



## SPARQL & GeoSPARQL Course



Tanja



Wouter

# Learning goals

## First hour:

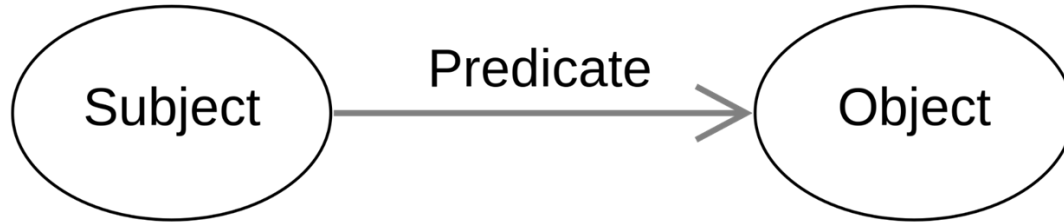
- Write and run basic SPARQL queries
- Learn SPARQL keywords; bind, filter, limit, order by, prefix, select
- Understand how SPARQL can be used to query linked data / graph data

## Second hour:

- Write and run basic GeoSPARQL queries
- Learn how GeoSPARQL data is represented: geo:hasGeometry, geo:asWKT
- Learn GeoSPARQL functions: geof:distance, geof:sfWithin
- Understand how GeoSPARQL can be used to query linked geodata

# First hour: SPARQL

# SPARQL is optimized for querying triples



# Triple pattern ([Try it out!](#))

```
select ?s ?p ?o {  
  ?s ?p ?o.  
}  
limit 10
```

Exercise: How many triples are in the dataset?

## Projection (columns)

```
select ?s ?p ?o
```

## Pattern (cells)

```
{ ?s ?p ?o. }
```

## Modifier (rows)

```
limit 10
```

# Triple pattern: IRI

```
select ?feature ?label {  
  ?feature <http://www.w3.org/2004/02/skos/core#prefLabel>  
  ?label.  
}
```

# Prefix declaration ([Try it out!](#))

```
prefix skos: <http://www.w3.org/2004/02/skos/core#>
select ?feature ?label {
  ?feature skos:prefLabel ?label.
}
```

## Order by ([Try it out!](#))

```
prefix skos: <http://www.w3.org/2004/02/skos/core#>
select ?feature ?label {
  ?feature skos:prefLabel ?label.
}
order by ?label
```



## Triple pattern: IRI + literal ([Try it out!](#))

```
prefix skos: <http://www.w3.org/2004/02/skos/core#>
select ?feature {
  ?feature skos:prefLabel "Hilversum Media Park"@nl.
}
```

Results should be dereferenceable, for example:

<https://data.osm.pldn.nl/node/7606656810>

## Filter ([Try it out!](#))

```
prefix skos: <http://www.w3.org/2004/02/skos/core#>
select ?feature ?label {
  ?feature skos:prefLabel ?label.
  filter(contains(str(?label), "Hilversum"))
}
```

Do it yourself: which features contain "Amsterdam" in their label?

# Graph pattern ([Try it out!](#))

```
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
prefix skos: <http://www.w3.org/2004/02/skos/core#>
select ?label ?url {
  ?feature skos:prefLabel ?label.
  ?feature rdfs:seeAlso ?url.
}
```

## Bind ([Try it out!](#))

```
prefix skos: <http://www.w3.org/2004/02/skos/core#>
select ?label ?length {
  ?feature skos:prefLabel ?label.
  bind(strlen(?label) as ?length)
}
limit 10
```

Exercise: Return the length of the ?feature IRIs.

Back at 14:45





# Second hour: GeoSPARQL



# What is geo data?

Is it only coordinates?

What about names?

