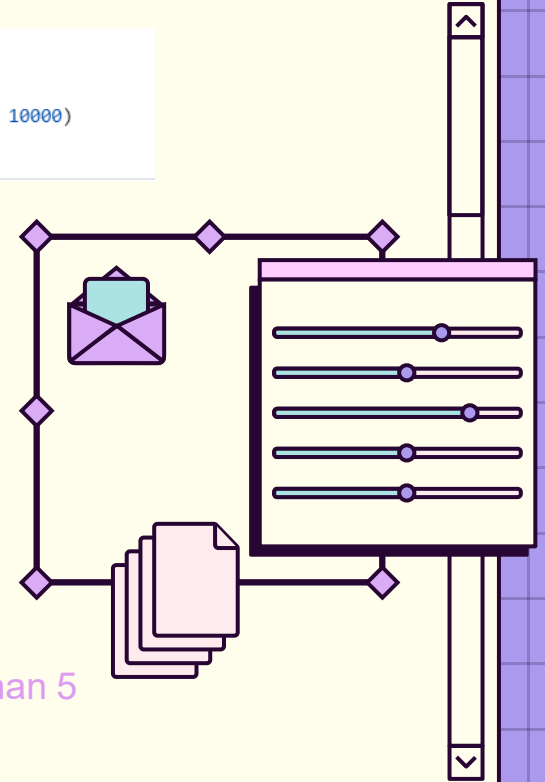
لو طلع لي الرقم :

number = 7589

وش اخر رقم ؟؟

اخر رقم هو 9 و 9 اكبر من 5 فا نطبع :

Last digit of 7589 is 9 and is greater than 5



2. I sometimes suffer from insomnia. And when I can't fall asleep, I play what I call the alphabet game

بيغانا نطبع حروف صغيره من

a to z

بدون مسافات وبدون سطر جديد

```
#!/usr/bin/python3
```

```
print("{}".format(".join(chr(i) for i in range(ord('a'), ord('z')+1))), end="")
```

- `ord('a')` يعطيك الكود الرقمي لحرف `a` يطلع 97
- `ord('z')` يعطيك الكود الرقمي لحرف `z` يطلع 122
- `range(123, 97)` معناها: من 97 إلى 122 (يعني من `a` إلى `z`)
- ياخذ كل الحروف اللي طلعت (`a` إلى `z`) ويجمعها في سلسلة وحدة بدون فواصل

`range(ord('a'), ord('z')+1)`

`join`

- كل رقم `i` راح يتحول لحرف باستخدام `chr(i)`
- يعني: 97 يصير `'a'`، 98 يصير `'b'`، وهكذا...

`chr(i) for i in range`



3. When I was having that alphabet soup, I never thought that it would pay off

يطبع كل الحروف الصغيره من:

a-z

: ماعدا

q and e

```
#!/usr/bin/python3
```

```
print("{}".format(''.join(chr(i) for i in range(ord('a'), ord('z')+1)  
if chr(i) not in ('e', 'q'))), end='')
```



4. Hexadecimal printing

راح يطبع الأرقام من 0 الى 98 بصيغتين :

Hexadecimal
الست عشريه

Decimal
العشريه

```
#!/usr/bin/python3
```

```
for i in range(99):  
    print("{} = 0x{:x}".format(i, i))
```

The output

0 = 0x0

1 = 0x1

2 = 0x2

...

10 = 0xa

11 = 0xb

...

15 = 0xf

16 = 0x10

...

98 = 0x62

• "{}" = نطبع الرقم كأنه عادي.

• "0x{:x}" = نطبع الرقم بنظام العد الست عشري (hex) باستخدام الحرف x الصغير (عشان يطبع الحروف a-f بحروف صغيرة).

• x0 هذي ثابتة، تعني إن الرقم بعدها مكتوب بصيغة hex.

5.00 ..99

يطبع الأرقام من 00 الى 99 مفصولة ب ,

```
#!/usr/bin/python3
```

```
for i in range(100):  
    if i < 99:  
        print("{:02d}, ".format(i), end="")  
    else:  
        print("{:02d}".format(i))
```

و ش يعني `{:02d}` ؟

خل الرقم يكون عرضه خانتين، وإذا
كان أصغر من 10، عبي بصفر قدامه

: بداية التنسيق

0 عبي الفراغات بصفر

2 العرض = 2 خانات

d يعني "decimal" (رقم

عشري)

اطبع الرقم i بصيغة رقم من رقمين لو كان أقل من 10، يضيف صفر قبله
أمثلة:

00 < --- 0

05 < --- 5

12 < --- 12

بعد الرقم، اطبع فاصلة ومسافة ,

The output: 00, 01, 02, 03, ..., 97, 98, 99

6. Inventing is a combination of brains and materials. The more brains you use, the less material you need

تخيلو لو عندنا رقمين نبغى نطبعهم مع بعض بشرط :
الرقمين مايكونون نفس الشئ زي : 00 و 11

ولا ينفع نطبع الرقم ومعكوسه زي : 10 و 01 لازم
نطبع واحد بس

نستخدم **nested loop**
واحد يمسك الرقم الأول i
والثاني يمسك الرقم الثاني j

لو كانت
 $i=3$

$j=4$

•نضمن إن j دائماً أكبر من i عشان نتجنب التكرار والعكس (يعني ما نطبع 21 بعد ما طبعنا 12).

