

## Guía de Ejercicios 1: Tipos de datos, expresiones y asignaciones

### Objetivos:

- Ejercitar el manejo de valores y expresiones de los tipos `int`, `float` y `str`.
  - Entender cómo funcionan las conversiones de tipo, tanto implícitas como explícitas.
  - Advertir sobre los errores de representación de números reales en la computadora.
  - Introducir los conceptos de variable, asignación y espacio de memoria de un programa.
- 

### Ejercicio 1. Evaluación de expresiones.

- (a) Para cada una de las siguientes expresiones, determinar cuál es su tipo y evaluarla a mano, realizando las conversiones de tipos que sean necesarias:

- |                               |                                      |
|-------------------------------|--------------------------------------|
| (I) <code>11 + 2</code>       | (VIII) <code>'ho' + 'la'</code>      |
| (II) <code>11 / 2</code>      | (IX) <code>'ja' * 3</code>           |
| (III) <code>11 // 2</code>    | (X) <code>len('hola') + 6</code>     |
| (IV) <code>11 % 2</code>      | (XI) <code>123 + '123'</code>        |
| (V) <code>1 / 0</code>        | (XII) <code>123 + 123.0</code>       |
| (VI) <code>1 % 0</code>       | (XIII) <code>123 + int('123')</code> |
| (VII) <code>11.0 + 2.0</code> | (XIV) <code>str(123) + '123'</code>  |

- (b) Revisar las respuestas del punto anterior, esta vez evaluando las expresiones en una consola `ipython` y averiguando su tipo con la operación `type()`.
- (c) ¿Qué diferencia hay entre los valores y los tipos de las expresiones (I) y (VII)?
- (d) Comparar el resultado de evaluar las expresiones (II) y (III). ¿Cuál símbolo corresponde a la división entera, y cuál a la división de `float`? ¿Qué rol tiene el símbolo `%` respecto de la división entera?
- (e) ¿A qué se deben los errores de las expresiones (V), (VI) y (XI)?
- (f) ¿Qué ocurrió exactamente al evaluar las expresiones (XII), (XIII) y (XIV)?

### Ejercicio 2. Representación de números reales en punto flotante.

- (a) Evaluar las siguientes expresiones, primero a mano y después en una consola `ipython`:

- (I) `(10 / 3 - 3) * 3 - 1`  
(II) `0.00000001 + 1000000000 - 1000000000`

- (b) ¿Los resultados son los esperados? ¿A qué se debe esto?
- (c) En (II), reemplazar el primer valor literal (`0.00000001`) por valores cada vez más grandes (ej.: `0.0000001`, `0.000001`, ..., hasta `0.1`), reevaluar y estudiar los resultados.

**Ejercicio 3.** A partir del string '123.45', escribir expresiones que arrojen como resultado los siguientes valores:

- (a) El float 123.45.
- (b) El entero 123.
- (c) El float 0.45.

**Ejercicio 4.** Cadenas de caracteres: tipo str.

- (a) Escribir el siguiente programa en un archivo `holamundo.py`:

```
1 print("hola mundo")
```

Ejecutar el programa y observar su salida, ya sea con el botón ► en Spyder, o bien con el comando “python holamundo.py” en la consola del sistema operativo (*Terminal, Command Prompt*, etc.).

- (b) Reemplazar el código por el siguiente y observar qué sucede al ejecutarlo.

```
1 print("hola
2 mundo")
```

- (c) Reemplazar el código por el siguiente y observar la salida de su ejecución.

```
1 print("hola\nmundo\nchau\tmundo")
```

- (d) Reemplazar el texto del programa por el siguiente y observar la salida de su ejecución.

```
1 # Esto es un comentario.
2 print("""En Python se pueden
3 escribir strings en varias líneas
4 """)
```

¿Qué se puede concluir de los símbolos `"""` y `#` en Python?

**Ejercicio 5.** En el ejercicio anterior apareció el carácter `\`. Se conoce a la barra invertida como *carácter de escape*, y su función dentro de una cadena de caracteres es darle un significado especial al carácter que figura inmediatamente a continuación. Por ejemplo, `\n` se usa como carácter de nueva línea. Escribir las siguientes instrucciones en la consola interactiva `ipython`:

```
1 print("\n")
2 print("""Cuidado con las "comillas".""")
3 print('Tengamos "mucho cuidado" por favor.')
```

**Ejercicio 6.** ¿Qué valor se imprime en la última instrucción del siguiente programa?

```
1  a:int = 1
2  b:int = a
3  a = 2
4  print(b)
```

¿Por qué no se imprime 2, si en la segunda instrucción indicamos que b valiera lo mismo que a?

**Ejercicio 7.** Completar el siguiente código en los lugares indicados con `...`, de modo que se imprima este texto: 9876 segundos son 2 horas, 44 minutos, 36 segundos.

```
1  n:int = 9876
2  horas:int = ...
3  minutos:int = ...
4  segundos:int = ...
5  mensaje:str = ...(n) + " segundos son " + \
6                  ...(horas) + " horas, " + \
7                  ...(minutos) + " minutos, " + \
8                  ...(segundos) + " segundos."
9  print(mensaje)
```

Observación: si una línea termina con `\`, la instrucción continúa en la siguiente línea. Este programa debe funcionar para cualquier valor inicial de `n`, no solo para 9876. :-)  
De ser necesario, pueden agregarse líneas adicionales de código.

**Ejercicio 8.** Completar el siguiente código, de manera que convierta una temperatura expresada en grados Fahrenheit a grados Celsius.

```
1  fahr:float = 80.0
2  cel:float = ...
3  print(...)
```

Cuando `fahr` se inicializa en 80.0, la salida por pantalla debe ser (aproximadamente):

```
80.0 grados Fahrenheit equivalen a 26.66666667 grados Celsius.
```

Este programa debe funcionar para cualquier valor inicial de `fahr`, no solo para 80. :-)