Graph Databases

Activity 3 - Cypher

You will be querying five Neo4j databases, provided to you. These databases are: (1) A graph representation of the Northwind operational database, denoted **northwindhg.db**; (2) A graph representation of the Northwind data warehouse database, called **northwindDW.db**; (3) A graph representation of the MusicBrainz database,   called **MusicBrainz.db**. This database contains a portion of the data in the web site of the same names, representing releases and events performed by artists, either individually or in collaborations; (4) a **trajectories** database, obtained from check-ins in 4-square, taken from Kaggle.com; (5) a **rivers** database, with data from the Flanders river system, in Belgium.

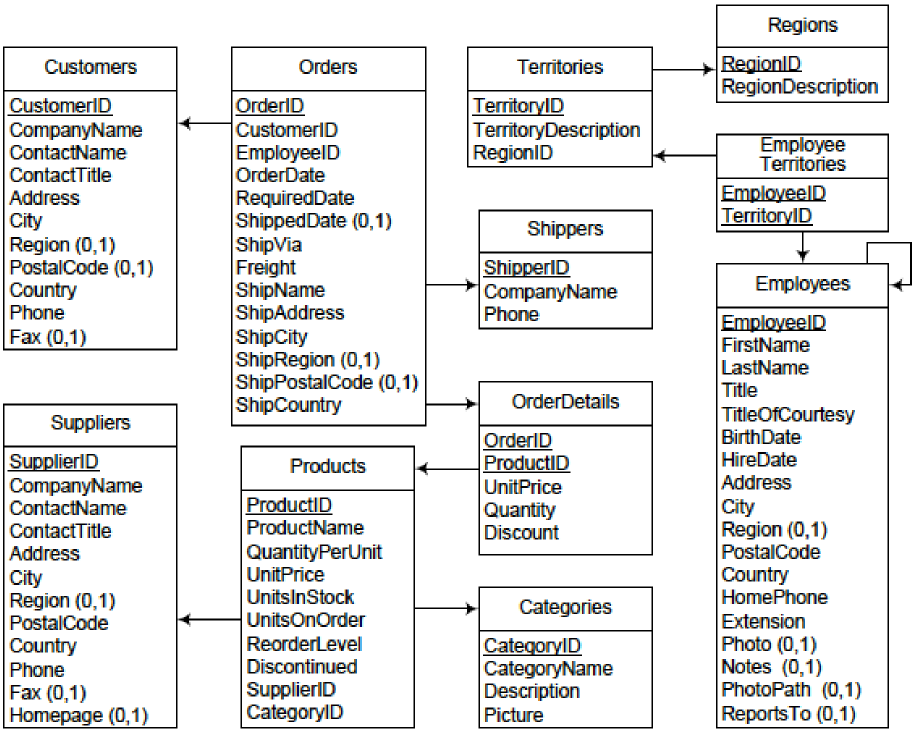
Before starting the Neo4j server, you need to choose the database you will work with. For this, you go to the **conf** folder, and edit the **neo4j.conf** file. You will find something like this:

```
#dbms.default_database=foodmartdw
#dbms.default_database=minigraphweb
#dbms.default_database=musicbrainz
#dbms.default_database=northwinddw
#dbms.default_database=northwindhg
#dbms.default_database=northwindoltp
dbms.default_database=rivers
#dbms.default_database=semantics
#dbms.default_database=neo4j
#dbms.default_database=trajectories
#dbms.default_database=webgraph3
#dbms.default_database=webdb
#dbms.default_database=telco
```

