

Základy syntaxe SPARQL

```
SELECT ?a ?b ?c
WHERE
{
  x y ?a.
  m n ?b.
  ?b f ?c.
}
```

jako například

```
SELECT ?budova
WHERE
{
  ?budova konstrukce pernikova.
  ?budova pocetPodlazi 1.
}
```

kde `?budova konstrukce pernikova.` vyjadřuje sémantickou vazbu *subjekt, predikát, objekt* (tzv. triple).

Tento (nebo jakýkoli jiný dotaz) se pak zadává v tzv. SPARQL endpoint. Jedná se o URL adresu místa, kde klient může komunikovat (dotazovat se) na server.

Základní rozhraní pro dotazy z DBpedia.org

<http://dbpedia.org/snorql/>

Úkol #1

Zkuste přepsat níže uvedený příklad do formy, která vám umožní zeptat se na významné rodáky vašeho města od roku 1900. Následně zkuste seřadit podle data narození.

```
SELECT ?name ?birth ?death ?person WHERE
{
  ?person dbo:birthPlace ?birthPlace .
  ?person dbo:birthDate ?birth .
  ?person foaf:name ?name .
  ?person dbo:deathDate ?death .
  FILTER (?birth < "1900-01-01"^^xsd:date) . } ORDER BY ?name
```

Nápovědy:

- 1) *Stávající syntaxe vám vypíše všechny rodáky všech měst v celém světě až po toho, kde dojde limit na délku odpovědi. Proto to chvíli trvá.*
- 2) *V syntaxi výše je nezbytné nahradit objekt místa narození konkrétní hodnotou. Konkrétní hodnoty se zapisují dvojtečkou a hodnotou.*
- 3) *Název města skládající se z více slov pojí podtržítko. Nepoužívejte diakritiku.*

Následující příklad umožní ukázat pouze hudební skladatele z Brna.

```
SELECT ?name ?birth ?description ?person WHERE {
  ?person a dbo:MusicalArtist .
  ?person dbo:birthPlace :Brno .
  ?person foaf:name ?name .
  ?person rdfs:comment ?description .
  FILTER (LANG(?description) = 'en') .
} ORDER BY ?name
```

Úkol #2

Zkuste přepsat výše uvedený příklad do formy, která vám umožní získat v odpovědi i datum narození hudebních skladatelů.

Sémantických databází je dnes neuvěřitelné množství, pro připomenutí informací z přednášky se podívejte na <http://lod-cloud.net/>.

V rámci těchto databází existuje také SPARQL endpoint <http://linkedgeodata.org/sparql>

Základní rozhraní pro prostorové dotazy

<http://linkedgeodata.org/sparql>

Syntaxe níže ukazuje dotaz na data z OpenStreetMap ve vzdálenosti 0,1 km od stanoveného místa (se souřadnicemi 16.597, 49.204).

```
Prefix lgdo: <http://linkedgeodata.org/ontology/>
Prefix geom: <http://geovocab.org/geometry#>
Prefix ogc:<http://www.opengis.net/ont/geosparql#>
Select *
From <http://linkedgeodata.org> {
  ?s
  a lgdo:Amenity ;
  rdfs:label ?l ;
  geom:geometry [
    ogc:asWKT ?g
  ] .

  Filter(bif:st_intersects (?g, bif:st_point (16.597, 49.204),
0.1)) .
}
```

Úkol #3

Zkuste přepsat výše uvedený příklad do formy, která vám umožní získat pekařství v místě vašeho zájmu.

Nápověda:

Amenity je univerzální název pro jakýkoli bod zájmu v OpenStreetMap. Tzv. Linked Geodata Ontology (<http://linkedgeodata.org/ontology>) odkazuje na tagy OpenStreetMap, viz <http://learnosm.org/en/hot-tips/tagging/>.