120A3051

Shreya Idate

Batch: E3

Experiment 2

AIM (2.a): To Study Semantic Web Open-Source Tool Protégé.

THEORY:

The Semantic Web is an evolving extension of the World Wide Web in which the semantics of information and services on the web is defined, making it possible for the web to understand and satisfy the requests of people and machines to use the Web content. It derives from W3C director Tim Berners -Lee vision of the Web as a universal medium for data, information and knowledge exchange.

The semantic web addresses this shortcoming, using the descriptive technologies Resource Description Framework (RDF) and Web Ontology Language (OWL), and the data-centric, customizable Extensible Mark-up Language (XML). These technologies are combined in order to provide descriptions that supplement or replace the content of Web documents.

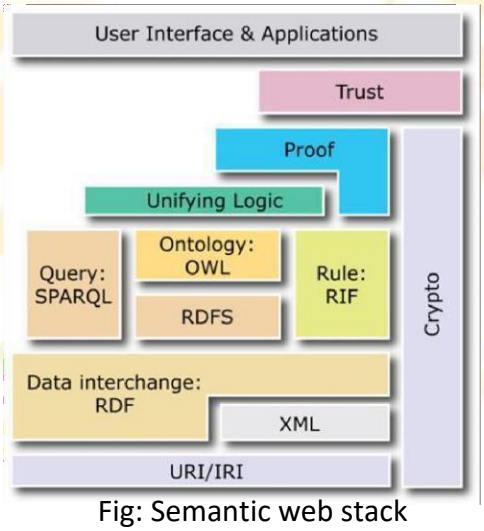


Fig: Semantic web stack

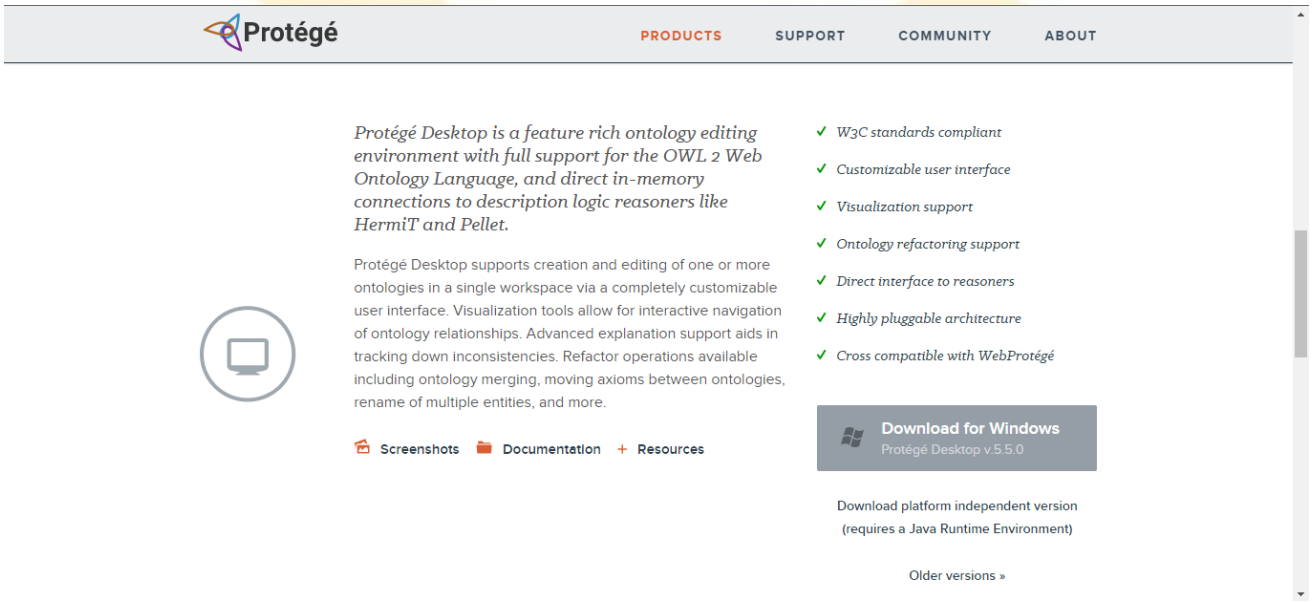
Web Ontology language: The W3C Web Ontology Language (OWL) is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things.

