

## MongoDB

O MongoDB Compass, utilizado neste projeto, foi baixado via site oficial do MongoDB. Para adicionar os dados no MongoDB, é necessário importar o arquivo .csv

Rodamos a seguinte query para nomear os países *null* como **Não especificado**

```
JavaScript
db.air_pollution.updateMany( { Country: null }, { $set: { Country: "Not
specified" } })
```

---

## Neo4j

O Homebrew foi utilizado para fazer o download do Neo4j foi utilizado. Para subir o serviço, basta rodar a seguinte linha no terminal:

```
JavaScript
neo4j start
```

Para adicionar os dados no Neo4j, é necessário integrar o banco de dados com o banco MongoDB criado anteriormente. Para isso, foi necessário instalar um conjunto de *procedures* chamado **apoc**.

Foram instaladas:

- apoc-5.22.0-core.jar
- apoc-5.22.0-extended.jar
- apoc-mongodb-dependencies-5.22.0-all.jar

Todos esses pacotes foram baixados do repositório do GitHub [neo4j-apoc-procedures](#). Para acessar os arquivos, basta clicar em releases. Lá existem várias releases de várias versões do apoc. É necessário baixar a versão correspondente com a versão do Neo4j instalada.

Após baixar os arquivos, é necessário adicioná-los na pasta plugins do Neo4j. Uma vez que isso tenha sido feito, é possível utilizar as *procedures* **apoc.mongo**.

Puxar dados do mongoDB, foi utilizada a query:

```
JavaScript
WITH "mongodb://localhost:27017/PMD2024_Gabriela.air_pollution" AS url
CALL apoc.mongo.find(url, {}, null) YIELD value
RETURN value
```

Onde localhost:27017 é o servidor e porta onde o MongoDB está rodando, PMD2024\_Gabriela é o nome do banco e air\_pollution a collection.

### Criação dos nós País

```
JavaScript
WITH "mongodb://localhost:27017/PMD2024_Gabriela.air_pollution" AS url
CALL apoc.mongo.find(url, {}, null) YIELD value
CREATE (p:Pais {name: value.Country})
```

### remoção de nós com mesmo valor

```
JavaScript
MATCH (n:Pais)
WITH n.name AS name, COLLECT(n) AS nodes
WHERE SIZE(nodes) > 1
WITH nodes[1..] AS duplicates
UNWIND duplicates AS duplicate
DETACH DELETE duplicate
```

```
JavaScript
CREATE INDEX pais FOR (a:Pais) ON (a.name)
```

### Criação dos nós Cidade

```
JavaScript
WITH "mongodb://localhost:27017/PMD2024_Gabriela.air_pollution" AS url
CALL apoc.mongo.find(url, {}, null) YIELD value
CREATE (c: Cidade {name: row.City})
```

```
JavaScript
CREATE INDEX cidade FOR (a:Cidade) ON (a.name)
```

### Criação do relacionamento entre Cidade e País

```
JavaScript
WITH "mongodb://localhost:27017/PMD2024_Gabriela.air_pollution" AS url
```

```
CALL apoc.mongo.find(url, {}, null) YIELD value
MATCH (p:Pais {name: value.Country}),(c: Cidade {name: value.City})
CREATE (c)-[:PERTENCE_A]->(p)
```

### Criação do nó AQI

```
JavaScript
CREATE (n:AQI)
```

### Criação do relacionamento de Cidade possui AQI

```
JavaScript
WITH "mongodb://localhost:27017/PMD2024_Gabriela.air_pollution" AS url
CALL apoc.mongo.find(url, {}, null) YIELD value
MATCH (c: Cidade {name: value.City}), (a: AQI)
CREATE (c)-[:POSSUI {categoria: value.`AQI Category`, valor:value.`AQI Value`, tipo: "geral"}]->(a)
```

### Criação do relacionamento de Cidade possui AQI (Carbono)

```
JavaScript
WITH "mongodb://localhost:27017/PMD2024_Gabriela.air_pollution" AS url
CALL apoc.mongo.find(url, {}, null) YIELD value
MATCH (c: Cidade {name: value.City}), (a: AQI)
CREATE (c)-[:POSSUI {categoria: value.`CO AQI Category`, valor:value.`CO AQI Value`, tipo: "CO"}]->(a)
```

### Criação do relacionamento de Cidade possui AQI (NO2)

```
JavaScript
WITH "mongodb://localhost:27017/PMD2024_Gabriela.air_pollution" AS url
CALL apoc.mongo.find(url, {}, null) YIELD value
MATCH (c: Cidade {name: value.City}), (a: AQI)
CREATE (c)-[:POSSUI {categoria: value.`NO2 AQI Category`, valor:value.`NO2 AQI Value`, tipo: "NO2"}]->(a)
```

### Criação do relacionamento de Cidade possui AQI (Ozone)

JavaScript

```
WITH "mongodb://localhost:27017/PMD2024_Gabriela.air_pollution" AS url
CALL apoc.mongo.find(url, {}, null) YIELD value
MATCH (c: Cidade {name: value.City}), (a: AQI)
CREATE (c)-[:POSSUI {categoria: value.`Ozone AQI Category`,
valor:value.`Ozone AQI Value`, tipo: "Ozone"}]->(a)
```

### Criação do relacionamento de Cidade possui AQI (PM2)

JavaScript

```
WITH "mongodb://localhost:27017/PMD2024_Gabriela.air_pollution" AS url
CALL apoc.mongo.find(url, {}, null) YIELD value
MATCH (c: Cidade {name: value.City}), (a: AQI)
CREATE (c)-[:POSSUI {categoria: value.`PM2`.`5 AQI Category`,
valor:value.`PM2`.`5 AQI Value`, tipo: "PM2.5"}]->(a)
```