SPARQL

SPARQL Protocol and
RDF Query Language

SPARQL

- SPARQL Query Language (1.1)
  - SELECT
  - CONSTRUCT
  - ASK
  - DESCRIBE

- SPARQL UPDATE (1.1)
  - INSERT
  - DELETE

SELECT

SELECT

WILDCARD - MATCH ALL

```
# QUERY 1: WILDCARD QUERY RETURNING ALL VALUES FOR ALL
VARIABLES MATCHING ALL TRIPLES
SELECT *
WHERE {
  ?subject ?predicate ?object .
}
```

SELECT
MATCH PUMP

```
# QUERY 2A: RETURNING ALL SUBJECTS THAT SOMEHOW IS RELATED TO A
PUMP
SELECT ?subject
WHERE {
  ?subject ?predicate <http://www.webstep.no/workshop/ens/Pump> .
}
```

SELECT
MATCH PUMP

```
# QUERY 2A: RETURNING ALL SUBJECTS THAT SOMEHOW IS RELATED TO A
PUMP
SELECT ?subject
WHERE {
  ?subject ?predicate <http://www.webstep.no/workshop/ens/Pump> .
}
```

```
# QUERY 2B: SAME AS ABOVE, BUT WITH PREFIX
PREFIX ens:   <http://www.webstep.no/workshop/ens/>

SELECT ?subject
WHERE {
  ?subject ?predicate ens:Pump .
}
```

SELECT

AGGREGATE - COUNT PREDICATE

```
# QUERY 3: COUNTING THE NUMBER OF TIMES A PREDICATE IS USED
SELECT ?predicate (count(?predicate) AS ?numberOf)
WHERE {
  ?subject ?predicate ?object .
}
GROUP BY ?predicate
# ORDER BY DESC(?numberOf)
```

SELECT

AGGREGATE - COUNT PUMP

```
# QUERY 4: CAN WE COUNT INSTANCES OF PUMP?
SELECT ?pump (count(?pump) AS ?numberOf)
WHERE {
  ?subject ?predicate ?pump .
}
GROUP BY ?pump
```

SELECT
AGGREGATE - COUNT PUMP

```
# QUERY 5A: COUNT INDIVIDUALS THAT IS AT LEAST OF TYPE PUMP
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
SELECT ?pump (count(?pump) AS ?numberOf)
WHERE {
  ?subject a ens:Pump .
  ?subject a ?pump .
}
GROUP BY ?pump
```

SELECT

AGGREGATE - COUNT PUMP

```
# QUERY 5A: COUNT INDIVIDUAL THAT IS AT LEAST OF TYPE PUMP
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
SELECT ?pump (count(?pump) AS ?numberOf)
WHERE {
  ?subject a ens:Pump .
  ?subject a ?pump .
}
GROUP BY ?pump
```

```
# QUERY 5B: GRAPH PATTERNS IN SPARQL CAN BE WRITTEN AS TURTLE
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
SELECT ?pump (count(?pump) AS ?numberOf)
WHERE {
  ?subject a ens:Pump , ?pump .
}
GROUP BY ?pump
```

SELECT
AGGREGATE - SUMMATION

```
# QUERY 6: SUMMATION OF PROPERTIES FOR THE SUBMERGED PUMPS
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
SELECT ?flowProperty  (sum(?normalFlow) AS ?totalFlow)
WHERE {
  ?subject a ens:Pump_PH ;
           ?flowProperty ?normalFlow .
}
GROUP BY ?flowProperty
```

SELECT
FILTER

```
# QUERY 7: FILTERING SO THAT ONLY NORMAL FLOW IS SUMMED
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
SELECT ?flowProperty  (sum(?normalFlow) AS ?totalFlow)
WHERE {
  ?subject a ens:Pump_PH ;
           ?flowProperty ?normalFlow .

  FILTER(?flowProperty = ens:normalFlow_m3h)
}
GROUP BY ?flowProperty
```

SELECT
OPTIONAL

```
# QUERY 8: ONLY GET PUMPS WHERE THE NORMAL FLOW HAS A VALUE
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
SELECT ?pump ?normalFlow
WHERE {
  ?pump a ens:Pump ;
          ens:normalFlow_m3h ?normalFlow .
}
```

SELECT
OPTIONAL

```
# QUERY 9: GET PUMPS WITH OR WITHOUT NORMAL FLOW VALUES
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
SELECT ?pump ?normalFlow
WHERE {
  ?pump a ens:Pump .
  OPTIONAL { ?pump ens:normalFlow_m3h ?normalFlow . }
}
```

CONSTRUCT

CONSTRUCT
COPY GRAPH

```
# QUERY 10: COPY THE ENTIRE GRAPH
CONSTRUCT { ?s ?p ?o . }
WHERE {
  ?s ?p ?o .
}
```

CONSTRUCT

RETURN TRIPLES FOR A NAMED INDIVIDUAL

```
# QUERY 11: RETURN TRIPLES FOR AN INDIVIDUAL
PREFIX data:       <http://www.webstep.no/workshop/data/>
CONSTRUCT { data:50_PH50B ?p ?o . }
WHERE {
  data:50_PH50B ?p ?o .
}
```

CONSTRUCT

RETURN NESTED TRIPLES FOR A NAMED INDIVIDUAL

```
# QUERY 12: RETURN NESTED TRIPLES FOR AN INDIVIDUAL
PREFIX data:      <http://www.webstep.no/workshop/data/>
CONSTRUCT
{
  ?individual ?firstLevelProperty ?firstLevelObject .
  ?firstLevelObject ?secondLevelProperty ?secondLevelObject
}
WHERE
{
  ?individual ?firstLevelProperty ?firstLevelObject .

  OPTIONAL { ?firstLevelObject ?secondLevelProperty?secondLevelObject . }

  FILTER(?individual = data:50_PH50B)
}
```

