	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	104/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

Guía práctica de estudio 09: Introducción a Python (I).




Elaborado por:

M.C. Edgar E. García Cano
Ing. Jorge A. Solano Gálvez

Autorizado por:

M.C. Alejandro Velázquez Mena

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	105/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

Guía práctica de estudio 09: Introducción a Python (I).

Objetivo:

Aplicar las bases del lenguaje de programación Python en el ambiente de Jupyter notebook.

Actividades:

- Insertar y ejecutar código en las celdas de la notebook
- Insertar texto en las celdas de la notebook
- Declarar variables
- Declarar cadenas
- Aplicar operadores
- Crear y manipular listas, tuplas y diccionarios
- Crear y ejecutar funciones


Repositorio de la guía:

Jupyter Notebook GitHub:

https://github.com/eegkno/FI_UNAM/blob/master/02_Estructuras_de_datos_y_algoritmos_1/P09/EDyA09_Lipynb

Jupyter Notebook Visualizador:

http://nbviewer.jupyter.org/github/eegkno/FI_UNAM/blob/master/02_Estructuras_de_datos_y_algoritmos_1/P09/EDyA09_Lipynb

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	106/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

Variables y tipos

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

```
#Iniciando variables
x = 10      #variable de tipo entero
print(x)   #función para imprimir los valores de las variables

#Se puede utilizar comillas dobles o simples para crear una cadena
cadena = "Hola Mundo"  #variable de tipo cadena
print(cadena)
```

```
10
Hola Mundo
```

```
#Asigna un mismo valor a tres variables
x = y = z = 10
print(x,y,z)
```


```
10 10 10
```

```
#La función type() permite conocer el tipo de una variable
type(x)
```

```
int
```

```
type(cadena)
```

```
str
```

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	107/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

```
#Se pueden cambiar los valores de las variables y el tipo se cambia automáticamente
x = "Hola Mundo"
cadena = 10
```

```
type(x)
```

```
str
```

```
type(cadena)
```

```
int
```

Cuando una variable tiene un valor constante, por convención, el nombre se escribe en mayúsculas.


```
SEGUNDOS_POR_DIA = 60 * 60 * 24
PI = 3.14
```

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.

```
#Inicializando cadenas
cadena1 = 'Hola '
cadena2 = "Mundo"
print(cadena1)
print(cadena2)
concat_cadenas = cadena1 + cadena2 #Concatenación de cadenas
print(concat_cadenas)
```

```
Hola
Mundo
Hola Mundo
```

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	108/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

```
#Para concatenar un número y una cadena se debe usar la función str()
num_cadena = concat_cadenas + ' ' + str(3) #Se agrega una cadena vacía para agregar un espacio
print(num_cadena)
```

Hola Mundo 3

Para concatenar cadenas se recomienda el uso de la función format(), en lugar del viejo estilo del operador '+'.

```
#El valor de la variable se va a imprimir en el lugar donde se encuentre {} en la cadena
num_cadena = "{} {} {}".format(cadena1, cadena2, 3)
print(num_cadena)
```


Hola Mundo 3

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>.

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	109/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

Operadores

Aritméticos: +, -, *, /

```
#Para el exponente se puede utilizar asterisco
print( 1 + 5 )
print( 6 * 3 )
print( 10 - 4 )
print( 100 / 50 )
print( 10 % 2 )
print( ((20 * 3) + (10 +1)) / 10 )
print( 2**2 )
```

```
6
18
6
2.0
0
7.1
4
```