

Step 1. Make sure neo4j.conf has the import clauses set as follows:

```
*****
# Miscellaneous configuration
*****

# Enable this to specify a parser other than the default one.
#cypher.default_language_version=3.0

# Determines if Cypher will allow using file URLs when loading data using
# `LOAD CSV`. Setting this value to `false` will cause Neo4j to fail `LOAD CSV`
# clauses that load data from the file system.
#dbms.security.allow_csv_import_from_file_urls=true
```

```
# This setting constrains all `LOAD CSV` import files to be under the `import` d
# directory. Remove or comment it out to
# allow files to be loaded from anywhere in the filesystem; this introduces poss
# ible security problems. See the
# `LOAD CSV` section of the manual for details.
#dbms.directories.import=import
```

Namely, (if not already commented) comment the line

#dbms.directories.import=import

and uncomment (or add) the line

dbms.security.allow\_csv\_import\_from\_file\_urls=true.

Step 2. Open cypher-shell

```
Ilirs-MacBook-Pro:bin ilirdema$ ./cypher-shell -u neo4j -p 1234
Connected to Neo4j 3.5.12 at bolt://localhost:7687 as user neo4j.
Type :help for a list of available commands or :exit to exit the shell.
Note that Cypher queries must end with a semicolon.
```

Step 3. Import actors:

```
neo4j> LOAD CSV FROM "file:///Users/ilirdema/tsv/actors.tsv" AS data FIELDTERMINATOR
'\t' CREATE (:Actor {actorId: data[1], name: data[2]});
0 rows available after 669 ms, consumed after another 0 ms
Added 26273 nodes, Set 52546 properties, Added 26273 labels
neo4j>
```

Step 4. Import movies:

```
neo4j> LOAD CSV FROM "file:///Users/ilirdema/tsv/movies.tsv" AS data FIELDTERMINATOR
'\t' CREATE (:Movies {movieId: data[1], name: data[3]});
0 rows available after 410 ms, consumed after another 0 ms
Added 65720 nodes, Set 131440 properties, Added 65720 labels
neo4j>
```

Step 5. Import relationships to a temporary collection:

```
neo4j> LOAD CSV FROM "file:///Users/ilirdema/tsv/relationships.tsv" AS data FIELDTERMINATOR '\t' CREATE(:TEMP {mid: data[0], aid: data[1]});
0 rows available after 847 ms, consumed after another 0 ms
Added 170407 nodes, Set 340814 properties, Added 170407 labels
neo4j>
```

Step 6. Create relationships:

```
neo4j> MATCH (t:TEMP),(m:Movies),(a:Actor) WHERE (t.mid=m.movieId AND t.aid=a.actorId) CREATE (a)-[:ActedIn]->(m);
0 rows available after 1312 ms, consumed after another 0 ms
Created 170398 relationships
neo4j>
```

Step 7. Remove TEMP nodes:

```
neo4j> MATCH (t:TEMP) DELETE t;
0 rows available after 142 ms, consumed after another 0 ms
Deleted 170407 nodes
neo4j>
```