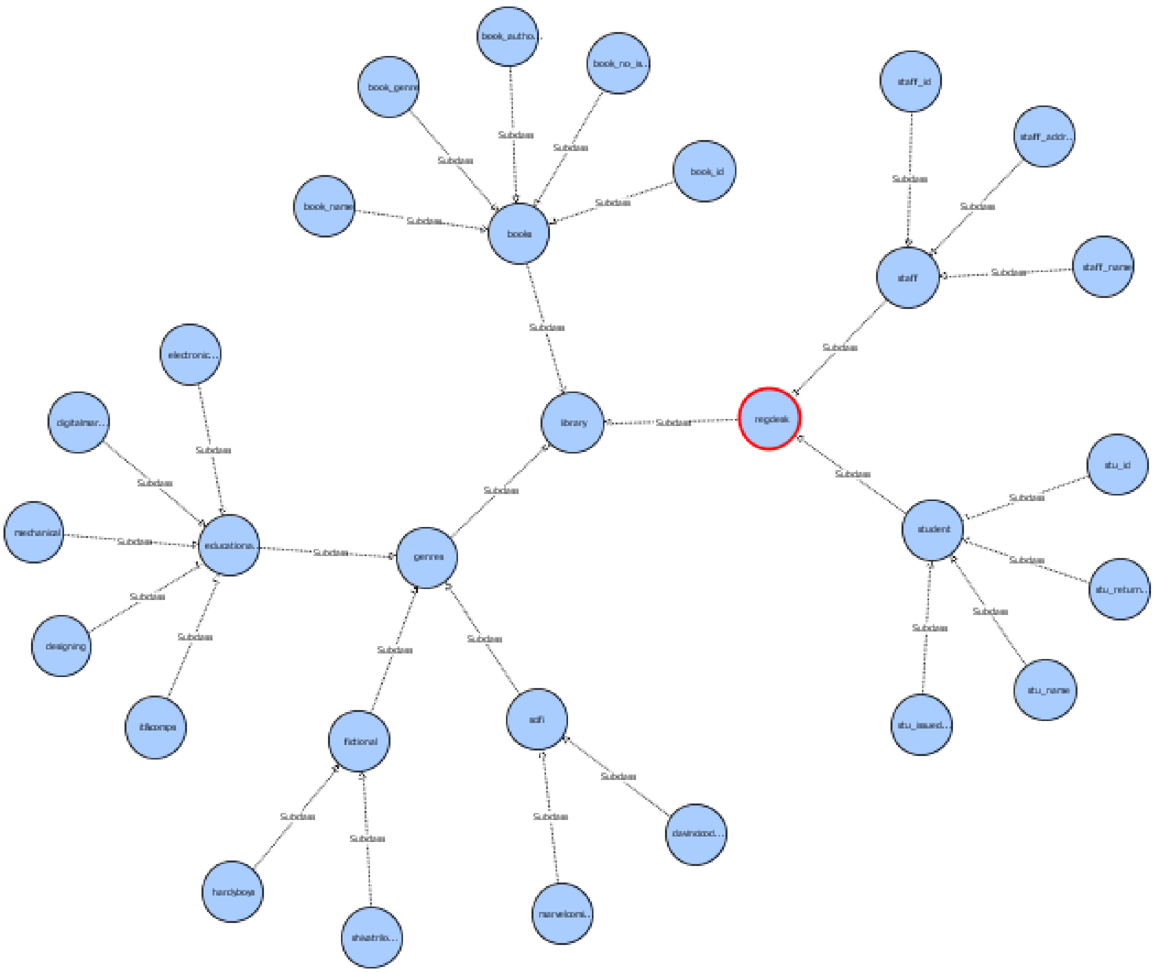
OWL tool Protégé: Protégé is a free, open source ontology editor and a knowledge management system. The Protégé meta-tool was first built by Mark Musen in 1987 and has since been developed by a team at Stanford University. The software is the most popular and widely used ontology editor in the world. Protégé is based on Java, is extensible, and provides a plug-and-play environment that makes it a flexible base for rapid prototyping and application development.

