**TP N° 3**

Ejercicio 1:

#1. Crear un programa que almacene en una lista las materias de esta u otra carrera y que las  
#muestre por pantalla. (La lista debe ser predefinida, no debe ser ingresada por el usuario)  
  
lista = ['Introducción a la informatica', 'Ingles', 'Programación','Programación de aplicaciones moviles', 'Desarrollo de software']  
  
print(lista)

Ejercicio 2:

#Pedir al usuario que ingrese 5 números para luego almacenarlos en una lista y ordenarlos.  
#Imprimir por pantalla el resultado.  
  
# Inicializar una lista vacía para almacenar los números  
numeros = []  
  
#Con la función range generamos 5 pedidos de ingreso de número al usuario  
for i in range(5):  
 while True:  
 try:  
 numeroUsuario = int(input("Ingrese un número: "))  
 numeros.append(numeroUsuario)  
 break  
 except ValueError:  
 print("Ingrese un número válido.")  
#El metodo sort() ordena la lista  
numeros.sort()  
  
print("Números ingresados y ordenados:", numeros)

Ejercicio 3:

# Dada la siguiente lista de frutas [“banana”, “manzana”, “pera”], permitir al usuario ingresar 3  
#frutas más para agregarlas al final de lista. Luego, mostrar por pantalla la lista completa y  
#debajo la misma lista pero sólo con las frutas que añadió el usuario  
  
frutas = ['banana', 'manzana', 'pera']  
frutasUsuario = []  
#Con la función range generamos 3 pedidos de ingreso de frutas al usuario  
for i in range(3):  
 while True:  
 frutaUsuario = input(f'Ingrese el nombre de una fruta, sin repetir las que ya están: ')  
 #Si la fruta ingresada se encuentra en la lista, se le va a pedir al usuario que ingrese nuevamente  
 #una fruta  
 if frutaUsuario in frutas or frutaUsuario in frutasUsuario:  
 print('La fruta ya se encuentra en la lista. Por favor, agregue una fruta diferente')  
 else:  
 frutas.append(frutaUsuario)  
 frutasUsuario.append(frutaUsuario)  
 break  
print(frutas)  
print(frutasUsuario)

Ejercicio 4:

#Dada la siguiente lista: países = [“Argentina,”Brasil”, “Bolivia”,”Paraguay”,”Venezuela”],  
#realizar lo siguiente:  
#a. Imprimir la cantidad de elementos que tiene la lista.  
#b. Imprimir el primer y último elemento.  
#c. Imprimir el resto.  
#d. Permitir que el usuario ingrese un país e imprimir el índice si el país se encuentra en  
#la lista. Si no se encuentra, imprimir un mensaje advirtiendo al usuario.  
#e. Permitir al usuario ingresar un número igual o menor a la cantidad de elementos de  
#la lista. Quitar el elemento correspondiente de esa posición.  
#f. Imprimir la lista en orden inverso.  
#g. Vaciar la lista  
  
paises = ['Argentina','Brasil', 'Bolivia','Paraguay','Venezuela']  
  
#A - Imprimir la cantidad de elementos que tiene la lista.  
  
print(f'Cantidad de elementos de la lista: ',len(paises))  
  
#B- Imprimir el primer y último elemento.  
print(f'Primer Elemento: ', paises[0])  
print(f'Ultimo Elemento: ', paises[4])  
  
#C - Imprimir el resto.  
print(paises[1:4])  
  
#D - Permitir que el usuario ingrese un país e imprimir el índice si el país se encuentra en  
#la lista. Si no se encuentra, imprimir un mensaje advirtiendo al usuario.  
  
paisUsuario = input('Ingrese el nombre de un país para buscar en la lista: ').strip().capitalize()  
if paisUsuario in paises:  
 print(f"El país '{paisUsuario}' se encuentra en la lista en la posición {paises.index(paisUsuario)}.")  
else:  
 print(f"El país '{paisUsuario}' no se encuentra en la lista.")  
  
#E - Permitir al usuario ingresar un número igual o menor a la cantidad de elementos de  
#la lista. Quitar el elemento correspondiente de esa posición.  
  
while True:  
 try:  
 numeroPosicion = int(input(f"Ingrese un número entre 0 y {len(paises)-1} para eliminar el elemento correspondiente: "))  
 if 0 <= numeroPosicion < len(paises):  
 eliminado = paises.pop(numeroPosicion)  
 print(f"Se ha eliminado el país: {eliminado}")  
 break  
 else:  
 print(f"Por favor, ingrese un número válido entre 0 y {len(paises)-1}.")  
 except ValueError:  
 print("Por favor, ingrese un número válido.")  
  
