

Managing Neo4j Instances

Table of Contents

About this module	1
Neo4j instance files	2
Changing the password for Neo4j.....	3
Exercise: Change the password for Neo4j	3
Starting the Neo4j instance	4
Exercise: Start the Neo4j instance	6
Stopping the Neo4j instance.....	9
Exercise: Stop the Neo4j instance	9
Renaming a Neo4j database.....	11
Copying a database	12
Dumping a database	12
Loading a database	13
Exercise: Copying a database	14
Modifying the location of the database.....	15
Exercise: Modifying the location of the database	16
Checking the consistency of a database	17
Exercise: Checking consistency of a database	18

About this module

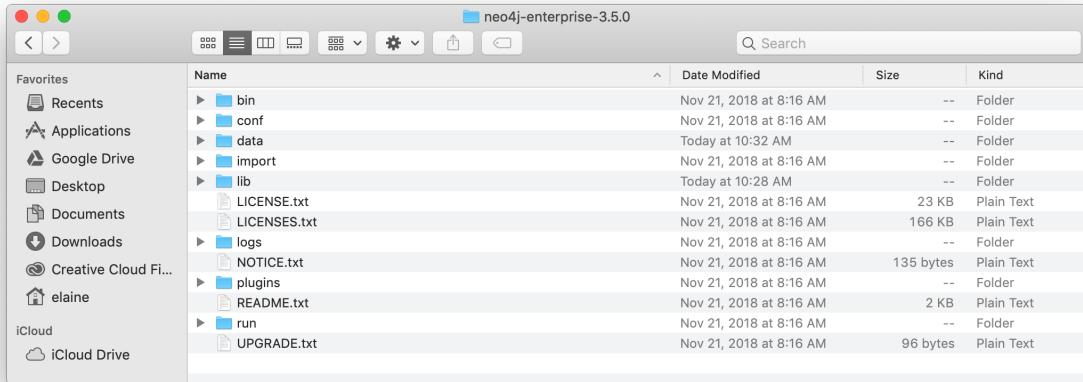
Now that you have installed the Neo4j Enterprise Edition, you will learn how to perform some administrative tasks with the Neo4j instance.

At the end of this module, you should be able to:

- Set the password for the *neo4j* user.
- Start a Neo4j instance.
- Stop the Neo4j instance.
- Copy a Neo4j database.
- Modify the location for a Neo4j database.
- Check the consistency of a Neo4j database.
- Troubleshoot problems starting a Neo4j instance.

Neo4j instance files

Here is the file structure for a Neo4j installation which is commonly referred to as *NEO4J_HOME*:



In this training, you will be working in some of these folders. The **bin** folder contains the tooling scripts you will typically run to manage the Neo4j instance. The **conf** folder contains configuration files that you will be modifying. The **data** folder contains the database(s). The **logs** folder contains log files that you can monitor.

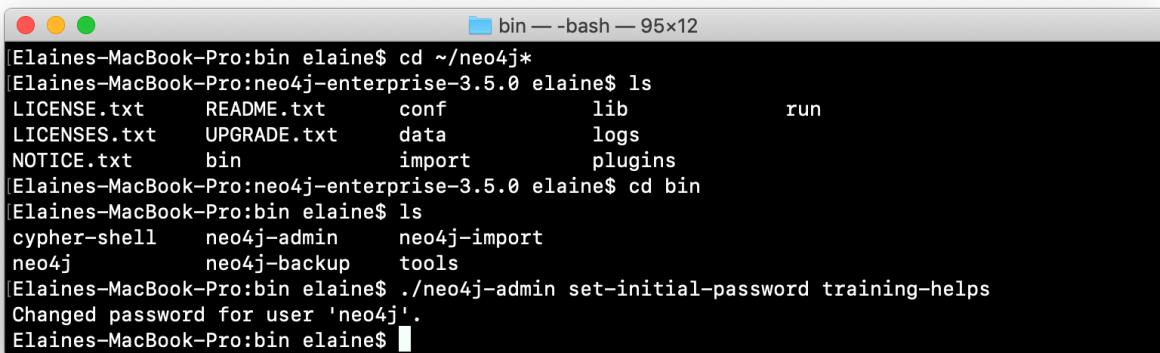
Changing the password for Neo4j

Before you do anything with Neo4j, you should change the default password for the user, *neo4j*. Doing so enables you to, at a minimum, control who can manage this Neo4j instance. Later in this training, you will learn more about securing Neo4j.

You change the password with the **neo4j-admin** tool that is found in the **bin** folder of your Neo4j Enterprise Edition installation (NEO4J_HOME). Here is the command you use to change the password:

```
neo4j-admin set-initial-password thePassword
```

Here is an example where in a terminal window, we have navigated to the **bin** folder in our Neo4j installation and have entered the command to set the initial password for the *neo4j* user to be *training-helps*.



The screenshot shows a terminal window on a Mac OS X system. The title bar says "bin — bash — 95x12". The terminal content is as follows:

```
[Elaines-MacBook-Pro:bin elaine$ cd ~/neo4j*
[Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ ls
LICENSE.txt      README.txt      conf          lib          run
LICENSES.txt     UPGRADE.txt    data          logs
NOTICE.txt       bin           import        plugins
[Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ cd bin
[Elaines-MacBook-Pro:bin elaine$ ls
cypher-shell     neo4j-admin    neo4j-import
neo4j           neo4j-backup   tools
[Elaines-MacBook-Pro:bin elaine$ ./neo4j-admin set-initial-password training-helps
Changed password for user 'neo4j'.
Elaines-MacBook-Pro:bin elaine$ ]
```

NOTE

Using **set-initial-password** to change the Neo4j password for the default *neo4j* user can be done at any time, not just initially as the command name might imply.

Exercise: Change the password for Neo4j

Before you begin

You should have installed Neo4j Enterprise Edition on your system and made a note of where you installed it. We refer to this location as NEO4J_HOME.

Exercise steps:

1. Open a terminal on your system.
2. Navigate to the **bin** folder of NEO4J_HOME.
3. Use the admin-tool to change the *neo4j* user password to something you will remember.