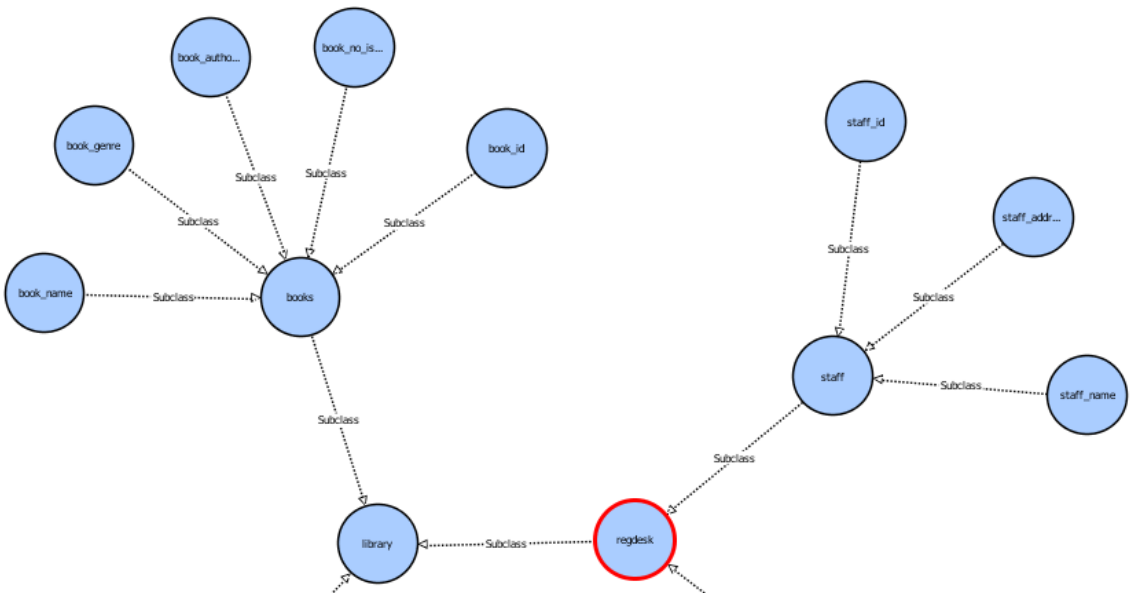
Download Protégé from the link below:
https://protege.stanford.edu/products.php#desktop-protege

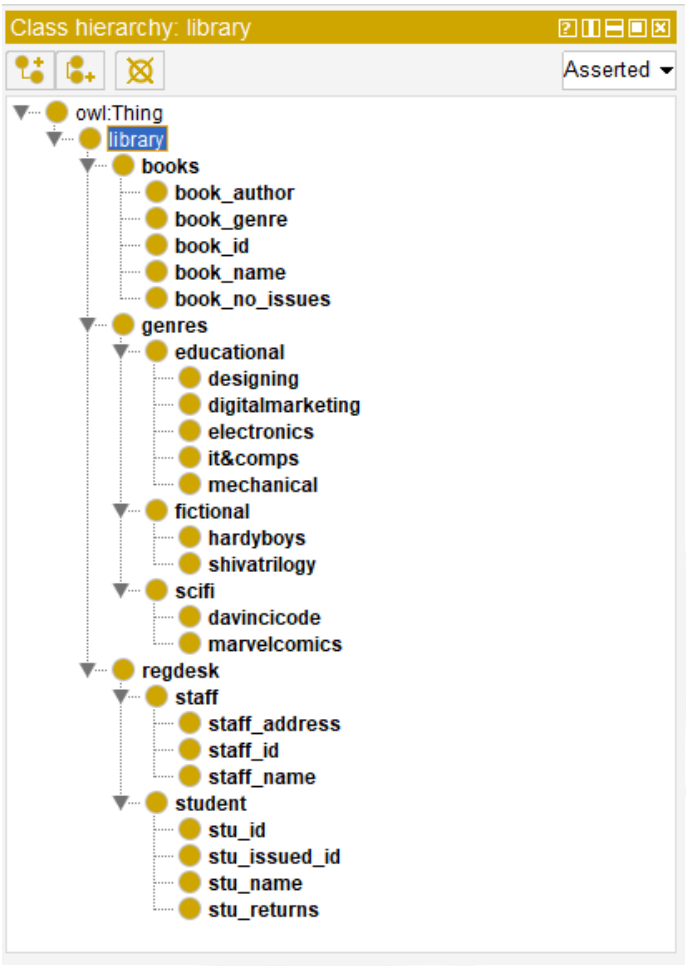
After installation, provide name to file and save as .owl extension.

