**Introduction**

"Excel Reader" is a Java-based application.

The project's primary purpose is to read data from a large Excel file, modify it as per user-defined requirements, and then insert the amended data into another Excel file.

**Build and Test**

To run the application, you have two options:

1. After downloading the "Demo\_Number\_2" file, refer to the "Excel Reader Tutorial" located in the "Excel Reader Tutorial" data folder (src/main/resources/Excel Reader Tutorial/Excel Reader Tutorial.docx).

Follow the instructions to run the application via the terminal.

1. If you have successfully cloned the project to your desired folder, you can execute the application using your preferred IDE by following the instructions provided below (#GettingStarted).

**Getting Started**

1. **Familiarizing Yourself with Excel Reader**

To get acquainted with the application, first, read the "Excel Reader Tutorial," which can be found in:

"src/main/resources/Excel Reader Tutorial/Excel Reader Tutorial.docx" within the codebase.

1. **Starting the Application from the Command Line**

Refer to the "RUNNING THE APPLICATION" section in the "Excel Reader Tutorial" to learn how to start the application from the command line (CMD).

1. **Understanding the Excel Reader Codebase**

To launch the application, start by cloning the master branch into a folder of your choice.

Once the cloning process is complete, follow these steps to run the application:

* 1. Create a new branch from the master branch with a name of your choice.
  2. In IntelliJ (or any other IDE you prefer), navigate to: Run -> Edit Configurations... and create a new configuration.

Click the "plus" button to "Add a New Configuration."

The configuration type should be "Application." After creating the configuration, select the main class by choosing "Main" in the "Main Class" box.

Next, paste the following line of code (src/main/resources/META-INF/Plik.txt) into the "Program Arguments" field.

* 1. Specify the input and output data files to perform the necessary operations.

To change the input and output files, open the src/main/resources/META-INF folder and the Plik.txt file.

For this demonstration, the input data file is located in: src/main/resources/DataFiles.

After cloning your application, provide the full path to this file by right-clicking on ChargedHoursNew.xlsb and selecting Copy Path/Reference, then choosing the "Absolute Path" option.

Based on the repository in use, the path will be:

C:\ExcelReader\src\main\resources\DataFiles\ChargedHoursNew.xlsb

**Important Note:**

Ensure the input and output paths are in the correct format. Notice that single backslashes (\) separate the directory parts.

To format the directory correctly, replace single backslashes with double backslashes (\\).

Copy the modified absolute path into the "inputFile" entry of the configuration file.

For example, the modified absolute path should be:

C:\ExcelReader\src\main\resources\DataFiles\ChargedHoursNew.xlsb

**Note:** If the configuration file contains the input file path in an incorrect format, the application will fail to run.

In this case, open the Plik.txt file in the META-INF folder (src/main/resources/META-INF) and paste the absolute path into the relevant field (e.g., the "inputFile" field).

After defining the input file, specify the location for the output file. You can choose any desired location and specify the output file format (xlsx is recommended).

To modify the output file, locate the "newFileName" entry in the configuration file at the bottom.

Provide the absolute path for this entry, using the same format as the input file path.

1. Once the required code is inserted, you can run the application from the "Main" method. To do this, select the "Main" Java class by double-clicking it in your IDE.

After selecting the class, run the application by clicking Run -> Run 'Main'.

1. Note that you need to install Maven on your computer to run the application.

After cloning the project to your desired folder, go to the "Maven" tab on the right side of your IntelliJ IDEA IDE.

Select the "Lifecycle" folder and then click "install."

Double-clicking the gear icon with the "install" label will install the Maven project in your chosen folder.

**Note:** This application was developed using IntelliJ IDEA. If you want to run it in another IDE, adjust the instructions accordingly.

1. To create a new jar file, use the Maven "package" option within the Lifecycle folder (refer to the point above).

Double-click the "package" option, and Maven will generate a new jar file in the "target" folder of your project directory (the output destination will be shown in the terminal).

You can copy and paste this jar file into your Demo folder to add additional functionality to the application.

**Contribute**

**WHAT HAS BEEN DONE:**

1. The application can copy and replace data based on the configuration file.
2. Excel functions have been written to check if a cell is EQUALS, NOT\_EQUALS, CONTAINS, NOT\_CONTAINS, LESS, or GREATER than another cell.

**WHAT NEEDS TO BE DONE:**

1. The application must be able to perform the operations mentioned above on large Excel files.
2. Excel functions must be written for the BEFORE, AFTER, and BETWEEN operations.