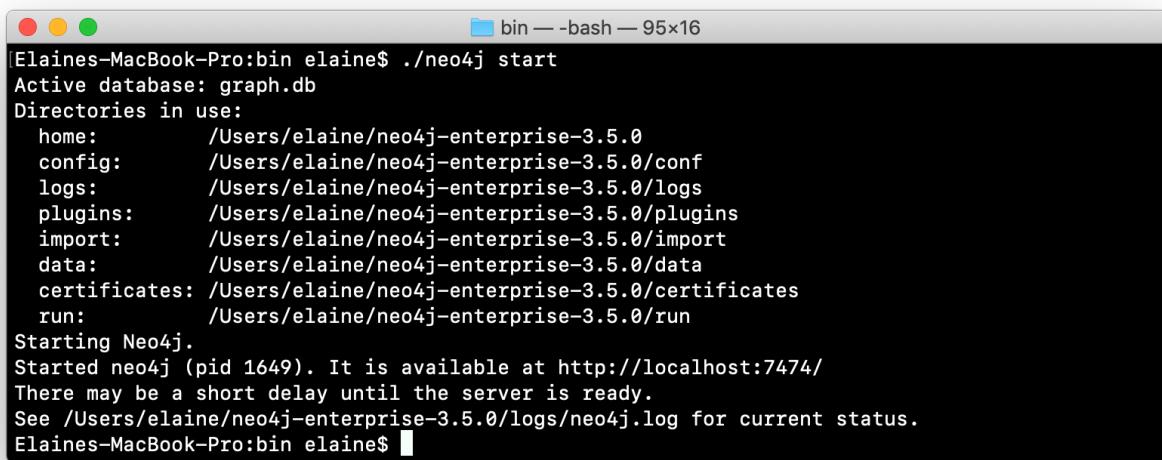
Starting the Neo4j instance

When you start the Neo4j instance for the first time, it automatically creates a database in the **data/databases** folder. Subsequent starts of the Neo4j instance simply uses the default database created.

You use the the **neo4j** script to start the Neo4j instance as follows:

```
neo4j start
```

Here is an example where in a terminal window, we start Neo4j instance:

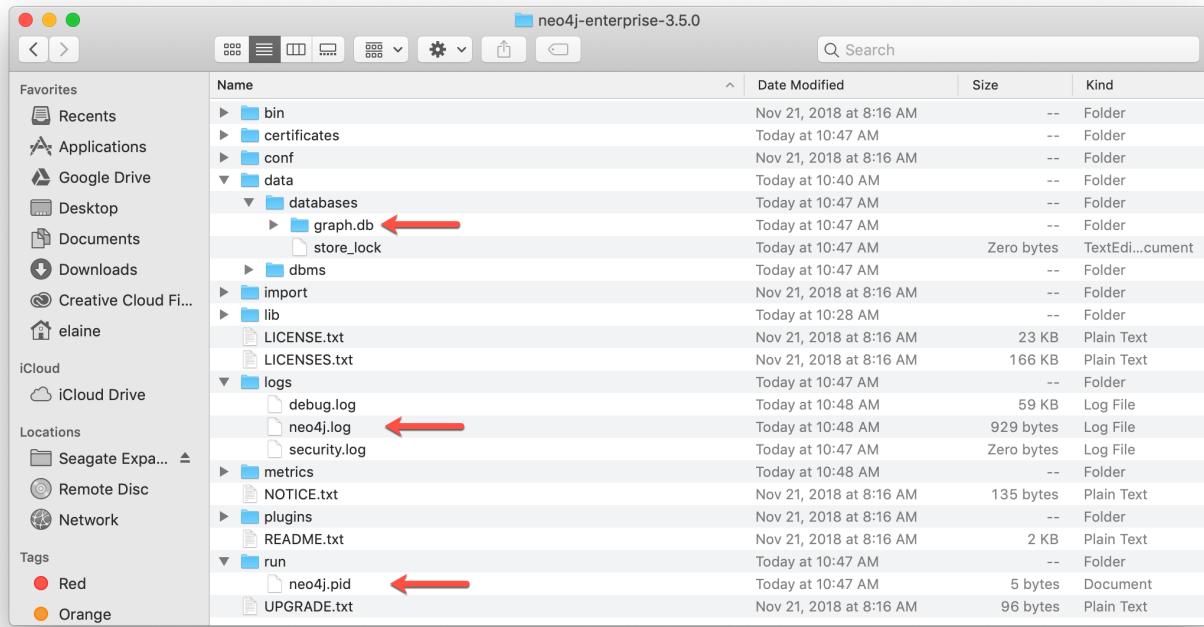


The screenshot shows a terminal window titled "bin — bash — 95x16". The command "Elaines-MacBook-Pro:bin elaine\$./neo4j start" is run. The output shows the active database is "graph.db" and lists the directories in use: home, config, logs, plugins, import, data, certificates, and run, all pointing to the Neo4j enterprise directory. It then starts the Neo4j server, noting the pid (1649) and the available URL (http://localhost:7474). It also suggests checking the log file for status. The prompt "Elaines-MacBook-Pro:bin elaine\$" is shown at the bottom.

```
Elaines-MacBook-Pro:bin elaine$ ./neo4j start
Active database: graph.db
Directories in use:
home:      /Users/elaine/neo4j-enterprise-3.5.0
config:     /Users/elaine/neo4j-enterprise-3.5.0/conf
logs:       /Users/elaine/neo4j-enterprise-3.5.0/logs
plugins:    /Users/elaine/neo4j-enterprise-3.5.0/plugins
import:     /Users/elaine/neo4j-enterprise-3.5.0/import
data:       /Users/elaine/neo4j-enterprise-3.5.0/data
certificates: /Users/elaine/neo4j-enterprise-3.5.0/certificates
run:        /Users/elaine/neo4j-enterprise-3.5.0/run
Starting Neo4j...
Started neo4j (pid 1649). It is available at http://localhost:7474/
There may be a short delay until the server is ready.
See /Users/elaine/neo4j-enterprise-3.5.0/logs/neo4j.log for current status.
Elaines-MacBook-Pro:bin elaine$
```

When the Neo4j instance starts, it creates the database and writes to the folders for the instance.

