CA Test Data Manager (TDM) Hand Book

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Version | Created by | Reviewed/ Approved by | Modifications |
| 27.07.2017 | 1.0 | Sushanta Kumar |  | Initial Draft |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**TABLE OF CONTENTS**

[CA Test Data Manager (TDM) Hand Book 1](#_Toc490249131)

[1. CA TEST DATA MANAGER 3](#_Toc490249132)

[1.1 Overview 3](#_Toc490249133)

[1.2 Key Components 3](#_Toc490249134)

[1.3 CA TDM Portal 4](#_Toc490249135)

[1.4 DIFFERENT TOOL USED FOR TEST DATA MANAGEMNET 4](#_Toc490249136)

[1.4.1 TDM Datamaker 4](#_Toc490249137)

[1.4.2 TDM Masking tool 4](#_Toc490249138)

[1.4.3 TDM Subset tool 5](#_Toc490249139)

[1.4.4 TDM Test data generation tool 5](#_Toc490249140)

[1.5 Test data manager Project tree 5](#_Toc490249141)

[2. Installation and configuration 11](#_Toc490249142)

[3. Provisioning TEST DATA manager 12](#_Toc490249143)

[1.1 Discover your data 12](#_Toc490249144)

[1.2 Sample and profile your data 12](#_Toc490249145)

[1.3 Subset production data for testing 17](#_Toc490249146)

[1.4 Mask production or data subset for testing 20](#_Toc490249147)

[1.5 Generate Synthetic Test Data to fill test coverage gap. 23](#_Toc490249148)

[4. DATA transformation 27](#_Toc490249149)

[5. CA TEST DATA MATCHING 28](#_Toc490249150)

[5. DATA GENERATION (20170) 36](#_Toc490249151)

# 1. CA TEST DATA MANAGER

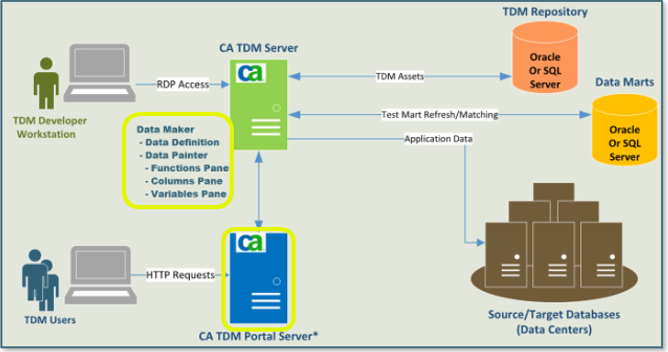
## Overview

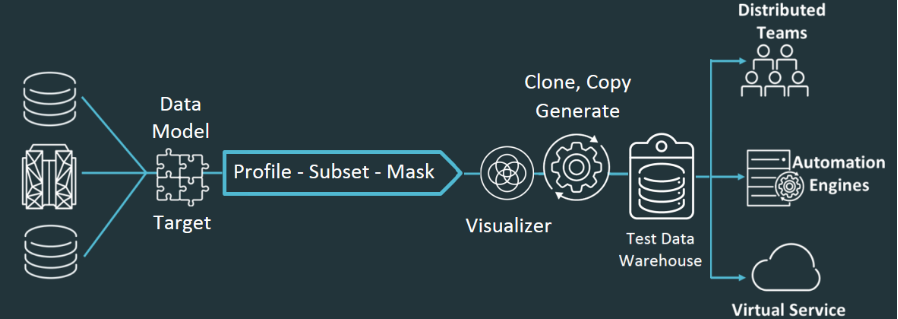
The CA TDM is one of the test data generation tool and the product owned by CA. The CA TDM provides a broad solution for test data management needs.

Benefits of CA TDM tool (**Right Data, in the Right Place, at the Right Time)**

* + Reduces time-to-market with a complete end-to-end test data management solution.
  + CA TDM uniquely combines elements of data sub setting, masking and synthetic, on-demand data generation to enable testing teams to meet the agile needs of the organization.
  + TDM is is used to Build test data warehouse from production environment.
  + TDM takes data from database, creates repository, masks, profiles, provide data from ‘Data management’ data base

**CA TDM Architecture**





## Key Components

The following key components provide the primary CA Test Data Manager capabilities.

* + **Repository**: The Repository is a database that stores product meta data, registered data, projects, and more
  + **Datamaker:** Datamaker provides a core project and data management interface and core data registration and generation capabilities. Data maker also links to several othercomponents that provide profiling, sub setting, and other capabilities.
  + **Fast Data Masker:** Provide core data masking capabilities. From Fast Data Masker, you can connect to a data source, select columns to mask, define masking rules, and mask the selected data to ensure data compliance.
  + **Data Subset:** Provide an interface for identifying criteria for taking meaningful subsets of a larger data source for testing.
  + **Data Profiler:** Data Profiler is an interface within Datamaker that lets you sample your data and better understand its characteristics**.**

## CA TDM Portal

CA TDM portal provides a modern web interface for key capabilities such as project and data management, data registration, data generation, and data reservation. The CA TDM Portal acquires new functionality each release.

* + **Administration**
  + **Data Modelling**
  + **Data generation**

Request Data from the CA TDM Portal: Login to TDM portal Administration

Search required Project -> go to “Self Service Catalog” -> New Request -> Submit 

## DIFFERENT TOOL USED FOR TEST DATA MANAGEMNET

### TDM Datamaker

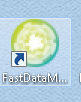
Datamaker tool used for test data management

***Important Navigations***

|  |  |  |
| --- | --- | --- |
|  |  | SQL window on data target |
|  | SQL window on data Source |
|  | Maintain Projects |
|  | Data Definition |
|  | Publish data |

### TDM Masking tool

When data is copied from production environment, it may contain sensitive content such as Personally Identifiable Information (PII). A breach of personally identifiable records can harm customers and businesses alike. Hence the data need to filter as a secure test data.



### TDM Subset tool

The Subset tool can then be used to:

* + Tag data objects and view possible data relationships
  + Model the relationships around a driving table
  + Pull data attributes from all profiled and sampled database systems to subset the driving table and all connected data relationships.

|  |  |  |
| --- | --- | --- |
|  |  |  |

### TDM Test data generation tool

Data generation is systematically creating the data so that it is valid and realistic

The goal is rapidly provision fit for purpose data in demand.

The below steps involves in data generation process

1. analyze existing data. discover where data exists. it formats attribute and quality.

the profiling part is called data discovery.

2. once data understood, mask the sensitive or PII data. this process is called data

transformation

3. once step 1nad 2 complete generate data to fill identified gap.

## Test data manager Project tree

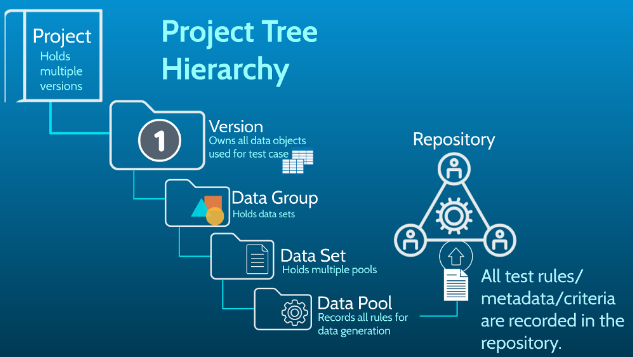
The highest (least granular) level of organization in CA TDM is the Project.

A Projects created for each test data model maintained using CA TDM.

Within each project, users create at least one version.

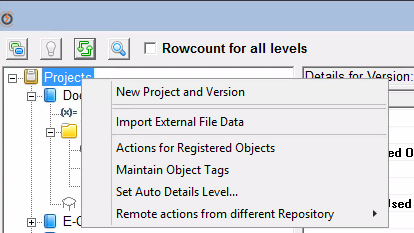
* + The Version Name is the name of the current release of the database or application.
  + Data Groups and Data Sets help you group your data categories.
  + The Data Pool contains all commands and the defined architecture for generating test data.
  + Test data can be published (generated) from groups, sets, or pools.
  + All the metadata from the version, the subfolders, and the data pool are stored in the repository.

