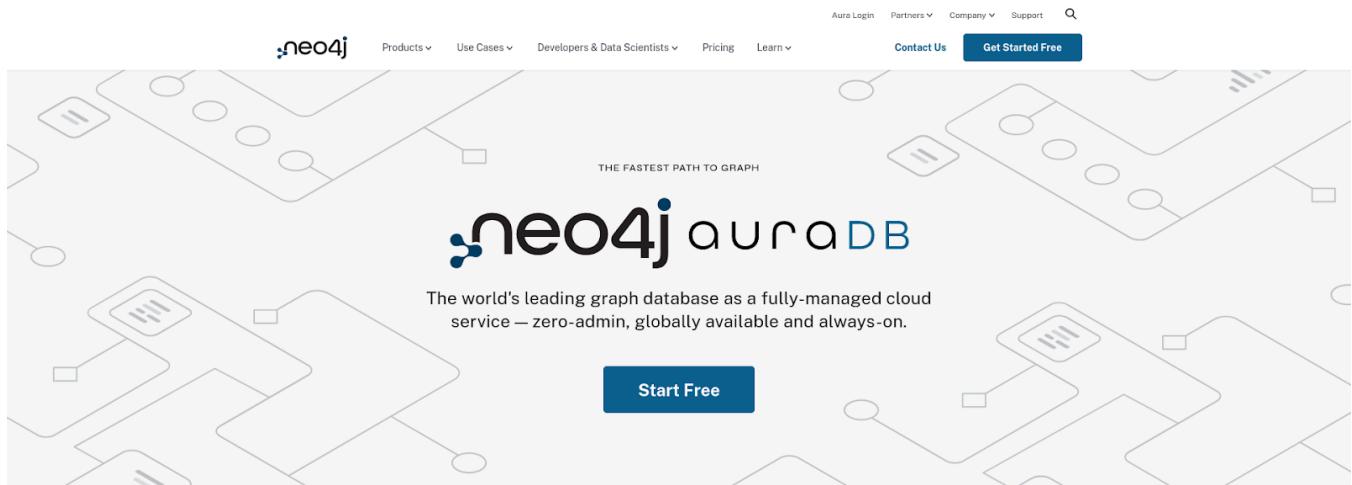


# CIS 5500: Database and Information Systems

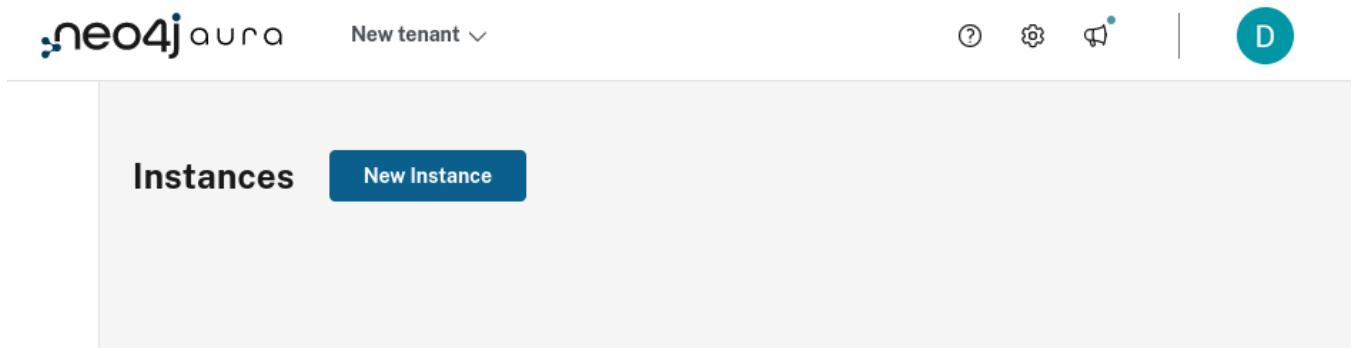
## Neo4j Aura Handout

### 1. Creating an Instance

1. Head over to the official [Neo4j Aura site](#) and create an account (click on Start Free).



2. Click on “New Instance”



3. Select “Create Free Instance”. Make sure to save the password that is given after creation. You’ll need this for entering your credentials in part 3.