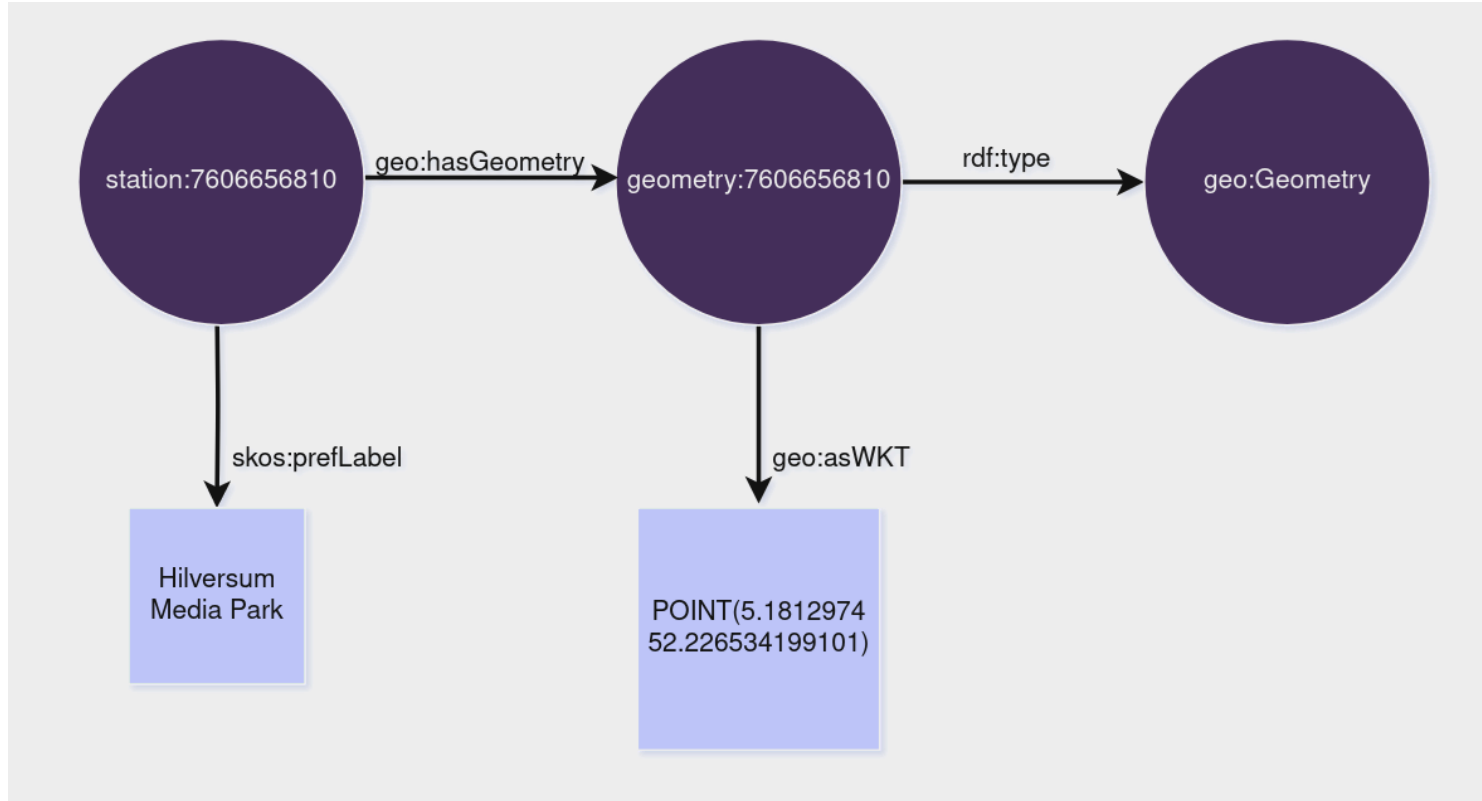


# How do we model geo data?

Let's look into the following predicates:

1. `geo:hasGeometry`
2. `geo:asWKT`

# Let's take a look at triples



# Let's look at the *geo:hasGeometry* relation ([Try it out](#))

```

1 v prefix geo: <http://www.opengis.net/ont/geosparql#>
2 v select * where {
3   ?feature geo:hasGeometry ?geometry.
4 }