Project->Version->Data Group -> Data Set -> Data Pool -> Repository

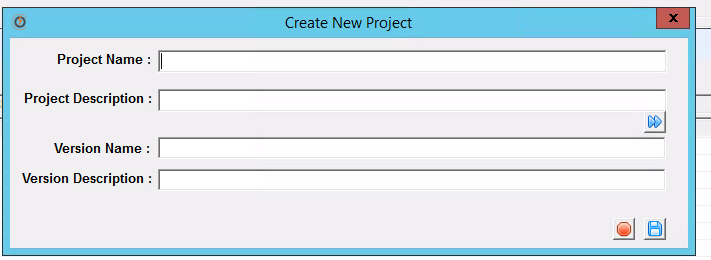


**Steps to create New Project:**

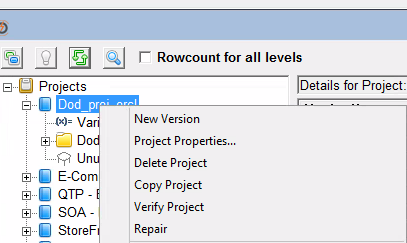
1. To create a new project, right click the Projects icon, and click New Project and Version

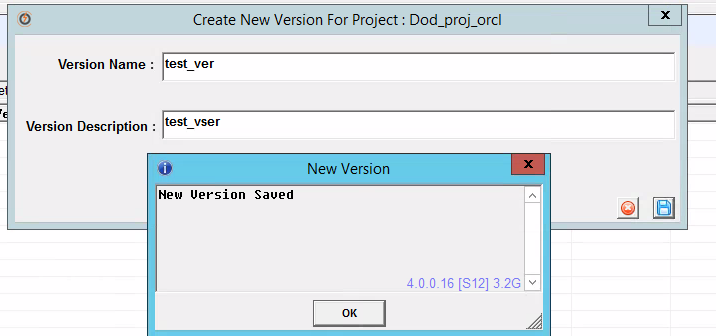


Enter the Project details;

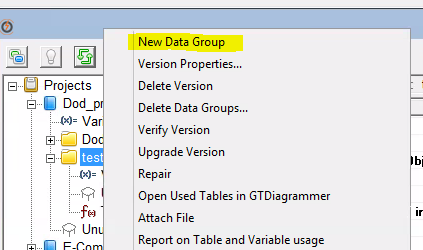


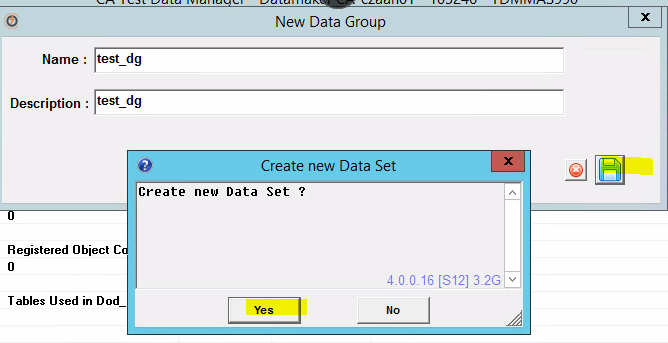
Right click on project and select Ne version



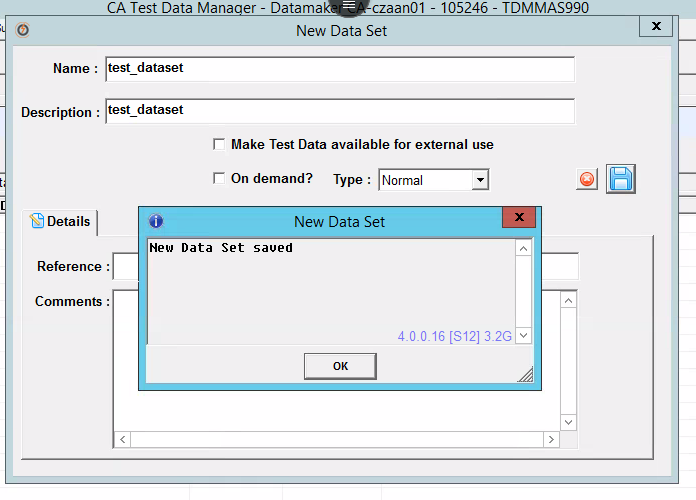


1. To create a new data group and data set, right click on new version and select new data group.

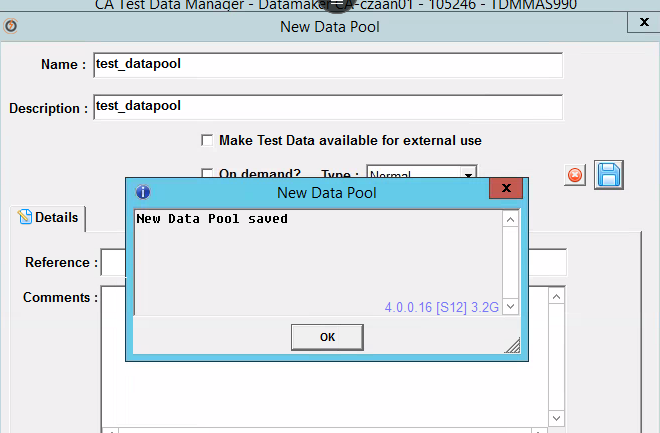




Data set created.

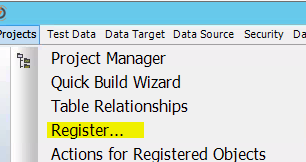


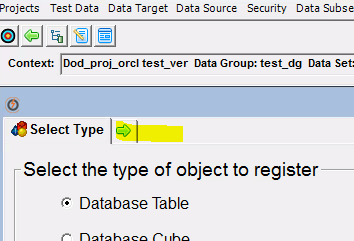
1. To create a new data pool, right click on dataset and select new data pool

.

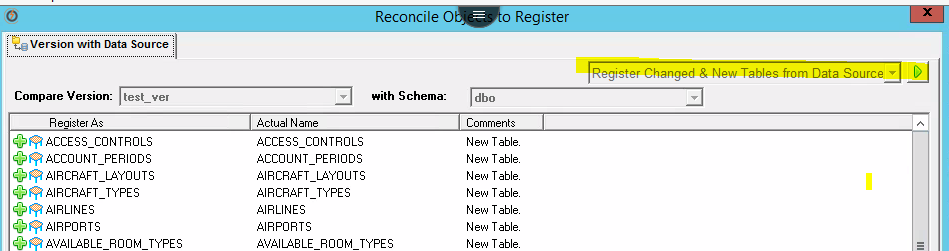
1. To register a data in data pool, double click on datapool or select datapool and go to

Projects -> Register and Opt the database table and click green arrow.

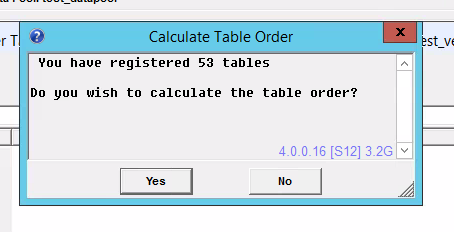




1. Select Register Tables from Data Sources -> All objects -> click Go( green button)



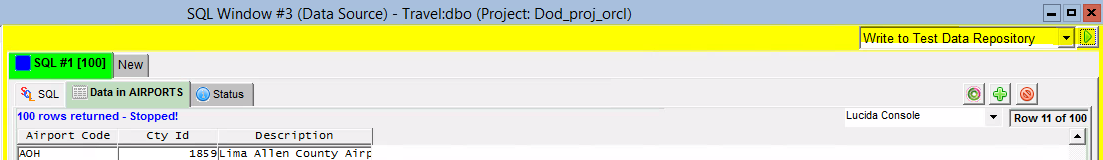
1. Tables registered

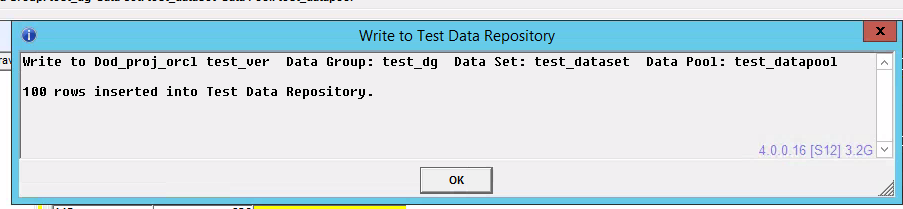


1. Publish data into data pool

Got to maintain project -> data pool -> double click on data pool -> then click source

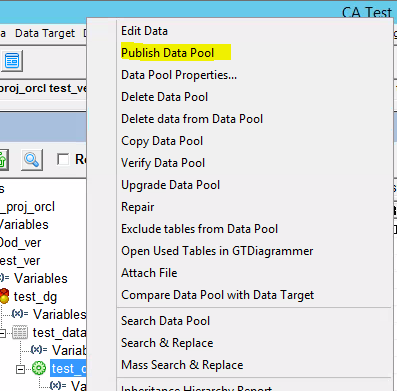
button and Select the required query and then click Go



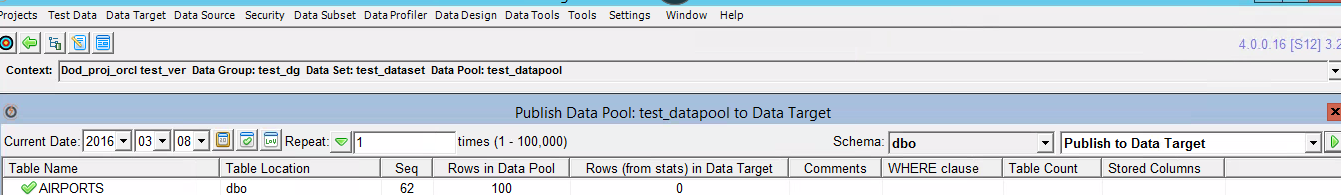


1. Publish data into target table

Right click on data pool and select *Publish data pool*



Publish to Target Table



1. Verify data in target table

# 2. Installation and configuration

The following processes need to be follow to complete the CA Test Data Manager Installation.

* + Review and meet system requirements
  + Install the repository
  + Install product components
  + Connect to repository
  + Obtain and activate product licences
  + Connect to other data sources
  + Install the CA TDM portal

Ref: <https://docops.ca.com/ca-test-data-manager/3-5/en/installing/install-ca-test-data-manager/system-requirements>

To be Draft..

