

SPARQL / FILTER

Федоренко Екатерина 22605
2021

FILTER

FILTER (условие) - это выражение, которое вы можете вставить в свой запрос SPARQL, чтобы отфильтровать результаты. Внутри круглых скобок вы можете поместить любое выражение логического типа, и будут использоваться только те результаты, в которых выражение возвращает истину.

List of SPARQL Filter Functions

Logical: !, &&, ||

- ! - not, negation
- Example: (!true) is the same as false
- &&: and
- Example: (?age > 10 && ?age < 15) includes anyone in the age range from 10-15.
- ||: or
- Example: (?age < 10 || ?age > 15) includes anyone younger than 10 or older than 15

Math: +, -, *, /

- +: addition
- -: subtraction
- *: multiplication
- /: division

Comparison: =, !=, >, <, IN, NOT IN...

- =: is equal to
- !=: is not equal to
- > <: greater than and less than
- IN, NOT IN: determine if an item is or is not in a set
- Example: (?Ford IN ?cars) should hopefully return true

SPARQL tests: isIRI, isURI, isBlank, isLiteral, isNumeric, bound

- isIRI, isURI: returns true if the term is an IRI or a URI
- isBlank: returns true if the term is a blank node
- isLiteral: returns true if the term is a literal
- isNumeric: returns true if the term is a numeric value
- bound: returns true if a variable is assigned a value (INF or infinite, and NaN or not a number are considered valid values)

List of SPARQL Filter Functions

- SPARQL accessors: str, lang, datatype
- Other: sameTerm, langMatches, regex, REPLACE
- Conditionals: IF, COALESCE, EXISTS, NOT EXISTS
- Constructors: URI, BNODE, STRDT, STRLANG, UUID, STRUUID
- Strings: STRLEN, SUBSTR, UCASE, LCASE, STRSTARTS, STRENDS, CONTAINS, STRBEFORE, STRAFTER, CONCAT, ENCODE_FOR_URI
- More math: abs, round, ceil, floor, RAND
- Date/time: now, year, month, day, hours, minutes, seconds, timezone, tz
- Hashing: MD5, SHA1, SHA256, SHA384, SHA512

FILTER on values

Чтобы получить список всех людей, родившихся в 2015 году, мы сначала получаем всех людей с датой их рождения. А затем отфильтруем, чтобы получить только те результаты, в которых год рождения - 2015. Это можно сделать двумя способами: извлечь год даты с помощью функции ГОД и проверить, что это 2015 год -

```
SELECT ?person ?personLabel ?dob
WHERE
{
  ?person wdt:P31 wd:Q5;
          wdt:P569 ?dob.
  SERVICE wikibase:label { bd:serviceParam wikibase:language "en". }
}
```

```
FILTER(YEAR(?dob) = 2015).
```

или убедиться, что это дата между 1 января (включительно) 2015 года и 1 января 2016 года:

```
FILTER("2015-01-01"^^xsd:date <= ?dob && ?dob < "2016-01-01"^^xsd:date).
```

FILTER on values

Кажется, что первый более простой, но, как оказалось, второй намного быстрее:

```
SELECT ?person ?personLabel ?dob
WHERE
{
  ?person wdt:P31 wd:Q5;
          wdt:P569 ?dob.
  FILTER("2015-01-01"^^xsd:dateTime <= ?dob && ?dob < "2016-01-01"^^xsd:dateTime).
  SERVICE wikibase:label { bd:serviceParam wikibase:language "en". }
}
```

FILTER EXAMPLES

Humans whose gender we know we don't know

```
#title: Humans whose gender we know we don't know  
# Demonstrates filtering for "unknown value"  
SELECT ?human ?humanLabel  
WHERE  
{  
  ?human wdt:P21 ?gender .  
  FILTER wikibase:isSomeValue(?gender)  
  SERVICE wikibase:label { bd:serviceParam wikibase:language "[AUTO_LANGUAGE],en"  
}  
}
```

Humans whose gender we know we don't know

FILTER EXAMPLES

Lexemes in English that match an expression

```
# Lexemes in English that match an expression  
select ?lexemeId ?lemma WHERE {  
  ?lexemeId dct:language wd:Q1860;  
  wikibase:lemma ?lemma.  
  # only those lemmas that begin with "pota", i.e. "potato"  
  FILTER (regex(?lemma, '^pota.*'))  
}
```

[Try it!](#)

FILTER EXAMPLES

Rock bands that start with "M"

```
SELECT ?band ?bandLabel
WHERE
{
  ?band wdt:P31 wd:Q5741069 .
        rdfs:label ?bandLabel .
  FILTER(LANG(?bandLabel) = "en") .
  FILTER(STRSTARTS(?bandLabel, 'M')) .
}
```

Try it!