

**16.- ¿Qué resultados se muestran al evaluar estas expresiones?****>>> True == True != False**

```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep  5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> True == True != False
True
>>>
```

**>>> 1 < 2 < 3 < 4 < 5**

```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep  5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 1 < 2 < 3 < 4 < 5
True
>>>
```

**>>> (1 < 2 < 3) and (4 < 5)**

```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep  5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> (1 < 2 < 3) and (4 < 5)
True
>>> |
```

>>> 1 < 2 < 4 < 3 < 5

```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep  5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 1 < 2 < 4 < 3 < 5
False
>>>
```

>>> (1 < 2 < 4) and (3 < 5)

```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep  5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> (1 < 2 < 4) and (3 < 5)
True
>>> |
```

## 17.- Evalúa estas expresiones:

a) 0xf + 0o17 + 0b1111 + 15

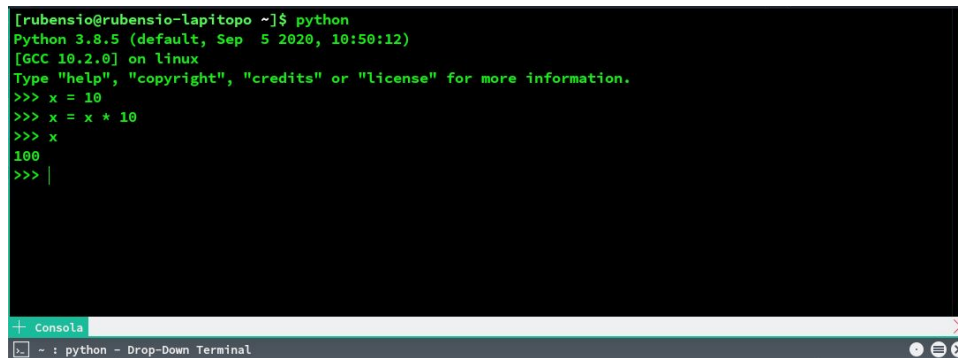
```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep  5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 0xf + 0o17 + 0b1111 + 15
60
>>>
```

b) 0xffff + 0b1

```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep  5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 0xffff + 0b1
65536
>>> |
```

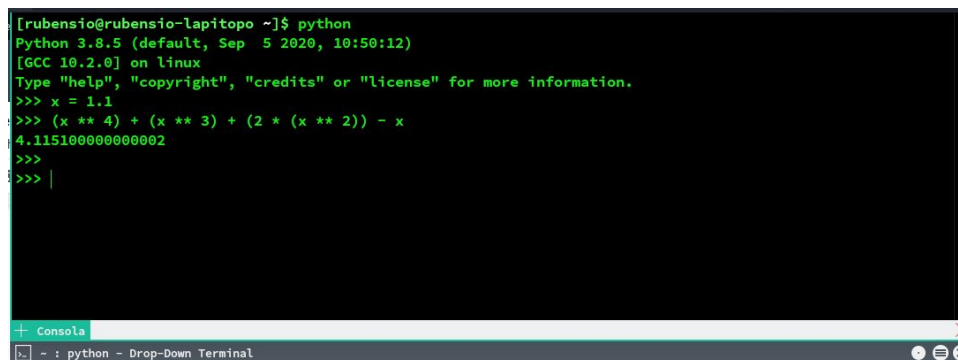
19.- ¿Qué resulta de ejecutar estas tres líneas?

```
>>> x = 10
>>> x = x * 10
>>> x
```



```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep 5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> x = 10
>>> x = x * 10
>>> x
100
>>> |
```

20.- Evalúa el polinomio  $x^4 + x^3 + 2x^2 - x$  en  $x = 1.1$ . Utiliza variables para evitar teclear varias veces el valor de  $x$ .



```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep 5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> x = 1.1
>>> (x ** 4) + (x ** 3) + (2 * (x ** 2)) - x
4.115100000000002
>>> |
```

21.- Evalúa el polinomio  $x^4 + x^3 + 1/2x^2 - x$  en  $x = 10$ .

```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep 5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> x = 10
>>> (x ** 4) + (x ** 3) + ((1/2) * (x ** 2)) - x
11040.0
>>> |
```

