

How to load data automatically to Neo4j and update data in Neo4j

Load data

```
deni@LAPTOP-EHV4QTMQ: $ docker cp /mnt/c/Users/deni/Downloads/Artist.csv testneo4j:/var/lib/neo4j/import/
docker cp /mnt/c/Users/deni/Downloads/Album.csv testneo4j:/var/lib/neo4j/import/
docker cp /mnt/c/Users/deni/Downloads/Track.csv testneo4j:/var/lib/neo4j/import/
Successfully copied 2.05kB to testneo4j:/var/lib/neo4j/import/
Successfully copied 2.05kB to testneo4j:/var/lib/neo4j/import/
Successfully copied 2.05kB to testneo4j:/var/lib/neo4j/import/
deni@LAPTOP-EHV4QTMQ: $ docker cp /mnt/c/Users/deni/Downloads/Genre.csv testneo4j:/var/lib/neo4j/import/
docker cp /mnt/c/Users/deni/Downloads/MediaType.csv testneo4j:/var/lib/neo4j/import/
docker cp /mnt/c/Users/deni/Downloads/InvoiceLine.csv testneo4j:/var/lib/neo4j/import/
docker cp /mnt/c/Users/deni/Downloads/Invoice.csv testneo4j:/var/lib/neo4j/import/
docker cp /mnt/c/Users/deni/Downloads/Employee.csv testneo4j:/var/lib/neo4j/import/
Successfully copied 2.05kB to testneo4j:/var/lib/neo4j/import/
Successfully copied 2.05kB to testneo4j:/var/lib/neo4j/import/
Successfully copied 2.05kB to testneo4j:/var/lib/neo4j/import/
Successfully copied 2.05kB to testneo4j:/var/lib/neo4j/import/
Successfully copied 2.05kB to testneo4j:/var/lib/neo4j/import/
Successfully copied 2.05kB to testneo4j:/var/lib/neo4j/import/
```

memindahkan file CSV dari sistem lokal ke dalam container Docker yang menjalankan Neo4j untuk memudahkan pemuatan data. Setelah file berada dalam direktori yang benar di dalam container, maka bisa menggunakan perintah Cypher untuk memuat dan memanipulasi data tersebut.

Cypher Query Language:

```
// Artist
LOAD CSV WITH HEADERS FROM 'file:///Artist.csv' AS row
MERGE (:Artist {ArtistId: toInteger(row.ArtistId), Name: row.Name});

// Album
LOAD CSV WITH HEADERS FROM 'file:///Album.csv' AS row
MERGE (:Album {AlbumId: toInteger(row.AlbumId), Title: row.Title});

// Track
LOAD CSV WITH HEADERS FROM 'file:///Track.csv' AS row
MERGE (:Track {
  TrackId: toInteger(row.TrackId),
  Name: row.Name
});

// Genre
LOAD CSV WITH HEADERS FROM 'file:///Genre.csv' AS row
MERGE (:Genre {GenreId: toInteger(row.GenreId), Name: row.Name});

// MediaType
LOAD CSV WITH HEADERS FROM 'file:///MediaType.csv' AS row
MERGE (:MediaType {MediaTypeId: toInteger(row.MediaTypeId), Name: row.Name});

// Invoice
LOAD CSV WITH HEADERS FROM 'file:///Invoice.csv' AS row
MERGE (:Invoice {
  InvoiceId: toInteger(row.InvoiceId),
```

```

InvoiceDate: row.InvoiceDate
});

// InvoiceLine
LOAD CSV WITH HEADERS FROM 'file:///InvoiceLine.csv' AS row
MERGE (:InvoiceLine {
  InvoiceLineId: toInteger(row.InvoiceLineId),
  Quantity: toInteger(row.Quantity)
});

// Customer
LOAD CSV WITH HEADERS FROM 'file:///Customer.csv' AS row
MERGE (:Customer {
  CustomerId: toInteger(row.CustomerId),
  FirstName: row.FirstName,
  LastName: row.LastName
});

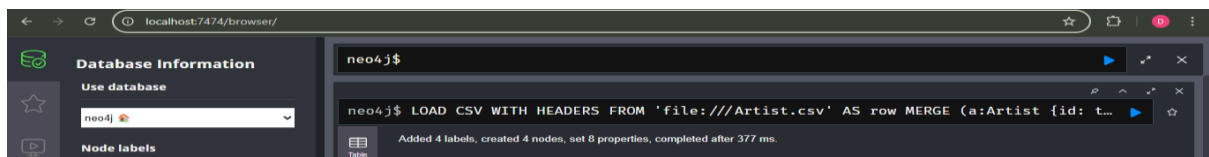
// Employee
LOAD CSV WITH HEADERS FROM 'file:///Employee.csv' AS row
MERGE (:Employee {
  EmployeeId: toInteger(row.EmployeeId),
  FirstName: row.FirstName,
  LastName: row.LastName
});

```

Contoh:

```
LOAD CSV WITH HEADERS FROM 'file:///Artist.csv' AS row
```

```
MERGE (:Artist {id: toInteger(row.ArtistId), name: row.Name});
```



Update Data

```
MATCH (a:Artist {ArtistId: 1})
```

```
SET a.Name = 'New Artist Name'
```

```
RETURN a
```