Here are some important files to note after you have started the Neo4j instance:



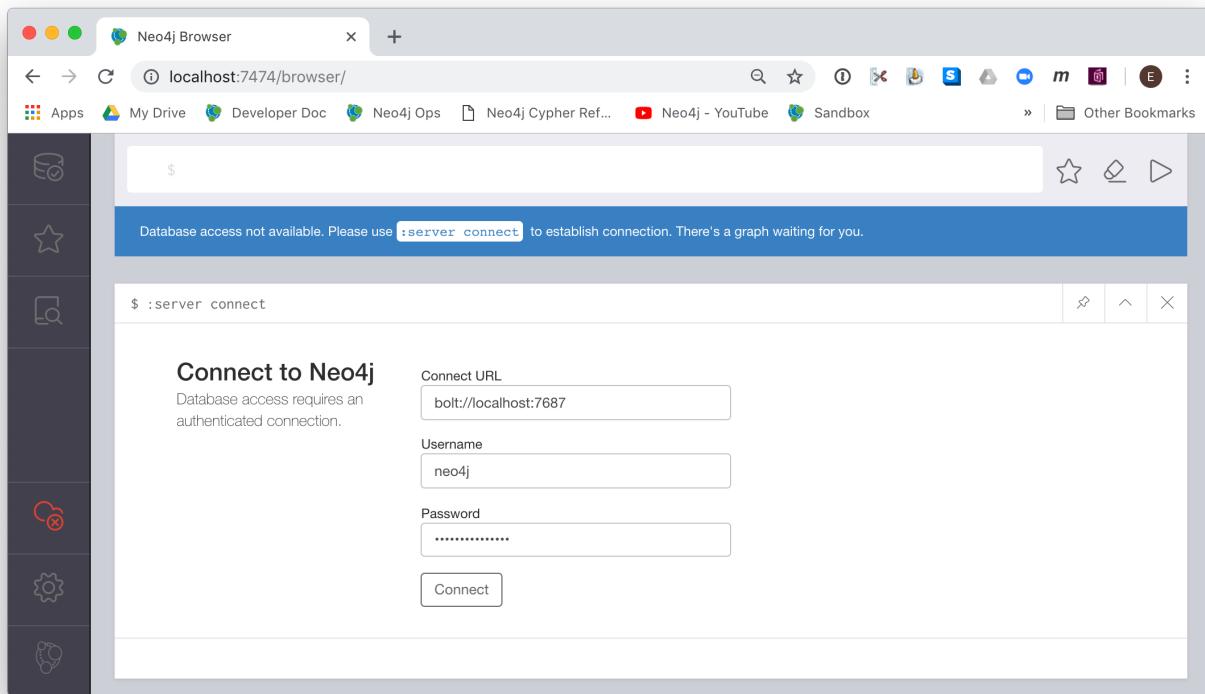
- A new database has been created which is named **graph.db**.
- Information has been written to **neo4j.log**.
- The PID for the Neo4j instance is written in the **run** folder. This is helpful since on systems with multiple JVMs, you need to know which one is the Neo4j instance JVM in the event that you need to kill it.

As an administrator, you should become familiar with the types of records that are written to **neo4j.log**. Here is the tail of that file:

```
logs — bash — 95x18
[Elaines-MacBook-Pro:bin elaine$ cd ..logs
[Elaines-MacBook-Pro:logs elaine$ ls
debug.log      neo4j.log      security.log
[Elaines-MacBook-Pro:logs elaine$ tail neo4j.log
2018-11-27 16:47:51.368+0000 INFO  Starting...
2018-11-27 16:47:57.655+0000 INFO  Initiating metrics...
2018-11-27 16:47:58.540+0000 INFO  Sending metrics to CSV file at /Users/elaine/neo4j-enterprise-3.5.0/metrics
2018-11-27 16:48:01.570+0000 WARN  Server thread metrics not available (missing neo4j.server.thread.reads.jetty.all)
2018-11-27 16:48:01.571+0000 WARN  Server thread metrics not available (missing neo4j.server.thread.reads.jetty.idle)
2018-11-27 16:48:04.374+0000 INFO  Bolt enabled on 127.0.0.1:7687.
2018-11-27 16:48:05.605+0000 INFO  Started.
2018-11-27 16:48:05.792+0000 INFO  Mounted REST API at: /db/manage
2018-11-27 16:48:05.847+0000 INFO  Server thread metrics have been registered successfully
2018-11-27 16:48:06.473+0000 INFO  Remote interface available at http://localhost:7474/
Elaines-MacBook-Pro:logs elaine$
```

When the Neo4j instance starts, you should confirm that it is started by seeing the **Started** record in the log file.

After the Neo4j instance is started, you can access the database with Neo4j Browser using port 7474. In Neo4j Browser, when you attempt to access the Neo4j database, you must first enter the credentials for the *neo4j* user that you configured when you set the password:



Exercise: Start the Neo4j instance

In this Exercise, you will start the Neo4j instance for the first time and confirm that it is started. You will connect to the started database with Neo4j Browser and load some data into the newly created database.

Before you begin

You should have changed the *neo4j* user password to something you will remember.

Exercise steps:

1. Open a terminal on your system.
2. Navigate to the **bin** folder of NEO4J_HOME.
3. Use the **neo4j** script to start the Neo4j instance.
4. Examine the files and folders created for this initial start of the Neo4j instance.
5. Examine the contents of **neo4j.log**.
6. In a Web browser, access the Neo4j database with Neo4j Browser at port 7474 on your local host.
7. Enter the password for the *neo4j* user and then click **Connect**. You now have access to a started and empty database.

8. In the query edit pane of Neo4j Browser enter `:play movie graph` which will open the Browser Guide for working with the Movie graph. It should look like this:

The screenshot shows the Neo4j Browser interface with the title bar "neo4j@bolt://localhost:7687 - 1" and the URL "localhost:7474/browser/". The main content area displays the "Movie Graph" guide. The title "Movie Graph" is bolded, followed by the subtitle "Pop-cultural connections between actors and movies". A descriptive text states: "The Movie Graph is a mini graph application containing actors and directors that are related through the movies they've collaborated on." Below this, a section titled "This guide will show you how to:" lists four steps: 1. Create: insert movie data into the graph, 2. Find: retrieve individual movies and actors, 3. Query: discover related actors and directors, 4. Solve: the Bacon Path. At the bottom of the guide, there is a horizontal navigation bar with several small blue dots and a right-pointing arrow icon.

