

## 1. Download the Code from GITHUB

1. Download the project from GITHUB using eclipse plugin. Repository URL is <https://github.com/aravindk7/fundReport.git>
2. Alternatively, by accessing the URL: <https://github.com/aravindk7/fundReport> , click “Clone or download” option and click “Download ZIP” option.

## 2. Building the project

Prerequisites:

1. Maven should be installed.
2. Ensure MVN\_HOME and JAVA\_HOME is set in class path.
3. Since the project is based in Spring Boot, it will download all the dependent jar files from external Artifactory. Please configure proxy if any.

Building the project from eclipse with STS plugin:

Import the project as existing maven project.

To build the fat jar file (Run the below command from eclipse or command prompt):  
mvn clean install

- Jar file will be in <local repo>\repository\com\lonsec\fundreport\0.0.1-SNAPSHOT\ fundreport-0.0.1-SNAPSHOT.jar
- Unit Test cases will be automatically executed during build. Reports are available in fundReport\target\surefire-reports

## 3. Running the project

Running the application from command line

```
>java -jar fundreport-0.0.1-SNAPSHOT.jar
```

*Enter the Option (1/2/3):*

- 1 - To Process the input files (**Give a csv file location or folder containing all CSV files**)
- 2 - To Generate Monthly Performance File
- 3 - To Exit

Note: Report file will be generated in the same location as the jar file

Running the application from Eclipse with STS plugin:

Select com.lonsec.App.java and run as → 3 Spring Boot App

## 4. Assumptions

- Assumption is made to design this as batch application.
- Data will be retained only for single run.
- Detailed error handling is not done.
- CSV records should have values for all the fields. If the values are missing, then the record is not processed.

- The headers of the CSV file are validated before parsing the contents to identify the file type. The input CSV file should have the same headers as in sample data.
- All the configurations for excess value and out Performance text is externalized to application.properties. This is part of jar file but it can be externalized as well.
- To add external config file use --spring.config.location=override.properties option when running the file
- App.log will also be generated in the same location as jar file.
- Detailed error/ exception handling is not done.

## 5. Application Overview

- Application is built using Spring Boot.
- Java 8 Lambda functions are used for all collection manipulations.
- Embedded H2 database is used. To view the data after option 1 use the web console
  - URL : <http://localhost:8082/>
  - JDBC URL : jdbc:h2:mem:testdb
  - User Name: sa

Note: if port 8082 is already used, then there will be exception in application startup.