**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. Es válido
2. No es Válido porque no puede empezar con un número
3. Es válido, pero es mala práctica no utilizar camel case
4. No es válido porque no puede empezar, ni tener una barra diagonal
5. Es válido, pero es mala practica
6. No es válido porque es una palabra reservada del lenguaje
7. No es válido porque no puede tener el símbolo $
8. No es válido porque es una palabra reservada del lenguaje
9. Es válido
10. Es válido
11. No es válido porque no puede tener un guion medio
12. No es válido porque no puede tener un @
13. No es válido porque no puede empezar con un número
14. No es válido porque es una palabra reservada del lenguaje
15. No es válido porque no puede tener un guion medio
16. No es válido porque no puede tener el símbolo %
17. Es válido
18. Es válido, pero es mala practica
19. Es válido, pero es mala práctica está en mayúscula
20. Es válido, pero es mala practica
21. Es válido
22. No es válido porque no puede tener el símbolo &
23. No es válido, empieza con mayúscula
24. No es válido porque no puede tener un signo de pregunta
25. 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
7. Indica qué tipo de dato se guarda en cada variable.



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



1. Es valida
2. Es valida
3. No es valida
4. No es valida
5. No es valida
6. No es valida
7. Es valido
8. Es valido
9. No es valida
10. No es valida
11. No es valida
12. No es valida
13. Declara una variable de cada tipo de dato y asígnale un valor.



num\_int = 2

num\_float = 0.2

num\_complex = 3 + 2j

caracter\_str = “hola”

tof\_boolean = 3 <= 5

nums\_lista = [1,2,3,4,5,6,7,8,9]

nums\_tupla = (1,2,3)

nombre\_dict = {

“Nombre 1” : “Sofia”

“Nombre 2” : “Carla”

“Nombre 3” : “Juan”

“Nombre 4” : “Pepe”

}

var\_vacia = 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”

frase[::-1]

1. ¿Cómo obtenemos la subcadena ‘hace’?

Posición = frase.find(“hace”)

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



1. Pon en mayúsculas la primera letra de cada palabra del siguiente nombre: ‘lucas mauricio barros’.

nombre\_primeraM = nombre.title()

1. Deja esta frase totalmente en letras minúsculas: ‘El qUe No arRiesGa, nO gANa.’

nombre\_minuscula = nombre.lower()

1. Deja esta frase totalmente en letras mayúsculas: ‘El qUe No arRiesGa, nO gANa.’

nombre\_mayuscula = nombre.upper()

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 \* (1/3)) + 34
10. (x/y) \* (x+w)\* π
11. (x+y) / (u +(w/b))
12. Convierte en expresiones algebraicas las siguientes expresiones algorítmicas. Coloca paréntesis solamente donde sean necesarios.



1. – 4ac
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 = 6.66
3. 8\*5 = 40
4. num % 2 = 0
5. 16 \* 2 = 32
6. (8 - 3) \* 6 = 30
7. (2\*6) – (4+3) = 5
8. n % 2 = 0 AND n % 3 = 0
9. precio >= 15 OR precio < 90
10. n += 12
11. n -= 5
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
5. Tipos *list*, *tuple* y *dict*.



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



Posición 3 = amarillo

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

rojo = posición 0

rosa = posición 7

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



lista = [‘tres’, ‘dos’, ‘cinco’, ‘cuatro’, ‘uno’]



1. Imprime la segunda posición de esta tupla.



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 operacion.





operacion = numeros[0] + numeros[2] + numeros[3] – numeros[1]

1. Cuenta la cantidad de elementos del siguiente diccionario.



elementos = 4

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

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.

num1 = int(input(“Ingrese el primer numero: ”))

num2 = int(input(“Ingrese el segundo numero: ”))

suma = num1 + num2

print(f”La suma de {num1} y {num2} es {suma}”)

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

edad = int(input(“Ingrese su edad: ”))

para\_los\_100 = 100 – edad

print(f“Le falta {para\_los\_100} años para que cumpla 100”)

1. Operadores ternarios.



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

1. Comprobar si un número es par o impar.

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

par\_impar = "Es par" if num % 2 == 0 else "Es impar"

print(par\_impar)

1. Obtener el valor absoluto de un número.

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

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

print(f“Valor absoluto {valor\_abs}”)

1. Comparar dos números y obtener el mayor.

num1 = int(input("Ingrese el primer numero: "))

num2 = int(input("Ingrese el segundo numero: "))

mayor = num1 if num1 > num2 else num2

print(f“El numero mayor es {mayor}”)