

"""Se requiere un sistema para registrar los turnos de un dia para una veterinaria.
La misma atiende de 9 a 19hs, los turnos son uno por hora.

Se debe contar con un menu tal:

- 1) Registrar turno: Se debe ingresar el nombre de la mascota, la edad, especie (Gato o Perro) y el horario. No dar dos turnos a la misma mascota. La carga Finaliza con mascota "FIN" (Se puede seguir cargando turnos)
- 2) Prioridad: Ordenar e insertar a continuacion de los turnos de las mascotas de mas de 10 años, un 99 en turno y edad, y "PRIORIDAD" en nombre y especie.
- 3) Ver turnos: Se muestra con formato adecuado y ordenados, los datos de cada turno (horario, mascota). Si la mascota tiene prioridad, mostrarlo.
- 4) Eliminar turno: Permite eliminar el turno de una mascota.
- 5) Ver estadisticas:
 - * Mascota más joven (mostrar especie, nombre y edad)
 - * Promedio de edad de cada especie.
 - * Porcentaje de mascotas que superan el promedio de edad, sobre el total de cada especie.
- 6) Salir."""

```
# FUNCIONES
def seleccionar_opcion():
    print("\nMenu de opciones:\n")
    print("1. Registrar turno")
    print("2. Prioridad")
    print("3. Ver turnos")
    print("4. Eliminar turno")
    print("5. Ver estadisticas")
    print("6. Salir")
    opcion = int(input("\nIngrese la opcion: "))
    while opcion < 1 or opcion > 6:
        opcion = int(input("Error - Ingrese la opcion: "))
    return opcion

def ingresar_nombre():
    mascota = input("\nIngrese el nombre de la mascota: ").upper()
    while mascota == "":
        mascota = input("Error - Ingrese el nombre de la mascota: ").upper()
    return mascota

def ingresar_especie():
    tipom = input("\nIngrese el tipo de mascota (P o G): ").upper()
    while tipom != "P" and tipom != "G":
        tipom = input("Error - Ingrese el tipo de mascota (P o G): ").upper()
    return tipom

def ingresar_edad():
    edadm = int(input("\nIngrese el edad (0-20): "))
    while edadm < 0 or edadm > 20:
        edadm = int(input("Error - Ingrese la edad (0-20): "))
    return edadm

def ingresar_horario():
    horariom = int(input("\nIngrese el horario (9-19): "))
    while horariom < 9 or horariom > 19:
        horariom = int(input("Error - Ingrese el horario (9-19): "))
    return horariom

def buscar_mascota(arr_nombres, nombre):
```

```

# REGISTRO ÚNICO: Busca la posición de una mascota en el arreglo de nombres.
i = 0
while i < len(arr_nombres) and nombre != arr_nombres[i]:
    i += 1
return i

def registrar_turnos(arr_nombres, arr_edades, arr_especies, arr_horarios):
    nombre = ingresar_nombre()
    while nombre != "FIN":
        pos = buscar_mascota(arr_nombres, nombre)
        if pos == len(arr_nombres):
            especie = ingresar_especie()
            edad = ingresar_edad()
            horario = ingresar_horario()

            arr_nombres.append(nombre)
            arr_especies.append(especie)
            arr_edades.append(edad)
            arr_horarios.append(horario)
        else:
            print(f"La mascota ya tiene turno hoy a las {arr_horarios[pos]}")
            nombre = ingresar_nombre()

def registrar_prioridad(arr_nombres, arr_edades, arr_especies, arr_horarios):
    i = 0 #Inserta turnos de prioridad después de cada mascota mayor de 10 años.
    print("\nBuscando mayores de 10 años...")
    while i < len(arr_nombres):
        es_prioridad = 0
        if arr_nombres[i] == "PRIORIDAD":
            es_prioridad = 1
        if es_prioridad == 0 and arr_edades[i] > 10:
            # Insertar los datos de prioridad *a continuación* (índice i+1)
            arr_nombres.insert(i + 1, "PRIORIDAD")
            arr_edades.insert(i + 1, 99)
            arr_horarios.insert(i + 1, arr_horarios[i])
            arr_especies.insert(i + 1, arr_especies[i])
        i += 1
        print(f"-> Prioridad asignada a: {arr_nombres[i-1]}")
    i += 1

def mostrar_arreglos (arr_nombres, arr_horarios):
    titulo = "Horario"
    titulo1 = "Mascota"
    print(f"{titulo:<10} {titulo1}")
    print("-----")
    for i in range(len(arr_nombres)):
        if i+1 < len(arr_nombres) and arr_nombres[i+1] == "PRIORIDAD":
            print(f"{arr_horarios[i]:<10} {arr_nombres[i]} - PRIORIDAD")
        elif arr_nombres[i] != "PRIORIDAD":
            print(f"{arr_horarios[i]:<10} {arr_nombres[i]}")

def eliminar_turno(arr_nombres, arr_edades, arr_especies, arr_horarios):
    nombre = ingresar_nombre()
    pos = buscar_mascota(arr_nombres, nombre)
    if pos != len(arr_nombres):
        print(f"-> Se elimino: {arr_nombres[pos]}")
        arr_nombres.pop(pos)
        arr_edades.pop(pos)

