**Testing and Validation**

**Notes:**

1. **The folders and the CSV files used for all the test cases are made available with the source.**
2. **Following are the steps to setup the database and the schema and the package to do the required as a first-time requirement. These and the other documentation and the usage are mentioned in the package specs and the body also.**

Login to the sys from the command prompt using the sqlplus:

SQLPLUS / AS SYSDBA;

Create the STOCK schema used for this task:

CREATE USER STOCK IDENTIFIED BY STOCK;

GRANT CONNECT, RESOURCE, CREATE ANY DIRECTORY TO STOCK;

CONNECT STOCK/STOCK@ORCL21C

Note: The complete process was tested on the 11g database also in addition to the 21c

**Create the schema and the database directories:**

create or replace directory EXCHANGEDATACSVPATH as 'd:\exchange\_data\';

create or replace directory DEPOSITORYDATACSVPATH as 'd:\depository\_data \';

|  |
| --- |
| The following four tables created are required to be created in this schema, before creating and compiling the package specs and the body, two for the logs and two for the stock data.  The two log tables are as follows:    create table TheMainComparisonLogOfTwoCSVs  (  MatchedStkCnt\_User\_ID VARCHAR2(50),  MatchedStkCnt\_Stock\_ID VARCHAR2(50),  MatchedStkCnt\_Stock\_Name VARCHAR2(200),  MatchedStkCnt\_Stock\_Count NUMBER(30),  MsMatchedStkCnt\_User\_ID VARCHAR2(50),  MsMatchedStkCnt\_Stock\_ID VARCHAR2(50),  MsMatchedStkCnt\_Stock\_Name VARCHAR2(200),  MsMatchedStkCnt\_Stock\_Count\_e NUMBER(30),  MsMatchedStkCnt\_Stock\_Count\_d NUMBER(30)  );  create table TheLogOfEdgeCasesAndCSVFileIssues  (  LogIDOrderedStep\_Serial NUMBER(10),  LogIDOrderedStep\_Desc VARCHAR2(400),  MissngOrIvldCSV\_exchangedata VARCHAR2(4000),  MissngOrIvldCSV\_depositorydata VARCHAR2(4000),  Dup\_User\_ID\_exchange\_data VARCHAR2(50),  Dup\_Stock\_ID\_exchange\_data VARCHAR2(50),  Dup\_Stock\_Name\_exchange\_data VARCHAR2(200),  Dup\_Stock\_Count\_exchange\_data VARCHAR2(50),  Dup\_User\_ID\_depository\_data VARCHAR2(50),  Dup\_Stock\_ID\_depository\_data VARCHAR2(50),  Dup\_Stock\_Name\_depository\_data VARCHAR2(200),  Dup\_Stock\_Count\_dep\_data VARCHAR2(50),  Incst\_Stock\_ID\_exchange\_data VARCHAR2(50),  Incst\_Stock\_Name\_exchange\_data VARCHAR2(50),  Incst\_Stock\_ID\_depstry\_data VARCHAR2(50),  Incst\_Stock\_Name\_depstry\_data VARCHAR2(50),  TotalRows\_exchange\_data NUMBER(30),  TotalRows\_depository\_data NUMBER(30),  MissingRows\_In\_exchange\_data NUMBER(30),  MissingRows\_In\_depository\_data NUMBER(30),  TotalRows\_Matched NUMBER(30),  TotalRows\_MtchdButDiffStkCntNm NUMBER(30)  );      The two stock tables are as follows. The "not null" constraints in the following two stock tables, check the - "Edge Cases - Incomplete rows in the CSV (e.g., missing stock data or user details", at the very root of the process  create table exchange\_data\_extrnl\_fromcsv  (  User\_ID VARCHAR2(50) not null,  Stock\_ID VARCHAR2(50) not null,  Stock\_Name VARCHAR2(200) not null,  Stock\_Count VARCHAR2(50) not null  )  organization external  (  type ORACLE\_LOADER  default directory EXCHANGEDATACSVPATH  access parameters  (  records delimited by newline  skip 1  fields terminated by ","  --missing field values are null  --reject rows with all null fields  )  location ('exchange\_data.csv')  )  reject limit unlimited;    create table depository\_data\_extrnl\_fromcsv  (  User\_ID VARCHAR2(50) not null,  Stock\_ID VARCHAR2(50) not null,  Stock\_Name VARCHAR2(200) not null,  Stock\_Count VARCHAR2(50) not null  )  organization external  (  type ORACLE\_LOADER  default directory DEPOSITORYDATACSVPATH  access parameters  (  records delimited by newline  skip 1  fields terminated by ","  --missing field values are null  --reject rows with all null fields  )  location ('depository\_data.csv')  )  reject limit unlimited; |