6. Inventing is a combination of brains and materials. The more brains you use, the less material you need

The output:

89 ,79 ,78 ,... ,13 ,12 ,... ,03 ,02 ,01

اللوبي الأول :

```
for i in range(10):
```

يمسك الرقم الأول من 0 إلى 9.

اللوبي الثاني :

```
for j in range (i+1 , 10) :
```

يمسك الرقم الثاني ويبدأ من i+1 عشان يكون اكبر من الرقم الأول ومايتكرر

إذا وصلنا لرقمين الأخير (8 و 9):

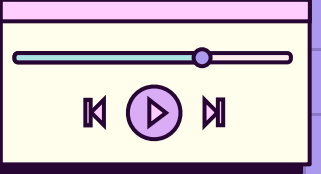
```
if i == 8 and j == 9:
```

```
print("{}{}".format(i, j))
```

نطبع 89 بدون فاصلة بعدها

```
print("{}{}", ".format(i, j), end=""
```

نطبع الرقمين مع فاصله ومسافه ونخليهم في نفس السطر



7. islower

نسوي داله اسمها
Islower(c)
وتشيك اذا الحرف هو صغير ولا لا ؟

اذا كان
الحرف
صغير

TRUE

اذا كان
الحرف
كبير

FALSE

```
return ord('a') <= ord(c) <= ord('z')
```

ببساطه اذا كان الحرف ما بين 98 و 122 راجع
True

Islower ('a') ?

TRUE

Islower ('Z') ?

FALSE

Islower ('g') ?

TRUE

Islower ('3') ?

FALSE



8. To uppercase

تحول الحروف الصغيره الى
الحروف الكبيره وتطبعها
ورا بعض على نفس السطر

Ord(c)-32

← ex

الفرق دايم بين أي حرف كبير و
الصغير هو 32
لذلك لما نجي نحول الحرف
الصغير للكبير نطرح -32

```
if ord('a') <= ord(char) <= ord('z'):  
    uppercase_char = chr(ord(char) - 32)
```

9. There are only 3 colors, 10 digits, and 7 notes; it's what we do with them that's important

المطلوب نسوي داله اسمها
`print_last_digit(number):`

تأخذ رقم سواء موجب او سالب
تطبع اخر رقم

`Print_last_digit(98)?`
8

`Print_last_digit(0)?`
0

`Print_last_digit(-1024)?`
4

```
#!/usr/bin/python3
```

```
def print_last_digit(number):  
    number = abs(number)  
    last_digit = number % 10  
    print(last_digit, end="")  
    return last_digit
```



10. $a + b$

مطلوب نسوي داله تستقبل
رقمين وترجع ناتج جمعهم

```
def add(a, b):  
    return a + b
```

Print (add(1,2))

3

Print (add(98,0))

98

Print (add(100,-2))

98



11. a^b

مطلوب نسوي داله تستقبل
رقمين وترجع ناتج الرقم
الأول اس الرقم الثاني

```
def pow (a, b)  
return a ** b
```

Print (pow(2,2))

4

Print (pow(98,2))

982

Print (pow(100,-2))

0.0001

Print (pow(-4,5))

-1024

إذا عندك رقم سالب
 $a^{(-b)}$
يساوي
 $a^{(b)/1}$

$$100^{-2} = 100^{2/1}$$



12. Fizz Buzz

مطلوب نسوي داله تطبع
الأرقام من 1 الى 100

إذا الرقم يقبل القسمة 3
نطبع :
Fizz

إذا الرقم يقبل القسمة 5
نطبع :
Buzz

إذا الرقم يقبل القسمة 3 و 5
نطبع :
Fizz
Buzz

```
#!/usr/bin/python3
```

```
def fizzbuzz():
```

```
for i in range(1, 101):
```

```
    if i % 3 == 0 and i % 5 == 0:
```

```
        print("FizzBuzz", end=" ")
```

```
    elif i % 3 == 0:
```

```
        print("Fizz", end=" ")
```

```
    elif i % 5 == 0:
```

```
        print("Buzz", end=" ")
```

```
    else:
```

```
        print(i, end=" ")
```





harry up

speed test



Print ("ضاقت فلما استحكمت حلقاتها فرجت وكنت اظنها لاتفرج ")

Print ("Hello ")

Print (" just PYTHON ")

Print (" Go GO ")





Thank you