9. Go to page two of the Browser Guide.

10. Click the Cypher code box. This should bring the Cypher code to the query edit pane. The Cypher statements for loading the database should now be in the query edit pane:

The screenshot shows the Neo4j Browser interface with the title bar "neo4j@bolt://localhost:7687 - 1" and the URL "localhost:7474/browser/". The main content area displays the "Movie Graph" guide. On the left, there is a sidebar with icons for file operations, a search bar, and a gear icon. The main content area has a header "The Movie Graph" and a "Create" section. The "Create" section contains a large block of Cypher code. Below the code, a note says: "To enjoy the full Neo4j Browser experience, we advise you to use Neo4j Browser Sync". At the bottom of the guide, there is a horizontal navigation bar with a right-pointing arrow icon. The query editor at the bottom has the command "\$:play movie graph". The results pane shows the Cypher code again, with the first few lines highlighted in a green box. A red box highlights the play button icon in the top right corner of the results pane.

11. Click the **Play** button. This loads data into the graph.db database for your Neo4j instance. You should have 171 nodes in this database.

neo4j@bolt://localhost:7687 - X

localhost:7474/browser/

Apps My Drive Developer Doc Neo4j Ops Neo4j Cypher Ref... Neo4j - YouTube Sandbox Other Bookmarks

Database Information

Node Labels

- (171) Movie Person

Relationship Types

- (253) ACTED_IN DIRECTED
- FOLLOWS PRODUCED
- REVIEWED WROTE

Property Keys

- born name rating released
- roles summary tagline
- title

Connected as

- Username: neo4j
- Roles: admin
- Admin: server user add

Database

- Version: 3.5.0
- Edition: Enterprise

To enjoy the full Neo4j Browser experience, we advise you to use [Neo4j Browser Sync](#)

\$ CREATE (TheMatrix:Movie {title:'The Matrix', released:1999...)

Graph Table Text Code

*(19) Person(9) Movie(10)

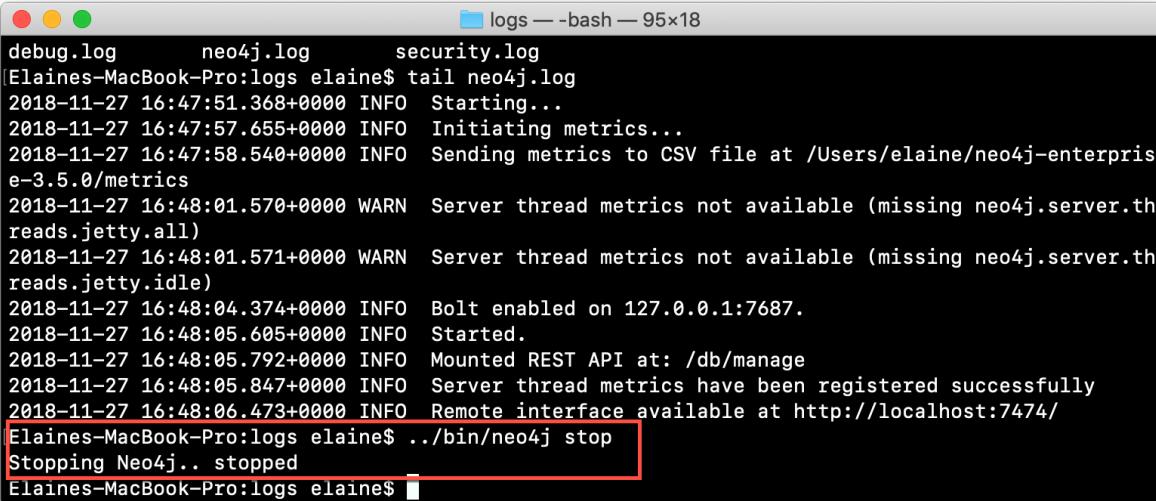
*(20) ACTED_IN(10) DIRECTED(10)

Displaying 19 nodes, 20 relationships.

Stopping the Neo4j instance

You use the same script for stopping the Neo4j instance, providing `stop` as the command:

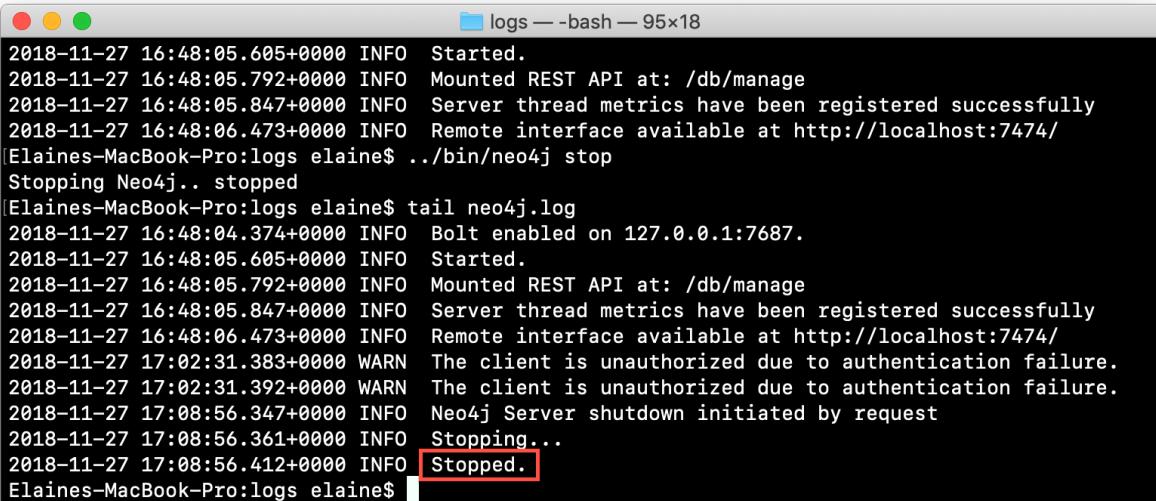
```
neo4j stop
```



A terminal window titled "logs — bash — 95x18" showing the Neo4j log file. The log output includes various INFO and WARN messages related to the server's startup and initial configuration. At the bottom, the command `neo4j stop` is entered, followed by the output "Stopping Neo4j.. stopped".

```
debug.log      neo4j.log      security.log
[Elaines-MacBook-Pro:logs elaine$ tail neo4j.log
2018-11-27 16:47:51.368+0000 INFO  Starting...
2018-11-27 16:47:57.655+0000 INFO  Initiating metrics...
2018-11-27 16:47:58.540+0000 INFO  Sending metrics to CSV file at /Users/elaine/neo4j-enterprise-3.5.0/metrics
2018-11-27 16:48:01.570+0000 WARN  Server thread metrics not available (missing neo4j.server.thread.reads.jetty.all)
2018-11-27 16:48:01.571+0000 WARN  Server thread metrics not available (missing neo4j.server.thread.reads.jetty.idle)
2018-11-27 16:48:04.374+0000 INFO  Bolt enabled on 127.0.0.1:7687.
2018-11-27 16:48:05.605+0000 INFO  Started.
2018-11-27 16:48:05.792+0000 INFO  Mounted REST API at: /db/manage
2018-11-27 16:48:05.847+0000 INFO  Server thread metrics have been registered successfully
2018-11-27 16:48:06.473+0000 INFO  Remote interface available at http://localhost:7474/
[Elaines-MacBook-Pro:logs elaine$ .. /bin/neo4j stop
Stopping Neo4j.. stopped
Elaines-MacBook-Pro:logs elaine$ ]
```

And we can view and confirm that the Neo4j instance is stopped in the log file:



A terminal window titled "logs — bash — 95x18" showing the Neo4j log file. The log output shows the instance starting up and then being stopped. The final message is "Stopped.".

```
2018-11-27 16:48:05.605+0000 INFO  Started.
2018-11-27 16:48:05.792+0000 INFO  Mounted REST API at: /db/manage
2018-11-27 16:48:05.847+0000 INFO  Server thread metrics have been registered successfully
2018-11-27 16:48:06.473+0000 INFO  Remote interface available at http://localhost:7474/
[Elaines-MacBook-Pro:logs elaine$ .. /bin/neo4j stop
Stopping Neo4j.. stopped
[Elaines-MacBook-Pro:logs elaine$ tail neo4j.log
2018-11-27 16:48:04.374+0000 INFO  Bolt enabled on 127.0.0.1:7687.
2018-11-27 16:48:05.605+0000 INFO  Started.
2018-11-27 16:48:05.792+0000 INFO  Mounted REST API at: /db/manage
2018-11-27 16:48:05.847+0000 INFO  Server thread metrics have been registered successfully
2018-11-27 16:48:06.473+0000 INFO  Remote interface available at http://localhost:7474/
2018-11-27 17:02:31.383+0000 WARN  The client is unauthorized due to authentication failure.
2018-11-27 17:02:31.392+0000 WARN  The client is unauthorized due to authentication failure.
2018-11-27 17:08:56.347+0000 INFO  Neo4j Server shutdown initiated by request
2018-11-27 17:08:56.361+0000 INFO  Stopping...
2018-11-27 17:08:56.412+0000 INFO  Stopped.
Elaines-MacBook-Pro:logs elaine$ ]
```

Exercise: Stop the Neo4j instance

Before you begin

The Neo4j instance should be started.

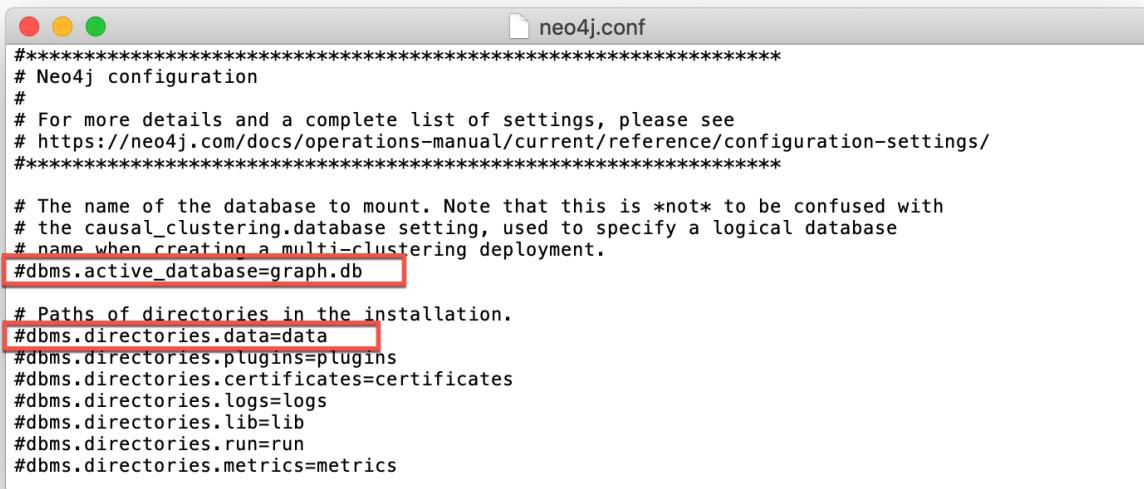
Exercise steps:

1. Open a terminal on your system.
2. Navigate to the **bin** folder of NEO4J_HOME.
3. Use the **neo4j** script to stop the Neo4j instance.
4. Examine the **neo4j.log** file to confirm that the Neo4j instance is stopped.
5. Confirm that you no longer have access to the database with Neo4j Browser.

Renaming a Neo4j database

By default, the Neo4j database is located in the `$NEO4J_HOME/data/databases` folder. The database is represented by a subfolder with the default name, `graph.db`. You should never modify, copy, or move any files or folders under `graph.db`.

A key file for a Neo4j instance is `$NEO4J_HOME/conf/neo4j.conf`. This file contains all settings used by Neo4j at runtime. Here is a portion of the default `neo4j.conf` file that is installed with Neo4j. Two settings here are the name of the active database and the path to the `data` folder. Since this is the default configuration as installed, these settings are commented out in the configuration file because Neo4j assumes that the defaults will be used at runtime.



```
neo4j.conf
*****
# Neo4j configuration
#
# For more details and a complete list of settings, please see
# https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/
*****  

# The name of the database to mount. Note that this is *not* to be confused with
# the causal_clustering.database setting, used to specify a logical database
# name when creating a multi-clustering deployment.
#dbms.active_database=graph.db  

# Paths of directories in the installation.
#dbms.directories.data=data
#dbms.directories.plugins=plugins
#dbms.directories.certificates=certificates
#dbms.directories.logs=logs
#dbms.directories.lib=lib
#dbms.directories.run=run
#dbms.directories.metrics=metrics
```

If you wanted to change the name of the Neo4j database, you could change the folder name `graph.db` to another name, but if you do so, you must uncomment the line in `neo4j.conf` for `dbms.active_database` to match what you have renamed the database folder to. You must make this type of change in the configuration when the Neo4j instance is stopped.