# 3. Provisioning TEST DATA manager

## Discover your data

CA TDM has built-in profiling features to assist with understanding data by extracting from different wide range of disparate source.

There are three actions focused on discovering of data.

1. **Discovering Metadata** – Registration of metadata to a test data repository

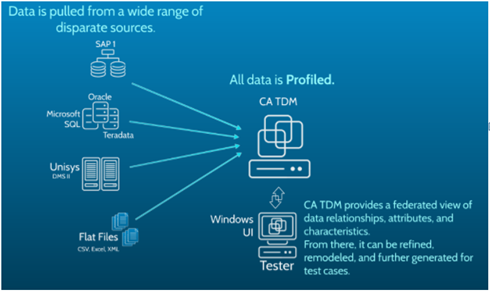
Metadata is data about data. Metadata includes table attributes, keys, number of records, links to other tables, and the like. Profiling is not simply copying a database and then scrubbing data. CA TDM examines metadata from chosen data sources like databases, flat files, mainframe files, and other file formats. The attributes are then extracted to the repository.

1. **Refining Data Relationship** – Importing, scanning and comparing data

CA TDM displays all data sources in a relational model. This model allows you to analyse, manipulate, and optimize data for testing by: Adding or deleting relationships for efficient data models, Setting cardinality and Adjusting rules and conditions for testing criteria.

1. **Discovering Content**  – Sample, profile, and analyze your data

CA TDM allows you to analyze sample data to help generate data for testing. For example, profiling helps you identify personally identifiable information (PII) that requires masking.



## Sample and profile your data

CA TDM extracts data and sampling through the Data profiler.

Benefits:

(i) To get better understanding of data structures through improvements of existing model.   
 Reduce the time taken to understand the model.  
(ii) To trace through complex data models and structures to discover all independent entities.

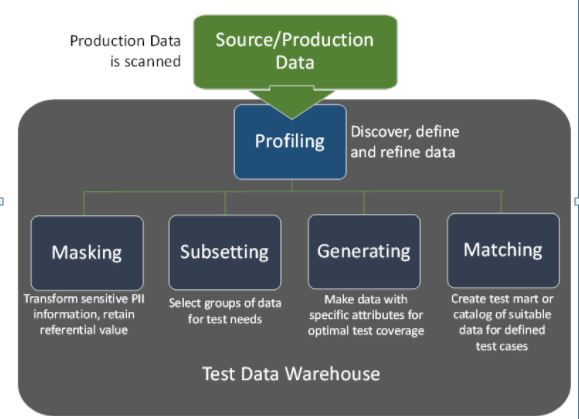
(iii) To Improve the design of data models to ensure they meet all requirements.

(iv) contains data sampling and analysis tools to accurately identify Personally Identifiable Data

Example: **–** banking Information, Medical History, Govt. issued document, Legal Records etc.

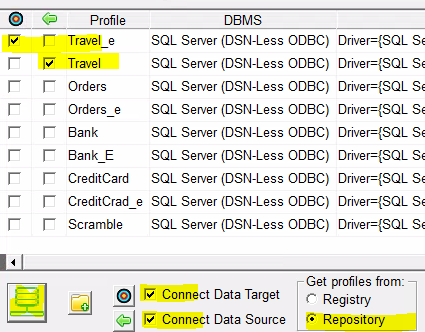
Step to Process PII data

* Move methodically through all data
* Tag specific elements of PII
* Use filtering categories to locate and then transform



**Exercise -1 (**Register Relational Data**)**

I) At start up, ensure all connection Profile settings are in place like 1) Source, 2) Target, 3) Repository and 4) Connection Checkboxes



II) Create Project and then create a version.

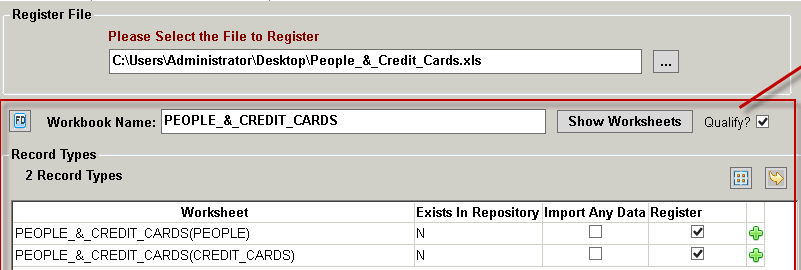
Right click on version -> Register -> Select database table -> Click register type -> select “Register Tables

from Data Source” -> All Objects -> Click GO -> All objects will registered.

**Exercise -2 (**Register Flat Files**)**

I) Select type of object Excel/CSV file

II) Select the file and check the worksheet (click Qualify)



III) Select the file and check the worksheet (click Qualify and import any data) -> click Register/Import

IV) Right click on the version and Select Action for registered tables and verify the data.

**Exercise -3 (**Register an XML Files**)**

I) Login to CA Portal

II) Select the appropriate project and version

III) Click Modelling -> Objects -> Select Object type XML -> Click import object data

IV) click CREATE and REGISTER tables

v) Submitted Request.

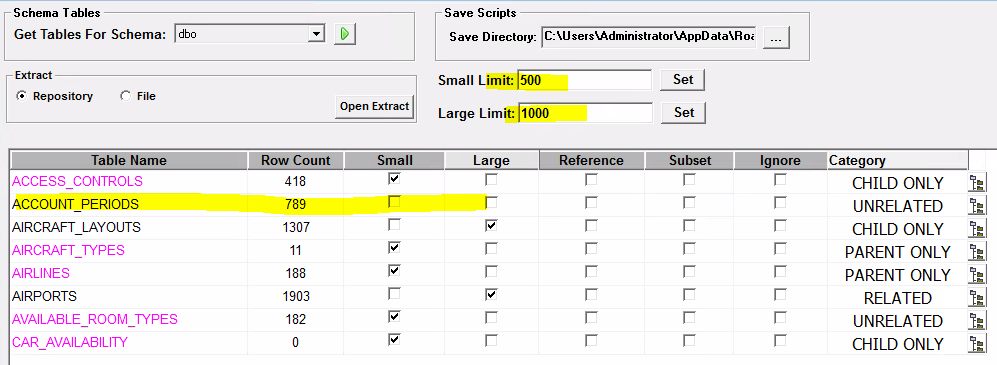
**Exercise -4 (**Manage Object Tag**)**

I) Login to CA DATA Maker

II) Register all Source objects

III) Click Data subset menu -> Verify and prepare subset schema

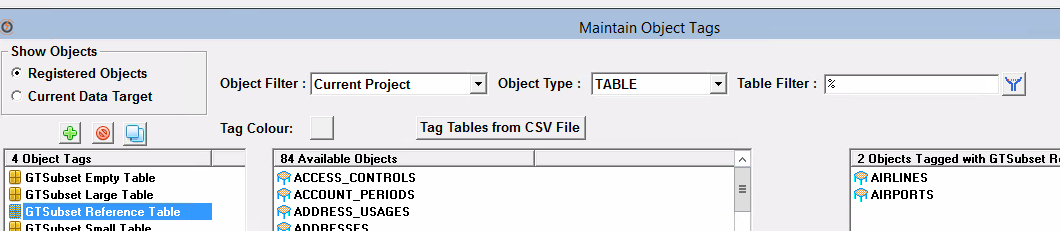
IV) Set small limit = 500 and Large limit = 1000



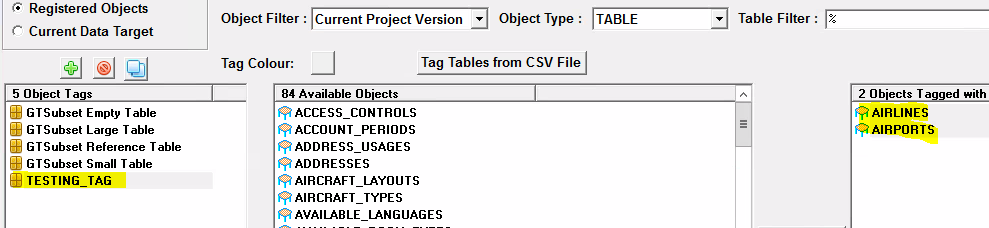
Note: ACCOUNT\_PERIODS not checked as count 789 > 500 and less than 1000

V) Click Update Tags – Tag Updated

VI) To verify, Right click on version -> Select maintain object Tags -> Select subset reference Table



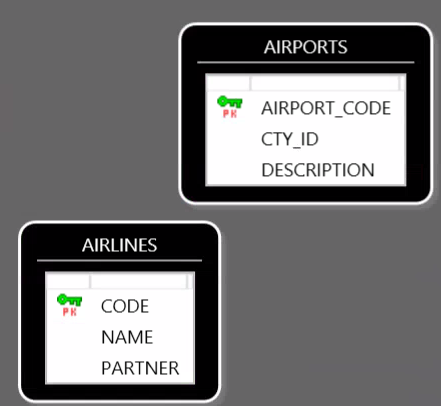
VII) Create a new Tag in current version project and select required objects and move to newly created TAG



VII) Click SAVE.