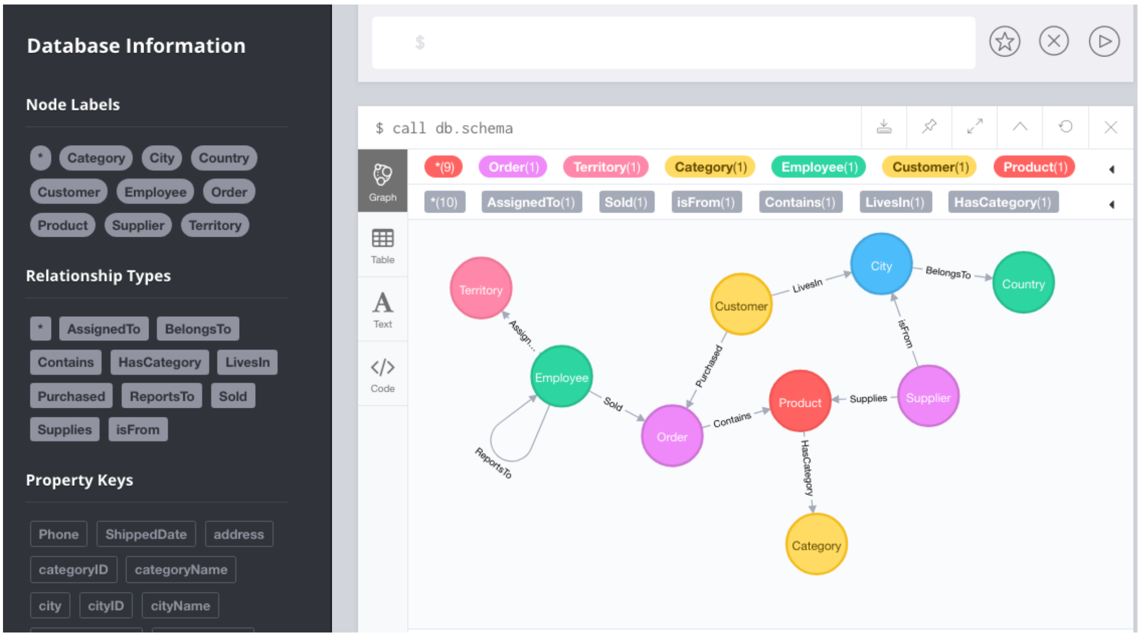
Since dbms.default_database =rivers is unmarked, to change the database to northwindhg, you mark #dbms.default_database =rivers, and unmark dbms.default_database = northwindhg. Save the changes, and quit the file. Then you run: `./bin/neo4j console` to start the Server. Then, open a browser, and type the following url: **localhost:7474.** Now you can start writing Cypher queries.

Exercise 1.

Consider the Northwind database, whose schema is:



This database has been exported to Neo4j, and you can find it at: /……../data/databases/northwindhg. The graph schema is:



```
$ call db.schema
```

Write in Cypher the following queries over the northwindhg.db database:

Query 1 - List products and their unit price.

Query 2 - List information about products 'Chocolade' & 'Pavlova'.

Query 3 - List information about products with names starting with a "C", whose unit price is greater than 50.

Query 4 - Same as 3, but considering the sales price, not the product's price.

Query 5 - Total amount purchased by customer and product.

Query 6 - Top ten employees, considering the number of orders sold.

Query 7 - For each employee, list the assigned territories.

Query 8 - For each city, list the companies settled in that city.

Query 9 - How many persons an employee reports to, either directly or transitively?

Query 10 - To whom do persons called "Robert" report to?

Query 11 - Who does not report to anybody?

Query 12 - Suppliers, number of categories they supply, and a list of such categories

Query 13 - Suppliers who supply beverages

Query 14 - Customer who purchases the largest amount of beverages

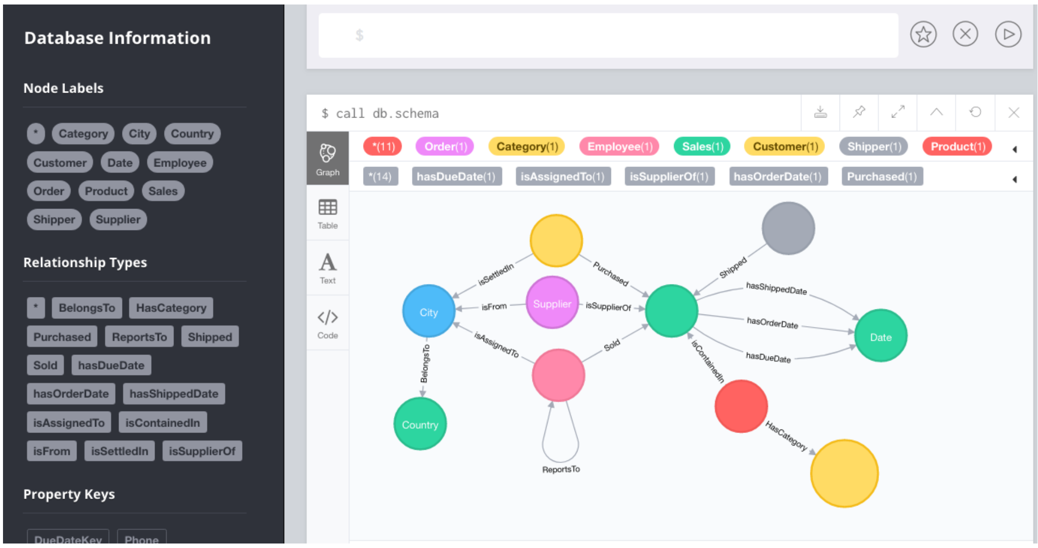
Query 15 - List the five most popular products (considering number of orders)

Query 16 - Products ordered by customers from the same country than their suppliers

Answer: In the lecture slides

Exercise 2.