Copying a database

The structure of a Neo4j database is proprietary and could change from one release to another. You should never copy the database from one location in the filesystem/network to another location using terminal shell commands or GUI tools such as File Explorer on Windows or Finder on OS X.

To copy a database that, perhaps you want to have as a backup or you want to give to another user for use on their system, you must:

1. Stop the Neo4j instance.
2. Ensure that the folder where you will dump the database exists.
3. Use the **dump** command of the **neo4j-admin** tool to create the dump file.

Then, if you want to create a database from the dump file to use in a Neo4j instance, you must:

1. Stop the Neo4j instance.
2. Determine what you will call the new database and adjust **neo4j.conf** to use this database as the active database.
3. Use the **load** command of the **neo4j-admin** tool to create the database from the dump file.
4. Start the Neo4j instance.

Dumping a database

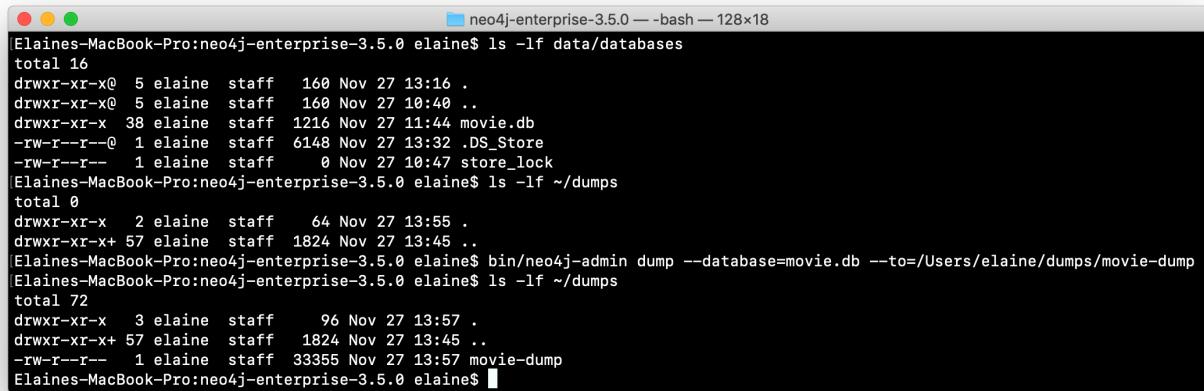
To dump a database, the Neo4j instance must be stopped. Here is how to use the **dump** command of the **neo4j-admin** tool to dump a database to a file:

```
neo4j-admin dump --database=db-folder --to=db-target-folder/db-dump-file
```

where:

<i>db-folder</i>	is the name of the folder representing source database to be dumped.
<i>db-target-folder</i>	is the folder in the filesystem where you want to place the dumped database. This folder must exist.
<i>db-dump-file</i>	is the name of the dump file that will be created.

Here is an example where we have renamed the database to be *movie.db* and we have created a folder named *dumps*. We dump the *movie.db* using **neo4j-admin**:



```

neo4j-enterprise-3.5.0 — bash — 128x18
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ ls -lf data/databases
total 16
drwxr-xr-x@ 5 elaine staff 160 Nov 27 13:16 .
drwxr-xr-x@ 5 elaine staff 160 Nov 27 10:40 ..
drwxr-xr-x 38 elaine staff 1216 Nov 27 11:44 movie.db
-rw-r--r--@ 1 elaine staff 6148 Nov 27 13:32 .DS_Store
-rw-r--r-- 1 elaine staff 0 Nov 27 10:47 store_lock
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ ls -lf ~/dumps
total 0
drwxr-xr-x 2 elaine staff 64 Nov 27 13:55 .
drwxr-xr-x+ 57 elaine staff 1824 Nov 27 13:45 ..
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ bin/neo4j-admin dump --database=movie.db --to=/Users/elaine/dumps/movie-dump
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ ls -lf ~/dumps
total 72
drwxr-xr-x 3 elaine staff 96 Nov 27 13:57 .
drwxr-xr-x+ 57 elaine staff 1824 Nov 27 13:45 ..
-rw-r--r-- 1 elaine staff 33355 Nov 27 13:57 movie-dump
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ 
```

After the dump file, *movie-dump* is created, you can move it anywhere on filesystem or network.

Loading a database

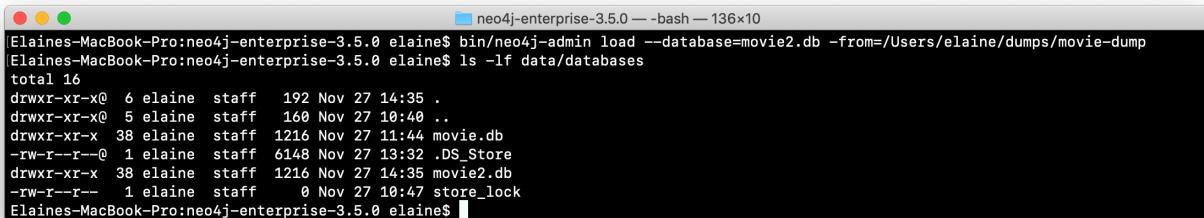
Assuming that you have a dump file to use, you must first determine what the name of the target database will be. If you use an existing database name, the load command, can overwrite the database. If you want to create a new database, then you specify a database name that does not already exist. To perform the load command, the Neo4j instance must be stopped.

Here is how to use the **load** command of the **neo4j-admin** tool to load a database from a file:

neo4j-admin load --from=path/db-dump-file --database=db-folder [-force=true] where:

<i>path</i>	is a folder in the filesystem where the dump file resides.
<i>db-dump-file</i>	is the file previously created with the dump command of neo4j-admin .
<i>db-folder</i>	is the name of the database that will be created or overwritten if --force is specified as true .

