

# Wikidata with SPARQL

David Arroyo Menéndez

February 2, 2022

# Wikidata Definition

Wikidata is a collaboratively edited knowledge base hosted by the Wikimedia Foundation. It is a common source of open data that Wikimedia projects such as Wikipedia can use, and anyone else, under a public domain license. This is similar to the way Wikimedia Commons provides storage for media files and access to those files for all Wikimedia projects, and which are also freely available for reuse. Wikidata is powered by the software Wikibase.

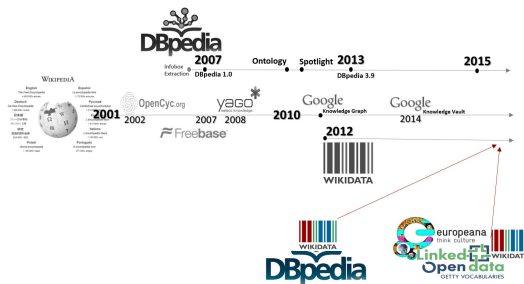
See: <https://www.wikidata.org>

# SPARQL definition

SPARQL is a recursive acronym for SPARQL Protocol and RDF Query Language) is an RDF query language—that is, a semantic query language for databases—able to retrieve and manipulate data stored in Resource Description Framework.

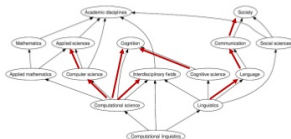
# Linked Open Data. History

## Knowledge Bases: Timeline

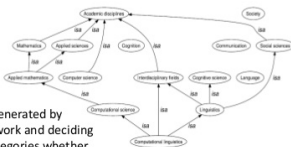


# Linked Open Data. History

(a) Wikipedia category graph with unspecified semantic relations.



(b) Category graph retaining *isa* semantic relations only.



WikiTaxonomy is generated by traversing the network and deciding for each pair of categories whether the sub-category *isa* a super-category.

Hovy et al., Combining open web semi-structured content and Artificial Intelligence: The story so far, *Artificial Intelligence* (2012)

# SPARQL in Python, dbpedia as example

```
from SPARQLWrapper import SPARQLWrapper, JSON

sparql = SPARQLWrapper("http://dbpedia.org/sparql")
sparql.setQuery("""
    PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
    SELECT ?label
    WHERE { <http://dbpedia.org/resource/Asturias> rdfs:label ?label }
""")

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

for result in results["results"]["bindings"]:
    print(result["label"]["value"])
```

# Wikidata: semantic model in a wikipedia page (I)

The image shows a Wikipedia page for Douglas Adams (Q42) with various semantic annotations. The annotations are as follows:

- label**: Points to the name "Douglas Adams (Q42)".
- description**: Points to the text "English writer and humorist".
- aliases**: Points to the text "Douglas Noel Adams | Douglas Noel Adams".
- property**: Points to the "educated at" property in the "Statements" section.
- value**: Points to the value "St John's College" in the "Statements" section.
- qualifiers**: Points to the "end line", "academic major", "academic degree", and "start line" qualifiers in the "Statements" section.
- rank**: Points to the "2 references" rank in the "Statements" section.
- statement group**: Points to the "2 references" statement group in the "Statements" section.
- opened references**: Points to the "opened references" section in the "Statements" section.
- collapsed reference**: Points to the "collapsed reference" section in the "Statements" section.

The "Statements" section contains the following information:

- educated at**: St John's College
- end line**: 1974
- academic major**: English literature
- academic degree**: Bachelor of Arts
- start line**: 1971
- 2 references**:
  - stated in**: Encyclopedia Britannica Online
  - reference URL**: <http://www.britb.com/people/731/000023962/>
  - original language of work**: English
  - retrieved**: 7 December 2013
  - publisher**: NNDG
  - title**: Douglas Adams (English)
- Brentwood School**:
  - end line**: 1970
  - start line**: 1969
- 0 references**

# Wikidata: semantic model in a wikipedia page (II)

- Properties (P) ([https://www.wikidata.org/wiki/Wikidata:List\\_of\\_properties/all\\_in\\_one\\_table](https://www.wikidata.org/wiki/Wikidata:List_of_properties/all_in_one_table))
- Querying for Values (Q)  
(<https://www.wikidata.org/w/index.php?search=>)