Switch to the northwinddw database, doing the **same steps as in Assignment 2.** Now, the database is **northwinddw**. The schema is:



Database Information

Node Labels

* Category City Country Customer Date Employee Order Product Sales Shipper Supplier

Relationship Types

* BelongsTo HasCategory Purchased ReportsTo Shipped Sold hasDueDate hasOrderDate hasShippedDate isAssignedTo isContainedIn isFrom isSettledIn isSupplierOf

Property Keys

DueDateKey Phone

```
$ call db.schema
```

*(11) Order(1) Category(1) Employee(1) Sales(1) Customer(1) Shipper(1) Product(1)

*(14) hasDueDate(1) isAssignedTo(1) isSupplierOf(1) hasOrderDate(1) Purchased(1)

Write in Cypher the following queries over the northwindDW.db DB

Query 1. Total sales amount per customer, year, and product category

Query 2. Yearly sales amount for each pair of customer and supplier countries

Query 3. Three best-selling employees

Query 4. Best-selling employee per product and year

Query 5. Total sales and average monthly sales by employee and year

Query 6. Total sales amount and total discount amount per product and month

Query 7. Monthly year-to-date sales for each product category

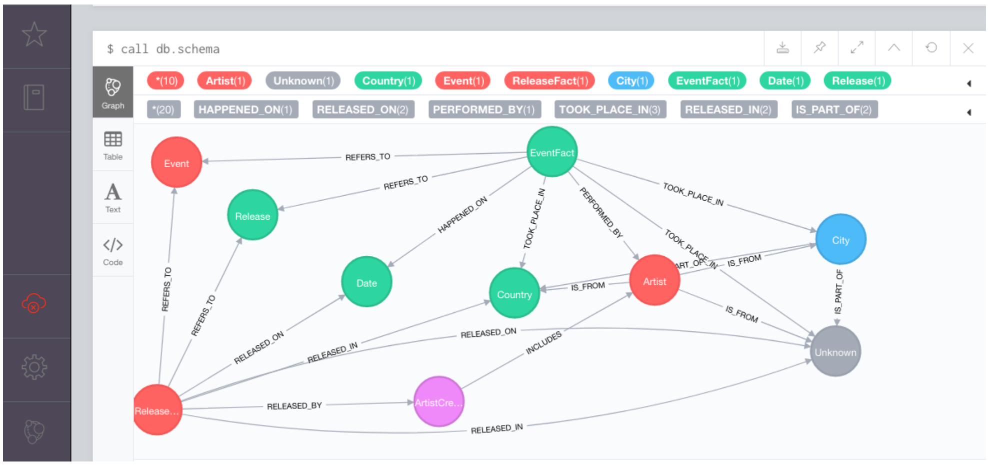
Query 8. Personal sales amount made by an employee compared with the total sales amount made by herself and her subordinates during 2017

Query 9. Total sales amount, number of products, and sum of the quantities sold for each order

Query 10. For each month, total number of orders, total sales amount, and average sales amount by order

Exercise 3.

Switch to the MusicBrainz database, doing the **same steps as in Assignment 2.** Now, the database is **musicbrainz**. The schema is:



```
$ call db.schema
```

Query 1. Compute the total number of releases per artist.

Query 2. Compute the total number of releases per artist and per year.

Query 3. Compute the number of times the artist performed in each event.

Query 4. For each (event, artist, year) triple, compute the number of times the artist performed in an event on an year.

Query 5. For each (event, artist, year) triple, compute the number of times an artist of the United Kingdom performed more than twice in an event occurred in 2006.

Query 6. Compute the number of releases, per language, in the UK.

Query 7. Compute, for each pair of artists, the number of times they performed together at least twice in an event, also listing the events' venues.

