**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:**

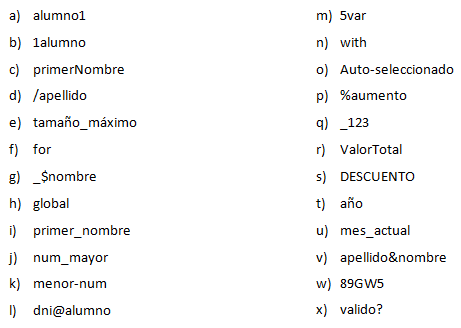
* **DOCENTES:**

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

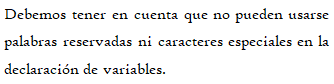
* **Alumno:**

|  |  |  |
| --- | --- | --- |
| **Nombre del Alumno** | **Periodo** | **Cantidad horas materia** |
| **Escudero Juan Cruz** |  | 6 horas |

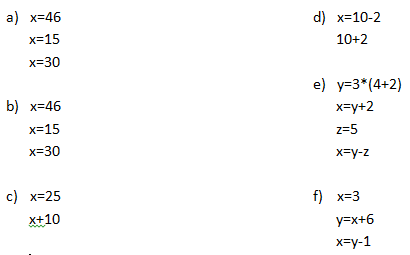
1. Indica si los siguientes identificadores son válidos en Python. En el caso de que el identificador no sea válido, explica el motivo.



|  |  |
| --- | --- |
| 1. A) Valido | M)Invalido / Empieza con numero |
| B) Invalido / Empieza con numero | N)Invalida / Palabra reservada |
| C)Valido | O) Invalida / Carácter especial - |
| D)Invalido / Carácter especial / | P) Invalida / Carácter especial % |
| E)Invalido / Carácter especial ñ-` | Q) Valida |
| F)Invalido / Palabra reservada | R) Valida |
| G)Invalido / Carácter especial | S) Valida |
| H)Invalido / palabra reservada | T) Invalida / Carácter especial ñ |
| I)Valida | U) Valida |
| J)Valido | V) Invalida / Carácter especial & |
| K)Invalido / Carácter especial - | W) Invalida / Empieza con numero |
| L)Valido | X) Invalida / Carácter especial |



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



1. A) 30

B)30

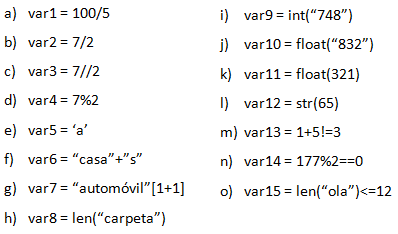
C)25

D)8

E)13

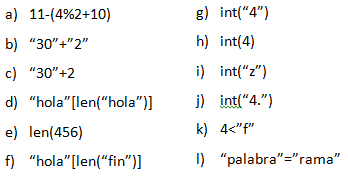
F)8

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



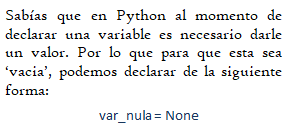
|  |  |
| --- | --- |
| A) Entero | I)Entero |
| 1. B) Float | J)Float |
| C) Entero | K)Float |
| D)Entero | L)String |
| E) String | M)Boleano |
| F)String | N)Boleano |
| G)String | O)Boleano |
| H)Entero |  |
|  |  |
|  |  |

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



|  |  |
| --- | --- |
| A)Valido | G)Valido |
| B) Valido | H)Valido |
| C) Invalido / no se puede conbinar string con números | I)Invalido / tienen que pertenecer a números |
| D)Valido | J) Invalido / no pueden ser números float |
| E)Valido | K)No se puede comparar strings con < > |
| F)Valido | L) Invalido una expresión no puede ser string |

1. Declara una variable de cada tipo de dato y asígnale un valor.
2. var1 = 23
3. var2 = 23.23
4. var3 = 23j
5. var4 = “23”
6. var5 = “true”
7. var6 = [23, 23]
8. var7 = 23, 23
9. var8 = {“23”: 23}
10. var9 = 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]
3. frase[-1]
4. frase[0:8]
5. frase[::3]
6. A) a

B) r

C) Caminan

D) Cin,oaci,ea molnr

1. Usando la variable del ejercicio anterior:
2. ¿Cómo obtenemos la cadena al revés? “.radna la onimac ecah es ,onimac yah on ,etnanimaC”

Print(frase[::-1])

1. ¿Cómo obtenemos la subcadena ‘hace’?

Print(frase[29:30])

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’

Title(frase)

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

frase = “El qUe No arRiesGa, nO gANa.”

Lower(frase)

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

frase = ‘El qUe No arRiesGa, nO gANa.’

Upper(frase)

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

9- A) b/2 – 4 \* a \* c J) x/y\*(z+w)\*pi

B) 3\*x\*y – 5\*x + 12\*x – 17 K) x+y/u+w/b

C) b+d/c+4

D) x\*y/y+2

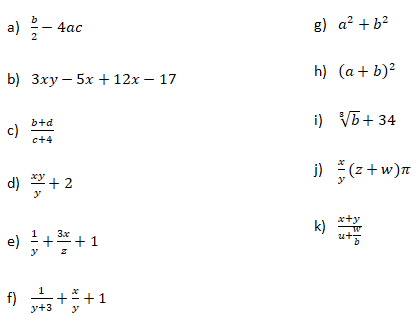
E) 1/y+3\*x/z+1

F)1/y+3+x/y+1

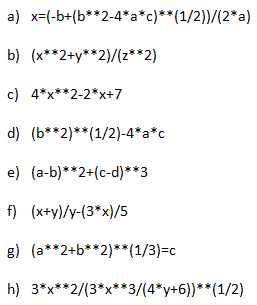
G)a\*\*2+b\*\*2

H)(2+2)\*\*2

I)b\*\*1/3+34



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



10) A) x =

B)

