

Test project specs.

Goal: Develop code to process an input data file and apply mapping data from a Vidrio mapping file to generate an account valuation and account performance output data sets to be imported to the Vidrio platform. Input data comes from an outside data vendor and contains their product/entity names. Those entity names must be converted to Vidio internal entity names to be accepted by the platform data import facilities. Specs below are to explain the entire data management process for this project.

Input Data File:

Ex_input_file_02.04.2021.xlsx

Mapping File:

Ex_mapping_file.xlsx

Tasks Description:

1. (Python) Extract a time stamp portion from the input file name and display it in MM/DD/YYYY format when the program starts.
2. (Python) Read data from the input file and save its content to database.
3. (Python) Read the mapping file and save its content to database.
4. (Python) Using “pandas” package to generate portfolio valuation output file (“DAV Proforma Acc Analy.xlsx”) that will have the following columns (col names in bold, specs below):

Reference Day: taken from the time stamp portion of the input data file

Periodicity: ‘Daily’

Investor Account UID: “HFRIILAU” from column B in the “Index Data” tab of the input data file

Investor Account Long Name: From column C in the “Index Data” tab of the input data file where ISIN = “HFRIILAU” and Date= **Reference Day**

Investment Account UID: Leave blank

Investment Account Long Name: Have to translate company name from “Constituents” tab of the data input file to Product Name from the mapping file using “ISIN” column from the “Constituents” tab and “Counterparty ID” column in the mapping file as matching keys. Use these keys to match all records in the output data set.

Attribution Gross: “Gross Contribution to Index” column from “Constituents” tab

Attribution Net: “Net Contribution to Index” column from “Constituents” tab

Opening Allocation: 'Beginning Weight %' column from “Constituents” tab

Closing Allocation: 'End Weight %' column from “Constituents” tab

Opening Equity: ‘Previous Day NAV’ column from Index Data tab for ‘HFRI-I Liquid Alt UCITS Index’ index for **Reference Day**

Closing Equity: ‘NAV’ column from Index Data tab for ‘HFRI-I Liquid Alt UCITS Index’ index for **Reference Day**

Investment Performance: '% Price Change' column from “Constituents” tab

Investment Adj Opening Balance: 'Opening Allocation' * 'Opening Equity'

Investment Closing Balance: 'Closing Allocation' * 'Closing Equity'

Portfolio Opening Balance: 'Investment Adj Opening Balance'

Portfolio Closing Balance: 'Investment Closing Balance'

5. (Python) Using “pandas” package to generate portfolio valuation output file (“Portfolio Account NAV.xlsx”) that will have the following columns:

Portfolio Account UID: “HFRIILAU” from column B in the “Index Data” tab of the input data file

Account Long Name: From column C in the “Index Data” tab of the input data file where ISIN = “HFRIILAU” and Date= **Reference Day**

Date: taken from the time stamp portion of the input data file

NAV/Share: ‘NAV’ column from Index Data tab for ‘HFRI-I Liquid Alt UCITS Index’

Final: ‘True’

6. (Python) Save output data written to “DAV Proforma Acc Analy.xlsx” and “Portfolio Account NAV.xlsx” files in database.
7. (Python) Convert output xlsx files into xml files (specs below):
 - a. XML structure for “DAV Proforma Acc Analy.xlsx” file (DAV_Proforma_Acc_Analy.xml):

```
<DAVPROFORMAACC>
  <DAVPROFORMAACCANALY
    col1_from_xlsx_file="value_from_xlsx_for_col_1
    col2_from_xlsx_file="value_from_xlsx_for_col_2
    .....
    col2_from_xlsx_file_N="value_from_xlsx_for_col_N" />
  </DAVPROFORMAACC>
```
 - b. XML structure for “Portfolio Account NAV.xlsx” file (Portfolio Account NAV.xml):

```
<DAVPROFORMAACC>
  <PORTFOLIOACCOUNTNAV
    account_long_name="HFRI-I Liquid Alt UCITS Index - Account" date="2021-02-
    04T00:00:00" nav_x002F_share="1.286961641823181e+003" final="TRUE"/>
  </DAVPROFORMAACC>
```
8. Create a SQL query to list Vidrio internal long names for all entity names from the input data file even though if they do not have counterparts in the mapping files.
9. Using input and mapping data saved in the database by the python module write Java code
To produce “DAV Proforma Acc Analy.xlsx” and “Portfolio Account NAV.xlsx” files based on the same columns specifications that was used for the Python portion of this project.
10. Please submit both “DAV Proforma Acc Analy.xlsx” and “Portfolio Account NAV.xlsx” files for a review as well as all Python, Java and SQL code used to complete this project.