Query 8. Compute the triples of artists, and the number of times they have performed together in an event, if this number is at least 3.

Query 9. Compute the quadruples of artists, and the number of times they have performed together in an event, if this number is at least 3.

Query 10. Compute the number of artists who released a record and performed in at least an event, and the year(s) this happened.

Exercise 4.

We will query the Flanders river system depicted in Figure 1. The schema and properties are shown in Figures 2 to 4. Segments are represented as nodes, with label :Segment (and their corresponding properties), and the relation between the nodes is called :flowsTo, defined as follows: there is a relation :flowsTo from node A to node B if the water flows to segment B from segment A. This is stored in the **rivers database.**

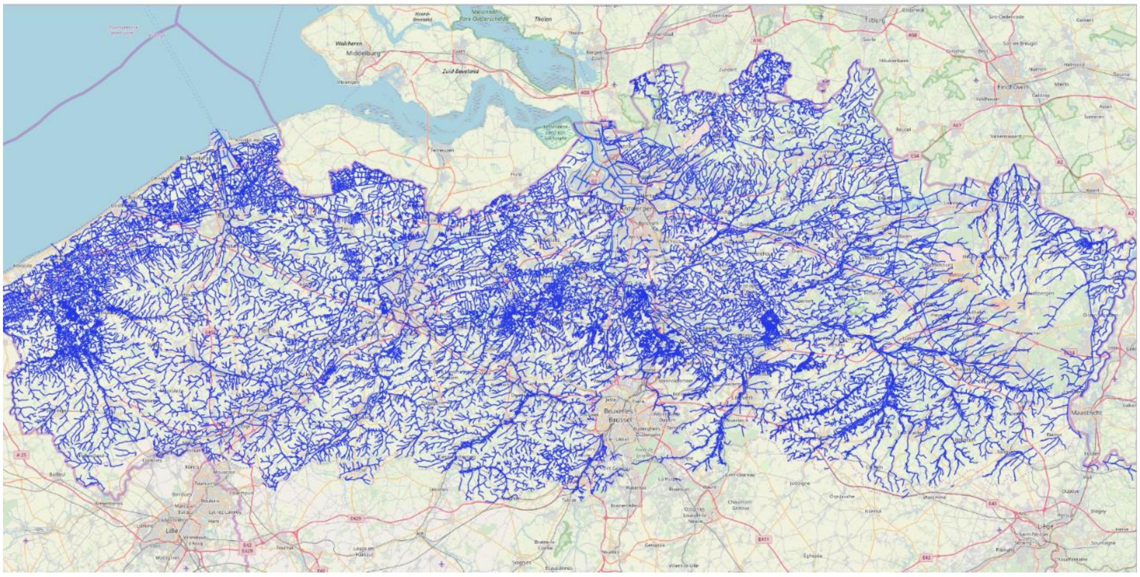
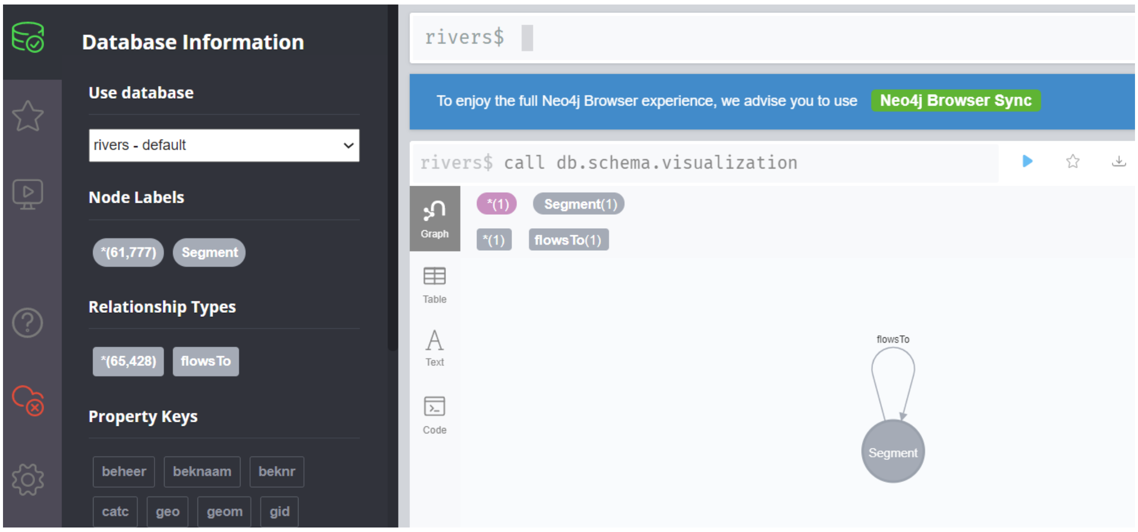


