SQL File Evaluation (SQLFE) Tool
Installation Instructions

1. The SQLFE software, for now available either as:
    a. a standalone executable, or
    b. as Java source code to be loaded into an integrated development environment (IDE) such as Eclipse or IntelliJ IDEA, is available for download through GitHub at: https://github.com/wagnerpj42/sql-file-evaluation .
    c. Additional files, such as sample SQL submission files, a sample assignment properties file, and this documentation, are also available at this GitHub location.
2. INSTALLATION PROCESS - Executable JAR file
    a. The executable JAR file named SQLFE.jar can be downloaded and run as a Java executable (for example, at the command line, one can run it with the command:
        i. java –jar SQLFE.jar
    b. All necessary libraries are packaged in the executable JAR, so the only precondition for running the SQLFE jar file is to have a Java 8 JRE already installed
3. INSTALLATION PROCESS – IDE
    a. SQLFE currently can be executed through an Integrated Development Environment (IDE). This might be done if you are modifying the source code for custom applications.
    b. SQLFE has been tested and run on the Windows 10 operating system under the Eclipse and IntelliJ IDEA IDE environments. A valid Java 8 JDK containing JavaFX support must be available in the project space, though newer versions may work as well.
        i. Since the SQLFE GUI uses JavaFX, support for JavaFX must be present in your Java execution environment. We have verified that the jdk1.8.0_abc library (where abc is the current version number) that is freely available from oracle.com has this support. Creating an oracle.com account will generally be required to get this package. This library must be installed on your system and added to the SQLFE project.
    c. Create a project within your IDE.
        i. The name of the project does not matter, but it always helps to use a descriptive name. e.g. SQLFE.
    d. You must download at least the src folder found at this GitHub location. The SQLFE source code must be placed within an IDE so that the SQLFE src folder is at the top level of the IDE project, possibly replacing any src folder automatically created by the IDE.

e. Any necessary database connector JAR files must be available in the IDE project structure. Below is a list of supported database management systems (DBMSs) and their database connector files:
    i. Oracle – ojdbc8.jar (preferred/tested), also possibly ojdbc7.jar (tested)
    ii. MySQL 5.x – mysql-connector-java-5.1.23-bin.jar (tested with MySQL Server 5.5), though other versions may work
    iii. MySQL 8.0.xx - mysql-connector-java-8.0.22.jar (tested with MySQL Community Server 8.0.22), though other versions may work.
f. If these database connector files are not already present within the SQLFE project, these can be added as follows, assuming that you've downloaded the files from an available online source to someplace on your target system:
    i. Eclipse – Right-click on your project name, choose Properties, choose Java Build Path, choose the Libraries tab, choose Add External JARs, browse to the location of the jar file (e.g. ojdbc8.jar) and choose Open, choose Apply and Close.
    ii. IntelliJ IDEA – Choose File, choose Project Structure, choose Project Settings / Modules, choose Dependencies, choose + on right, choose JARS or Directories, browse to path of the jar file (e.g. ojbc8.jar), OK as needed to finish.
g. If you want to run the available JUnit tests, SQLFE must also have access to a JUnit 4 library. This must be added to the project structure of SQLFE.
    i. Eclipse – Right-click on your project name, choose Properties, choose Java Build Path, choose the Libraries tab, choose Add Library, choose JUnit and Next, choose JUnit 4 from drop-down list, choose Finish, choose Apply and Close.
    ii. IntelliJ IDEA – Go into the sqlfe/junit package and open one of the JUnit test files (e.g. CondAndCountTests). If one of the test annotations (that is, @Test) is red and signaling an error, that means JUnit is not available. Click on that annotation, which should then show a red bulb with suggestions. Within that, choose Add JUnit 4 Library and finish.
    iii. You will also need to edit AbstractTest.java in the sqlfe/junit package to instantiate the correct SQLFE data access object for your environment (placeholder line assumes Oracle) and pass the correct connection information to that object.
h. At this point, SQLFE should be ready to run, by opening the Main.java file in the sqlfe/general package and choosing Run As/Java Application.