Unidad 5

Repaso

En la unidad pasada estuvimos trabajando sobre estructuras de control tanto   
condicionales como de repetición**.** En esta unidad abordaremos la forma de desarrollar nuestras propias funciones con el objetivo de reutilizar código en forma sistemática.

Introducción

Una función, es la forma de agrupar expresiones y sentencias (algoritmos) que realicen determinadas acciones, pero que éstas, solo se ejecuten cuando son llamadas. Es decir, que, al colocar un algoritmo dentro de una función, al ejecutar el programa, el algoritmo no será invocado si no se ha hecho una referencia a la función que lo contiene.

En Python, la definición de funciones se realiza mediante la palabra clave **def** más un nombre de función descriptivo, para el cuál, aplican las mismas reglas que para el nombre de las variables seguido de paréntesis de apertura y cierre. Como toda estructura de control en Python, la definición de la función finaliza con dos puntos (:) y el   
algoritmo que la compone llevará indentación.

def mi\_funcion():   
 # aquí el algoritmo

Una función no será ejecutada hasta tanto no sea invocada. Para invocar una función, simplemente se la llama por su nombre:   
def mi\_funcion():   
 print("Hola Mundo")

mi\_funcion()   
Cuando una función realice un retorno de datos, éstos, pueden ser asignados a una variable:   
def funcion():   
 return("Hola Mundo")

frase = funcion()   
print(frase)

Parámetros de una función

Un parámetro es un valor que la función espera recibir cuando sea llamada (invocada), a fin de ejecutar acciones en base al mismo. Una función puede esperar uno o más parámetros (que irán separados por una coma) o ninguno.