Figure 1



Database Information

Use database

rivers - default

Node Labels

*(61,777) Segment

Relationship Types

*(65,428) flowsTo

Property Keys

beheer beknaam beknr catc geo geom gid

```
rivers$
```

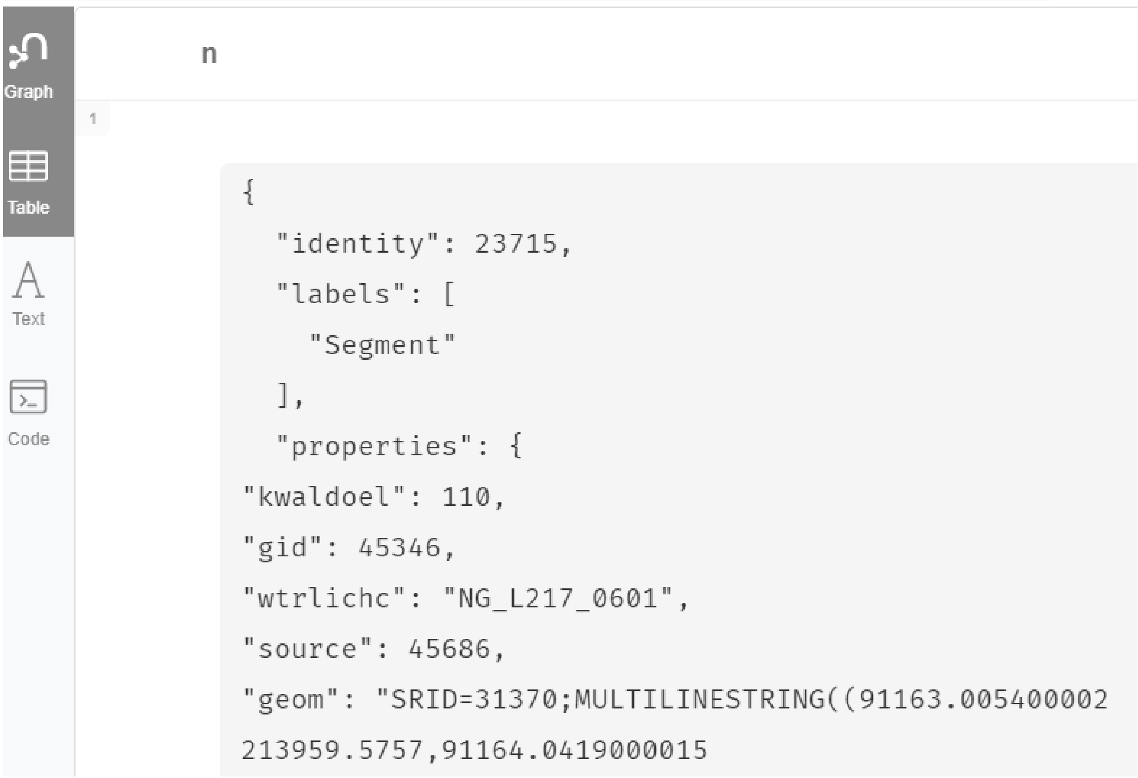
To enjoy the full Neo4j Browser experience, we advise you to use Neo4j Browser Sync

```
rivers$ call db.schema.visualization
```

*(1) Segment(1)