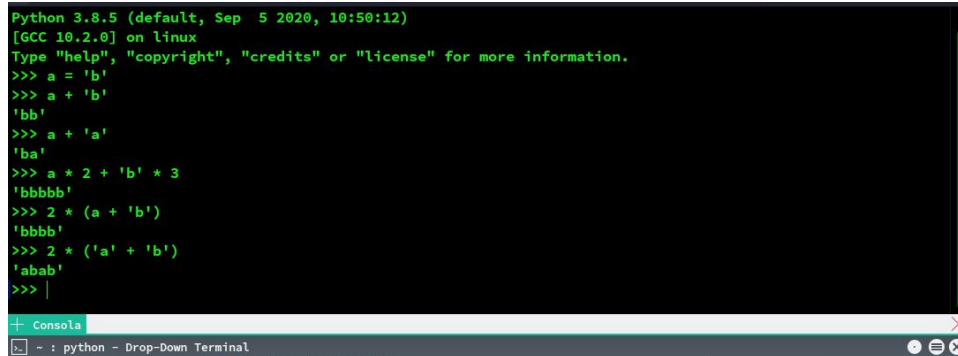
22.- ¿Qué valor tiene z tras evaluar estas sentencias?

```
>>> z = 2
>>> z += 2
>>> z += 2 - 2
>>> z *= 2
>>> z *= 1 + 1
>>> z //= 2
>>> z %= 3
>>> z /= 3 - 1
>>> z -= 2 + 1
>>> z -= 2
>>> z **= 3
>>> z
```

```
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> z = 2
>>> z += 2
>>> z += 2 - 2
>>> z *= 2
>>> z *= 1 + 1
>>> z //= 2
>>> z %= 3
>>> z /= 3 - 1
>>> z -= 2 + 1
>>> z -= 2
>>> z **= 3
>>> z
-64.0
>>> |
```

23.- Evalúa estas expresiones y sentencias en el mismo orden en el que aparecen e indica lo que muestra el intérprete de Python como respuesta

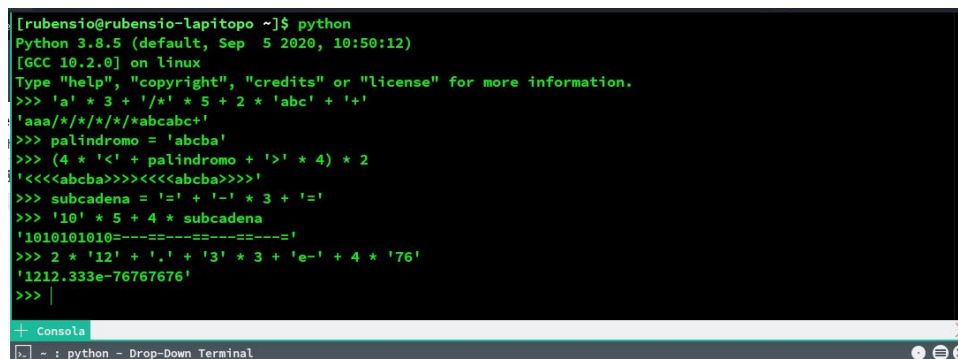
```
>>> a = 'b'
>>> a + 'b'
>>> a + 'a'
>>> a * 2 + 'b' * 3
>>> 2 * (a + 'b')
>>> 2 * ('a' + 'b')
```



```
Python 3.8.5 (default, Sep 5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> a = 'b'
>>> a + 'b'
'bb'
>>> a + 'a'
'ba'
>>> a * 2 + 'b' * 3
'bbbbb'
>>> 2 * (a + 'b')
'bbbb'
>>> 2 * ('a' + 'b')
'abab'
>>> |
```