IX) Right click on version -> Select Actions on selected Objects -> Select the TAG “TESTING TAG” ->

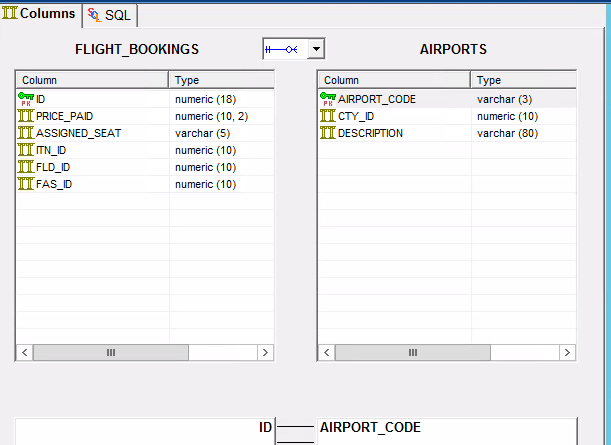
Select Open tables in GTDigrammer -> Click GO



**Exercise -5 (**Build and modify Relationship**)**

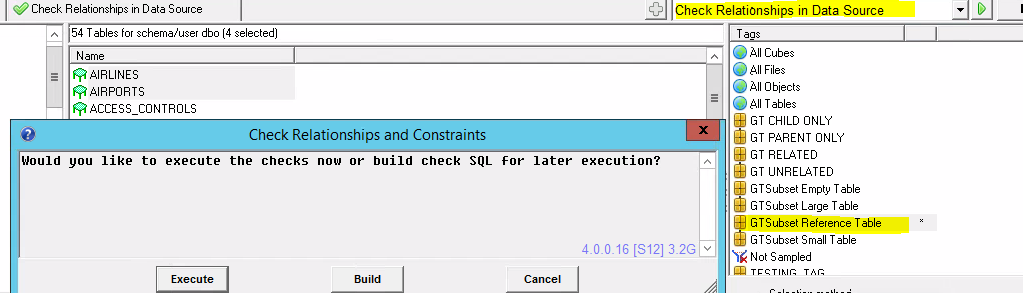
Go to Projects -> Table Relationship -> Select one of the registered table -> Add Relationship (right click) ->

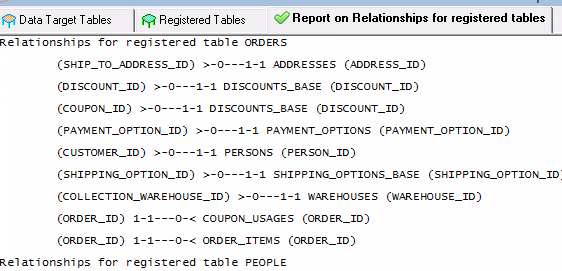
Create relationship from selected tables by selecting respective columns -> Put the description and click OK button



**Exercise -6 (**Validate and Publish Data**)**

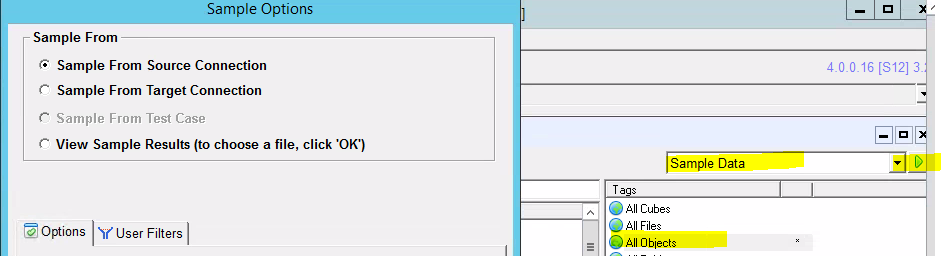
Go to Projects -> Table Relationship -> Select one of the registered Tag(GT Subset Reference Table) -> Select “Check Relationship in data Source” -> GO -> e relationship from selected tables by selecting respective columns -> Save the Report

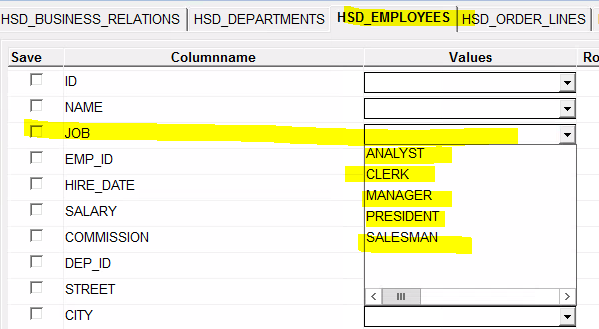




**Exercise -6 (**Data Sampling and verifying Data**)**

Register data in Project version -> Data Profiler -> Sample Table data -> Select *All Objects* -> Type “Sample Data” -> GO

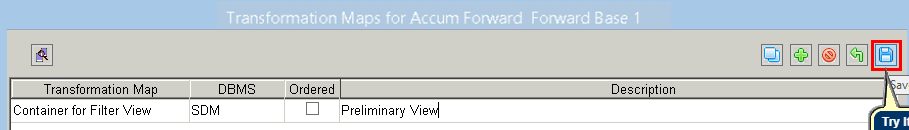




**Exercise -7 (**Filtering PII Data**)**

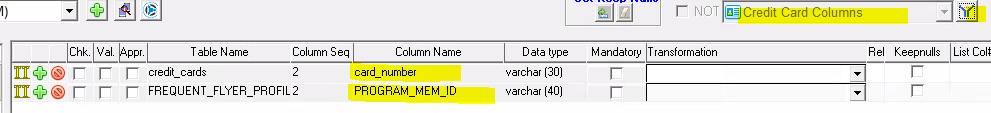
Projects -> Transformation map -> Click New Transformation Map -> Insert Row ->

Edit transformation map, select DBMS to SDM and description as preliminary view -> Save -> Return to selected Map.



Select Credit Card Number column (PII data) and click Filter as Example in Transformation map

Note: Program\_MEM\_ID is similar pattern like Credit card column definition.



## Subset production data for testing

There are many benefits on sub setting production data

* Reduces infrastructure costs on storage and maintenance of full copies of production
* Allows greater control over your data with more manageable sets of tables
* Allows to select exactly the set of data which need for testing
* Allows to save time creating subsets through an automated process
* Allows to improve the quality of testing by extracting covered and federated subsets
* Enables to work with the optimal data needed for your test case

**Database sub setting**

**(i) Preparing Subset Schema**

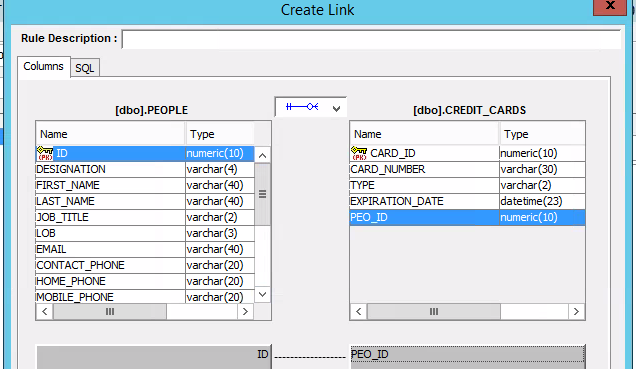
Data Maker -> Data Subset -> Design Extract and Transactions –> Select (Built SQL Insert Script)

Select correct Schema – dbo

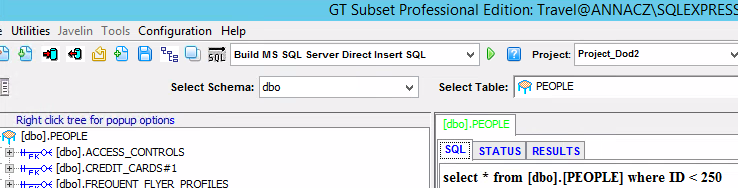
Select table PEOPLE.

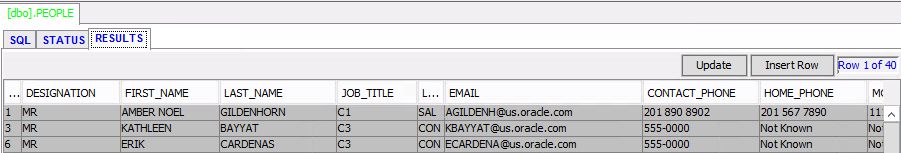
–> Right click on table PEOPLE and select Add link -> select link PEOPLE.ID and

CREDIT\_CARDS.PEO\_ID -> Press OK.

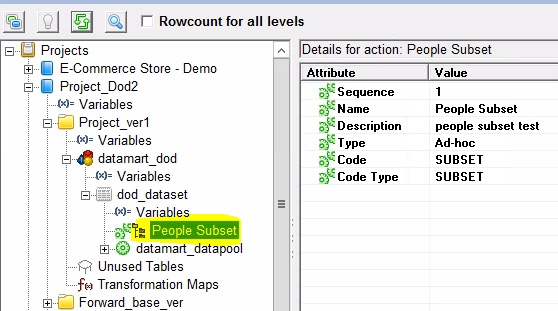


Execute the SQL Query with condition ID < 250 -> Click Preview Row count -> close ->





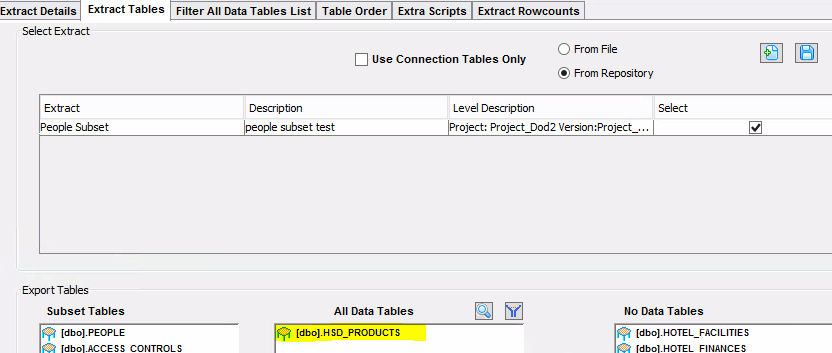
File -> Save Extract to Repository -> provide extract -> opt save as extract -> save.