4. A new instance is being set up.

Instances    New Instance

Instance01 Free

Creating Instance...  
(This may take a few minutes)

5. After a few minutes, the instance will start

Get started    Developer hub

Data services

Instances 1

Import

Graph Analytics

Data APIs

Agents Preview

Tools

Query

Explore

Dashboards

Operations >

Instances

Aura (1)    Self-managed (0)

Search

Connect ...

Instance01 RUNNING

ID: e7a887bf

Type: AuraDB Free    Nodes: 105 (0%)    Relationships: 180 (0%)

Monthly usage: \$0.00    Upgrade

Metrics

6. Click on the inspect option inside options menu to see instance overview.

The screenshot shows the Neo4j Aura Instances interface. At the top, there are tabs for 'Aura (1)' and 'Self-managed (0)'. Below the tabs is a search bar with placeholder text 'Search' and a 'Create instance' button. The main area displays 'Instance01' which is 'RUNNING' with ID 'e7a887bf'. It shows metrics: Type: AuraDB Free, Nodes: 105 (0%), Relationships: 180 (0%). A 'Metrics' section is partially visible. To the right of the instance card is a vertical options menu with the following items: Inspect (highlighted), Snapshots, Backup & restore, Clone to, IP filtering, Reset to blank, and Delete. The 'Clone to' item has a dropdown arrow indicating it has sub-options. The 'Delete' item is preceded by a trash icon.

## 2. Data Loading

### 2.1. Method 1-Manually Running CREATE Queries

7. Connect to the instance (see instructions in part 3)
8. Connect to a database in the horizontal menu
9. Open “flights.cyp” and run all the CREATE queries in the query console (found in the TOOLS section of the sidebar).

We recommend you take a look at the flights.cyp file provided with this handout to get an idea of the queries that are occurring to create the relations.

Neo4j has rate-limiting issues that prevent you from running all of the queries at once so please run them chunk by chunk. The chunks are separated with newlines in the file for your convenience.

### 2.3. Check Import Success

10. The instance should have 105 nodes and 180 relationships after importing all data.

The screenshot shows the Neo4j Browser interface. On the left, there's a sidebar with various options like 'Get started', 'Developer hub', 'Data services' (with 'Instances', 'Import', 'Graph Analytics', 'Data APIs', and 'Agents'), and 'Tools' (with 'Query' selected). The main area displays 'Database information' with 'Nodes (105)' (City and Flight types) and 'Relationships (180)' (FLYING\_TO and HAS\_FLIGHT types). Below this, 'Property keys' are listed: arrival, carrier, code, country, departure, destination\_airport\_code, duration, name, and source\_airport\_code. A timestamp 'Last update: 2:08:24 PM' is at the bottom. To the right, a query console window titled 'neo4j\$' shows the command '\$ :welcome'. A 'GUIDE' section features a 3D graph icon, and a 'Query fundamentals' section lists learning objectives: writing basic queries, viewing graph and tabular results, and performing queries to answer questions.

### 3. Querying the Database

11. After loading the DB into neo4j aura, you can execute queries in the same console.

The screenshot shows the neo4j aura interface. On the left, a sidebar menu includes 'Get started', 'Developer hub', 'Data services' (with 'Instances', 'Import', 'Graph Analytics', 'Data APIs', and 'Agents' listed), 'Tools' (with 'Query' selected), and 'Explore'. The main area displays 'Database information' for 'Instance01'. It shows 'Nodes (105)' (highlighted in green) and 'Relationships (180)' (highlighted in purple). A query editor at the top right contains the following code:

```
1 MATCH (n)-[r]-(m)
2 RETURN n, r, m|
```

The results overview on the right indicates there are 105 nodes and 180 relationships. The nodes are categorized as City (15) and Flight (90). The relationships are categorized as FLYING\_TO (90) and HAS\_FLIGHT (90). Below the results is a network graph visualization.