

Ejecución

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

/p16_investigaciones

[query](#) [add data](#) [edit](#) [info](#)

Available Graphs

[list current graphs](#)

name	count
default	2716

[«](#) [<](#) **1** [>](#) [»](#)

graph

default

```
1 PREFIX ex: <http://example.org#>
2 PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
3
4 ex:investigation_0 a ex:Indictment;
5 ex:american true;
6 ex:cp_date "1973-01-30"^^xsd:date;
7 ex:cp_days -109;
8 ex:indictment_days -246;
9 ex:investigation ex:watergate;
10 ex:investigation_days 1492;
11 ex:investigation_end "1977-06-19"^^xsd:date;
12 ex:investigation_start "1973-05-19"^^xsd:date;
```

[X discard changes](#) [✓ save](#)

```
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#Lance_H._Wilson, Presidente: http://example.org#George_H.W._Bush
```

Código

```
import requests
from rdflib import Graph
from SPARQLWrapper import SPARQLWrapper, JSON

def cargar_datos(fuseki_url, archivo_ttl):
    # Cargar el archivo Turtle con rdflib
    g = Graph()
    g.parse(archivo_ttl, format='turtle')

    # Serializar los datos en formato Turtle
    turtle_data = g.serialize(format='turtle')

    # Configurar los headers para la solicitud HTTP
    headers = {
        'Content-Type': 'text/turtle'
    }

    # Hacer la solicitud HTTP PUT para cargar los datos en el dataset
    response = requests.put(fuseki_url, data=turtle_data,
headers=headers)

    # Verificar el estado de la respuesta
    if response.status_code == 200:
        print(f'Datos de {archivo_ttl} cargados exitosamente en Fuseki.')
    else:
        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):
    # Crear un objeto SPARQLWrapper y configurar la consulta y el formato de retorno
    sparql = SPARQLWrapper(fuseki_sparql_url)
    sparql.setQuery(consulta_sparql)
    sparql.setReturnFormat(JSON)

    # Ejecutar la consulta SPARQL
    try:
        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}")
    else:
        print("No se encontraron resultados.")
except Exception as e:
    print(f"Error al ejecutar la consulta SPARQL: {e}")

def menu():
    dataset = 'p16_investigaciones'
    archivo_ttl = 'investigaciones_kg.ttl'
    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 = """
                PREFIX ex: <http://example.org#>
                PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

                SELECT ?name ?president
                WHERE {
                    ?investigation a ex:Indictment ;
                                ex:name ?name ;
                                ex:president ?president ;
                                ex:outcome ex:conviction ;
                                ex:overturned true .
                }
            """
            ejecutar_consulta_sparql_wrapper(fuseki_sparql_url,
consulta_sparql)

```

```
        elif opcion == '3':
            break
        else:
            print("Opción no válida, por favor seleccione nuevamente.")

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