```



[Table](#)
[Response](#)
[Gallery](#)
[Chart](#)
[Geo](#)
[Geo-3D](#)
[Geo events](#)
[Markup](#)
[Network](#)
[Pivot](#)
[Timeline](#)

15 results in 0.096 seconds

Simple view  Ellipse

Filter query results

Page size: 50



|   | feature  | geometry  |
|---|--|---|
| 1 | <https://data.labs.kadaster.nl/cbs/wbk/id/gemeente/0307> | <https://data.labs.kadaster.nl/geometry/0307>       |
| 2 | <https://data.osm.pldn.nl/way/136447263>                 | <https://data.osm.pldn.nl/geometry/way/136447263>   |
| 3 | <https://data.labs.kadaster.nl/cbs/wbk/id/gemeente/0402> | <https://data.labs.kadaster.nl/geometry/0402>       |
| 4 | <https://data.osm.pldn.nl/node/2672775963>               | <https://data.osm.pldn.nl/geometry/node/2672775963> |
| 5 | <https://data.osm.pldn.nl/node/2839615144>               | <https://data.osm.pldn.nl/geometry/node/2839615144> |

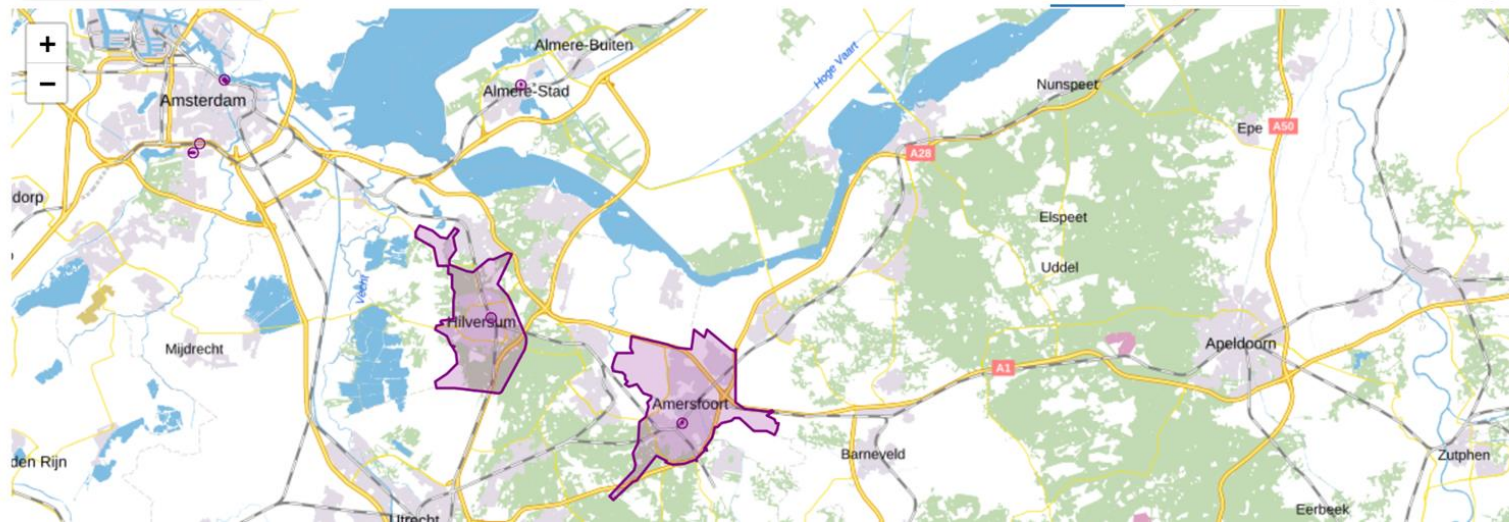
# Let's see objects on a map ([Try it out!](#))

```
1 v prefix geo: <http://www.opengis.net/ont/geosparql#>
2 v select * where {
3   ?feature geo:hasGeometry ?geometry.
4   ?geometry geo:asWKT ?geo.
5   bind ("purple" as ?geoColor)
6 }
```

Table Response Gallery Chart **Geo** Geo-3D Geo events Markup Network Pivot Timeline

15 results in 0.144 seconds

Normal Grouped Heatmap Layers ?



