

MAIN.PY

```
from funciones import *

ARCHIVO = "países.csv"

def menu():
    países = cargar_países(ARCHIVO)

    while True:
        print("\n=== MENÚ ===")
        print("1. Agregar país")
        print("2. Actualizar datos de un país")
        print("3. Buscar país")
        print("4. Filtrar países")
        print("5. Ordenar países")
        print("6. Mostrar estadísticas")
        print("7. Salir")

        opcion = input("Seleccione una opción: ")

        if opcion == "1":
            agregar_pais(países)
            guardar_países(ARCHIVO, países)
        elif opcion == "2":
            actualizar_pais(países)
            guardar_países(ARCHIVO, países)
        elif opcion == "3":
            buscar_pais(países)
        elif opcion == "4":
            filtrar_países(países)
        elif opcion == "5":
            ordenar_países(países)
        elif opcion == "6":
            mostrar_estadísticas(países)
        elif opcion == "7":
            print("Saliendo del programa.")
            break
        else:
            print("Opción inválida.")

if __name__ == "__main__":
    menu()
```

FUNCIONES.PY

```
import csv

def cargar_países(nombre_archivo):
    """
    Lee el archivo CSV y devuelve una lista de diccionarios.
    Cada país se representa como:
    {"nombre": str, "poblacion": int, "superficie": int, "continente": str}
    """
    países = []
    try:
        with open(nombre_archivo, newline='', encoding='utf-8') as archivo:
            lector = csv.DictReader(archivo)
            for fila in lector:
                try:
                    pais = {
                        "nombre": fila["nombre"].strip(),
                        "poblacion": int(fila["poblacion"]),
                        "superficie": int(fila["superficie"]),
                        "continente": fila["continente"].strip()
                    }
                    países.append(pais)
                except (KeyError, ValueError):
                    print("Error en el formato de una fila, se omitió.")
    except FileNotFoundError:
```

```

        print("Archivo no encontrado. Se creará uno nuevo al guardar.")
    return paises

def guardar_paises(nombre_archivo, paises):
    """
    Guarda la lista de países en el archivo CSV.
    Sobrescribe todo el archivo con los datos actuales.
    """
    with open(nombre_archivo, 'w', newline='', encoding='utf-8') as archivo:
        campos = ["nombre", "poblacion", "superficie", "continente"]
        escritor = csv.DictWriter(archivo, fieldnames=campos)
        escritor.writeheader()
        escritor.writerows(paises)
    print("Datos guardados correctamente.")

def agregar_pais(paises):
    """
    Solicita los datos de un nuevo país y lo agrega a la lista.
    No permite campos vacíos ni duplicados.
    """
    nombre = input("Nombre del país: ").strip()
    if not nombre:
        print("El nombre no puede estar vacío.")
        return

    # Esta evita duplicados
    for p in paises:
        if p["nombre"].lower() == nombre.lower():
            print("Ese país ya existe en la lista.")
            return

    try:
        poblacion = int(input("Población: "))
        superficie = int(input("Superficie (km²): "))
    except ValueError:
        print("Debe ingresar números válidos.")
        return

    continente = input("Continente: ").strip()
    if not continente:
        print("El continente no puede estar vacío.")
        return

    nuevo = {
        "nombre": nombre,
        "poblacion": poblacion,
        "superficie": superficie,
        "continente": continente
    }
    paises.append(nuevo)
    print(f"País '{nombre}' agregado con éxito.")

def actualizar_pais(paises):
    """
    Permite modificar la población y superficie de un país existente.
    """
    nombre = input("Ingrese el nombre del país a actualizar: ").strip()
    for pais in paises:
        if pais["nombre"].lower() == nombre.lower():
            try:
                nueva_poblacion = int(input("Nueva población: "))
                nueva_superficie = int(input("Nueva superficie (km²): "))
                pais["poblacion"] = nueva_poblacion
                pais["superficie"] = nueva_superficie
                print(f"Datos actualizados para {pais['nombre']}.")
                return
            except ValueError:
                print("Ingrese números válidos.")
                return
    print("País no encontrado.")

