

Práctico 5 : Modularización

1. Crear una función llamada cuadrado que tome un número como parámetro e imprima en pantalla dicho número al cuadrado.
 - a) Imprimir, desde dentro de la función, las variables locales que ésta posea. ¿Qué se imprime? * *utilizar locals()*
 - b) Agregar dos o tres variables extras con valores inventados, y volver a mostrar todas las variables locales. ¿Hubo cambio alguno?
2. Crear una variable llamada n, que será global, en el código del ejercicio anterior y asignarle un valor inventado. Realizar las siguientes acciones:
 - a) Mostrar el valor de n, elevado al cuadrado desde dentro del procedimiento.
 - b) Modificar el valor de n, dentro del procedimiento. ¿Qué ocurre?
 - c) Si necesitase modificar el valor de n, que es una variable global, dentro del procedimiento cuadrado, ¿Qué debería hacer?
3. ¿Qué imprimirá en pantalla el siguiente código? ¿Cuál es el alcance de la variable frase?

```
frase = "Hola"
```

```
def f():
    frase = "Es un lindo dia"
    print(frase)
```

4. ¿Qué imprimirá en pantalla el siguiente código? Determine el alcance de cada variable.

```
x = 3
```

```
def f():
    y = x + 1
    print(x)

    def g():
        x = 1
        print(y)
        print(x)
    g()
f()
```

5. Crear una función que reciba un número como parámetro el cual representa el lado de un cuadrado y muestre en pantalla el perímetro y la superficie del mismo.
6. Crear una función que reciba como parámetro un número entero, y retorne la tabla de multiplicar de ese número. Imprimir tabla (*por ejemplo, para el 3 deberá mostrar desde $3 \times 0 = 0$ hasta $3 \times 10 = 30$*).
7. Escriba una función denominada *cuadrante(x, y)*, donde *x* e *y* son valores enteros recibidos como parámetros los cuales representan un punto, y que retorne un valor entre 1, 2, 3 o 4 de acuerdo al cuadrante que se encuentre el punto *(x, y)*, ingresado como parámetro, en los ejes cartesianos.

