**ACTIVIDADES PRÁCTICAS**

**Actividad 1:**

Resuelva

1. Considere el siguiente código e modo interactivo, ¿cuál es el resultado?

|  |
| --- |
| **2+1 \_\*2 \_\*\*2** |

1. Considere la expresión:

|  |
| --- |
| **True and not False is True** |

1. Considere la siguiente tupla e índice, ¿qué devolverá al ejecutarse?

|  |
| --- |
| **(1,2,3)[-0]** |

1. Considere la siguiente tupla e índice, ¿qué devolverá al ejecutarse?

|  |
| --- |
| **(1,2,3)[-0:0]** |

1. Considere la siguiente lista **x=[1,2,3]**, ingrese el código para adicionar el número 4 a **x.**

|  |
| --- |
|  |

1. Considere la siguiente tupla **x=(1,2,3)**, ingrese el código para adicionar el número 4 a **x.**

|  |
| --- |
|  |

1. ¿Qué muestra después de ejecutarse la siguiente línea de código?

|  |
| --- |
| **str(range(10)** |

1. ¿Qué realiza el método siguiente en las líneas de código dadas?

|  |
| --- |
| **X = "125,000"**  **X.isdigit()** |

1. Sea los sets  **x={1,2,3}** y  **y={2,3,4}**. ¿Cómo se puede obtener **{4}** de  **x** y **y** usando las operaciones básicas de conjuntos (set)?

|  |
| --- |
|  |

1. Sea los sets  **x={1,2,3}** y  **y={2,3,4}**. ¿Cómo se puede obtener **{2, 3}** de  **x** y **y** usando las operaciones básicas de conjuntos (set)?

|  |
| --- |
|  |

1. Sea los sets  **x={1,2,3}** y  **y={2,3,4}**. ¿Cómo se puede obtener **{1, 4}** de  **x** y **y** usando las operaciones básicas de conjuntos (set)?

|  |
| --- |
|  |

1. Considere el siguiente diccionario **estudiantes={'Tim':29, 'Jim':31, 'Pam':27, 'Sam':35}**, ¿qué retornará **estudiantes[0]**?

|  |
| --- |
|  |

**Actividad 2:**

Resuelva

1. Considere el siguiente código. ¿A qué es igual **x**?

|  |
| --- |
| **x=3  y=x  y=y-1** |

1. Considere el siguiente código. ¿A qué es igual **L1**?

|  |
| --- |
| **L1 = [2,3,4]  L2 = L1  L2[0] = 24** |

1. Considere el siguiente código, ¿qué imprimirá?

|  |
| --- |
| **if False:      print("Falso!")  elif True:      print("Ahora Verdad")  else:      print("Finalmente Verdad")** |

1. Considere el siguiente diccionario: **osos = {"Grizzly":"gruñon", "Brown":"amigable","Polar":"amigable"}**.

Completa el código para imprimir un mensaje solo a los osos amigables

|  |
| --- |
| **osos = {"Grizzly":"gruñon","Brown":"amigable","Polar":"amigable"}**  **for oso in osos:**  **if #completar aquí#:**  **print("Hola, "+oso+" oso!")**  **else:**  **print("----")** |

1. ¿Qué imprime al ejecutar este código?

|  |
| --- |
| **n=100**  **veces = 0**  **while n >= 1:**  **n //= 2**  **veces += 1**  **print(veces)** |

1. Considere el siguiente código, ¿cuál será la salida?

|  |
| --- |
| **sum([i\*\*2 for i in range(3)])** |

1. Considere el siguiente código, ¿cuál será la salida?

|  |
| --- |
| **sum([i for i in range(10) if i%2 == 1])** |

1. Considere el siguiente código, ¿qué imprimirá?

|  |
| --- |
| **F = open("archivo.txt", "w")**  **F.write("Hola\n Mundo")**  **F.close()**  **lineas = []**  **for linea in open("archivo.txt"):**  **lineas.append(linea.strip())**  **print(lineas)** |

1. Considere la siguiente función, ¿cuál es el valor final de la línea?

|  |
| --- |
| **def modifica(milista):**  **milista[0] \*= 10**  **return(milista)**  **L = [1, 3, 5, 7, 9]**  **M = modifica(L)**  **M is L** |

**Actividad 3:**

Resuelva:

1. Considere el siguiente código, ¿qué es lo que imprime?

|  |
| --- |
| **x = numpy.array([[3,6],[5,7]])  y = x.transpose()  print(y)** |

1. Considere el siguiente código, ¿qué devuelve?

|  |
| --- |
| **x = np.array([1,2,5])  x[1:2]** |

1. Considere el siguiente código, ¿qué devuelve?

|  |
| --- |
| **a = np.array([1,2])  b = np.array([3,4,5])  b[a]** |

1. Considere el siguiente código, ¿qué devuelve?

|  |
| --- |
| **a = np.array([1,2])  b = np.array([3,4,5])  b[a]**  **c = b[1:]  b[a] is c** |

**Actividad 4:**

Resuelva:

¿Qué gráfico produce lo siguiente?

|  |
| --- |
| **plt.plot([0,1,2],[0,1,4],"rd-")** |

**Actividad 5:**

Descargar la data iris y construye:

* Histogramas para cada variable en un panel
* Gráficos de dispersión entre pares de variables

*Nota:*

* Puede descargar la información desde la web a través del buscador
* La data iris está en la librería de Python y puede descargarla de la forma siguiente: **from sklearn.datasets import load\_iris**