SQL Query API Documentation

Contents

[**Starting the API application** 2](#_Toc4265387)

[**Editing Configuration File** 2](#_Toc4265388)

[**Using Front End HTML Page** 3](#_Toc4265389)

[Database Query Page: 3](#_Toc4265390)

[Compare Scheduled Files: 4](#_Toc4265391)

[**API Endpoints:** 4](#_Toc4265392)

# **Starting the API application**

To start the application you must run the .jar file with the directory of the configuration file as an argument.

java -jar API\_Project.jar “C:\Users\user\MihialProject\config.json”

# **Editing Configuration File**

Configuration file should be in JSON format as below:

|  |
| --- |
| {  "scheduled\_file\_path":"C:\\Users\\Ray\\Desktop\\MihialProject\\scheduled\_files\\",  "xlsx\_file\_path":"C:\\Users\\Ray\\Desktop\\MihialProject\\xlsx\_files\\",  "sql":{  "sql\_db\_url":"localhost:1450",  "sql\_db\_name":"testDatabase",  "sql\_db\_user":"usernametest",  "sql\_db\_pass":"passwordtest",  "sql\_hidden\_query":"SELECT \* FROM testTable ORDER BY age ASC",  "sql\_db\_query1":"SELECT \* FROM testTable ORDER BY firstname ASC",  "sql\_query1\_interval\_days": 1  }  } |

|  |  |
| --- | --- |
| scheduled\_file\_path | Path to save automatically scheduled files to |
| xlsx\_file\_path | Path to save xslx files to for downloading |
| sql | SQL database details |
| sql\_db\_url | SQL database URL (127.0.0.1:1450) |
| sql\_db\_name | SQL database name |
| sql\_db\_user | SQL user name |
| sql\_db\_pass | SQL user password |
| sql\_hidden\_query | SQL statement which can be run from html page without user seeing it. |
| sql\_db\_query1 | SQL statement which will run automatically and save file locally (path defined in “scheduled\_file\_path”) |
| sql\_query1\_interval\_days | Number days between scheduled “sql\_db\_query1” |

**NOTE:** The value of sql\_query1\_interval\_days, should not be in quotes.

**NOTE:** Ensure any directories defined in the configuration file exist.

# **Using Front End HTML Page**

If run locally, javascript/scripts/activeX permissions may need to be enabled in your browser for the page to run correctly. If run on a server this should not be an issue.



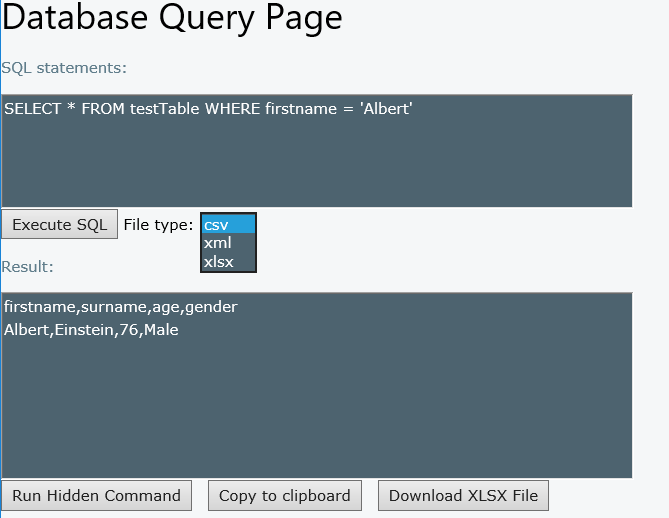
**NOTE:**

I suggest restricting access to this page, for security purposes, as it allows direct access to the database.

## Database Query Page:

Here you can manually enter any MS SQL statement, select a file type from the dropdown menu and hit the “Execute SQL” button.

* If csv or xml is selected then the return fields will appear in the Result box.
* If xlsx was select, the .xlsx can then be downloaded by pressing the Download button.



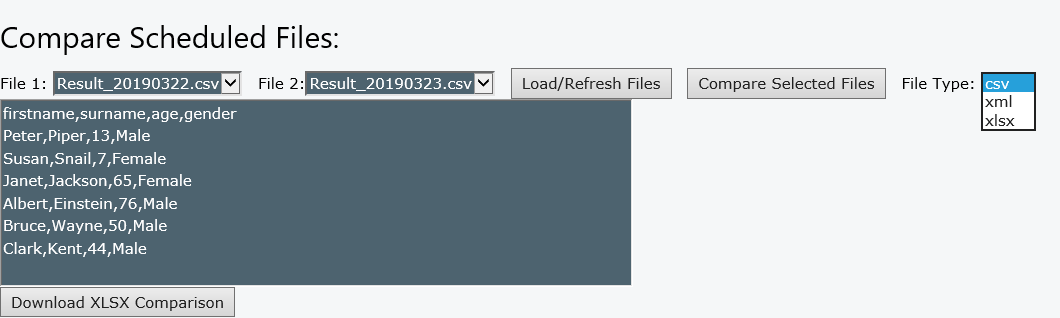
* The ‘Run Hidden Command’ button will run the “sql\_hidden\_query” line, defined in the configuration file.
* The ‘Copy to clipboard’ button will copy the Result Textbox to the clipboard.

## Compare Scheduled Files:

This section allows you to select 2 files, from a list of automatically scheduled SQL query results.

* Note filenames contain dates for easy comparison.

The 2 files can then be compared, and result will show only rows from the database which are different in the 2 queries.



# **API Endpoints:**

The HTML page uses javascript to run ‘CURL’ methods, which access the API endpoints.

However these endpoints can be accessed with the HTML page should you wish.

By default the application runs on port 4567. This can be altered in the Application.java file by adding the port to the main method e.g.:

|  |
| --- |
| public static void main(String[] args) throws Exception {  *port*(4555); |

**Default base url:**

http://localhost:4567

**The Endpoint details are:**

/alive

|  |  |
| --- | --- |
| Method | Get |
| Response | “I am alive!” |

/query/sql/csv

|  |  |
| --- | --- |
| Method | Post |
| Request Body | SQL Query e.g. “SELECT \* FROM table” |
| Response | SQL query result in CSV format |
| NOTE | Adding ‘/hidden’ to the endpoint will run the hidden sql statement defined in the configuration file.  /query/sql/csv/hidden |

/query/sql/xml

|  |  |
| --- | --- |
| Method | Post |
| Request Body | SQL Query e.g. “SELECT \* FROM table” |
| Response | SQL query result in XML format |
| NOTE | Adding ‘/hidden’ to the endpoint will run the hidden sql statement defined in the configuration file.  /query/sql/xml/hidden |

/query/sql/xlsx

|  |  |
| --- | --- |
| Method | Post |
| Request Body | SQL Query e.g. “SELECT \* FROM table” |
| Response | Path to xlsx file containing result of query. |
| NOTE | Adding ‘/hidden’ to the endpoint will run the hidden sql statement defined in the configuration file.  /query/sql/csv/hidden |

/fileList

|  |  |
| --- | --- |
| Method | Post |
| Response | A list of comma separated file names. Files contain results of scheduled SQL statements |

/compareFiles

|  |  |
| --- | --- |
| Method | Post |
| Request Body | 2 comma separated file names followed by filetype e.g. “file1.csv,file2.csv,xml” |
| Response | SQL query result in CSV/XML format or the name of generated XSLX file. |

**Example curl can be run in command prompt / terminal:**

curl -d "SELECT \* FROM testTable" -X POST http://localhost:4567/query/sql/csv