**Trabajo práctico nro. 1**

|  |  |  |
| --- | --- | --- |
|  | **Asignatura:** | |
|  | |
| **Cursado:** | **Horas** **semanales**: |
|  | **Horas semestrales:** |
| **Carrera**: *Tecnicatura Universitaria en Programación* | **Nivel (Año):** |
| **Ciclo Lectivo: 2023** |

**Integrantes de la Cátedra:**

* **DOCENTES:**
* Indica si los siguientes identificadores son válidos en Python. En el caso de que el identificador no sea válido, explica el motivo.



a) Es valido

b) No es valido porque empieza con un numero

c) Es valido

d) No es valido porque empieza con un simbolo

e) Es valido

f) No es valido porque "for" es una palabra reservada

g) No es valido porque contiene un caracter especial en su declaracion

h) No es valido porque "global" es una palabra reservada

i) Es valido

j) Es valido

k) No es valido porque tiene guion medio

l) No es valido porque contiene un caracter especial en su declaracion

m) No es valido porque empieza con un numero

n) No es valido porque "with" es una palabra reservada

o) No es valido

p) No es valido porque empieza con un simbolo

q) Es valido pero no recomendable

r) Es valido pero seria mejor escribirlo en snake\_case

s) Es valido pero no recomendable

t) No es valido porque tiene "ñ"

u) Es valido

v) No es valido porque contiene un caracter especial en su declaracion

w) No es valido porque empieza con un numero

x) No es valido porque contiene un caracter especial en su declaracion



* Indica qué dato se guarda en la variable **x** en cada caso, suponiendo una ejecución secuencial del programa.



a) Se guarda el valor "30"

b) Se guarda el valor "30"

c) Se guarda el valor "25"

d) Se guarda el valor "8"

e) Se guarda el valor "13"

f) Se guarda el valor "8"

* Indica qué tipo de dato se guarda en cada variable.



a) Se guarda un entero

b) Se guarda un float

c) Se guarda un entero

d) Se guarda un entero

e) Se guarda un string

f) Se guarda un string

g) Se guarda un string

h) Se guarda un entero

i) Se guarda un entero

j) Se guarda un float

k) Se guarda un float

l) Se guarda un string

m) Se guarda un boolean

n) Se guarda un boolean

o) Se guarda un boolean

* Indica cuáles de las siguientes operaciones no son válidas.



c) Operacion no valida

d) Operacion no valida

e) Operacion no valida

i) Operacion no valida

j) Operacion no valida

k) Operacion no valida

l) Operacion no valida

* Declara una variable de cada tipo de dato y asígnale un valor.

a)variable\_int = 1

b)variable\_float = 2.2

c)variable\_complex = 3 + 2j

d)variable\_string = "Hola"

e) variable\_bool = True

f) variable\_list = [1, 2, 3, 4, 5]

g) variable\_tuple = (3, 4)

h) variable\_dict = {

"nombre": "Juan",

"edad": 30,

"email": "juan@example.com"

}

i) variable\_nula = None

* Teniendo la variable de tipo **string:** frase = “Caminante, no hay camino, se hace camino al andar.”, indica qué obtendríamos si aplicáramos:
* frase[5]
* frase[-1]
* frase[0:8]
* frase[::3]

a) Obtenemos la posicion numero 5 de la cadena almacenada en la variable frase, que en este caso es la letra "a"

b) Obtenemos la ultima letra de la cadena almacenada en la variable "frase". En este caso es la letra "r".

c)Obtenemos la cadena almacenada en la variable frase desde la posicion 0 hasta la posicion 8 sin incluir esta ultima. En este caso seria "Caminant".

d) Obtenemos la posicion numero 3 desde el principio de la variable frase hasta el final. Va saltando de 3 en 3, nos da como resultado "Cin,oaci,ea molnr"

* Usando la variable del ejercicio anterior:
* ¿Cómo obtenemos la cadena al revés? “.radna la onimac ecah es ,onimac yah on ,etnanimaC”
* ¿Cómo obtenemos la subcadena ‘hace’?

Para obtener la cadena al reves vamos a recorrerla desde el final al principio utilizando un indice negativo:

var\_frase = "Caminante, no hay camino, se hace camino al andar"

print(var\_frase[ : :-1])

Para obtener la subcadena "hace", recorremos la cadena de la variable frase desde la posicion "29" que es donde empieza subcadena hasta la posicion "33" para que nos tome hasta la posicion 32 inclusive.

var\_frase = "Caminante, no hay camino, se hace camino al andar"

print(var\_frase[29:33])

* Métodos upper(), lower() y title().



* Pon en mayúsculas la primera letra de cada palabra del siguiente nombre: ‘lucas mauricio barros’.
* Deja esta frase totalmente en letras minúsculas: ‘El qUe No arRiesGa, nO gANa.’
* Deja esta frase totalmente en letras mayúsculas: ‘El qUe No arRiesGa, nO gANa.’

a) var\_nombre = "lucas mauricio barros"

print(var\_nombre.title())

b) var\_frase="El qUe No arRiesGa, nO gANa."

print(var\_frase.lower())

c) var\_frase="El qUe No arRiesGa, nO gANa."

print(var\_frase.upper())

* Convierte en expresiones algorítmicas las siguientes expresiones algebraicas. Coloca paréntesis solamente donde sean necesarios.



a) var\_resultado = (b / 2) - (4 \* a \* c)

b) var\_resultado = 3 \* x \* y - 5 \* x + 12 \* x - 17

c) var\_resultado = (b + d) / (c + 4)

d) var\_resultado = (x \* y) / y + 2

e) var\_resultado = 1 / y + (3 \* x) / z + 1

f) var\_resultado = 1 / (y + 3) + x / y + 1

g) var\_resultado = a \*\* 2 + b \*\* 2

h)var\_resultado = (a + b) \*\* 2

i)var\_resultado = b \*\* (1/3) + 34

j)var\_resultado = (x / y) \* (z + w) \* pi

k)var\_resultado = (x + y) / (u + (w / b))

* Convierte en expresiones algebraicas las siguientes expresiones algorítmicas. Coloca paréntesis solamente donde sean necesarios.







