



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Marco Antonio Martínez Quintana

Estructura de Datos y Algoritmos 1

Asignatura:

16

Grupo:

Practica 9

No de Práctica(s):

Tierrablanca Oviedo Evelyn

Integrante(s):

*No. de Equipo de
cómputo empleado:*

No. de Lista o Brigada:

Segundo Semestre

Semestre:

Fecha de entrega:

Observaciones:

CALIFICACIÓN: _____

Objetivo:

Aplicar las bases del lenguaje de programación Python en el ambiente de Jupyter notebook. Actividades: *f*
Insertar y ejecutar código en las celdas de la notebook *f*
Insertar texto en las celdas de la notebook *f*
Declarar variables *f*
Declarar cadenas *f*
Aplicar operadores *f*
Crear y manipular listas, tuplas y diccionarios *f*
Crear y ejecutar funciones

Variables y tipos *f*

Los nombres de las variables son alfanuméricos (a-z, A-Z, 0-9) y empiezan con una letra en minúscula. *f*
No se especifica el tipo de valor que una variable contiene, está implícito al momento de asignar un valor. *f*
No se necesita poner ; al final de cada instrucción. *f*
Mantener las indentaciones al momento de escribir código.

Nombres reservados en Python

and, as, assert, break, class, continue, def, del, elif, else, except, exec, finally, for, from, global, if, import, in, is, lambda, not, or, pass, print, raise, return, try, while, with, yield.

Cadenas

Las cadenas pueden ser definidas usando comilla simple (') o comilla doble ("). Una característica especial de las cadenas es que son inmutables, esto quiere decir que no se pueden cambiar los caracteres que contiene. El carácter \ sirve para escapar caracteres como \n o \t.

Para concatenar cadenas se recomienda el uso de la función format(), en lugar del viejo estilo del operador '+'.
Por medio de la función format, se puede cambiar el orden en que se imprimen las variables:

```
#Cuando se agrega un número dentro de {}, el valor la variable que se encuentra en esa posición  
#dentro de la función format(), será impreso.  
num_cadena = "Cambiando el orden: {1} {2} {0} #".format(cadena1, cadena2, 3)  
print(num_cadena)  
  
Cambiando el orden: Mundo 3 Hola #
```

Las funciones que están integradas en Python para trabajar con cadenas se pueden ver en el siguiente link <https://docs.python.org/3/library/stdtypes.html#string-methods>.