# Let's add labels ([Try it out!](#))

```

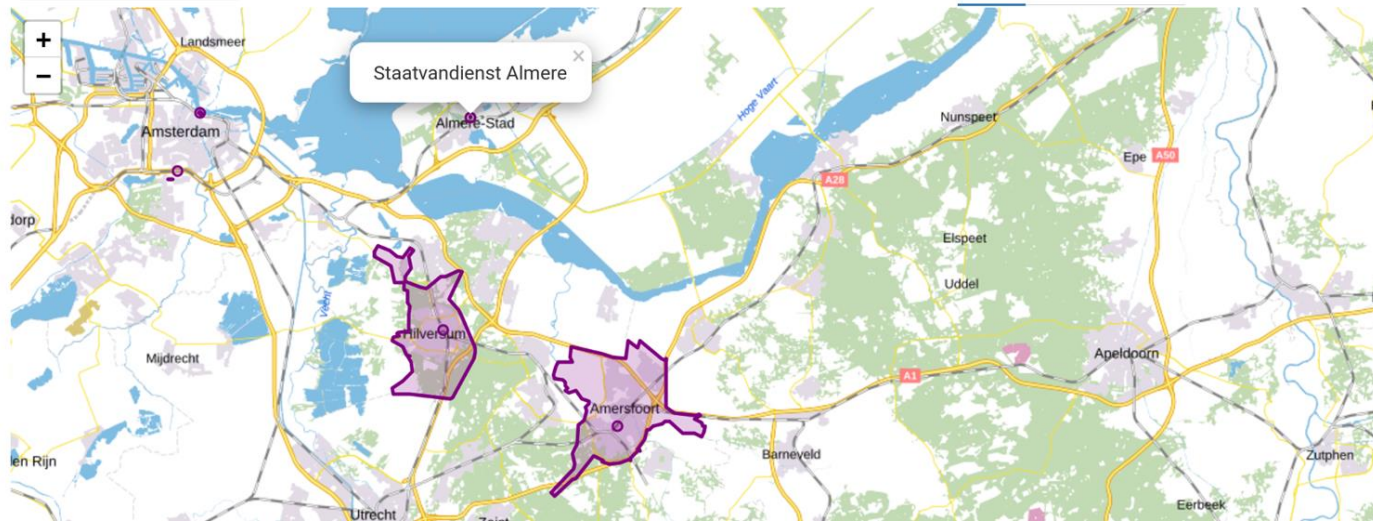
1 prefix geo: <http://www.opengis.net/ont/geosparql#>
2 prefix skos: <http://www.w3.org/2004/02/skos/core#>
3 select * where {
4   ?feature geo:hasGeometry ?geometry;
5           skos:prefLabel ?geoLabel.
6   ?geometry geo:asWKT ?geo.
7   bind ("purple" as ?geoColor)
8 }

```

Table Response Gallery Chart **Geo** Geo-3D Geo events Markup Network Pivot Timeline

13 results in 0.128 seconds

Normal Grouped Heatmap Layers ?





# Let's find all *the train stations* ([Try it out!](#))

```
1 PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
2 prefix geo: <http://www.opengis.net/ont/geosparql#>
3 prefix sdo: <https://schema.org/>
4 select * where {
5   ?feature geo:hasGeometry/geo:asWKT ?geo;
6     a sdo:TrainStation;
7     skos:prefLabel ?geoLabel.
8   bind ("red" as ?geoColor)
9 }
```

Table Response Gallery Chart **Geo** Geo-3D Geo events Markup Network Pivot Timeline

3 results in 0.075 seconds

Normal Grouped Heatmap Layers ?



