**Trabajo práctico nro. 1**

|  |  |  |
| --- | --- | --- |
| Logo Departamento 2.jpg | **Asignatura: Programación I** | |
|  | |
| **Cursado:**Primer Trimestre | **Horas** **semanales**: |
|  | **Horas semestrales:**  *Cantidad estimada de horas semestrales/anuales.* |
| **Carrera**: *Tecnicatura Universitaria en Programación* | **Nivel (Año):** |
| **Ciclo Lectivo: 2023** |

**Integrantes de la Cátedra:**

* **DOCENTES:**

|  |  |  |
| --- | --- | --- |
| **Nombre del Profesor** | **Periodo** | **Cantidad horas materia** |
|  |  | 6 horas |

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



1. Valido
2. Invalido – Empieza con un numero
3. Valido – Mala Práctica - No utilizar snake case
4. Invalido – Toma / como un operador
5. Valido – Mala Práctica - Utiliza Ñ
6. Invalido – Palabra reservada
7. Valido
8. Invalido – Palabra reservada
9. Valido
10. Valido
11. Invalido – Toma el – como una operación
12. Invalido – No se puede utilizar @
13. Invalido – No puede empezar con un numero
14. Invalido – Palabra Reservada
15. Invalido – Tiene una mayúscula lo cual es mala práctica y el – la convierte en una operación
16. Invalido – El % la convierte en una operación
17. Valido
18. Valido – Mala Práctica – Tiene mayúsculas
19. Valido – Mala Práctica – Tiene mayúsculas
20. Valido – Mala Práctica – Utiliza Ñ
21. Valido
22. Invalido – Toma el & como una condición
23. Invalido – Empieza con un número
24. Invalido – No se puede utilizar ?



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



1. 30
2. 30
3. 25
4. 8
5. 13
6. 8

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



1. Float
2. Float
3. Int
4. Int
5. String
6. String
7. String
8. Int
9. Int
10. Float
11. Float
12. String
13. Boolean
14. Boolean
15. Boolean
16. Indica cuáles de las siguientes operaciones no son válidas.



No validas: C – D – E – F – i – J – K - I

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



var\_int = 5

var\_float = 5.25

var\_complex = complex(3, 4)

var\_string = "string"

var\_bool = False

var\_list = [1,2,3,4,5]

var\_tuple = (1,2,3,4,5)

var\_dict = {

    "rojo": "#FF0000",

    "verde": "#00FF00",

    "azul": "#0000FF"

}

var\_null = None

1. Teniendo la variable de tipo **string:** frase = “Caminante, no hay camino, se hace camino al andar.”, indica qué obtendríamos si aplicáramos:
2. frase[5] = a
3. frase[-1] = .
4. frase[0:8] = Caminant
5. frase[::3] = Cin,oaci,ea molnr
6. Usando la variable del ejercicio anterior:
7. ¿Cómo obtenemos la cadena al revés? “.radna la onimac ecah es ,onimac yah on ,etnanimaC”
8. ¿Cómo obtenemos la subcadena ‘hace’?

cadena = "Caminante, no hay camino, se hace camino al andar."

cadena\_invertida=cadena[::-1]

inicio\_palabra = cadena.find("hace")

fin\_palabra = cadena.find(" ", inicio\_palabra)

palabra\_final = cadena[inicio\_palabra:fin\_palabra]

print(cadena\_invertida)

print(palabra\_final)

1. Métodos upper(), lower() y title().



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

nombre ="lucas mauricio barros"

nombre = nombre.title()

print(nombre)

frase\_mayuscula = "El qUe No arRiesGa, nO gANa."

frase\_mayuscula = frase\_mayuscula.upper()

frase\_minuscula = frase\_mayuscula.lower()

print(frase\_mayuscula)

print(frase\_minuscula)

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



1. (b/2) – (4\*a \*c)
2. (3\*x\*y) – (5\*x) + (12\*x) – 17
3. (b+d) / (c+4)
4. ((x\*y) / y) + 2
5. (1/y) + ((3\*x) /z) + 1
6. (1/(y+3)) + (x/y) + 1
7. (a^2) + (b^2)
8. (a + b) ^2
9. (b^1/8) + 34
10. (x/y) \* (z+w) \* π
11. (x+y) / u+ (w/b)
12. Convierte en expresiones algebraicas las siguientes expresiones algorítmicas. Coloca paréntesis solamente donde sean necesarios.



2. = c
3. Dada la siguiente expresión aritmética:



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

= 10.625

1. Escribe las expresiones algorítmicas equivalentes a los siguientes enunciados:



1. 5 + 3 = 8
2. (4+7+9) /3 =
3. 8\*5 =
4. num % 2 == 0
5. 16\*2 = 32
6. (8-3 )\*6 =30
7. (2\*6) – (4+3)=
8. num %2 == 0 AND num%3 == 0
9. precio >=15 AND precio<=90
10. n += 12
11. n -= 12
12. n \*= 3
13. n /= 2
14. ¿Qué resultado (True/False) dan las siguientes operaciones?



1. False
2. True
3. False
4. False
5. False
6. True
7. True
8. True
9. True
10. Siendo x una variable de tipo entera, con valor 5, determine qué se mostrará por pantalla en cada caso.



1. 6
2. 3
3. 25
4. 1



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



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



RTA: Se encuentra el color “amarillo”

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

RTA: “rojo” se encuentra en la posición 0 y “rosa” en la posición 7

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



RTA: números = [“tres”, “dos”, “cinco”, “cuatro”, “uno”]



1. Imprime la segunda posición de esta tupla.



RTA: print(colores [1])

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 operación.



RTA: num = numeros[0] + numeros[3] + numeros[2] – numeros[1]



1. Cuenta la cantidad de elementos del siguiente diccionario.



RTA: Tiene 4 elementos

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

RTA: print(diccionario["c"])

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

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



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

primer\_numero = input("Ingresa el primer número ")

segundo\_numero = input("Ingresa el segundo número ")

primer\_numero = int(primer\_numero)

segundo\_numero = int(segundo\_numero)

print("El resultado de la suma de ambos numeros es: ", (primer\_numero + segundo\_numero))

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

edad = input("Ingrese la edad de la persona ")

edad\_faltante = 100 - int(edad)

print(edad\_faltante)

1. Operadores ternarios.



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

1. Comprobar si un número es par o impar.
2. Obtener el valor absoluto de un número.
3. Comparar dos números y obtener el mayor.

A)

num = input("ingrese un numero ")

num = int(num)

mensaje = "El número es par" if num%2==0 else "El número no es par"

print(mensaje)

B)

B) num = input("ingrese un numero ")

num = int(num)

valor\_absoluto = num if num >= 0 else -num

print(valor\_absoluto)

C)

num = input("ingrese un numero ")

num = int(num)

num\_segundo = input("Ingrese un segundo número ")

num\_segundo = int(num\_segundo)

num\_mayor = num if num>=num\_segundo else num\_segundo

print("El mayor número fue: " , num\_mayor)