C)

D)

E)

F)

G)

H)

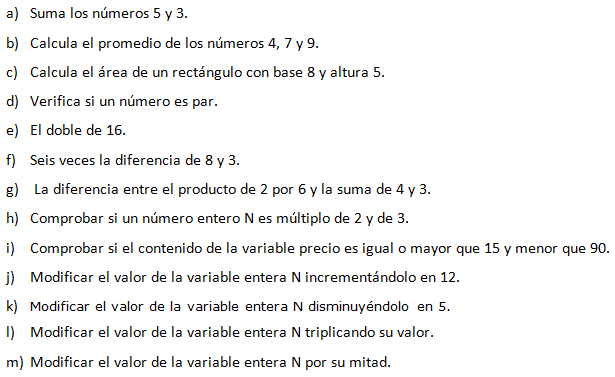
1. Dada la siguiente expresión aritmética:



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

Rta: 10.625

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



1. A = 5 + 3

B = (4 + 7 + 9) / 3

C = 8 \* 5

D = n%2 == 0 (si es true es par)

E = 16\*2

F = (8-3) \* 6

G = (2\*6)-(4+3)

H) def multiplos(N):

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

return "El numero: ", N, "es multiplo de 2 y 3"

else:

return "El numero: ", N, "NO es multiplo de 2 y 3"

I = precio >= 16 and precio < 90

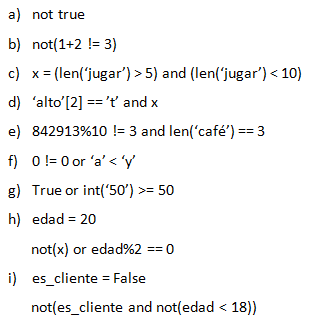
J =) N += 12

K =) N -= 5

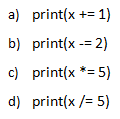
L =) N \*= 3

M =) N /= 2

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

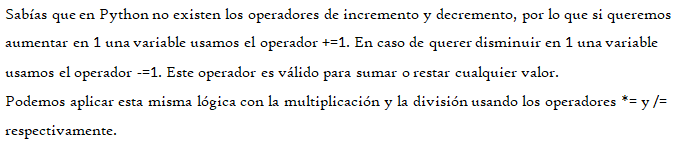


1. False
2. True
3. False
4. True
5. False
6. True
7. True
8. True
9. True
10. Siendo x una variable de tipo entera, con valor 5, determine qué se mostrará por pantalla en cada caso.

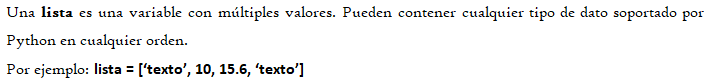


1. Error, “file main.py”, line 4 …. Print(x+=1)

Aparece eso en todos los casos, con sus respectivos escritos, print(x \*= 5), print(x /= 5), etc etc.



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



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

Esta el color “amarillo”, y se accede con: colores[3]

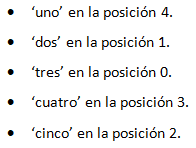


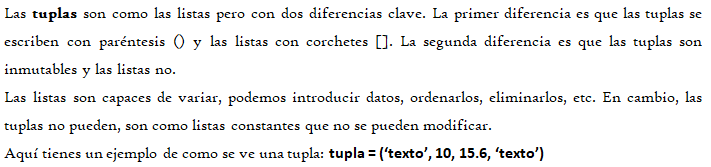
1. ¿En qué posición se encuentra el color ‘rojo’? ¿Y el ‘rosa’?

Rojo en la posición 0 y rosa en la -1

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

Rta: lista = [“tres”, “dos”, “cinco”, “cuatro”, “uno”]





1. Imprime la segunda posición de esta tupla.

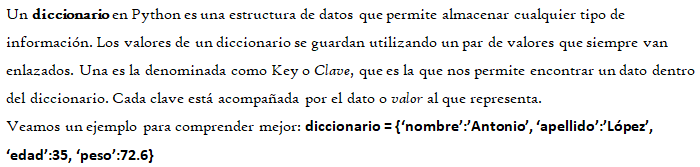
Rta: print(colores[1])



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: operación = numeros[0] + numeros[0] + numeros[1]





1. Cuenta la cantidad de elementos del siguiente diccionario.

Rta: tiene 4 eleentos



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

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.

a = input("Ingrese el primer numero: ")

b = input("Ingrese el segundo numero: ")

c = float(a) + float(b)

print("la suma de los dos numeros es:", c)

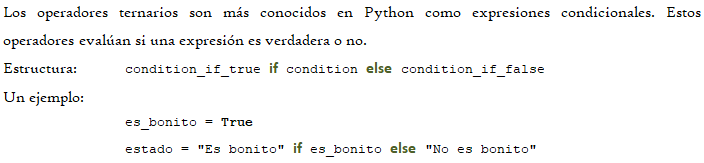
1. Solicita la edad de una persona, calcula cuántos años faltan para que cumpla 100 años e imprime el resultado.

a = input("Ingrese su edad: ")

c = 100 - float(a)

print("estas solo a:",c,"años de cumplir 100 años")

1. Operadores ternarios.



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

1. Comprobar si un número es par o impar.

If numero%2 == 0:

print(“el numero es par”)

else:

print(“el numero es inpar)

1. Obtener el valor absoluto de un número. (es mas fácil usar abs())

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

if n >= 0:

abs = n

else:

abs = -n

print("El valor absoluto del número es:", abs)

1. Comparar dos números y obtener el mayor.

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

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

If a > b

print(“el numero”, a, “es mayor que”, b,)

else:

print(“el numero”, b, “es mayor que”, a,)