

# SPARQL



**SPARQL**  
**Protocol**  
**And**  
**RDF**  
**Query**  
**Language**





# Introduction -

- ▶ **SPARQL** (pronounced "sparkle", 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 (RDF) format.
- ▶ SPARQL allows for a query to consist of triple patterns, conjunctions, disjunctions and optional patterns.
- ▶ The results of SPARQL queries can be result sets or RDF graphs.



The SPARQL language specifies four different query variations for different purposes -

**SELECT query**

Used to extract raw values from a SPARQL endpoint, the results are returned in a table format.

**CONSTRUCT query**

Used to extract information from the SPARQL endpoint and transform the results into valid RDF.

**ASK query**

Used to provide a simple True/False result for a query on a SPARQL endpoint.

**DESCRIBE query**

Used to extract an RDF graph from the SPARQL endpoint, the content of which is left to the endpoint to decide based on what the maintainer deems as useful information.

# Document Conventions

- ▶ To make URI's simpler to write, RDF's TURTLE Serialization format uses abbreviated prefixes for URIs.

```
@prefix dc: <http://purl.org/dc/elements/1.1/> .  
@prefix : <http://example.org/book/> .  
:book1 dc:title "SPARQL Tutorial" .
```

- ▶ Result sets are illustrated in tabular form

x	y	z
"Alice"	<http://example/a>	

- ▶ A 'binding' is a pair - Variable and Value
  - X → Alice
  - Y → <http://example/a>
  - Z → not bound to an RDF term. Variables are not required to be bound in a solution.

Schema information (the ontology) is often provided externally, though, to allow different datasets to be joined in an unambiguous manner.

```
@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .
@prefix sn: <http://www.snee.com/hr/> .

sn:emp1 vcard:given-name "Heidi".
sn:emp1 vcard:family-name "Smith".
sn:emp1 vcard:title "CEO".
sn:emp1 sn:hireDate "2015-01-13".
sn:emp1 sn:completedOrientation "2015-01-30".

sn:emp2 vcard:given-name "John".
sn:emp2 vcard:family-name "Smith".
sn:emp2 vcard:title "Engineer".
sn:emp2 sn:hireDate "2015-01-28".
sn:emp2 sn:completedOrientation "2015-01-30".

sn:emp3 vcard:given-name "Francis".
sn:emp3 vcard:family-name "Jones".
sn:emp3 vcard:title "Vice President".
sn:emp3 sn:hireDate "2015-02-13".
```

Standard 'vcard' ontology

```
@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .
@prefix sn: <http://www.snee.com/hr/> .

sn:emp1 vcard:given-name "Heidi".
sn:emp1 vcard:family-name "Smith".
sn:emp1 vcard:title "CEO".
sn:emp1 sn:hireDate "2015-01-13".
sn:emp1 sn:completedOrientation "2015-01-30".

sn:emp2 vcard:given-name "John".
sn:emp2 vcard:family-name "Smith".
sn:emp2 vcard:title "Engineer".
sn:emp2 sn:hireDate "2015-01-28".
sn:emp2 sn:completedOrientation "2015-01-30".

sn:emp3 vcard:given-name "Francis".
sn:emp3 vcard:family-name "Jones".
sn:emp3 vcard:title "Vice President".
sn:emp3 sn:hireDate "2015-02-13".
```

Customized domain name with property names and values

## Structure of a SPARQL Query -

A SPARQL query comprises, in order:

- *Prefix declarations*, for abbreviating URIs
- *Dataset definition*, stating what RDF graph(s) are being queried
- A *result clause*, identifying what information to return from the query
- The *query pattern*, specifying what to query for in the underlying dataset
- *Query modifiers*, slicing, ordering, and otherwise rearranging query results

# prefix declarations

PREFIX foo: <http://example.com/resources/> ...

# dataset definition

FROM ...

# result clause

SELECT ...

# query pattern

WHERE { ... }

# query modifiers

ORDER BY ...



# Example – Dataset of Employees

empNum	given-name	family-name	title	hireDate	completedOrientation
emp1	Heidi	Smith	CEO	2015-01-13	2015-01-30
emp2	John	Smith	Engineer	2015-01-28	2015-01-30
emp3	Francis	Jones	Vice President	2015-02-13	
emp4	Jane	Berger	Sales	2015-03-10	

List the names of all the Employees whose family name is  
"Smith"

The diagram illustrates an SPARQL query and its results. The query is shown in a box labeled "Query" on the left. The results are shown in a box labeled "Result" on the left. Red arrows point from labels to specific parts of the query and results.

**Query:**

```
PREFIX vcard: <http://www.w3.org/2006/vcard/ns#>  
SELECT ?person  
WHERE  
{  
  ?person vcard:family-name "Smith".  
}
```

**Result:**

?person
sn:emp1
sn:emp2

**Annotations:**

- Subject:** Points to `?person` in the `SELECT` clause.
- Predicate:** Points to `vcard:family-name` in the `WHERE` clause.
- Object:** Points to `"Smith"` in the `WHERE` clause.
- Value of the Variable to be Listed:** Points to `?person` in the `SELECT` clause.
- Prefix – No need to use URI's in Query:** Points to the `PREFIX` line.



## Query

```
PREFIX vcard: <http://www.w3.org/2006/vcard/ns#>
PREFIX sn: <http://www.snee.com/hr/>

SELECT ?givenName ?familyName ?fullName
WHERE
{
  ?person vcard:given-name ?givenName .
  ?person vcard:family-name ?familyName .
  BIND(concat(?givenName," ",?familyName) AS ?fullName)
}
```

## Result

?givenName	?familyName	?fullName
John	Smith	John Smith
Heidi	Smith	Heidi Smith
Jane	Berger	Jane Berger
Francis	Jones	Francis Jones