* Dada la siguiente expresión aritmética:



Determinar qué resultado obtendremos si a=5, b=2, c=6, x=(-6) y y=4.

El resultado que obtendremos de 5+2\*(5-6/2)+(7-(-6))/(4+4) es 10.625

* Escribe las expresiones algorítmicas equivalentes a los siguientes enunciados:



a) var\_numero1 = 5

var\_numero2 = 3

resultado = var\_numero1 + var\_numero2

b) lista\_numeros = [4,7,9]

sum\_total = sum(lista\_numeros)

cantidad\_numeros = len(lista\_numeros)

promedio = sum\_total / cantidad\_numeros

print(promedio)

c) var\_base = 8

var\_altura = 5

var\_area = var\_base\*var\_altura

print("El area del rectangulo es: ",var\_area)

d) numero = int(input("Ingrese un número: "))

if numero % 2 == 0:

print("El número es par.")

else:

print("El número no es par.")

e) var\_doble= 16\*2

print(var\_doble)

f) var\_diferencia = 6 \*(8-3)

g) var\_diferencia = (2\*6)-(4+3)

h) numero = int(input("Ingrese un numero:"))

if numero % 3 == 0 and numero % 2 == 0:

print("El numero ", numero," es multiplo de 2 y 3")

else:

print("El numero ", numero," no es multiplo de 2 y 3")

i)var\_precio = int(input("Ingrese un precio:"))

if var\_precio >= 15 and var\_precio <= 90:

print("El precio esta dentro del rango de precios")

else:

print("El precio esta fuera del rango de precios")

j)var\_N=int(input("Ingrese un numero:"))

var\_N+=12

print(var\_N)

k)var\_N=int(input("Ingrese un numero:"))

var\_N-=5

print(var\_N)

l)var\_N=int(input("Ingrese un numero:"))

var\_N\*=3

print(var\_N)

m)var\_N=int(input("Ingrese un numero:"))

var\_N//=2

print(var\_N)

* ¿Qué resultado (True/False) dan las siguientes operaciones?



a) False

b) True

c) False

d) Error porque la variable "x" no esta definida

e) False

f) True

g) True

h) Error porque la variable "x" no esta definida

i) True

* Siendo x una variable de tipo entera, con valor 5, determine qué se mostrará por pantalla en cada caso.



a) Se muestra por pantalla el valor 6 ya que estamos incrementando el valor de x en 1

b) Se muestra el valor 3 ya que estamos disminuyendo el valor de x en 2

c) Se muestra el valor 25 ya que estamos multiplicando a X por 5

d) Se muestra el valor 1.0 ya que estamos dividiendo X en 5



* Tipos *list*, *tuple* y *dict*.



* De la siguiente lista, ¿qué color está en la posición 3?, ¿cómo accedemos a esta posición?



En la posicion 3 se encuentra el color Amarillo

* ¿En qué posición se encuentra el color ‘rojo’? ¿Y el ‘rosa’?

El color rojo se encuentra en la posicion 0 y el rosa en la posicion 7

* Crea una lista que contenga los siguientes valores en las posiciones indicadas.



lista = [3,2,5,4,1]



* Imprime la segunda posición de esta tupla.



print(colores[1])

* Utiliza los símbolos de suma y resta para obtener el resultado 25 a partir de los elementos de la siguiente tupla en una variable llamada operacion.



numeros = (10,1,5,11)

operacion = numeros[0] + numeros[0] + numeros[2]

print(operacion)



* Cuenta la cantidad de elementos del siguiente diccionario.



diccionario = {'a': 1, 'b': 2, 'c': 3, 'd': 4}

cantidad\_elementos = len(diccionario)

print("La cantidad de elementos en el diccionario es:", cantidad\_elementos)

* Accede al valor de la clave ‘c’ en el diccionario.

print(diccionario['c'])

* Vamos a practicar el uso de las funciones **input()** y **print()**.

Ejemplo: Solicita el nombre de una persona e imprime un mensaje de bienvenida.



* Solicita dos números al usuario, súmalos e imprime el resultado.

var\_numero = int(input("Ingresa un numero: "))

var\_numero2= int(input("Ingresa otro numero: "))

resultado = var\_numero + var\_numero2

print(resultado)

* Solicita la edad de una persona, calcula cuántos años faltan para que cumpla 100 años e imprime el resultado.

edad = int(input("Ingresa tu edad: "))

cuanto\_falta = 100 - edad

print("Te faltan", cuanto\_falta, "años para cumplir los 100")

* Operadores ternarios.



*¡Practiquemos!* Crear las variables necesarias para realizar la ejercitación.

* Comprobar si un número es par o impar.

numero\_par=int(input("Ingrese un numero: "))

if numero\_par%2 == 0:

print("El numero es par")

else: print("El numero es impar")

* Obtener el valor absoluto de un número.

numero = float(input("Ingresa un número: "))

valor\_absoluto = abs(numero)

* Comparar dos números y obtener el mayor.

numero\_uno = int(input("Ingrese el primer numero: "))

numero\_dos = int(input("Ingrese el segundo numero: "))

if numero\_uno > numero\_dos:

print("El numero uno es mayor al numero dos")

else:

print("El numero dos es mayor al numero uno")