OUTPUT:




book_genre
book_name
book_id
books
library
regdesk
staff_id
staff
staff_name
stu_id
student
stu_name
electronic...
digitalmar...
mechanical
designing
genres
fictional
hardyboys


book_autho...
book_no_is...
book_genre
book_id
book_name
Subclass
books
library
regdesk
staff_id
staff_addr...
staff
staff_name



CONCLUSION: Web ontology language visualization is implemented using protégé framework successfully.

AIM (2.b): Demo of SPARQL using SPARQL Wrapper – is a simple Python wrapper around a SPARQL service to remotely execute your queries

THEORY:

SPARQL is the standard query language and protocol for Linked Open Data and RDF databases. Having been designed to query a great variety of data, it can efficiently extract information hidden in non-uniform data and stored in various formats and sources. SPARQL has four types of queries. It can be used to: ASK whether there is at least one match of the query pattern in the RDF graph data; SELECT all or some of those matches in a tabular form (including aggregation, sampling and pagination through OFFSET and LIMIT); CONSTRUCT an RDF graph by substituting the variables in these matches in a set of triple templates.

DESCRIBE the matches found by constructing a relevant RDF graph.

Querying DBpedia: The DBpedia data set enables some astonishing queries against Wikipedia data to be answered. You can query the DBpedia data set online via a SPARQL endpoint (described on this page) and as Linked Data.

Public SPARQL Endpoint : A public SPARQL endpoint for querying the DBpedia data set is at http://dbpedia.org/sparql. OpenLink Virtuoso serves as both the back-end database SPARQL query engine and the front- end HTTP/SPARQL server with an nginx overlay primarily to cache results for each submitted query string.The public endpoint does NOT include all available DBpedia data sets. The Loaded Datasets subsection below provides a list of all DBpedia data sets currently loaded into the public SPARQL endpoint.

You can run queries against DBpedia using:

- Rhe OpenLink Interactive SPARQL Query Builder (iSPARQL) at http://dbpedia.org/isparql
- The SNORQL query explorer at http://dbpedia.org/snorql (does not work with Internet Explorer)
- Any other SPARQL-aware client tool

SPARQLWrapper: SPARQLWrapper is a simple Python wrapper around a SPARQL service to remotelly execute your queries. It helps in creating the query invokation and, possibly, convert the result into a more manageable format. How to use You can use SPARQLWrapper either as a Python command line script or as a Python package. Command Line Script To use as a command line script, you will need to install SPARQLWrapper and then a command line script called rqw (spaRQl Wrapper) will be available within the Python environment into which it is installed.

- run $ rql -h to see all the script's options.
- !pip install SPARQLWrapper
- From rdflib import Graph from SPARQLWrapper
- Import SPARQLWrapper, JSON, N3 from pprint Sample execution of Musical Instruments Knowledge base og SPAEQL in RDF using Python language.

Python package

ASK,SELECT,DESCRIBE like queries executed via SPARQLWrapper as a python package.

Steps:

Use jupyter notebook to implement SPARQL using RDF library in python.