24.- ¿Qué resultados obtendrán al evaluar las siguientes expresiones y asignaciones Python? Calcula primero a mano el valor resultante de cada expresión y comprueba, con la ayuda del ordenador, si tu resultado es correcto.

```
>>> "a" * 3 + "/" * 5 + 2 * "abc" + "+"
>>> palindromo = "abcba"
>>> (4 * "<" + palindromo + ">" * 4) * 2
>>> subcadena = "=" + "-" * 3 + "="
>>> "10" * 5 + 4 * subcadena
>>> 2 * "12" + "." + "3" * 3 + "e-" + 4 * "76"
```



```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep 5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 'a' * 3 + '/' * 5 + 2 * 'abc' + '+'
'aaa//*/*/*/abcabc+'
>>> palindromo = 'abcba'
>>> (4 * '<' + palindromo + '>' * 4) * 2
'<<<<abcba>>>><<<<abcba>>>>'
>>> subcadena = '=' + '-' * 3 + '='
>>> '10' * 5 + 4 * subcadena
'1010101010=====!'
>>> 2 * '12' + '.' + '3' * 3 + 'e-' + 4 * '76'
'1212.333e-76767676'
>>> |
```

26.- Calcula con una única expresión el valor abs del redondeo de -3.2.

```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep  5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> abs(round(-3.2))
3
>>> |
```

27.- Convierte (en una única expresión) a una cadena el resultado de la división 5011/10000 redondeado con 3 decimales

```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep  5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> str(round(5011 / 10000))
'1'
>>> |
```

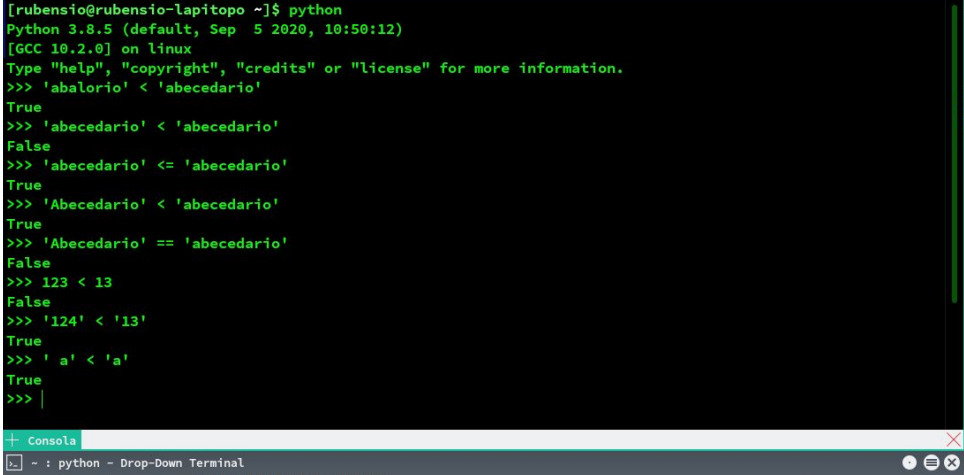
28.- ¿Qué resulta de evaluar estas expresiones?

```
>>> str(2.1) + str(1.2)
>>> int(str(2) + str(3))
>>> str(int(12.3)) + '0'
>>> int('2' + '3')
>>> str(2 + 3)
>>> str(int(2.1) + float(3))
```

```
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> str(2.1) + str(1.2)
'2.11.2'
>>> int(str(2) + str(3))
23
>>> str(int(12.3)) + '0'
'120'
>>> int('2' + '3')
23
>>> str(2 + 3)
'5'
>>> str(int(2.1) + float(3))
'5.0'
>>> |
```

**29.- ¿Qué resultados se muestran al evaluar estas expresiones?**

```
>>> 'abalorio' < 'abecedario'
>>> 'abecedario' < 'abecedario'
>>> 'abecedario' <= 'abecedario'
>>> 'Abecedario' < 'abecedario'
>>> 'Abecedario' == 'abecedario'
>>> 123 < 13
>>> '124' < '13'
>>> 'a' < 'a'
```



```
[rubensio@rubensio-lapitopo ~]$ python
Python 3.8.5 (default, Sep 5 2020, 10:50:12)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 'abalorio' < 'abecedario'
True
>>> 'abecedario' < 'abecedario'
False
>>> 'abecedario' <= 'abecedario'
True
>>> 'Abecedario' < 'abecedario'
True
>>> 'Abecedario' == 'abecedario'
False
>>> 123 < 13
False
>>> '124' < '13'
False
>>> 'a' < 'a'
True
>>> |
```