#F - Imprimir la lista en orden inverso.  
print("Lista en orden inverso:", paises[::-1])  
  
#g - Vaciar la lista.  
paises.clear()  
print("Lista vacía:", paises)

Ejercicio 5:

#Escriba un programa Python que solicite al usuario el ingreso de números enteros. Cuando el  
#usuario ingrese la palabra “fin” el programa deberá concluir con la carga de datos. Cada uno  
#de los números ingresados por el usuario deberá ser almacenado en una lista. A  
#continuación, realice las siguientes tareas:  
#a. Determinar el máximo.  
#b. Obtener segundo valor máximo. Es decir el que le sigue al máximo.  
#c. Determinar el mínimo.  
#d. Calcular la multiplicación de todos los números de la lista.  
#e. Contar los valores pares e impares.  
#f. Remover los elementos repetidos.  
  
def solicitarNumeros():  
 numeros = []  
 while True:  
 entrada = input("Ingrese un número entero (o 'fin' para terminar): ")  
 if entrada.lower() == 'fin':  
 break  
 try:  
 numero = int(entrada)  
 numeros.append(numero)  
 except ValueError:  
 print("Por favor, ingrese un número entero válido.")  
 return numeros  
  
#A - Determinar el máximo.  
def determinarMaximo(numeros):  
 return max(numeros)  
  
#B - Obtener segundo valor máximo. Es decir el que le sigue al máximo.  
def obtenerSegundoMaximo(numeros):  
 maximo = max(numeros)  
 numerosSinMaximo = [num for num in numeros if num != maximo]  
 if numerosSinMaximo:  
 segundoMaximo = max(numerosSinMaximo)  
 return segundoMaximo  
 else:  
 return None  
  
#C - Determinar el mínimo.  
def determinarMinimo(numeros):  
 return min(numeros)  
  
#D - Calcular la multiplicación de todos los números de la lista.  
def multiplicacionLista(numeros):  
 resultado = 1  
 for numero in numeros:  
 resultado \*= numero  
 return resultado  
  
#E - Contar los valores pares e impares.  
def contarParesImpares(numeros):  
 pares = sum(1 for num in numeros if num % 2 == 0)  
 impares = sum(1 for num in numeros if num % 2 != 0)  
 return pares, impares  
  
#F - Remover los elementos repetidos.  
def removerRepetidos(numeros):  
 return list(set(numeros))  
  
def main():  
 numeros = solicitarNumeros()  
 if numeros:  
 maximo = determinarMaximo(numeros)  
 segundoMaximo = obtenerSegundoMaximo(numeros)  
 minimo = determinarMinimo(numeros)  
 multiplicacion = multiplicacionLista(numeros)  
 pares, impares = contarParesImpares(numeros)  
 numeros\_unicos = removerRepetidos(numeros)  
  
 print(f'Los números ingresados son: {numeros}')  
 print(f'El número más alto ingresado es: {maximo}')  
 if segundoMaximo is not None:  
 print(f'El segundo número más alto ingresado es: {segundoMaximo}')  
 else:  
 print('No hay segundo máximo ya que todos los números son iguales.')  
 print(f'El número mínimo ingresado es: {minimo}')  
 print(f'La multiplicación de todos los números es: {multiplicacion}')  
 print(f'Números de valores pares: {pares}')  
 print(f'Números de valores impares: {impares}')  
 print(f'Lista sin números repetidos: {numeros\_unicos}')  
 else:  
 print('No se ingresaron números.')  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

Ejercicio 6:  
#Programe una aplicación de consola Python que brinde al usuario la posibilidad de a partir  
#de una lista vacía:  
#a. Agregar un elemento al final.  
#b. Agregar un elemento al principio.  
#c. Quitar un elemento al final.  
#d. Quitar un elemento al principio  
  
#Definimos las opciones del menu  
def menu():  
 print("\nMenú de opciones:")  
 print('1. Agregar un elemento al final')  
 print('2. Agregar un elemento al principio')  
 print('3. Quitar un elemento al final')  
 print('4. Quitar un elemento al principio')  
 print('5. Mostrar lista')  
 print('6. Salir')  
  
def agregarAlFinal(lista):  
 elemento = input('Ingrese el elemento a agregar al final: ')  
 lista.append(elemento)  
 print(f"Elemento '{elemento}' agregado al final.")  
  