**Details about the structure and the contents of the two log tables and the details of the operation of the loggings:**

|  |  |
| --- | --- |
| **The table TheMainComparisonLogOfTwoCSVs which logs the outcome of the comparison (i.e., matches, mismatches)** | |
| **Column in the table** | **Description** |
| MatchedStkCnt\_User\_ID VARCHAR2(50), | Thesed four coulms store the the row which are found exactly matching in the two CSV files |
| MatchedStkCnt\_Stock\_ID VARCHAR2(50), |
| MatchedStkCnt\_Stock\_Name VARCHAR2(200), |
| MatchedStkCnt\_Stock\_Count NUMBER(30), |
| MsMatchedStkCnt\_User\_ID VARCHAR2(50), | Thesed five coulms store the the row which are found not matching in the two CSV files by Stock\_Count. The coumn MsMatchedStkCnt\_Stock\_Count\_e shows the Stock\_Count value in the exchange\_data.CSV while the coumn MsMatchedStkCnt\_Stock\_Count\_d shows the Stock\_Count value in the depository\_data.CSV  When the data is inserted in this main log table, a summary data is also logged in the other log table, TheLogOfEdgeCasesAndCSVFileIssues, in the following coulns which are the sumaary of the count of the comparison operation (refe to the details of the each columns of this secondary log table):  LogIDOrderedStep\_Serial  LogIDOrderedStep\_Desc  TotalRows\_exchange\_data TotalRows\_depository\_data MissingRows\_In\_exchange\_data MissingRows\_In\_depository\_data TotalRows\_Matched TotalRows\_MtchdButDiffStkCntNm |
| MsMatchedStkCnt\_Stock\_ID VARCHAR2(50), |
| MsMatchedStkCnt\_Stock\_Name VARCHAR2(200), |
| MsMatchedStkCnt\_Stock\_Count\_e NUMBER(30), |
| MsMatchedStkCnt\_Stock\_Count\_d NUMBER(30) |

|  |  |
| --- | --- |
| **The table TheLogOfEdgeCasesAndCSVFileIssues which logs the outcome of the summary of the comparison and the Edge Cases (Corrupt files or data)** | |
| **Column in the table** | **Description** |
| LogIDOrderedStep\_Serial NUMBER(10), |  |
| LogIDOrderedStep\_Desc VARCHAR2(400), |  |
| MissngOrIvldCSV\_exchangedata VARCHAR2(4000), | Type 1 of the Edge Cases. Value 1 and the description are stored in the columns LogIDOrderedStep\_Serial & LogIDOrderedStep\_Desc accordingly. No value is inserted in the main log table TheMainComparisonLogOfTwoCSVs in this case |
| MissngOrIvldCSV\_depositorydata VARCHAR2(4000), |
| Dup\_User\_ID\_exchange\_data VARCHAR2(50), | Type 2 of the Edge Cases. Value 2 and the description are stored in the columns LogIDOrderedStep\_Serial & LogIDOrderedStep\_Desc accordingly. No value is inserted in the main log table TheMainComparisonLogOfTwoCSVs in this case |
| Dup\_Stock\_ID\_exchange\_data VARCHAR2(50), |
| Dup\_Stock\_Name\_exchange\_data VARCHAR2(200), |
| Dup\_Stock\_Count\_exchange\_data VARCHAR2(50), |
| Dup\_User\_ID\_depository\_data VARCHAR2(50), |
| Dup\_Stock\_ID\_depository\_data VARCHAR2(50), |
| Dup\_Stock\_Name\_depository\_data VARCHAR2(200), |
| Dup\_Stock\_Count\_dep\_data VARCHAR2(50), |
| Incst\_Stock\_ID\_exchange\_data VARCHAR2(50), | Type 3 of the Edge Cases. Value 3 and the description are stored in the columns LogIDOrderedStep\_Serial & LogIDOrderedStep\_Desc accordingly. No value is inserted in the main log table TheMainComparisonLogOfTwoCSVs in this case |
| Incst\_Stock\_Name\_exchange\_data VARCHAR2(50), |
| Incst\_Stock\_ID\_depstry\_data VARCHAR2(50), |
| Incst\_Stock\_Name\_depstry\_data VARCHAR2(50), |
| TotalRows\_exchange\_data NUMBER(30), | Type 4 of the log status. Value 4 and the description are stored in the columns LogIDOrderedStep\_Serial & LogIDOrderedStep\_Desc accordingly. The outcome of the comparison, is inserted in the main log table TheMainComparisonLogOfTwoCSVs only in this case and the summary count of the result is stored in these columns in this table |
| TotalRows\_depository\_data NUMBER(30), |
| MissingRows\_In\_exchange\_data NUMBER(30), |
| MissingRows\_In\_depository\_data NUMBER(30), |
| TotalRows\_Matched NUMBER(30), |
| TotalRows\_MtchdButDiffStkCntNm NUMBER(30) |

