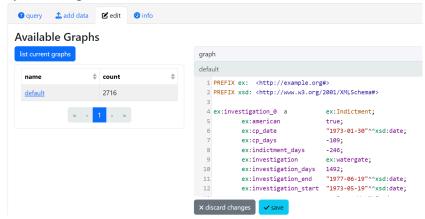
Ejecución

```
    Cargar datos en Fuseki
    Ejecutar consulta SPARQL
    Salir
    Seleccione una opción: 1
    Datos de investigaciones_kg.ttl cargados exitosamente en Fuseki.
```

/p16_investigaciones



```
Menú
1. Cargar datos en Fuseki
2. Ejecutar consulta SPARQL
3. Salir
Seleccione una opción: 2
Nombre: http://example.org#Robert_C._Mardian, Presidente: http://example.org#Richard_Nixon
Nombre: http://example.org#Bernard_L._Barker, Presidente: http://example.org#Richard_Nixon
Nombre: http://example.org#Eugenio_R._Martinez, Presidente: http://example.org#Richard_Nixon
Nombre: http://example.org#Ed_Reinecke, Presidente: http://example.org#Richard_Nixon
Nombre: http://example.org#Stuart_Russell, Presidente: http://example.org#Richard_Nixon
Nombre: http://example.org#Oliver_North, Presidente: http://example.org#Ronald_Reagan
Nombre: http://example.org#John_Poindexter, Presidente: http://example.org#Ronald_Reagan
Nombre: http://example.org#Lyn_Nofziger, Presidente: http://example.org#Ronald_Reagan
Nombre: http://example.org#Lyn_Nofziger, Presidente: http://example.org#Ronald_Reagan
Nombre: http://example.org#Lyn_Nofziger, Presidente: http://example.org#Ronald_Reagan
Nombre: http://example.org#Lyn_Nofziger, Presidente: http://example.org#Ronald_Reagan
```

```
import requests
from rdflib import Graph
from SPARQLWrapper import SPARQLWrapper, JSON
def cargar datos(fuseki url, archivo ttl):
   g.parse(archivo ttl, format='turtle')
   turtle data = g.serialize(format='turtle')
   headers = {
    response = requests.put(fuseki url, data=turtle data,
headers=headers)
   if response.status code == 200:
       print(f'Datos de {archivo_ttl} cargados exitosamente en
Fuseki.')
       print(f'Error al cargar los datos de {archivo ttl}:
response.status_code} - {response.text}')
def ejecutar consulta sparql wrapper(fuseki sparql url,
consulta sparql):
    sparql = SPARQLWrapper(fuseki sparql url)
   sparql.setQuery(consulta sparql)
   sparql.setReturnFormat(JSON)
       sparql results = sparql.query().convert()
       if 'results' in sparql results:
            for result in sparql results['results']['bindings']:
```

```
name = result['name']['value'] if 'name' in result else
'N/A'
                president = result['president']['value'] if 'president'
in result else 'N/A'
                print(f"Nombre: {name}, Presidente: {president}")
            print("No se encontraron resultados.")
        print(f"Error al ejecutar la consulta SPARQL: {e}")
def menu():
    dataset = 'p16 investigaciones'
    while True:
       print("\nMenú")
        print("1. Cargar datos en Fuseki")
        print("2. Ejecutar consulta SPARQL")
        print("3. Salir")
        opcion = input("Seleccione una opción: ")
        if opcion == '1':
            fuseki url = f'http://localhost:3030/{dataset}/data'
            cargar datos(fuseki url, archivo ttl)
        elif opcion == '2':
            fuseki sparql url =
f'http://localhost:3030/{dataset}/sparql'
            consulta sparql = """
            ejecutar consulta sparql wrapper(fuseki sparql url,
consulta sparql)
```