Here is an example where we load the contents of **movie-dump** into a database named **movie2.db**.



```

neo4j-enterprise-3.5.0 — bash — 136x10
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ bin/neo4j-admin load --database=movie2.db --from=/Users/elaine/dumps/movie-dump
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ ls -lf data/databases
total 16
drwxr-xr-x@ 6 elaine staff 192 Nov 27 14:35 .
drwxr-xr-x@ 5 elaine staff 160 Nov 27 10:40 ..
drwxr-xr-x 38 elaine staff 1216 Nov 27 11:44 movie.db
-rw-r--r--@ 1 elaine staff 6148 Nov 27 13:32 .DS_Store
drwxr-xr-x 38 elaine staff 1216 Nov 27 14:35 movie2.db
-rw-r--r-- 1 elaine staff 0 Nov 27 10:47 store_lock
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ 
```

In order to access this newly created and loaded database, we must modify **neo4j.conf** to use **movie2.db** as the active database before starting the Neo4j instance.

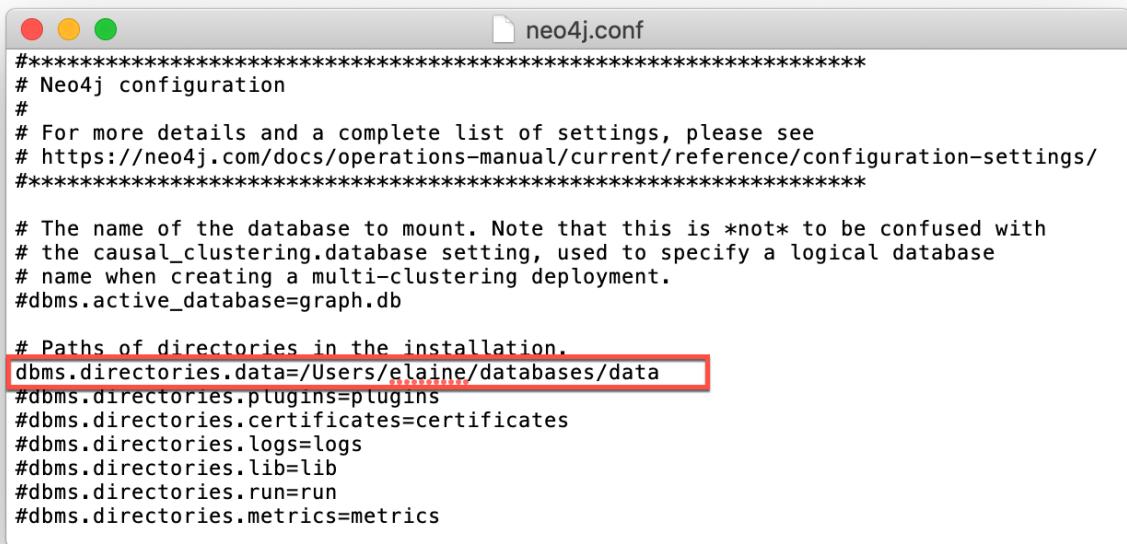
Exercise: Copying a database

TBD

Modifying the location of the database

If you do not want the database used by the Neo4j instance to reside in the same location as the Neo4j installation, you can modify it in the **neo4j.conf** file. If you specify a new location for the data, it must exist in the filesystem.

Here we have specified a new location for the data in the configuration file:



```
#*****
# Neo4j configuration
#
# For more details and a complete list of settings, please see
# https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/
#*****  
  
# The name of the database to mount. Note that this is *not* to be confused with
# the causal_clustering.database setting, used to specify a logical database
# name when creating a multi-clustering deployment.
#dbms.active_database=graph.db  
  
# Paths of directories in the installation.
dbms.directories.data=/Users/elaine/databases/data
#dbms.directories.plugins=plugins
#dbms.directories.certificates=certificates
#dbms.directories.logs=logs
#dbms.directories.lib=lib
#dbms.directories.run=run
#dbms.directories.metrics=metrics
```

We ensure that the location for the data exists and then we can start the Neo4j instance. If this is the first time Neo4j has been started for this location, a new database named **graph.db** will be created.

```
neo4j-enterprise-3.5.0 — bash — 80x25
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ mkdir ~/databases
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ mkdir ~/databases/data
[Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ bin/neo4j start
Active database: graph.db
Directories in use:
  home:          /Users/elaine/neo4j-enterprise-3.5.0
  config:        /Users/elaine/neo4j-enterprise-3.5.0/conf
  logs:          /Users/elaine/neo4j-enterprise-3.5.0/logs
  plugins:       /Users/elaine/neo4j-enterprise-3.5.0/plugins
  import:        /Users/elaine/neo4j-enterprise-3.5.0/import
  data:          /Users/elaine/databases/data
  certificates: /Users/elaine/neo4j-enterprise-3.5.0/certificates
  run:           /Users/elaine/neo4j-enterprise-3.5.0/run
Starting Neo4j.
Started neo4j (pid 2897). It is available at http://localhost:7474/
There may be a short delay until the server is ready.
See /Users/elaine/neo4j-enterprise-3.5.0/logs/neo4j.log for current status.
[Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ ls -lf ~/databases/data/da-
bases
total 0
drwxr-xr-x  4 elaine  staff   128 Nov 27 15:05 .
drwxr-xr-x  4 elaine  staff   128 Nov 27 15:05 ..
-rw-r--r--  1 elaine  staff     0 Nov 27 15:05 store_lock
drwxr-xr-x  37 elaine  staff  1184 Nov 27 15:05 graph.db
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$ ]
```

If you have an existing database that you want to reside in a different location for the Neo4j instance, remember that you must dump and load the database to safely copy it to the new location.

Exercise: Modifying the location of the database

TBD

Checking the consistency of a database

A database's consistency could be compromised if a software or hardware failure has occurred that affects the Neo4j instance. You will learn later in this training about live backups and replicating databases, but if you have reason to believe that a specific database has been corrupted, you can perform a consistency check on it.

The Neo4j instance must be stopped to perform the consistency check.