Next, the package, Pkg\_AnlzeTwoStkCSVs\_LogResults is required to be created (Execute the PKG\_ANLZETWOSTKCSVS\_LOGRESULTS.pks and PKG\_ANLZETWOSTKCSVS\_LOGRESULTS.pdb provided along with.

**The test cases:**

1. All Data Match Perfectly

The data in the exchange\_data.csv and the depository\_data.csv are exactly same in this case scenario.

Data folders created for test:

D:\TestCases\1\_AllDataMatchPerfectly\exchange\_data\exchange\_data.csv

D:\TestCases\1\_AllDataMatchPerfectly\depository\_data\depository\_data.csv

The execution usage to execute the package for this and all the tests are as follows:

Step1: First create the physical folders in the OS for the data and then create the database definition of the folders (execute the package procedures 1 & 2 with the folder path as input:

**exec Pkg\_AnlzeTwoStkCSVs\_LogResults.OrdredStp1\_DirFor\_Exchange('''D:\TestCases\1\_AllDataMatchPerfectly\exchange\_data\''');**

**exec Pkg\_AnlzeTwoStkCSVs\_LogResults.OrdredStp2\_DirFor\_depository('''D:\TestCases\1\_AllDataMatchPerfectly\depository\_data\''');**

Step2: Execute the package procedure 3, **exec Pkg\_AnlzeTwoStkCSVs\_LogResults.OrdredStp3\_LogTheResults;**

Check the results output delivered in the two log files:

select \* from TheMainComparisonLogOfTwoCSVs;

select \* from TheLogOfEdgeCasesAndCSVFileIssues;

Note about the two logs:

The table, TheMainComparisonLogOfTwoCSVs contains the log of the outcome of the comparison (i.e., matches, mismatches), as required in a separate table.

The table, TheLogOfEdgeCasesAndCSVFileIssues contains the the log of the outcome of the following edge cases.

1. Missing or corrupt files and format errors (e.g., invalid CSV structure).
2. duplicate data / rows

for example,

|  |
| --- |
| User\_ID,Stock\_ID,Stock\_Name,Stock\_Count  1003,STK001,Apple,40  1003,STK001,Apple,45 |

is an instance of corrupt data and can not be processed, as there cannot be two Stock\_Count figures for the same User\_ID and Stock\_ID

iii. Varying Stock\_Name for a Stock\_ID in the rows

for example,

|  |
| --- |
| User\_ID,Stock\_ID,Stock\_Name,Stock\_Count  1003,STK001,Apple,40  1003,STK001,Oracle,45 |

is instance of corrupt data

Note that the incomplete rows in the CSV (e.g., missing stock data or user details) are not accepted in the data tables because of the "not null" constraints in the table structures.

Execute the package likewise for all the following test cases detailed, and review the output. The folders and the CSV files contained inside the folders are provided along with.