```

```

        arr_especies.pop(pos)
        arr_horarios.pop(pos)
    else:
        print("\nLa mascota no tiene turno asignado. ")

def buscar_joven(arr_edades):
    pos_min = 0
    for i in range(len(arr_edades)):
        if arr_edades[i] < arr_edades[pos_min]:
            pos_min = i
    return pos_min

def calcular_promedio(arr_especies, arr_edades, tipo):
    acum = 0
    cont = 0
    promedio = -1 # cero indica que no hay mascotas de ese tipo
    for i in range(len(arr_edades)):
        if arr_especies[i] == tipo:
            acum += arr_edades[i]
            cont += 1
        if cont !=0:
            promedio = acum/cont
    return promedio

def calcular_porcentaje(arr_especie, arr_edades, tipo, promedio):
    cont_total = 0
    cont_supera = 0

    for i in range(len(arr_edades)):
        if arr_especie[i] == tipo:
            if arr_edades[i] > promedio:
                cont_supera += 1
            cont_total += 1
    if cont_total != 0:
        porcentaje = (cont_supera/cont_total)* 100
    else:
        porcentaje = 0
    return porcentaje

def ver_estadisticas(arr_nombres, arr_edades, arr_especie):
    # Mascota más joven (mostrar especie, nombre y edad), promedio de edad de cada especie.
    # Porcentaje de mascotas que superan el promedio de edad, sobre el total de cada especie.
    if len (arr_nombres) > 0:
        pos_mas_joven = buscar_joven(edades)
        promedio_gato = calcular_promedio(arr_especie, arr_edades, "G")
        promedio_perro = calcular_promedio(arr_especie, arr_edades, "P")
        if arr_especie[pos_mas_joven] == "P":
            arr_especie[pos_mas_joven] = "Perro"
        else:
            arr_especie[pos_mas_joven] = "Gato"
        print(f"La mascota mas joven es un {arr_especie[pos_mas_joven]} llamado {arr_nombres[pos_mas_joven]} y tiene {arr_edades[pos_mas_joven]} años.")
        if promedio_gato != -1:
            print("El promedio de edades de los gatos es: ", promedio_gato)
            porcentaje = calcular_porcentaje(arr_especie, arr_edades, "G", promedio_gato)
            print("El porcentaje que supera dicha edad es:", porcentaje)
        if promedio_perro != -1:

```

```

        print("El promedio de edades de los perros es: ", promedio_perro)
        porcentaje = calcular_porcentaje(arr_especie, arr_edades, "P", promedio_perro)
        print("El porcentaje que supera dicha edad es:", porcentaje)
    else:
        print("No hay mascotas registradas.")

def ordenar_arreglos(arr_nombres, arr_edades, arr_especie, arr_horario):
    for i in range(len(arr_horario)):
        for j in range(len(arr_horario)):
            if arr_horario[i] < arr_horario[j]:
                intercambiar(arr_horario, i, j)
                intercambiar(arr_nombres, i, j)
                intercambiar(arr_edades, i, j)
                intercambiar(arr_especie, i, j)

def intercambiar(arreglo, i, j):
    aux = arreglo[i]
    arreglo[i] = arreglo[j]
    arreglo[j] = aux

# PROGRAMA PRINCIPAL

nombres = []
edades = []
especies = []
horarios = []

opcion = seleccionar_opcion()
while opcion != 6:
    if opcion == 1:
        print("\n--- Registro de turnos ---")
        registrar_turnos(nombres, edades, especies, horarios)
        print("\nTurnos registrados con éxito.")
    if opcion == 2:
        print("\n--- Asignación de prioridad ---")
        registrar_prioridad(nombres, edades, especies, horarios)
        print("\nFin de la asignación de prioridad.")
    if opcion == 3:
        print("\n--- Turnos asignados ---\n")
        mostrar_arreglos(nombres, horarios)
        print("\nFin de la lista de turnos.")
    if opcion == 4:
        print("\n--- Eliminar turno ---")
        eliminar_turno(nombres, edades, especies, horarios)
        print("\nFin de la eliminación de turno.")
    if opcion == 5:
        print("\n--- Estadísticas ---\n")
        ver_estadisticas(nombres, edades, especies)
        ordenar_arreglos(nombres, edades, especies, horarios)
        mostrar_arreglos(nombres, horarios)
        print("\nFin de las estadísticas.\n")
    opcion = seleccionar_opcion()
print("\nSaliendo del programa...")
print("\nFin del programa.\n")

```