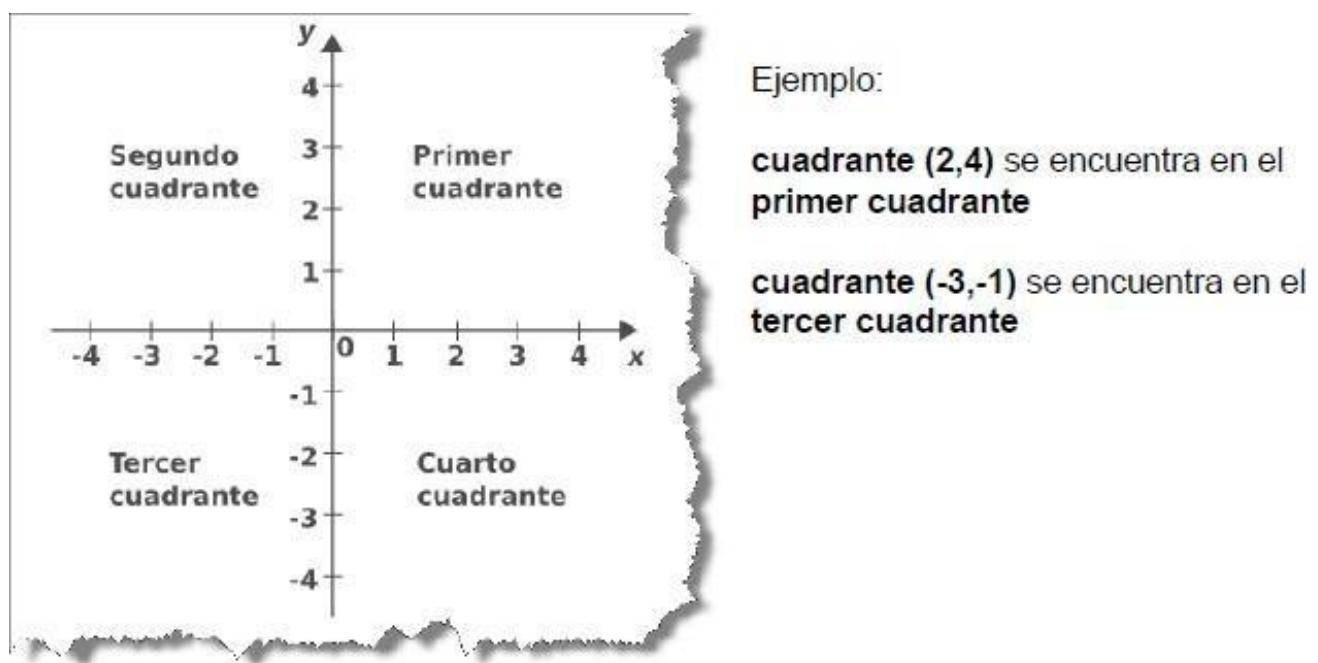


Figura 1: Cuadrantes.

8. Escriba una función que calcule la enésima potencia de un número, recibiendo como parámetro un número real base y otro entero llamado exponente.
La definición de la función es: $y = x^n$
donde *x* representa la base y *n* representa el exponente.
Nota: tener en cuenta que n puede ser un número negativo.
Ejemplo: $2^3 = 8$ y $2^{-3} = 0.125$
9. Crear una función que reciba una cadena de caracteres y una letra como parámetros, y devuelva la cantidad de veces que dicha letra aparece en la cadena.
Por ejemplo, si la cadena es 'Barcelona' y la letra es 'a', debería devolver 2 (*aparece 2 veces*).
10. Con la función creada en el ejercicio anterior, elabore un programa en donde se ingresa un carácter y 10 palabras; y muestre la cantidad total de veces que apareció el carácter en las 10 palabras.

11. Crear una función lógica (*función que retorna un valor lógico*) que determine si un número entero es par o impar.
12. Crear una función que reciba un carácter y un número como parámetros e imprima en pantalla un triángulo formado por ese carácter que tenga como ancho inicial el número recibido como parámetro.

Por ejemplo, si el carácter es * y el ancho es 4, debería escribir:

```
****  
***  
**  
*
```

13. Crear una función que reciba un número entero como parámetro y retorne verdadero si es un número primo o falso en caso contrario.
14. Crear una función que dados dos valores distintos, ingresados por parámetro, retorne el mayor de ellos.
15. Desarrollar una función que dados cinco números, recibidos por parámetro, devuelva el promedio de ellos. Se puede generalizar para n parámetros devolviendo el promedio de los mismos.
16. Dado un número entero formado sólo por los dígitos 0 (cero) y 1 (uno), diseñe una función que compruebe si el número tiene o no la misma cantidad de ceros que de unos.
17. Desarrollar una función que retorne la posición de un carácter (*la primera vez que aparezca*) dentro de la cadena de N caracteres de longitud, donde se reciben como parámetro la cadena y el carácter respectivamente.
18. Realizar un procedimiento que tome como parámetro una longitud e imprima en pantalla un rectángulo de numerales, hueco por dentro. Por ejemplo, si se ingresó 4, se verá en pantalla: *Tip: Puede ser útil pensarlo por línea horizontal*
Generalizarlo, luego, en una versión 2, para un parámetro extra: el carácter que se usará para dibujar el rectángulo, en vez de usar siempre un numeral.

```
####  
# #  
# #  
####
```

19. Crear una función que reciba un String y se encargue de poner en mayúscula la primera letra de cada palabra. No se pueden utilizar operaciones del lenguaje que lo resuelvan directamente.
20. Crear una función que reciba dos array, un booleano y retorne un array.
 - Si el booleano es verdadero buscará y retornará los elementos comunes de los dos array.
 - Si el booleano es falso buscará y retornará los elementos no comunes de los dos array.
 - No se pueden utilizar operaciones del lenguaje que lo resuelvan directamente.

21. Crear un programa que calcule quien gana más partidas al piedra, papel, tijera.

- El resultado puede ser: "Player 1", "Player 2", "Tie" (empate)
- La función recibe un listado que contiene pares, representando cada jugada.
- El par puede contener combinaciones de "R" (piedra), "P" (papel) o "S" (tijera). Ejemplo. Entrada: [("R","S"), ("S","R"), ("P","S")]. Resultado: "Player 2".