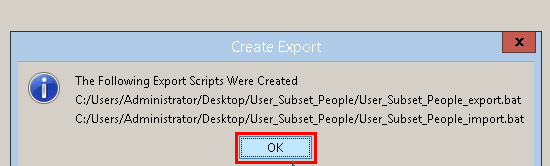
Subset data created.

**(ii) Generate and save scripts**

Data Maker -> Data Subset -> Design Extract and Transactions –> Select (Built MS SQL Server Export/Import) -> Select a table to *All data table* -> Extract Script ->



Script will generate in the below path



**(iv) Verify the target table**

Run the below script in the command window to ensure target table has no data.

**sqlplus travel\_e/travel\_e@XE @actionname\_deletetables.sql**



**(v) Run the Export/Import**

Run this command in command window

**Export:** User\_Subset\_People\_export.bat travel/travel/XE

**Import:** User\_Subset\_People\_import.bat travel/travel/XE travel\_e

**(vi) Verify the physical data in Target table**

**Exercise:** Create smaller, targeted databases from large, complex databases

* **Scenario:**

Test Data Engineer’s assignment is to provide data to test a Employee Management application. The data for the Employee Management application is spread across various tables in Employee database in the Microsoft SQL Server. The number of tables in the database and the amount of the data is so huge to manage this assignment. **So need to create a subset of the data.**

Steps followed:

* Establish Database Connection
* Create Extract Definition from Data Source
* Generate Extract and Import Scripts
* Run the Export Script to Extract the Data from Data Source
* Run the Import Script to Import the Data into Data Target

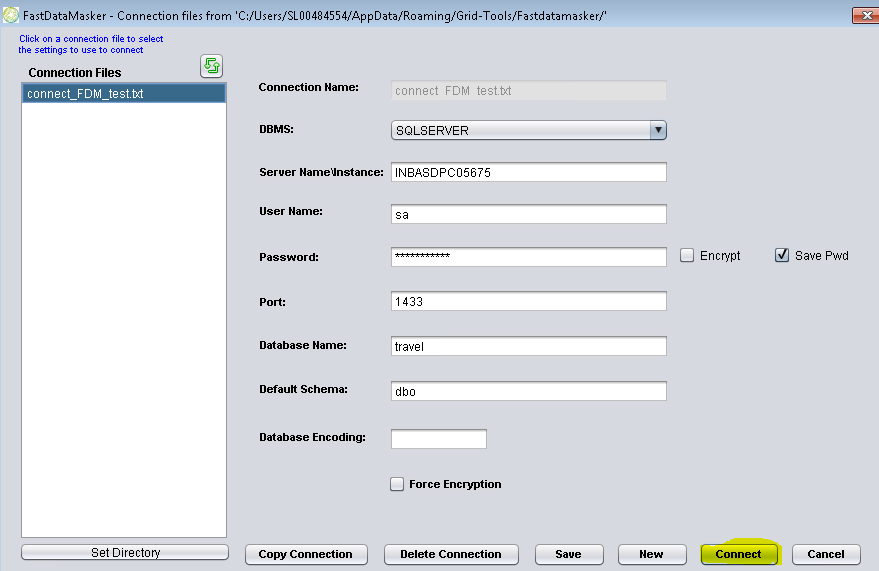
## Mask production or data subset for testing

Masking engines secure millions of rows of data in minutes using automated data profiling and high performance. Masking of confidential data will be done here. FASTDATA Masker App is used for Masking.

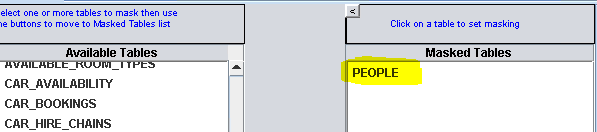
(I) Launch FastDataMasker

(II) Click New button to create new connection by providing below details or select already

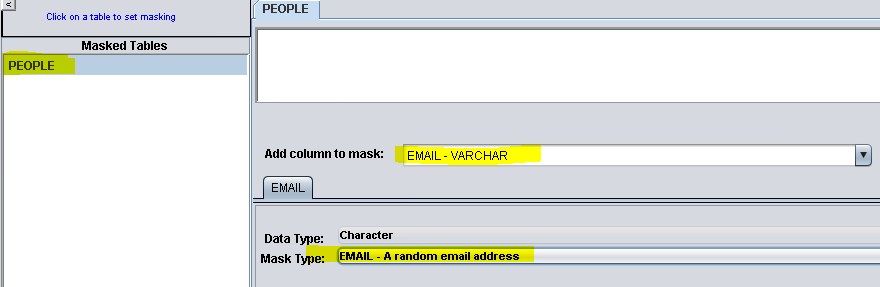
existing connection to login. Uncheck Encrypt check box

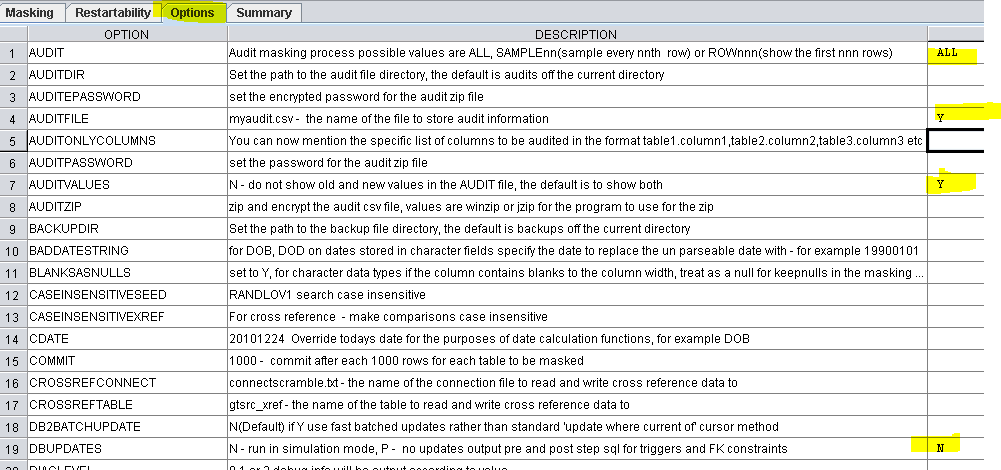


(III) Select a Table from Available Tables list and add to Masked Tables list

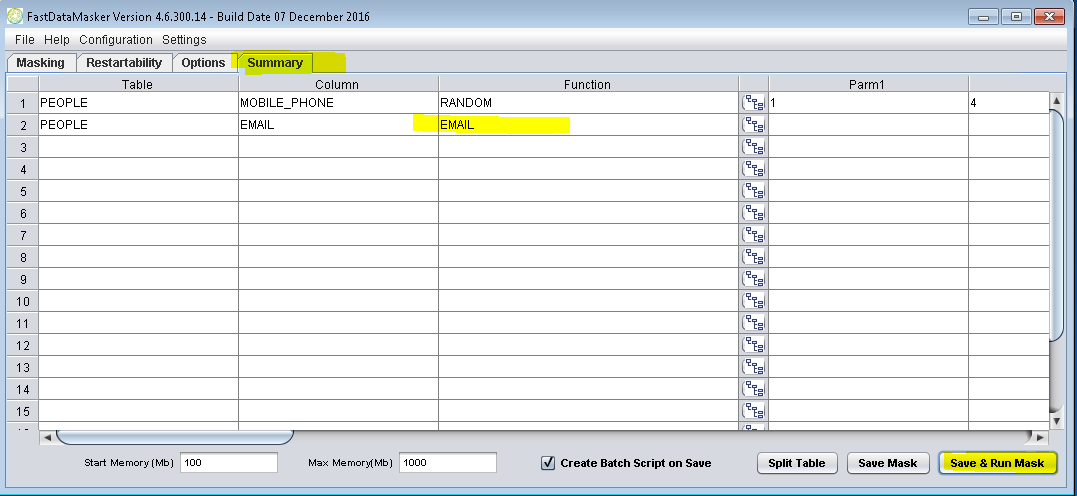


(IV) Select a column from Add column to mask drop down and mask type

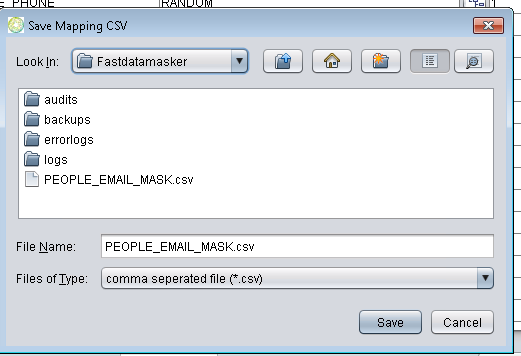


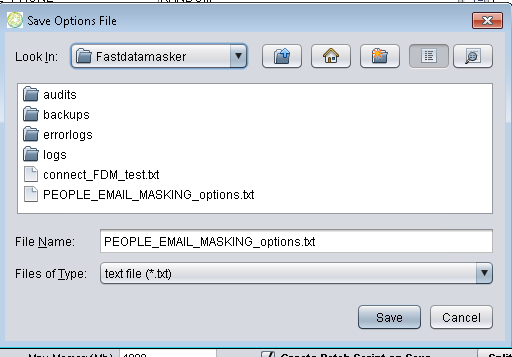
(V) Navigate to options tab and add values in the below respective rows 

(VI) Navigate to Summary Tab and click on Save & Run Mask button.

