**Prerequisites**

1. Load the source files ‘Positions.csv’ and ‘Time.csv’ into your Filestore and update the path accordingly in your Positions\_class\_sourcecode.dbc or Positions\_Main\_eggPackage.dbc

Positions\_location = "/FileStore/tables/Positions.csv"

Time\_location = "/FileStore/tables/Time.csv"

1. .dbc files are to be imported in Databricks and can be tested accordingly

**Additional Information:**

1. Positions Class (Positions\_Package.py) is exported as egg file (Python 3.7) and installed in Databricks as a library.

Under Positions\_Main\_eggPackage.dbc the module is imported as below

from Positions\_Package import Positions\_Package

1. Date validation is implemented
2. GetTrades function accepts date and location as parameters
3. AuditLogs function also accepts date and location as parameters
4. GetTrades function Output is stored as CSV and also in delta format.

For CSV please use the respective YYYYMMDD\_HHMM.csv to view the data.

1. AuditLogs function records the below data for logs

Position\_date

Load\_date\_time

User\_name

Total\_rows\_count

GetTrades:

Input

# read GetTrades Output delta or csv (for csv please add the YYYYMMDD\_HHSS.csv to the Position location)

Positions\_location = "/FileStore/df/PowerPosition\_20220921\_1730.csv"

file\_type = "csv"

# CSV options

infer\_schema = "false"

first\_row\_is\_header = "true"

delimiter = ","

# read spark dataframe.

read\_df = spark.read.format(file\_type) \

.option("inferSchema", infer\_schema) \

.option("header", first\_row\_is\_header) \

.option("sep", delimiter) \

.load(Positions\_location)

display(read\_df)

Output

Table

Description automatically generated

AuditLogs:

Input

# read Audit logs Output delta

Audit\_location = "/FileStore/df/Auditlogs\_PowerPosition"

file\_type = "delta"

# CSV options

infer\_schema = "false"

first\_row\_is\_header = "true"

delimiter = ","

# read spark dataframe.

read\_df = spark.read.format(file\_type) \

.option("inferSchema", infer\_schema) \

.option("header", first\_row\_is\_header) \

.option("sep", delimiter) \

.load(Audit\_location)

display(read\_df)

Output

Table

Description automatically generated