Here is how you use the `neo4j-admin` tool to check the consistency of the database:

```
neo4j-admin check-consistency --database=db-name --report-dir=report-location [--verbose=true]
```

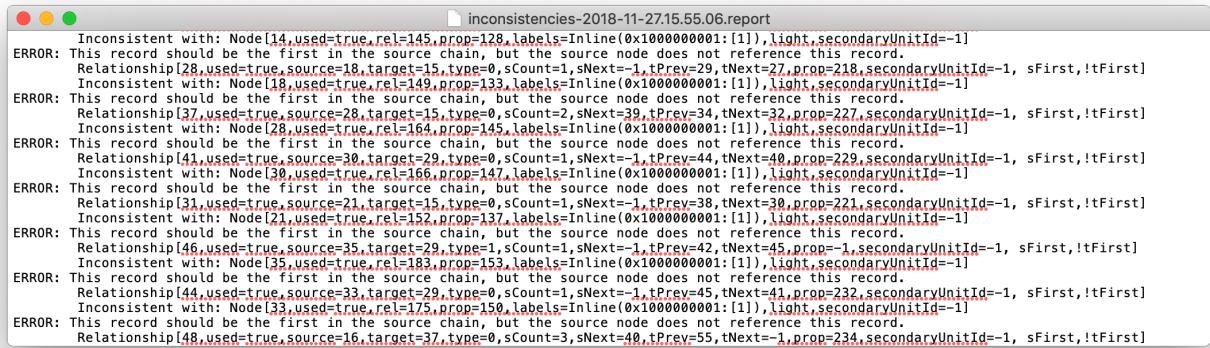
The database named *db-name* is found in the data location specified in **neo4j.conf** file. If the tools comes back with no error, then the database is consistent. Otherwise, an error is returned and a report is written to *report-location*. You can specify verbose reporting. See the *Operations Manual* for more options. For example, you can check the consistency of a backup.

Here is what a successful run of the consistency checker should produce:

Here is an example of what an unsuccessful run of the consistency checker should produce:

```
neo4j-enterprise-3.5.0 — bash — 159x19
Inconsistent with: 53
..... 80%
2018-11-27 21:55:07.705+0000 ERROR [o.n.c.ConsistencyCheckService] The relationship key entries in the store does not correspond with the expected number.
CountsEntry[RelationshipKey[()-->()]: 22]
Inconsistent with: 28
..... 98%
..... 100%
2018-11-27 21:55:07.705+0000 WARN [o.n.c.ConsistencyCheckService] Inconsistencies found: ConsistencySummaryStatistics{
    Number of errors: 539
    Number of warnings: 0
    Number of inconsistent NODE records: 63
    Number of inconsistent RELATIONSHIP records: 319
    Number of inconsistent LABEL_SCAN_DOCUMENT records: 138
    Number of inconsistent COUNTS records: 19
}
2018-11-27 21:55:07.721+0000 WARN [o.n.c.ConsistencyCheckService] See '/Users/elaine/neo4j-enterprise-3.5.0/reports/inconsistencies-2018-11-27.15.55.06.report'
for a detailed consistency report.
command failed: Inconsistencies found. See '/Users/elaine/neo4j-enterprise-3.5.0/reports/inconsistencies-2018-11-27.15.55.06.report' for details.
Elaines-MacBook-Pro:neo4j-enterprise-3.5.0 elaine$
```

If inconsistencies are found, a report is generated and placed in the folder specified for the report location:



```
Inconsistent with: Node[14,used=true,rel=145,prop=128,labels=Inline(0x100000001:[1]),light,secondaryUnitId=-1]
ERROR: This record should be the first in the source chain, but the source node does not reference this record.
Relationship[28,used=true,source=18,target=15,type=0,sCount=1,sNext=-1,tPrev=29,tNext=27,prop=218,secondaryUnitId=-1, sFirst,!tFirst]
Inconsistent with: Node[18,used=true,rel=149,prop=133,labels=Inline(0x100000001:[1]),light,secondaryUnitId=-1]
ERROR: This record should be the first in the source chain, but the source node does not reference this record.
Relationship[37,used=true,source=28,target=15,type=0,sCount=2,sNext=39,tPrev=34,tNext=32,prop=227,secondaryUnitId=-1, sFirst,!tFirst]
Inconsistent with: Node[28,used=true,rel=164,prop=145,labels=Inline(0x100000001:[1]),light,secondaryUnitId=-1]
ERROR: This record should be the first in the source chain, but the source node does not reference this record.
Relationship[41,used=true,source=30,target=29,type=0,sCount=1,sNext=-1,tPrev=44,tNext=40,prop=229,secondaryUnitId=-1, sFirst,!tFirst]
Inconsistent with: Node[30,used=true,rel=166,prop=147,labels=Inline(0x100000001:[1]),light,secondaryUnitId=-1]
ERROR: This record should be the first in the source chain, but the source node does not reference this record.
Relationship[31,used=true,source=21,target=15,type=0,sCount=1,sNext=-1,tPrev=38,tNext=30,prop=221,secondaryUnitId=-1, sFirst,!tFirst]
Inconsistent with: Node[21,used=true,rel=152,prop=137,labels=Inline(0x100000001:[1]),light,secondaryUnitId=-1]
ERROR: This record should be the first in the source chain, but the source node does not reference this record.
Relationship[46,used=true,source=35,target=29,type=1,sCount=1,sNext=-1,tPrev=42,tNext=45,prop=1,secondaryUnitId=-1, sFirst,!tFirst]
Inconsistent with: Node[35,used=true,rel=183,prop=153,labels=Inline(0x100000001:[1]),light,secondaryUnitId=-1]
ERROR: This record should be the first in the source chain, but the source node does not reference this record.
Relationship[44,used=true,source=33,target=29,type=0,sCount=1,sNext=-1,tPrev=45,tNext=41,prop=232,secondaryUnitId=-1, sFirst,!tFirst]
Inconsistent with: Node[33,used=true,rel=175,prop=150,labels=Inline(0x100000001:[1]),light,secondaryUnitId=-1]
ERROR: This record should be the first in the source chain, but the source node does not reference this record.
Relationship[48,used=true,source=16,target=37,type=0,sCount=3,sNext=40,tPrev=55,tNext=-1,prop=234,secondaryUnitId=-1, sFirst,!tFirst]
```

Inconsistencies in a database are a serious matter that should be looked into with the help of Technical support. Later in this training you will learn more about troubleshooting problems that are detected.

Exercise: Checking consistency of a database

TBD