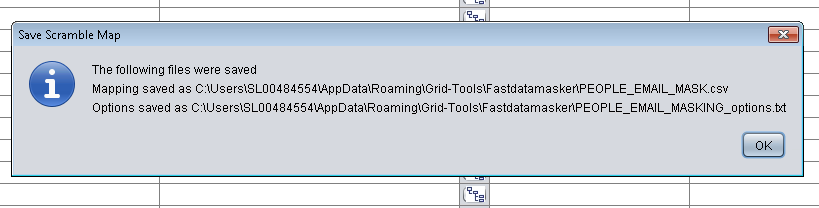


(VII) Specify the file name and Save the file

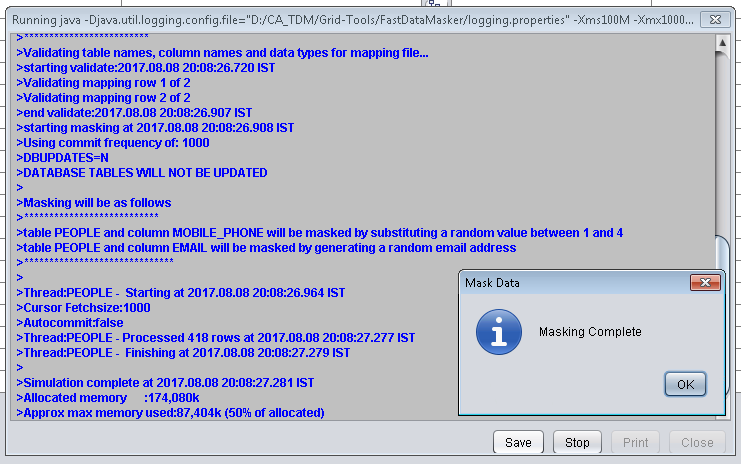




(IX) Masked file saved in selected file location



(IX) Masking Completed



(XI) Masking Completed

To view the masked file, navigate to following path:

C:\Users\SL00484554\AppData\Roaming\Grid-Tools\Fastdatamasker\

**Exercise:**  Data masking is nothing but obscuring specific records within the database

* **Scenario:** Test Data Engineer’s assignment is to mask the first name and email ID of an employee in Employee Table in Employee Database by using Fast Data Masker tool

Steps followed:

* Select a table to mask
* Select a column to mask for the selected table
* Select the mask type depending on the categories
* Specify the option to resume the process at the fail point if the masking job fails
* Set options for the masking process
* Review the masking definitions, save the mask, or save and run the mask

## Generate Synthetic Test Data to fill test coverage gap.

Creates rich synthetic data which covers 100% of possible tests. It creates large

volumes of data for performance and load tests. Also It Improve the efficiency of

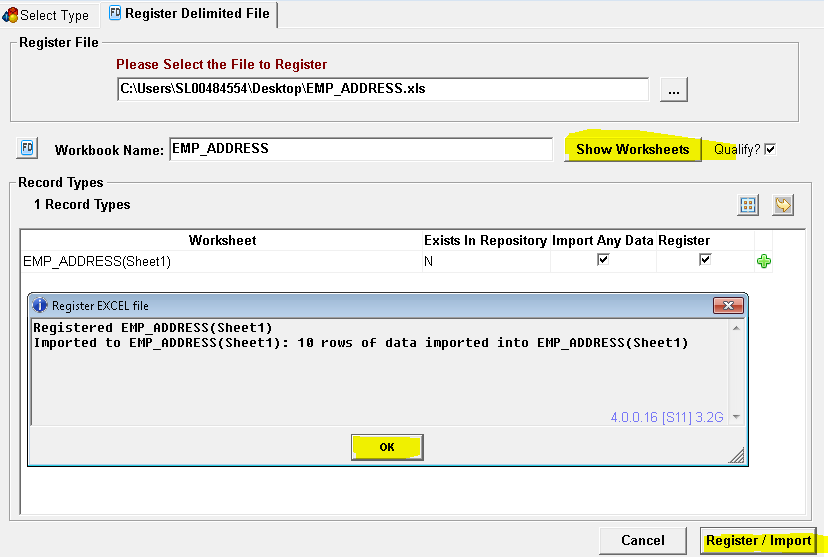
your testing.

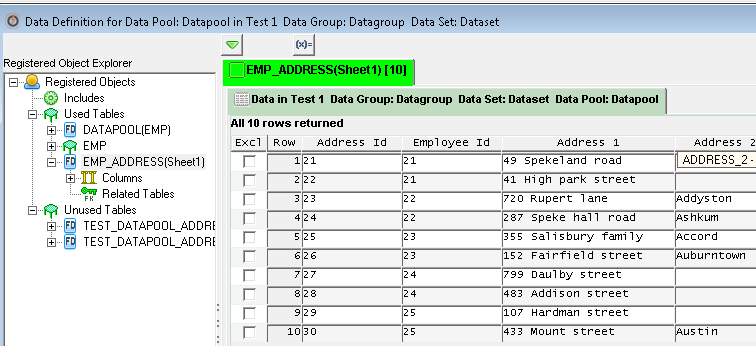
(I) Open Data Maker Application

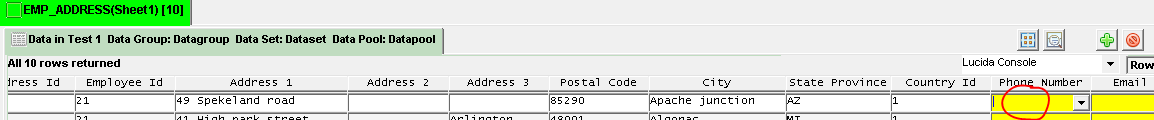
(II) Select the project, click on version and create new data pool. After creating data pool go to

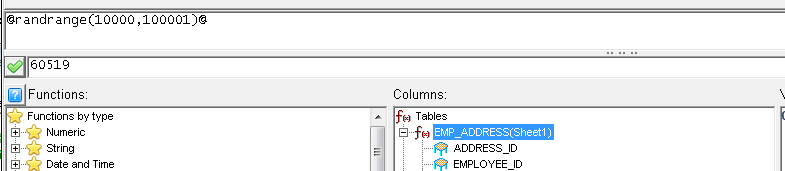
projects click on Register & select Excel/CSV file

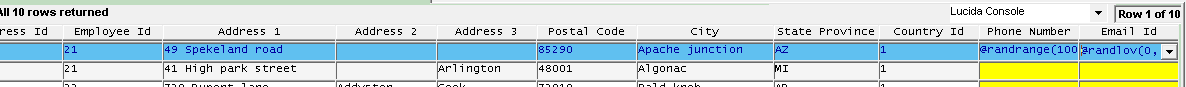
(III) Browse the excel file from the file location, check all the checkboxes then click on Register / Import button

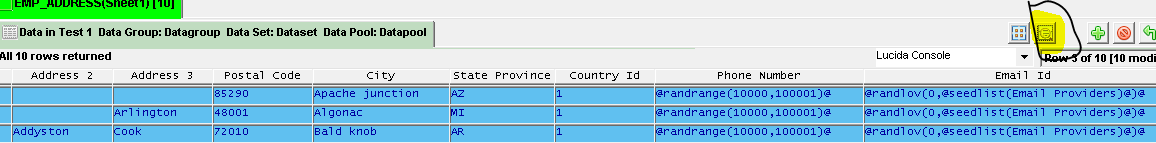


(IV) Select Projects > Actions for Registered Objects & then click on go to data pool and double click on excel file 

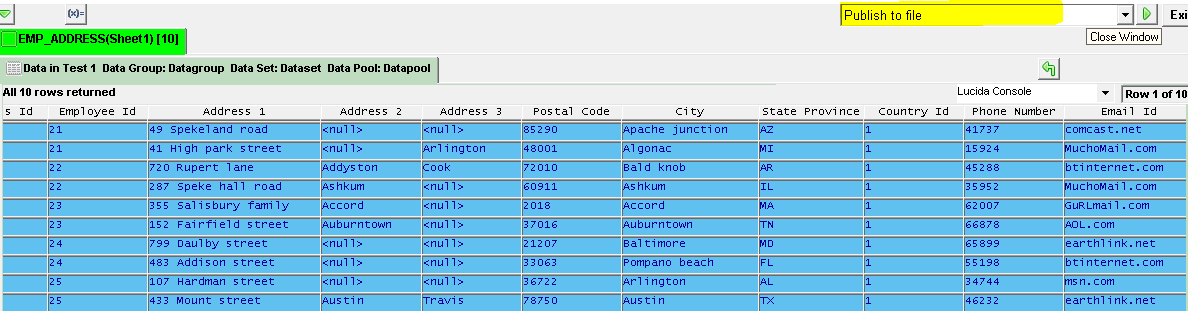
(V) Double click on field phone number and then email. select @randrange(min,max) for phone\_number and @randlov(0,@seedlist(Email Providers)@)@ for email and then test the rule and SAVE. Right click and copy column down -> copy down all. 