def agregarAlPrincipio(lista):  
 elemento = input('Ingrese el elemento a agregar al principio: ')  
 lista.insert(0, elemento)  
 print(f"Elemento '{elemento}' agregado al principio.")  
  
def quitarAlFinal(lista):  
 if lista:  
 elemento = lista.pop()  
 print(f"Elemento '{elemento}' quitado del final.")  
 else:  
 print('La lista está vacía. No hay elementos para quitar.')  
  
def quitarAlPrincipio(lista):  
 if lista:  
 elemento = lista.pop(0)  
 print(f"Elemento '{elemento}' quitado del principio.")  
 else:  
 print('La lista está vacía. No hay elementos para quitar.')  
  
def mostrarLista(lista):  
 print('Lista actual:', lista)  
  
def main():  
 lista = []  
 while True:  
 menu()  
 opcion = input('Seleccione una opción: ')  
  
 #Creamos el acceso al menu de opciones  
 if opcion == "1":  
 agregarAlFinal(lista)  
 elif opcion == "2":  
 agregarAlPrincipio(lista)  
 elif opcion == "3":  
 quitarAlFinal(lista)  
 elif opcion == "4":  
 quitarAlPrincipio(lista)  
 elif opcion == "5":  
 mostrarLista(lista)  
 elif opcion == "6":  
 print('Saliendo del programa.')  
 break  
 else:  
 print('Opción no válida. Por favor, intente de nuevo.')  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

Ejercicio 7:

#7. Escriba un programa Python que cuente con dos listas, la primera de ellas almacenará strings  
#con tareas pendientes y la segunda almacenará strings con tareas terminadas. Permita al  
#usuario:  
#a. Agregar nuevas tareas pendientes.  
#b. Marcar las tareas pendientes como terminadas. Al hacer esto, la tarea deberá pasar  
#de la lista de pendientes a la de terminadas.  
#Nota: posterior a cada operación deberá mostrar por pantalla el estado actual de ambas  
#listas.  
  
#Definimos el menu de opciones  
def Menu():  
 print("\nMenú de opciones:")  
 print('1. Agregar nueva tarea pendiente')  
 print('2. Marcar tarea pendiente como terminada')  
 print('3. Mostrar listas de tareas')  
 print('4. Salir')  
  
  
def agregarTareaPendiente(tareasPendientes):  
 tarea = input('Ingrese una nueva tarea pendiente: ')  
 tareasPendientes.append(tarea)  
 print(f"Tarea '{tarea}' agregada a la lista de pendientes.")  
  
  
def marcarTareaTerminada(tareasPendientes, tareasTerminadas):  
 if not tareasPendientes:  
 print('No hay tareas pendientes para marcar como terminadas.')  
 return  
  
 print('Lista de tareas pendientes:')  
 for idx, tarea in enumerate(tareasPendientes, start=1):  
 print(f"{idx}. {tarea}")  
  
 try:  
 numeroTarea = int(input('Ingrese el número de la tarea que desea marcar como terminada: '))  
 if 1 <= numeroTarea <= len(tareasPendientes):  
 tarea = tareasPendientes.pop(numeroTarea - 1)  
 tareasTerminadas.append(tarea)  
 print(f"Tarea '{tarea}' marcada como terminada.")  
 else:  
 print('Número de tarea inválido.')  
 except ValueError:  
 print('Por favor, ingrese un número válido.')  
  
  
def mostrarListas(tareasPendientes, tareasTerminadas):  
 print("\nTareas pendientes:")  
 if tareasPendientes:  
 for tarea in tareasPendientes:  
 print(f"- {tarea}")  
 else:  
 print('No hay tareas pendientes.')  
  
 print("\nTareas terminadas:")  
 if tareasTerminadas:  
 for tarea in tareasTerminadas:  
 print(f"- {tarea}")  
 else:  
 print('No hay tareas terminadas.')  
  
  
def main():  
 tareasPendientes = []  
 tareasTerminadas = []  
  
 while True:  
 Menu()  
 opcion = input('Seleccione una opción: ')  
 #Creamos el acceso al menu de opciones  
 if opcion == "1":  
 agregarTareaPendiente(tareasPendientes)  
 elif opcion == "2":  
 marcarTareaTerminada(tareasPendientes, tareasTerminadas)  
 elif opcion == "3":  
 mostrarListas(tareasPendientes, tareasTerminadas)  
 elif opcion == "4":  
 print('Saliendo del programa.')  
 break  
 else:  
 print('Opción no válida. Por favor, intente de nuevo.')  
  
 mostrarListas(tareasPendientes, tareasTerminadas)  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()