Install RDF library and SPARQL Wrapper by below command pip install rdflib.

```
#!/usr/bin/env python
# coding: utf-8

# In[1]:


get_ipython().system('pip install rdflib')


# In[2]:


get_ipython().system('pip install SPARQLWrapper')


# In[3]:


from rdflib import Graph
from SPARQLWrapper import SPARQLWrapper,JSON,N3
from pprint import pprint
```

```
# In[4]:


sparql = SPARQLWrapper('https://dbpedia.org/sparql')
sparql.setQuery('''
    SELECT ?object
    WHERE { dbr:Barack_Obama rdfs:label ?object .}
    # WHERE { dbr:Barack_Obama dbo:abstract ?object .}
''')
sparql.setReturnFormat(JSON)
qres = sparql.query().convert()

# pprint(qres)
for result in qres['results']['bindings']:
    # print(result['object'])

    lang, value = result['object']['xml:lang'], result['object']['value']
    print(f'Lang: {lang}\tValue: {value}')
    # if lang == 'en':
        # print(value)
```

```
sparql = SPARQLWrapper("http://dbpedia.org/sparql")
sparql.setQuery('''
CONSTRUCT { dbc:Machine_learning skos:broader ?parent .
            dbc:Machine_learning skos:narrower ?child .}
WHERE {
    { dbc:Machine_learning skos:broader ?parent . }
UNION
    { ?child skos:broader dbc:Machine_learning . }
}''')

sparql.setReturnFormat(N3)
qres = sparql.query().convert()

g = Graph()
g.parse(data=qres, format='n3')
print(g.serialize(format='ttl'))


# In[37]:


import io
from IPython.display import display
from io import BytesIO
from PIL import Image
import requests

sparql = SPARQLWrapper('https://dbpedia.org/sparql')

instruments = ['Trombone', 'Viola','Drum', 'Saxophone','Pan_flute']

for Musical_instrument in instruments:
    print('##########################################')
    sparql.setQuery(f'''
    SELECT ?name ?comment ?image
    WHERE {{ dbr:{Musical_instrument} rdfs:label ?name.
             dbr:{Musical_instrument} rdfs:comment ?comment.
             dbr:{Musical_instrument} dbo:thumbnail ?image.

        FILTER (lang(?name) = 'en')
        FILTER (lang(?comment) = 'en')
    }}''')

    sparql.setReturnFormat(JSON)
    qres = sparql.query().convert()

    result = qres['results']['bindings'][0]
    name, comment, image_url = result['name']['value'], result['comment']['value'], result['image']['value']

    print(name)
    response = requests.get(image_url)
    display(Image.open(BytesIO(response.content)))
    print(f'{comment}...')
```

```
Collecting rdflib
  Downloading https://files.pythonhosted.org/packages/d0/6b/6454aa1db753c0f8bc265a5bd5c10b5721a4bb24160fb4faf758cf6be8a1/rdflib-5.0.0-py3-none-any.whl (231kB)
Requirement already satisfied: pyparsing in c:\programdata\anaconda3\lib\site-packages (from rdflib)
Collecting isodate (from rdflib)
  Using cached https://files.pythonhosted.org/packages/b6/85/7882d311924cbcfc70b1890780763e36ff0b140c7e51c110fc59a532f087/isodate-0.6.1-py2.py3-none-any.whl
Requirement already satisfied: six in c:\programdata\anaconda3\lib\site-packages (from rdflib)
Installing collected packages: isodate, rdflib
Successfully installed isodate-0.6.1 rdflib-5.0.0

You are using pip version 9.0.1, however version 23.0 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

Collecting SPARQLWrapper
  Downloading https://files.pythonhosted.org/packages/00/9b/443fbe06996c080ee9c1f01b04e2f683b2b07e149905f33a2397ee3b80a2/SPARQLWrapper-1.8.5-py3-none-any.whl
Requirement already satisfied: rdflib>=4.0 in c:\programdata\anaconda3\lib\site-packages (from SPARQLWrapper)
Requirement already satisfied: six in c:\programdata\anaconda3\lib\site-packages (from rdflib>=4.0->SPARQLWrapper)
Requirement already satisfied: pyparsing in c:\programdata\anaconda3\lib\site-packages (from rdflib>=4.0->SPARQLWrapper)
Requirement already satisfied: isodate in c:\programdata\anaconda3\lib\site-packages (from rdflib>=4.0->SPARQLWrapper)
Installing collected packages: SPARQLWrapper
Successfully installed SPARQLWrapper-1.8.5

You are using pip version 9.0.1, however version 23.0 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

Lang: en        Value: Barack Obama
Lang: ar        Value: باراك أوباما
Lang: ca        Value: Barack Obama
Lang: cs        Value: Barack Obama
Lang: el        Value: Μπαράκ Ομπάμα
Lang: de        Value: Barack Obama
Lang: eo        Value: Barack Obama
Lang: es        Value: Barack Obama
Lang: eu        Value: Barack Obama