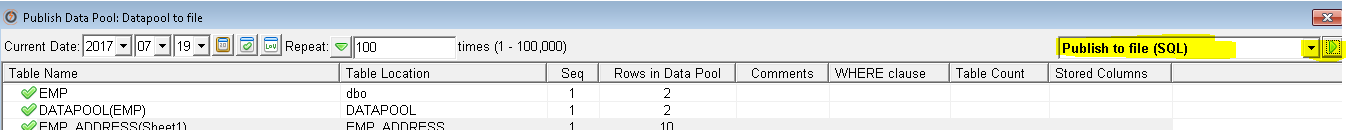


(VI) After validating the function, click on Preview icon from window 

(VII) Exporting data: Select *Publish to file* form the drop down list



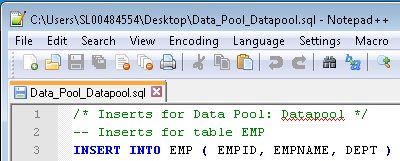
(VIII) If required to generate more than 1 record enter the value in Repeat and select Publish to file from dropdown list and run – to generate SQL file



(IX) Select appropriately value and then Choose the folder and run

|  |  |
| --- | --- |
|  |  |

(X) File (Data\_Pool\_Datapool.sql) created



(XI) To generate the synthetic data in EXCEL file.

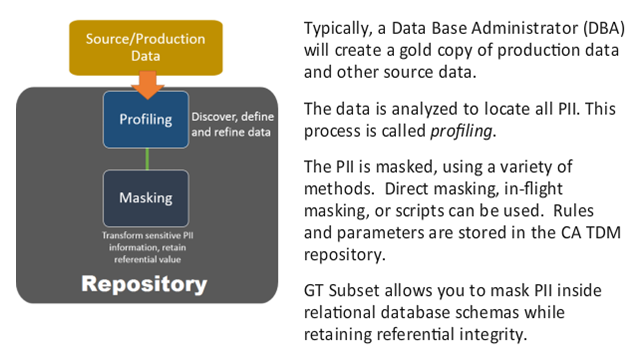
(XI) Published data

**Exercise**

* **Definition**: Synthetic test data is data that contains all the characteristics of production, but with none of the sensitive content. CA TDM uses data profiling techniques to take an accurate picture of your data model.
* **Scenario**: Test Data Engineer received an incomplete data for testing, so he need to populate the data for incomplete columns by using generation rules.
* Register the data
* Generate Rules
* Publish the data

# 4. DATA transformation

Data Transformation is the process of converting data from one format to another, usually from the format of source system (production database) into required format of a new destination system(target/lower) database. And during transformation data can be transforming into data scrubbing, data masking, obfuscation, cleaning or anonymization



**In-Place masking** – FDM connects directly to DB and masks all data based on masking rules.

**In-Flight masking** – Masking rules created in CA TDM and can be applied to a subset routine or design extract. The subset routine creates the export/import scripts and dump file which contains masking rules.

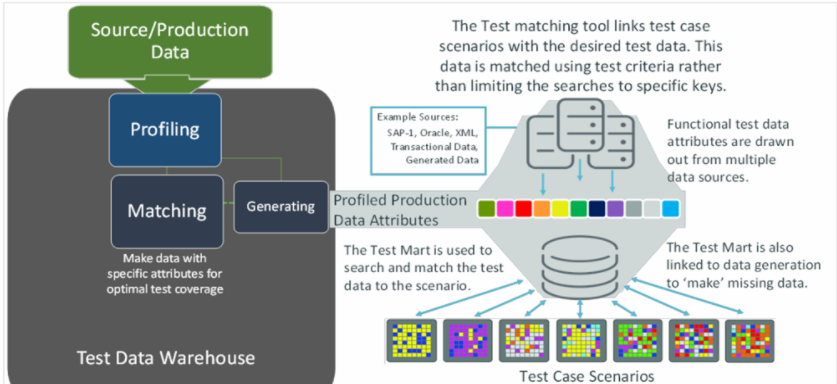
Whenever production data is copied, these rules can be applied so data is masked in flight.

Example of masking function: HASHLOV, EMAIL, CONCAT, RANDLOV

# 5. CA TEST DATA MATCHING

The test matching tool links test case scenarios with the desired test data. This data is

Matched using test criteria rather than limiting the searches to specific keys.



**Process Flow to Prepare Test Match**

* Create a View to map the required data. A view can point to multiple sources across your profiled data.
* A "flat" table is created as a Test Mart. The Test Mart pulls the data selected by the view. Like the view, it also contains no real data. The mart uses CA TDM's sequential list of values function (SEQLOV) to pull the data through the view for analysis.
* The Test Mart table is registered to a version, but remains unrelated to other data objects. This provides version control and data inheritance.
* A data Pool is created for the Test Mart. This allows the mart to pull and provide data from any source, based on attributes rather than specific keys.
* The view is linked to the Test Mart. Both are built to mirror the other. A "model row' is built in the Data Mart with the SEQLOV function, automatically connecting it to all intended sources.
* Finally, another Data Pool is created, using a Test Match setting. This populates columns inside the pool with test data criteria used for Test Match settings and reporting.

**Example**:

**Step1- Create a view in source database (travel)**

create or replace VIEW web\_user

(id,first\_name, last\_name, email,card\_number,credit\_card\_type, web\_user\_name)

as select p.id,p.first\_name, p.last\_name, p.email,c.card\_number credit\_card\_number,c.type credit\_card\_type,

access\_controls\_web\_user\_name

from people p, credit\_cards c, access\_controls a

where c.peo\_id = p.id and a.peo\_id = p.id;

**Step2- Build a test data mart**

(i) Connect to TARGET database and execute the below Query to create a data mart

CREATE TABLE TEST\_MART\_WEB\_USER (

ID NUMERIC(10,0),

FIRST\_NAME VARCHAR(40),

LAST\_NAME VARCHAR(40),

EMAIL VARCHAR(40),

CARD\_NUMBER VARCHAR(30),

TYPE VARCHAR(2),

WEB\_USER\_NAME VARCHAR(10),

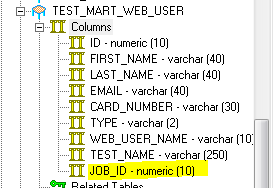
TEST\_NAME VARCHAR(250));

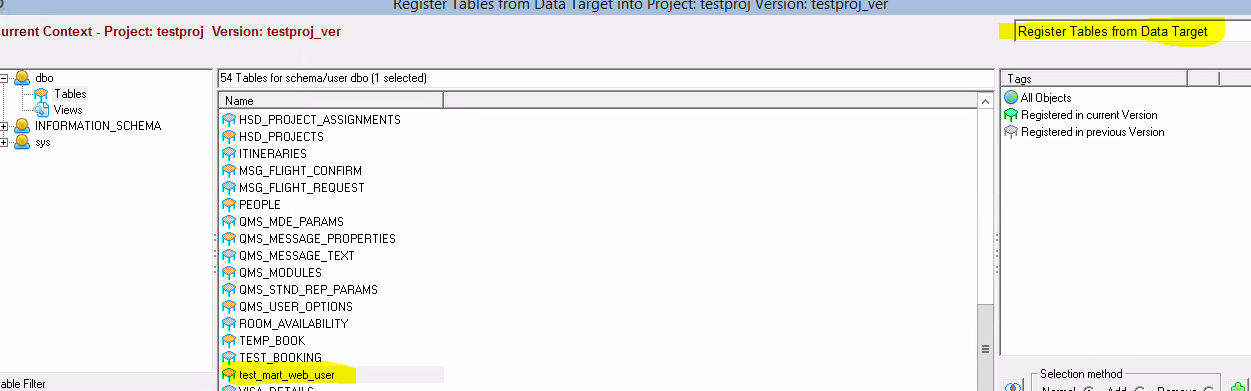
(ii) Once table created add a new column by right clicking on table -> Alter table -> add column

Column\_name – JOB\_ID

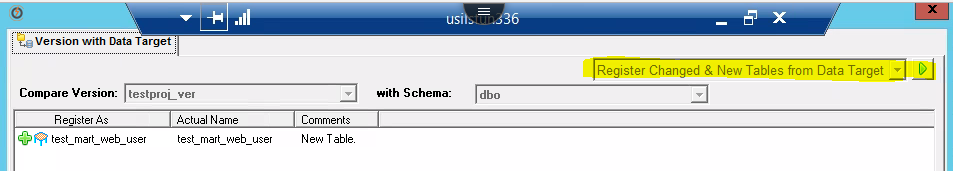
Data type – NUMBER

Width – 10 and then Execute and register tables from Data Target





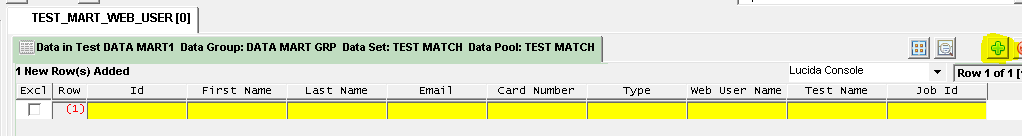
(iii) Next select Register Changed & New Tables from Data Target -> GO