# Let's find the distance between objects ([Try it out!](#))

```

1 PREFIX sdo: <https://schema.org/>
2 prefix skos: <http://www.w3.org/2004/02/skos/core#>
3 prefix geof: <http://www.opengis.net/def/function/geosparql/>
4 prefix geo: <http://www.opengis.net/ont/geosparql#>
5 prefix uom: <http://www.opengis.net/def/uom/OGC/1.0/>
6 select ?label_1 ?label_2 ?distance where {
7   ?feature_1 geo:hasGeometry/geo:asWKT ?geo_1;
8             skos:prefLabel ?label_1;
9             a sdo:TrainStation.
10  ?feature_2 geo:hasGeometry/geo:asWKT ?geo_2;
11            skos:prefLabel ?label_2.
12  bind (geof:distance(?geo_1, ?geo_2, uom:meter) as ?distance)
13 }
14 limit 5

```

[Table](#)
[Response](#)
[Gallery](#)
[Chart](#)
[Geo](#)
[Geo-3D](#)
[Geo events](#)
[Markup](#)
[Network](#)
[Pivot](#)
[Timeline](#)

5 results in 0.086 seconds

Simple view  Ellipse   Page size: 50

| label_1              | label_2                   | distance |
|----------------------|---------------------------|----------|
| Hilversum Media Park | Kadaster, locatie De Brug | 53272.3  |
| Hilversum Media Park | Hilversum                 | 0.0      |
| Hilversum Media Park | W&N                       | 24328.7  |
| Hilversum Media Park | Staatvondienst Almere     | 16831.1  |
| Hilversum Media Park | Amsterdam Centraal        | 25606.6  |



# Let's find objects in the municipality of Hilversum ([Try it out!](#))

```

1 PREFIX geof: <http://www.opengis.net/def/function/geosparql/>
2 prefix geo: <http://www.opengis.net/ont/geosparql#>
3 prefix skos: <http://www.w3.org/2004/02/skos/core#>
4 select ?geo_2 where {
5   ?feature_1 geo:hasGeometry/geo:asWKT ?geoGemeente;
6     skos:prefLabel "Hilversum"@nl.
7   ?feature_2 geo:hasGeometry/geo:asWKT ?geo_2.
8   filter(geof:sfWithin(?geo_2, ?geoGemeente))
9 } limit 10

```

Table Response Gallery Chart Geo Geo-3D Geo events Markup Network Pivot Timeline

3 results in 0.379 seconds

Normal Grouped Heatmap Layers Download Help



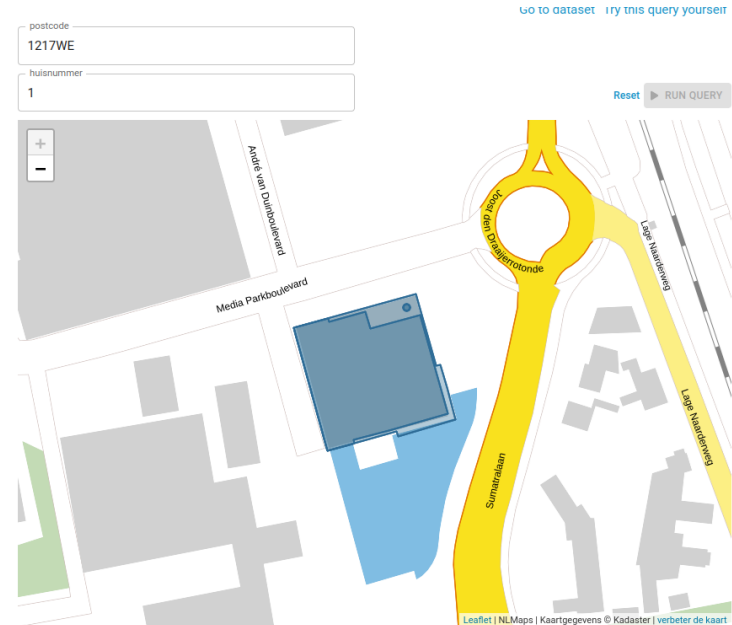
# Let's explore the BAG Datastory! ([Try it out!](#))

- Let's dive into the first two queries.
- Question time:

*Find B&G building:*

*zip code: 1217WE;*

*house number: 1.*



Figuur 1. Postcode-gebaseerde bevraging van de BAG.