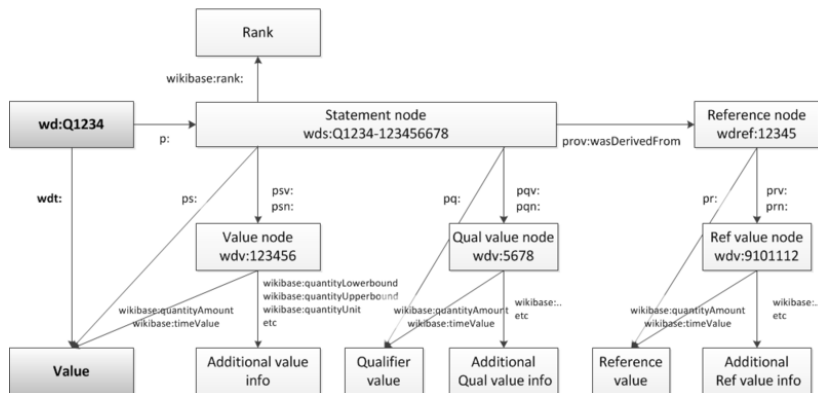


# Wikidata: semantic model in a wikipedia page (III)

	Example
Statements	<code>wd:Q42 wdt:P69 wd:Q691283.</code> <code>or wd:Q42 p:P69 ?s. ?s ps:P69</code> <code>wd:Q691283.</code> <code>or wd:Q42 p:P69 [ ps:P69</code> <code>wd:Q691283 ].</code>
Rank	<code>wd:Q42 p:P69 [ wikibase:rank ?rank</code> <code>].</code>
Qualifier	<code>wd:Q42 p:P69 [ pq:P580 ?qualifier</code> <code>].</code>
Reference	<code>wd:Q42 p:P69 [</code> <code>prov:wasDerivedFrom [ pr:P248 ?</code> <code>ref ] ].</code>

# Wikidata: semantic model in a wikipedia page (IV)

## SPARQL data representation



Item **Johann Sebastian Bach** (q1339)

## Johann Sebastian Bach

Jean-Sébastien Bach | Ėran Бaх | Бaх, Йоганн Себастиан | Бaх | Бaх, Йоганн Себастиан | Bach | J. S. Bach | JS Bach | حac

German composer, organist, harpsichordist, violist, and violinist

male composer /organist from Germany

[See the full family tree](#)

### Relatives

#### Parents

**father** ♂ [Johann Ambrosius Bach](#) wd

**mother** ♀ [Maria Elisabeth Lämmerhirt](#) wd

#### Siblings

**brother** ♂ [Johann Jacob Bach](#) wd

♂ [Johann Christoph Bach](#) wd

#### Children

**child** ♂ [Wilhelm Friedemann Bach](#) wd

♂ [Carl Philipp Emanuel Bach](#) wd

♂ [Johann Christian Bach](#) wd

♂ [Johann Gottfried Bernhard Bach](#) wd

♂ [Johann Christoph Friedrich](#)

#### Other

**spouse** ♀ [Anna Magdalena Bach](#) wd

♀ [Maria Barbara Bach](#) wd

**grandparent** ♂ [Christoph Bach](#) wd

of ♂ [Johann Sebastian Bach](#) wd



*Johann Sebastian Bach.*

GND 11850553X

LCCN n79021425

ISNI 0000 0001 2276 4157

BNF 118897907

IMDb nm0001925

VIAF 12304462

SUDOC 026699656

NDL 00432003

NI A 000035011573

# Wikidata as Language

If you take a look at Germany (Q183), then you can see a whole host of properties like population (P1082), median income (P3529) or even images with the image (P18) property.

```
SELECT
```

```
  ?country ?countryLabel ?population ?area ?medianIncome
```

```
WHERE {
```

```
  ?country wdt:P463 wd:Q458.
```

```
  ?country wdt:P1082 ?population.
```

```
  ?country wdt:P2046 ?area.
```

```
  ?country wdt:P3529 ?medianIncome.
```

```
  SERVICE wikibase:label { bd:serviceParam wikibase:language "es" }
```

# Wikidata in Python (I)

```
import requests
url = 'https://query.wikidata.org/sparql'
query = """
SELECT
    ?country ?countryLabel ?population ?area ?medianIncome
WHERE {
    ?country wdt:P463 wd:Q458.
    ?country wdt:P1082 ?population.
    ?country wdt:P2046 ?area.
    ?country wdt:P3529 ?medianIncome.
    SERVICE wikibase:label { bd:serviceParam wikibase:language "es"
}
}
"""

r = requests.get(url, params = {'format': 'json', 'query': query})
data = r.json()
print(data)
```

# Wikidata in Python (II)

Print ten females in json.

```
import requests
```

```
url = "https://query.wikidata.org/sparql"
```

```
query = """
```

```
SELECT ?name ?nombre ?sexo_o_g_nero ?sexo_o_g_neroLabel WHERE {  
    ?human wdt:P31 wd:Q5.
```

```
    OPTIONAL { ?human wdt:P21 ?nombre. }
```

```
    OPTIONAL { ?human wdt:P21 ?sexo_o_g_nero. }
```

```
}
```

```
LIMIT 10"""
```

```
r = requests.get(url, params = {'format': 'json', 'query': query})
```

```
data = r.json()
```

```
print(data)
```

# Wikidata in Python (III)

Print ten cats in json:

```
import requests
url = "https://query.wikidata.org/sparql"
query = """#added before 2016-10
#Cats
SELECT ?item ?itemLabel
WHERE
{
    ?item wdt:P31 wd:Q146.
    SERVICE wikibase:label { bd:serviceParam wikibase:language "
}
LIMIT 10
"""
r = requests.get(url, params = {'format': 'json', 'query': query})
data = r.json()
print(data['results']['bindings'])
```

# Wikidata in Python (IV)

Full examples!

```
$ python3 wikidata-female-artists.py  
$ python3 wikidata-female-scientists.py  
$ python3 female-streets-fr.py  
$ python3 wikispecies.py
```



- Wikidata: a free collaborative knowledge base
- <https://www.wikidata.org>
- <https://tools.wmflabs.org/reasonator/>