1. Stock Count Differences in Two CSV Files

Data folders created for test:

D:\TestCases\2\_A-StockCountDifferencesInTwoCSVFiles\exchange\_data\exchange\_data.csv

D:\TestCases\2\_A-StockCountDifferencesInTwoCSVFiles\depository\_data\depository\_data.csv

1. Missing Rows in the exchange Data

Data folders created for test:

D:\TestCases\2\_B-MissingRows\_In\_exchange\_data\exchange\_data\exchange\_data.csv

D:\TestCases\2\_B-MissingRows\_In\_exchange\_data\depository\_data\depository\_data.csv

1. Missing Rows in the depository Data

Data folders created for test:

D:\TestCases\2-C-MissingRows\_In\_depository\_data\exchange\_data\exchange\_data.csv

D:\TestCases\2-C-MissingRows\_In\_depository\_data\depository\_data\depository\_data.csv

1. Missing Rows in Both the CSVs

Data folders created for test:

D:\TestCases\2-D-MissingRows\_In\_BothCSVs\exchange\_data\exchange\_data.csv

D:\TestCases\2-D-MissingRows\_In\_BothCSVs\depository\_data\depository\_data.csv

1. Missing Rows in Both the CSVs But Data Match

Data folders created for test:

D:\TestCases\2-E-MissingRows\_In\_BothCSVsButDataMatch\exchange\_data\exchange\_data.csv

D:\TestCases\2-E-MissingRows\_In\_BothCSVsButDataMatch\depository\_data\depository\_data.csv

1. Edge Case 1 - Missing CSV Files

Data folders created for test:

D:\TestCases\3-A\_EdgeCase1-MissingCSVfiles\exchange\_data\exchange\_data.csv

D:\TestCases\3-A\_EdgeCase1-MissingCSVfiles\depository\_data\depository\_data.csv

1. Edge Case 1 – Missing CSV folders

Any false folder can be used fir this case which does not exist

Eg.

D:\XYZ\exchange\_data\exchange\_data.csv

D:\XYZ\depository\_data\depository\_data.csv

1. Edge Case 2 – Duplicate Data or Rows

Data folders created for test:

D:\TestCases\3-C-EdgeCase2-DuplicateDataOrRows\exchange\_data\exchange\_data.csv

D:\TestCases\3-C-EdgeCase2-DuplicateDataOrRows\depository\_data\depository\_data.csv

1. Edge Case 2 – Duplicate Data or Rows in exchange Only

Data folders created for test:

D:\TestCases\3-D-EdgeCase2-DuplicateDataOrRows-InexchangeOnly\exchange\_data\exchange\_data.csv

D:\TestCases\3-D-EdgeCase2-DuplicateDataOrRows-InexchangeOnly\depository\_data\depository\_data.csv

1. Edge Case 2 – Duplicate Data or Rows in depository Only

Data folders created for test:

D:\TestCases\3-E-EdgeCase2-DuplicateDataOrRows-IndepositoryOnly\exchange\_data\exchange\_data.csv

D:\TestCases\3-E-EdgeCase2-DuplicateDataOrRows-IndepositoryOnly\depository\_data\depository\_data.csv

1. Edge Case 3 – Corrupt Data - Varying Stock\_Name for a Stock\_ID - only in Exchange

Data folders created for test:

D:\TestCases\3-F-EdgeCase3-CorruptDataVaryingStock\_NameForaStock\_ID-OnlyInExcgange\exchange\_data\exchange\_data.csv

D:\TestCases\3-F-EdgeCase3-CorruptDataVaryingStock\_NameForaStock\_ID-OnlyInExcgange\depository\_data\depository\_data.csv

1. Edge Case 3 – Corrupt Data Varying Stock\_Name for a Stock\_ID – only in Depositiry

Data folders created for test:

D:\TestCases\3-G-EdgeCase3-CorruptDataVaryingStock\_NameForaStock\_ID-OnlyInDepositiry\exchange\_data\exchange\_data.csv

D:\TestCases\3-G-EdgeCase3-CorruptDataVaryingStock\_NameForaStock\_ID-OnlyInDepositiry\depository\_data\depository\_data.csv

1. Corrupt Unreadable Data Files - in exchange Only

Data folders created for test:

D:\TestCases\3-H\_CorruptUnreadableDataFiles-InexchangeOnly\exchange\_data\exchange\_data.csv

D:\TestCases\3-H\_CorruptUnreadableDataFiles-InexchangeOnly\depository\_data\depository\_data.csv

1. Corrupt Unreadable Data Files - in depository Only

Data folders created for test:

D:\TestCases\3-I\_CorruptUnreadableDataFiles-IndepositoryOnly\exchange\_data\exchange\_data.csv

D:\TestCases\3-I\_CorruptUnreadableDataFiles-IndepositoryOnly\depository\_data\depository\_data.csv