```

```

def buscar_pais(paises):
    """
    Busca un país por coincidencia parcial o exacta del nombre.
    """
    termino = input("Ingrese el nombre (o parte) del país a buscar: ").strip().lower()
    resultados = [p for p in paises if termino in p["nombre"].lower()]

    if resultados:
        print("\n Resultados encontrados:")
        for p in resultados:
            print(f"- {p['nombre']} | Población: {p['poblacion']:,} | Superficie: {p['superficie']:,} km")
    else:
        print("No se encontraron países con ese nombre.")

def filtrar_paises(paises):
    """
    Permite filtrar países por continente, rango de población o superficie.
    """
    print("\n=== FILTROS ===")
    print("1. Por continente")
    print("2. Por rango de población")
    print("3. Por rango de superficie")
    opcion = input("Seleccione una opción: ")

    if opcion == "1":
        cont = input("Ingrese el continente: ").strip().lower()
        filtrados = [p for p in paises if p["continente"].lower() == cont]
    elif opcion == "2":
        try:
            min_p = int(input("Población mínima: "))
            max_p = int(input("Población máxima: "))
            filtrados = [p for p in paises if min_p <= p["poblacion"] <= max_p]
        except ValueError:
            print("Ingrese números válidos.")
            return
    elif opcion == "3":
        try:
            min_s = int(input("Superficie mínima: "))
            max_s = int(input("Superficie máxima: "))
            filtrados = [p for p in paises if min_s <= p["superficie"] <= max_s]
        except ValueError:
            print("Ingrese números válidos.")
            return
    else:
        print("Opción inválida.")
        return

    if filtrados:
        print("\n Países filtrados:")
        for p in filtrados:
            print(f"- {p['nombre']} | {p['continente']} | Pob: {p['poblacion']:,} | Sup: {p['superficie']:,} km")
    else:
        print("No hay países que cumplan con ese filtro.")

def ordenar_paises(paises):
    """
    Ordena los países por nombre, población o superficie.
    """
    print("\n=== ORDENAR POR ===")
    print("1. Nombre")
    print("2. Población")
    print("3. Superficie")
    opcion = input("Seleccione una opción: ")

    if opcion == "1":
        paises.sort(key=lambda x: x["nombre"].lower())
    elif opcion == "2":
        paises.sort(key=lambda x: x["poblacion"])
    elif opcion == "3":
        sentido = input("Ascendente (A) o Descendente (D)? ").strip().lower()

```

```

        reverse = sentido == "d"
        paises.sort(key=lambda x: x["superficie"], reverse=reverse)
    else:
        print("■ Opción inválida.")
        return

    print("\n■ Países ordenados:")
    for p in paises:
        print(f"- {p['nombre']} | Población: {p['poblacion']:,} | Superficie: {p['superficie']:,}")

def mostrar_estadisticas(paises):
    """
    Muestra indicadores básicos:
    - País con mayor y menor población
    - Promedio de población y superficie
    - Cantidad de países por continente
    """
    if not paises:
        print("■ No hay datos cargados.")
        return

    mayor = max(paises, key=lambda x: x["poblacion"])
    menor = min(paises, key=lambda x: x["poblacion"])

    prom_pob = sum(p["poblacion"] for p in paises) / len(paises)
    prom_sup = sum(p["superficie"] for p in paises) / len(paises)

    # Contar países por continente
    continentes = {}
    for p in paises:
        c = p["continente"]
        continentes[c] = continentes.get(c, 0) + 1

    print("\n■ ESTADÍSTICAS")
    print(f"- País con mayor población: {mayor['nombre']} ({mayor['poblacion']:,})")
    print(f"- País con menor población: {menor['nombre']} ({menor['poblacion']:,})")
    print(f"- Promedio de población: {prom_pob:,.0f}")
    print(f"- Promedio de superficie: {prom_sup:,.0f} km²")
    print("- Cantidad de países por continente:")
    for cont, cant in continentes.items():
        print(f"    • {cont}: {cant}")

```