**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** |
| **Cinthia Rigoni** |  | 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. Valida
2. Invalida debido a que los nombres de variables no deben de comenzar con un numero
3. Valido pero no es lo recomendable, en Python se aconseja utilizar un \_ para separar palabras y no el camelCase
4. Invalido ya que las variables no deben comenzar con caracteres especiales
5. Invalido, las palabras no deben contener la letra ñ ni acentos, ya que escribimos las variables en ingles
6. Invalido debido a que es una palabra reservada del lenguaje
7. Invalido porque contiene caracteres especiales
8. Invalido porque es una palabra reservada
9. Valido
10. Valido
11. Invalido ya que el guion medio es un operador
12. Invalido porque contiene un carácter especial
13. Invalido ya que utiliza un numero al inicio
14. Invalido ya que es una palabra reservada
15. Invalido porque el guion medio es un operador
16. Invalido, comienza con un carácter especial
17. Valido pero no es recomendable ya que es poco intuitivo
18. Valido pero no se recomienda utilizar ese tipo de nombres en Python
19. Valido para variables que no queremos que sean modificadas.
20. Invalido, no debemos de usar la letra ñ de ser posible (El programa lo tomara como valido pero no es aconsejable)
21. Valido
22. Invalido, no se recomienda utilizar caracteres de ese tipo
23. Invalido, comienza con números
24. Invalido, contiene un carácter especial



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



1. X = 30 d) x = 8
2. X = 30 e) x = 13
3. X = 25 f) x = 8
4. Indica qué tipo de dato se guarda en cada variable.



1. float **b)** float **c)** float **d)** int **e)** str **f)** str **g)** str **h)** Int **i)** int

**j)** float **k)** float **l)** str **m)** bool **n)** bool **o)** bool

1. Indica cuáles de las siguientes operaciones no son válidas.



1. Valida
2. Valida
3. Invalida porque concatena un str con un int
4. Invalido arroja error “string index out of range”
5. Invalido ya que no se puede utilizar el método len() con un int
6. Valido
7. Valido
8. Valido
9. Invalido ya que “z” no corresponde al sistema numérico.
10. Invalido porque el . no corresponde a un número.
11. Invalido ya que el termino < no puede ser utilizado entre un int(4) y un str(“f”)
12. Invalido ya que no se puede asignar
13. Declara una variable de cada tipo de dato y asígnale un valor.





• **var\_entera** = 4 • **var\_decimal** = 10.5 • **var\_compleja** = 2+3j • **var\_cadena** = (“hola”)

• **var\_booleana** = 5<3==2 • **var\_lista** = [1, 2, 3, 'a', 'b', 'c'] • **var\_tupla** = (1, 2, 'a', 'b')

• **var\_diccionario** = {'nombre': 'Maxi', 'edad': 30, 'ciudad': 'Mendoza'} • **var\_nula** = 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’?

frase = ("Caminante, no hay camino, se hace camino al andar.")

var = frase.find("hace")

print(var) #Esto me devuelve el índice de donde comienza la subcadena “hace”.

print(frase[29:33])

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



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

frase = ("lucas mauricio barros")

frase\_modificada = frase.title()

print(frase\_modificada)

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

frase = ("El qUe No arRiesGa, nO gANa.")

frase\_modified = frase.lower()

print(frase\_modified)

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

frase = ("El qUe No arRiesGa, nO gANa.")

frase\_modified = frase.upper()

print(frase\_modified)

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



1. 4
2. Dada la siguiente expresión aritmética:



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

Obtendremos como resultado el numero 10,625

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



1. 5+3
2. 4+7+9/3
3. 8\*5
4. x%2 == 0 es par
5. 16\*2
6. 6\*(8-3)
7. (2\*6) – (4+3)

n\_multiplo = 10

if n\_multiplo % 2 == 0 and n\_multiplo % 3 == 0:

    print("Es múltiplo de 2 y 3")

else:

    print("No es múltiplo de 2 y 3")

1. if n < 15 and n > 90:

print(“Es mayor a 15 y menor a 90”)

else:

print(“Es menor o igual a 15 o menor o igual a 90”)

1. n = 1

n = n+12

k) n = 5

n = n-5

1. n = 3

n \*= 3; n = 9

1. n = 10

n = 10/2

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



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



1. Se imprimirá 6
2. Se imprimirá 3
3. Se imprimirá 25
4. Se imprimirá 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. Accedemos a esta posición por medio de la indexación.

Creamos la variable:

indice = colores[3] print(indice)

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

Rta = El color rojo está en la posición 0 y el rosa en la 7

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



Rta = mi\_lista [“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 operacion.



Rta = numeros = (10,1,5,11)

operacion = numeros[0] + numeros[1] + numeros[2] + (numeros[3] - 2)

print(operacion)



1. Cuenta la cantidad de elementos del siguiente diccionario.



Rta = Contiene 4 elementos

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

Rta = Para acceder a un valor en un diccionario debemos de asignar dicho valor a una variable para asi poder imprimirla.

tercer\_valor = ["c"]

print(tercer\_valor)

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.

pedido = input("Ingrese un numero: ")

numero1 = int(pedido)

pedido2 = input("Ingrese el segundo número: ")

numero2 = int(pedido2)

resultado\_suma = numero1 + numero2

print("La suma de ambos números es: " ,resultado\_suma)

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

Rta = Para conocer el faltante utilice el método abs que me devuelve el valor absoluto de un número, y a su edad la reste a 100.

pedido = input("Ingrese su edad: ")

numero = int(pedido)

resultado\_edad = abs(numero - 100)

print("Restan:",resultado\_edad, "para que llegues a los 100 años!!")

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 número: "))

def par (num):

    return num % 2 == 0

if par(num):

    print("El número es par")

else:

    print("El número es impar")

1. Obtener el valor absoluto de un número.

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

def num\_positive(num):

    return num > 0

if num\_positive(num):

    print("El número ingresado es positivo")

else:

    print("Su número negativo es: ",num)

    num = abs(num)

    print("Su numero en positivo es: ",num)

1. Comparar dos números y obtener el mayor.

num\_1 = int(input("Ingrese el primer número: "))

num\_2 = int(input("Ingrese el segundo número: "))

def es\_mayor(num\_1, num\_2):

    return num\_1 > num\_2

def iguales(num\_1,num\_2):

    return num\_1 == num\_2

if iguales(num\_1, num\_2):

    print("Los numeros ingresados son iguales")

else:

    if es\_mayor(num\_1, num\_2):

        print("El primero número es mayor")

    else:

        print("El segundo número es mayor")