Lang: fr        Value: Barack Obama
Lang: ga        Value: Barack Obama
Lang: in        Value: Barack Obama
Lang: it        Value: Barack Obama
Lang: ko        Value: 버락 오바마
Lang: ja        Value: バラク・オバマ
Lang: nl        Value: Barack Obama
Lang: pt        Value: Barack Obama
Lang: pl        Value: Barack Obama
Lang: ru        Value: Обама, Барак
Lang: sv        Value: Barack Obama
Lang: uk        Value: Барак Обама
Lang: zh        Value: 贝拉克·奥巴马
b'@prefix dbc: <http://dbpedia.org/resource/Category:> .\n@prefix skos: <http://www.w3.org/2004/02/skos/core#> .\n\ndbc:Machine_learning skos:broader dbc:Artificial_intelligence,\n        dbc:Learning ;\n    skos:narrower dbc:Applied_machine_learning,\n        dbc:Artificial_intelligence_conferences,\n        dbc:Artificial_neural_networks,\n        dbc:Bayesian_networks,\n        dbc:Blockmodeling,\n        dbc:Classification_algorithms,\n        dbc:Cluster_analysis,\n        dbc:Computational_learning_theory,\n        dbc:Data_mining_and_machine_learning_software,\n        dbc:Datasets_in_machine_learning,\n        dbc:Deep_learning,\n        dbc:Dimension_reduction,\n        dbc:Ensemble_learning,\n        dbc:Evolutionary_algorithms,\n        dbc:Genetic_programming,\n        dbc:Inductive_logic_programming,\n        dbc:Kernel_methods_for_machine_learning,\n        dbc:Latent_v
```

```
_programming,\n        dbc:Inductive_logic_programming,\n        dbc:Kernel_methods_for_machine_learning,\n        dbc:Latent_v
ariable_models,\n        dbc:Learning_in_computer_vision,\n        dbc:Log-linear_models,\n        dbc:Loss_functions,\n
dbc:Machine_learning_algorithms,\n        dbc:Machine_learning_researchers,\n        dbc:Machine_learning_task,\n        dbc:Ma
rkov_models,\n        dbc:Natural_language_processing_researchers,\n        <http://dbpedia.org/resource/Category:Ontology_lear
ning_(computer_science)>,\n        dbc:Reinforcement_learning,\n        dbc:Semisupervised_learning,\n        dbc:Signal_proces
sing_conferences,\n        dbc:Statistical_natural_language_processing,\n        dbc:Structured_prediction,\n        dbc:Superv
ised_learning,\n        dbc:Support_vector_machines,\n        dbc:Unsupervised_learning .\n\n'
##########################################
Trombone
```

```
The trombone (German: Posaune, Italian, French: trombone) is a musical instrument in the brass family. As with all brass instru
ments, sound is produced when the player's vibrating lips (embouchure) cause the air column inside the instrument to vibrate. M
ost brass instruments use valves to alter the pitch, but trombones have a telescoping slide mechanism instead. Many modern trom
bone models also have a valve attachment which lowers the pitch of the instrument. Variants such as the valve trombone and supe
rbone have three valves similar to those on the trumpet....
##########################################
Viola
```

```
##########################################
Viola
```

```
The viola (/viˈoʊlə/ vee-OH-lə, also UK: /vaɪˈoʊlə/ vy-OH-lə, Italian: [ˈvjɔːla, viˈɔːla]) is a string instrument that is bowe
d, plucked, or played with varying techniques. Slightly larger than a violin, it has a lower and deeper sound. Since the 18th c
entury, it has been the middle or alto voice of the violin family, between the violin (which is tuned a perfect fifth above) an
d the cello (which is tuned an octave below). The strings from low to high are typically tuned to C3, G3, D4, and A4....
##########################################
Drum
---------------------------------------------------------------------------
OSError                                   Traceback (most recent call last)
<ipython-input-1-0f447d63dc9b> in <module>()
     98     print(name)
     99     response = requests.get(image_url)
--> 100     display(Image.open(BytesIO(response.content)))
    101     print(f'{comment}...')
    102

C:\ProgramData\Anaconda3\lib\site-packages\PIL\Image.py in open(fp, mode)
   2517         fp.close()
   2518     raise IOError("cannot identify image file %r"
-> 2519                   % (filename if filename else fp))
   2520
   2521 #

OSError: cannot identify image file <_io.BytesIO object at 0x0000026BB00ADFC0>
```

Conclusion: Successfully studied SPARQL to remotely execute your queries