



Trabajo Práctico III

Variables y Funciones Built-in

Al utilizar variables en Python, tenga en cuenta los siguientes lineamientos:

- Los nombres de variables sólo pueden contener letras, números, y guiones bajos.
- Los nombres de variables deben comenzar con una letra o un guión bajo.
- Los nombres de variables no pueden ser palabras reservadas por el lenguaje Python (puede consultar el listado de palabras reservadas en https://www.w3schools.com/python/python_ref_keywords.asp).
- Es recomendable que los nombres de las variables sean descriptivos. Por ejemplo, "edad" es un mejor nombre de variable que "e".
- Los nombres de las variables se escriben siempre en minúscula (hay excepciones, pero por el momento las ignoraremos).

Puede consultar el listado completo de funciones predefinidas (built-in) de Python en <https://docs.python.org/3/library/functions.html>.

Aclaración importante: si bien los ejercicios listados a continuación tienen varias soluciones posibles, utilice, siempre que sea posible, una alternativa que haga uso de la función built-in de Python pensada para el caso.

1. Cree un script que almacene un número entero en una variable, y luego muestre en pantalla su valor absoluto, con el mensaje "El valor absoluto de *N* es *|N|*". Finalmente, verifique que su programa funciona correctamente, ejecutándolo con el valor 10 en la variable (la salida debería ser 10), y luego con el valor -10 (la salida debería ser 10 nuevamente).
2. Cree un script que almacene su nombre de pila en una variable, y luego muestre en pantalla la cantidad de letras de ese nombre, con el mensaje "El nombre [*NOMBRE*] tiene [*N*] letras."
3. Cree un script que almacene, en dos variables, una *base* y un *exponente*, y luego muestre en pantalla el resultado de elevar el número *base* a la potencia *exponente*.
4. Implemente un algoritmo en Python para calcular el perímetro de un rectángulo, conociendo su base y altura. Los datos se deben almacenar en variables, y el resultado se debe mostrar en pantalla.
$$\text{perímetro} = 2 * (\text{base} + \text{altura})$$
5. Implemente un algoritmo en Python para calcular el área de un rectángulo, conociendo su base y altura. Los datos se deben almacenar en variables, y el resultado se debe mostrar en pantalla.



11071: Introducción a la Programación
Departamento de Ciencias Básicas
Universidad Nacional de Luján

área = base * altura

6. Implemente un algoritmo que intercambie los valores entre dos variables *a* y *b* cualesquiera. Por ejemplo, si *a* = 10 y *b* = 5, luego de ejecutar el algoritmo, la variable *a* debería ser igual 5, y la variable *b* debería ser igual a 10.
7. En Python es posible resolver el problema del intercambio de valores sin hacer uso de variables adicionales, mediante la sintaxis de **asignación múltiple**. Investigue de qué se trata dicha funcionalidad, y utilízela para resolver el ejercicio anterior sin utilizar variables auxiliares/adicionales.
8. Escriba un algoritmo que, conociendo las notas de los dos parciales de un alumno de la asignatura Introducción a la Programación, muestre en pantalla su promedio.
9. Cree un script que, sabiendo cuántos pesos argentinos tiene una persona ahorrada en su cuenta (almacenando ese monto en una variable), muestre en pantalla los montos convertidos en dólares (U\$1 = \$80.5), reales (\$R1 = \$14.1), y euros (€1 = \$69.5). La salida del programa debe tener el siguiente formato:

```
Usted tiene $XXX pesos argentinos, los cuales se convierten en:  
- U$XXX dólares.  
- R$XXX reales.  
- €XXX euros.
```