CONSTRUCT
VARIABLE ASSIGNMENT

```
# QUERY 13: CREATE AN IRI AND BIND IT TO A NEW VARIABLE
PREFIX data:  <http://www.webstep.no/workshop/data/>
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
CONSTRUCT
{
  ?systemIRI a ens:System
}
WHERE
{
  ?pump a ens:Pump ;
        ens:partOfSystem ?system .

  BIND(IRI(CONCAT(STR(data:), ?system)) AS ?systemIRI )
}
```

CONSTRUCT

CREATE NEW DATA

```
# QUERY 14: EXTEND QUERY 13 TO PRODUCE MORE TRIPLES
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX data:  <http://www.webstep.no/workshop/data/>
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
CONSTRUCT
{
  ?systemIRI a ens:System ;
             rdfs:label ?system .
  ?pump ens:partOfSystem ?systemIRI .
}
WHERE
{
  ?pump a ens:Pump ;
        ens:partOfSystem ?system .

  BIND(IRI(CONCAT(STR(data:), ?system)) AS ?systemIRI )
}
```

ASK

ASK

DOES GRAPH PATTERN EXIST?

```
# QUERY 15: DOES THE SYSTEM DATA CREATED IN QUERY 14 EXIST?
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX data:  <http://www.webstep.no/workshop/data/>
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
ASK
WHERE
{
  ?systemIRI a ens:System ;
             rdfs:label ?system .
  ?pump ens:partOfSystem ?systemIRI .
}
```

ASK

DOES GRAPH PATTERN EXIST?

```
# QUERY 16: DOES AT LEAST 1 PUMP THAT IS PART OF A SYSTEM AND AT A
LOCATION DEPTH OF 16.0 EXIST?
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
ASK
WHERE
{
  ?pump a ens:Pump ;
        ens:partOfSystem ?system ;
        ens:locationDepth [
             ens:hasValue 16.0
         ]
}
```

DESCRIBE

DESCRIBE

WHAT IS KNOWN ABOUT AN INDIVIDUAL?

```
# QUERY 17: WHAT IS KNOWN ABOUT A SPECIFIC INDIVIDUAL
PREFIX data:  <http://www.webstep.no/workshop/data/>
DESCRIBE data:50_PH50B
```

DESCRIBE
BLANK NODES

```
# QUERY 18: WHAT IS KNOWN ABOUT A SPECIFIC INDIVIDUAL WITH BLANK NODES?
PREFIX data:  <http://www.webstep.no/workshop/data/>
DESCRIBE data:50_PH50A
```

DESCRIBE
MULTIPLE INDIVIDUAL

```
# QUERY 18: WHAT IS KNOWN ABOUT A GIVEN PUMP AND ITS LOCATION DEPTH?
PREFIX data:  <http://www.webstep.no/workshop/data/>
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
DESCRIBE ?pump ?locationDepth
WHERE
{
     ?pump a ens:Pump_PH ;
           ens:locationDepth ?locationDepth .

     FILTER(?pump = data:50_PH50B)
}
```

UPDATE

INSERT
CREATE NEW DATA

```
# QUERY 19: INSERT NEW DATA
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX data:  <http://www.webstep.no/workshop/data/>
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
INSERT
{
  data:System50 a ens:System ;
                rdfs:label "System 50"
}
```

INSERT

PREPARATION : CREATE DATA FROM GRAPH PATTERNS

```
# QUERY 20: PREPARATIONS FOR CREATING NEW RELATIONSHIPS
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
CONSTRUCT
{
    ?pump ens:partOfSystem ?systemIri .
}
WHERE
{
  ?system a ens:System .
  ?pump ens:partOfSystem ?systemLabel .

  BIND(IF(CONTAINS(STR(?system), ?systemLabel), ?system, ?undefined) AS ?systemIri)
}
```

INSERT

CREATE DATA FROM GRAPH PATTERNS

```
# QUERY 21: INSERTING NEW RELATIONSHIPS
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
INSERT
{
    ?pump ens:partOfSystem ?systemIri .
}
WHERE
{
  ?system a ens:System .
  ?pump ens:partOfSystem ?systemLabel .

  BIND(IF(CONTAINS(STR(?system), ?systemLabel), ?system, ?undefined) AS ?systemIri)
}
```

DELETE

PREPARATION : DELETE DATA FROM GRAPH PATTERNS

```
# QUERY 22: PREPARATIONS FOR DELETING OLD RELATIONSHIPS
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
CONSTRUCT
{
     ?pump ens:partOfSystem ?system .
}
WHERE
{
  ?pump a ens:Pump ;
        ens:partOfSystem ?system .

  #FILTER ISLITERAL(?system)
}
```

DELETE

DELETE DATA FROM GRAPH PATTERNS

```
# QUERY 23: DELETING OLD RELATIONSHIPS
PREFIX ens:   <http://www.webstep.no/workshop/ens/>
CONSTRUCT
{
     ?pump ens:partOfSystem ?system .
}
WHERE
{
  ?pump a ens:Pump ;
        ens:partOfSystem ?system .

  FILTER ISLITERAL(?system)
}
```

CONSTRUCT
REVIEWING RESULT

```
# QUERY 24(10): REVIEW THE GRAPH
CONSTRUCT { ?s ?p ?o . }
WHERE {
  ?s ?p ?o .
}
```