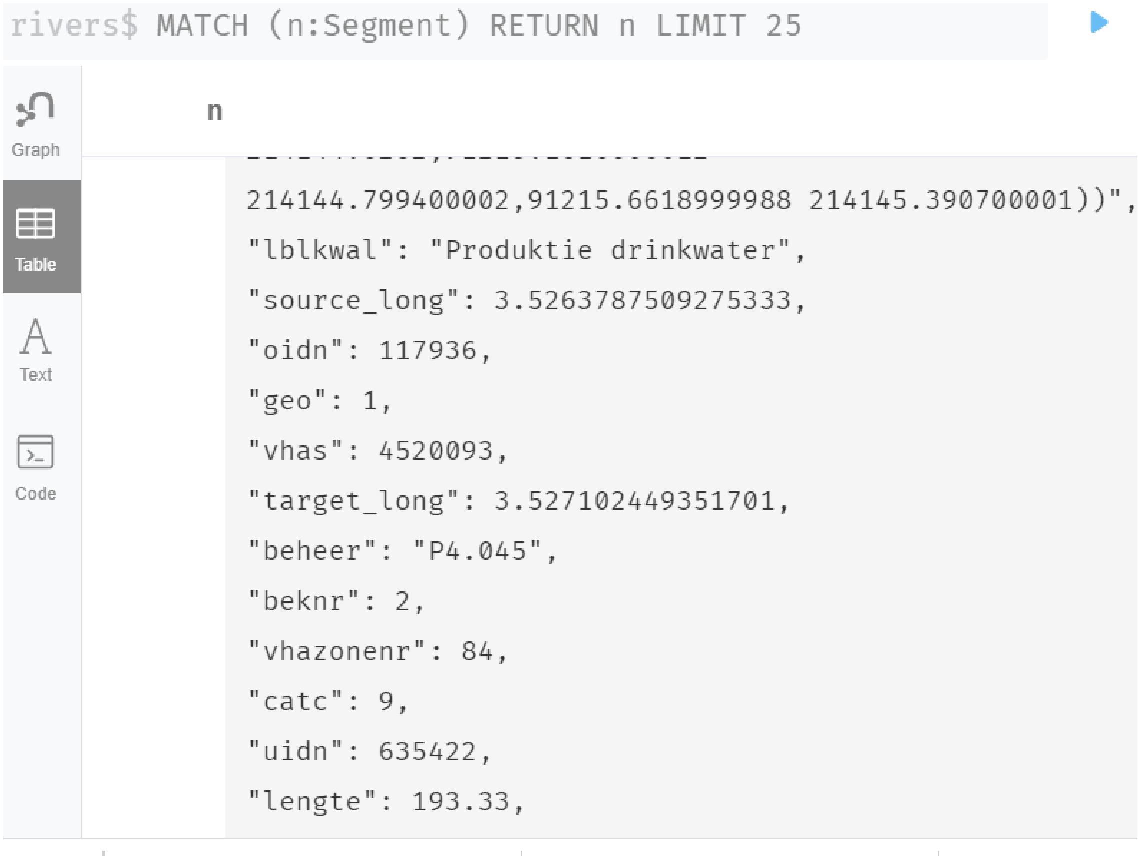
*(1) flowsTo(1)

Figure 2. Schema



```
{
  "identity": 23715,
  "labels": [
    "Segment"
  ],
  "properties": {
"kwaldoel": 110,
"gid": 45346,
"wtrlichc": "NG_L217_0601",
"source": 45686,
"geom": "SRID=31370;MULTILINESTRING((91163.005400002
213959.5757,91164.0419000015
```

Figure 3. Properties



```
rivers$ MATCH (n:Segment) RETURN n LIMIT 25
```

```
214144.799400002,91215.6618999988 214145.390700001))",
"lblkwal": "Produktie drinkwater",
"source_long": 3.5263787509275333,
"oidn": 117936,
"geo": 1,
"vhas": 4520093,
"target_long": 3.527102449351701,
"beheer": "P4.045",
"beknr": 2,
"vhazonenr": 84,
"catc": 9,
"uidn": 635422,
"lengte": 193.33,
```

Figure 4. Properties

Query 1. Compute the average segment length. (property: lengte)

Query 2. Compute the average segment length by segment category (property:catc)

Query 3. Find all segments that have a length within a 10% margin of the length of segment with ID 6020612. (segmentID = vhas)

Query 4. For each segment find the number of incoming and outgoing segments.

Query 5. Find the segments with the maximum number of incoming segments.

Query 6. Find the nodes where there is a split in the downstream path of segment 6020612

Query 7. Find the number of in-flowing segments in the downstream path of segment 6020612.

Query 8. Determine if there is a loop in the downstream path of segment 6031518. For every loop, list the path.

Query 9. Find the length, the # of segments, and the IDs of the segments, of the longest branch of upstream flow starting from a given segment.

Query 10. How many paths exist between two given segments X and Y?

Query 11. Find all segments reachable from the segment closest to Antwerpen's Groenplaats