

EJERCICIOS DE INTRODUCCIÓN A PYTHON PARTE I

TALLER

1.1. INTRODUCCIÓN

El propósito de esta actividad es afianzar los conocimientos adquiridos durante la unidad: "Introducción a Python".

1.2. HERRAMIENTAS

Tiempo estimado: 1.00 h.

REQUISITOS	
Técnicos	<ul style="list-style-type: none">■ Disponer de Jupyter Notebook.■ Buenas prácticas.
OBJETIVOS	
<ul style="list-style-type: none">■ Afianzar los conocimientos aprendidos de Python.	
MATERIAL Y HERRAMIENTAS NECESARIAS PARA EL TALLER	
<ul style="list-style-type: none">■ Tener la instalación de Anaconda (con una versión de Python 3.4 o superior)	

1.3. INSTRUCCIONES PARA REALIZAR LOS EJERCICIOS

Intenta no ayudarte del temario para la realización de estos ejercicios. Si finalmente necesitas repasarlo para terminarlos te recomiendo que los repitas cada día hasta que ya no te haga falta utilizar el temario. Así interiorizarás las bases de Python, así como su sintaxis.

Puedes utilizar un Notebook de Jupyter para realizar los ejercicios. Recuerda que estos ejercicios son autocorregibles y que una vez los termines, en el campus puedes pulsar sobre la opción “marcar para terminar” para que se actualice la página y así te aparezca el documento con las soluciones.

Las soluciones se entregarán como un archivo .ipynb (notebook de Jupyter)

1. Crea un comentario.

```
# Soy un comentario breve
'''
Soy un comentario que puede tener varias líneas
'''
```

2. Realiza las siguientes operaciones utilizando números enteros: Una suma, una resta y una multiplicación.

```
# Suma de dos números enteros
7 + 11
18
```

```
# Resta de dos números enteros
4 - 8
-4
```

```
# Multiplicación de dos números enteros
3 * 5
15
```

3. Realiza una división de números enteros cuyo resultado sea un entero.

```
# División cuyo resultado sea un entero
17 // 3
5
```

4. Calcula la potencia de un número entero.

```
# Potencia de un número entero
2 ** 3 # Dos elevado al cubo
8
```

```
# Potencia de un número decimal
2.2 ** 3
10.648000000000003
```

5. Realiza las siguientes operaciones utilizando números decimales o enteros y decimales: Una suma, una resta, una multiplicación y una división.

```
# Suma de decimales  
2.1 + 3
```

5.1

```
# Resta de decimales  
5.4 - 2
```

3.4000000000000004

```
# Multiplicación con decimales  
3.4 * 2.1
```

7.14

6. Calcula el valor absoluto de dos números enteros y de un número decimal.

```
# Valor absoluto de un número decimal  
abs(-3.4)
```

3.4

```
# Valor absoluto de un número entero  
abs(-4)
```

4

```
# Valor absoluto de otro número entero  
abs(7)
```

7

7. Calcula el resto de una división.

```
# Resto de dos números  
13 % 2
```

1

8. Redondea un número decimal para que sólo tenga cifra decimal. Repite el ejercicio para que tenga hasta dos cifras decimales.

```
# Una cifra decimal  
round(3.141592, 1)
```

3.1

```
# 2 cifras decimales  
round(3.141592, 2)
```

3.14

9. Realiza las siguientes operaciones de comparación utilizando números enteros o decimales: Mayor que; menor que; igual que; no igual que; igual o mayor que; igual o menor que

```
# Mayor que
```

```
3.7 > 5
```

```
False
```

```
# Menor que
```

```
3.7 < 5
```

```
True
```

```
# Igualdad
```

```
3.7 == 5
```

```
False
```

```
# No igual que
```

```
3.7 != 5
```

```
True
```

```
# Menor o igual que
```

```
3.7 <= 5
```

```
True
```

```
# Mayor o igual que
```

```
3.7 >= 5
```

```
False
```

10. Define una variable y una constante.

```
# Definimos una variable
```

```
x = 4
```

```
# Definimos una constante
```

```
PI = 3,1415926535897
```

```
'''
```

```
Recuerda que en Python las constantes no existen como tal, son variables cuyo valor SI puede cambiar.  
Cuando definimos una variable en mayúsculas los programadores la trataremos como una constante cuyo valor  
no debería poder cambiar
```

```
'''
```

11. Crea dos variables y comprueba si alguna de ellas tiene el valor "None" asignado.

```
estudiante = None  
profesor = "Patricio Martinez"
```

```
profesor is None
```

```
True
```

```
estudiante is None
```

```
True
```

12. Define e instancia dos variables e intercambia sus valores sin utilizar una variable intermedia (o auxiliar).

```
color1 = "rojo"
color2 = "amarillo"

color1, color2 = color2, color1
print("El color 1 ahora es: ", color1, "\nEl color 2 ahora es: ", color2)

El color 1 ahora es:  amarillo
El color 2 ahora es:  rojo
```

13. Crea dos variables cuyo valor sea verdadero y falso. Utiliza las variables para realizar las operaciones lógicas habituales (AND y OR).

```
tengo_gato = True
tengo_perro = False

# Tener ambos animales sería la (Y) lógica
tengo_gato and tengo_perro

False

# Tener al menos uno de los dos animales sería la (O) lógica
tengo_gato or tengo_perro

True
```