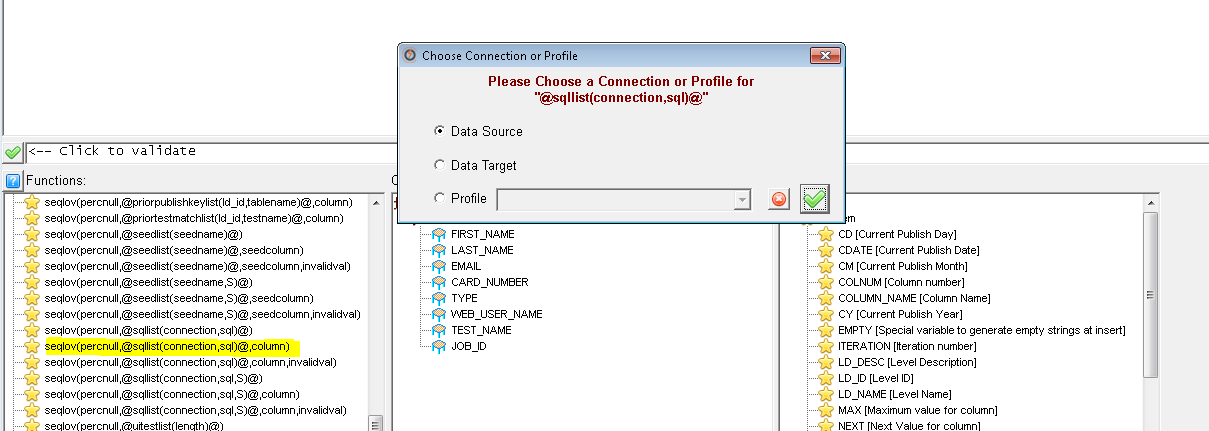


**Step3- Link Test Mart to View**

(iv) Create test data group -> data set -> data pool -> right clicked (edit data) ->

Select Table **TEST\_MART\_WEB\_USER** -> (+) Insert one new rows -> double click on first column





Set the rule for all columns one by one and finally **SAVE**

ID -> @seqlov(0,@sqllist(S,select \* from travel.web\_user)@,ID)@

FIRST\_NAME-> @seqlov(0,@sqllist(S,select \* from web\_user)@,LAST\_NAME)@

LAST\_NAME -> @seqlov(0,@sqllist(S,select \* from web\_user)@,LAST\_NAME)@

EMAIL -> @seqlov(0,@sqllist(S,select \* from web\_user)@,EMAIL)@

CARD -> @seqlov(0,@sqllist(S,select \* from web\_user)@,CARD)@

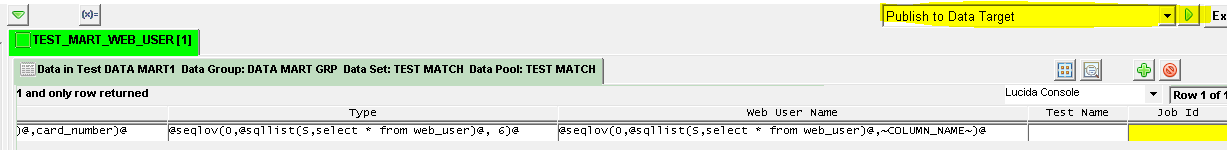
EMAIL -> @seqlov(0,@sqllist(S,select \* from web\_user)@,LAST\_NAME)@

CARD\_NUMBER -> @seqlov(0,@sqllist(S,select \* from web\_user)@, CARD\_NUMBER)@

TYPE -> @seqlov(0,@sqllist(S,select \* from web\_user)@, 6)@

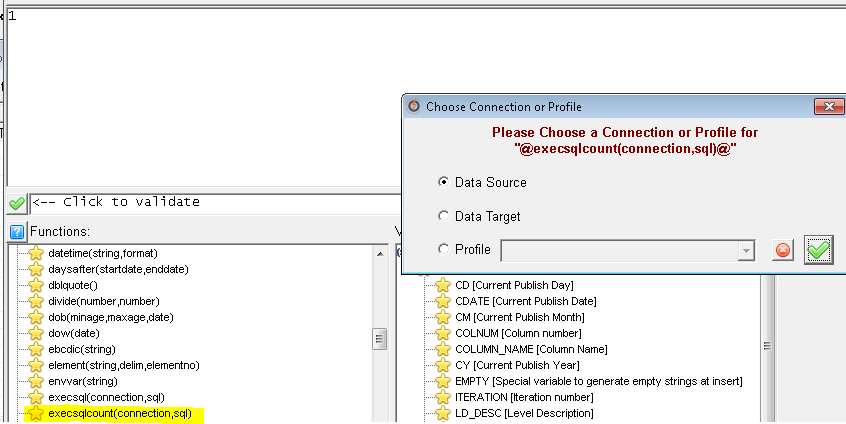
WEB\_USER\_NAME -> @seqlov(0,@sqllist(S,select \* from web\_user)@, ~COLUMN\_NAME~)@

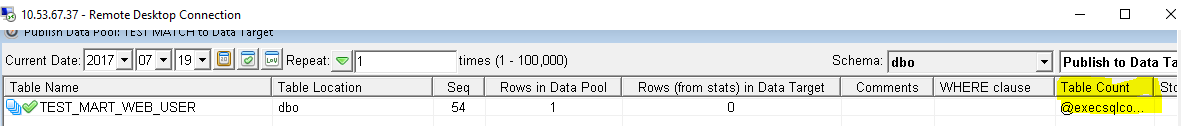
V) Select “Publish to Data Target” and click on GO button.

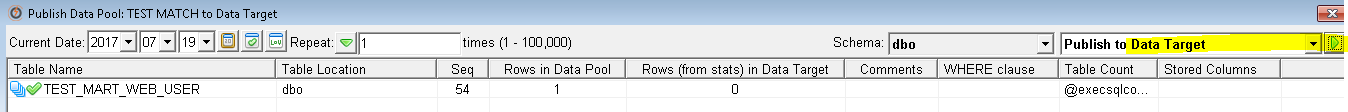


(VI) Right click on table and select Table Repeat Count.  ****

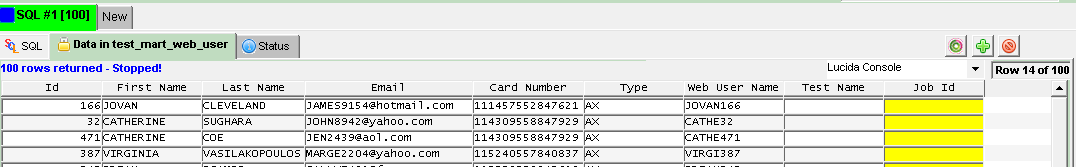
(VII) select Rule execsqlcount(connection,sql) ->data Source -> GO -> create rule “@execsqlcount(S,select \* from web\_user)@” -> Publish to Target -> GO







(VIII) Go to Target and check the table

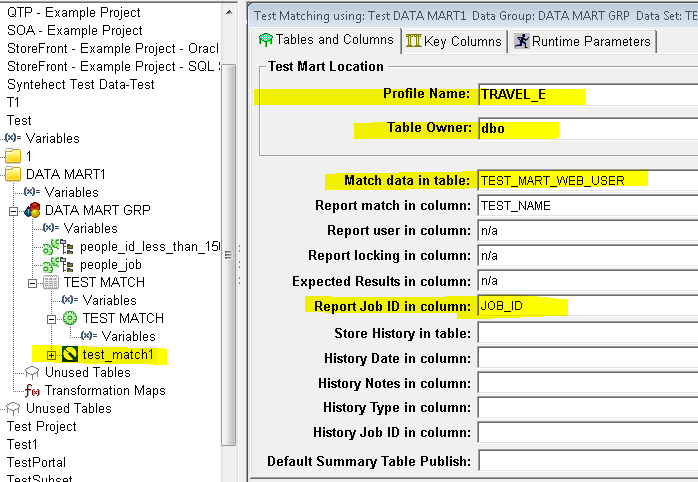


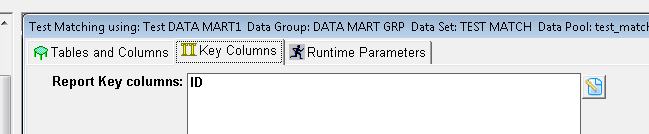
**Step4- User Test Case to Build Test Criteria**

(i) Create a new data pool under the data set “TEST MATCH” with type ”Test Match” and opt On demand

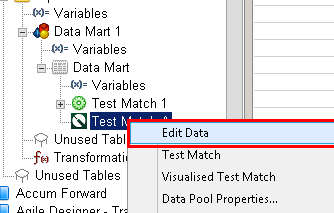
Profile Name – TRAVEL\_E and Table Owner – dbo and Key Columns – ID and Report Job ID in column is

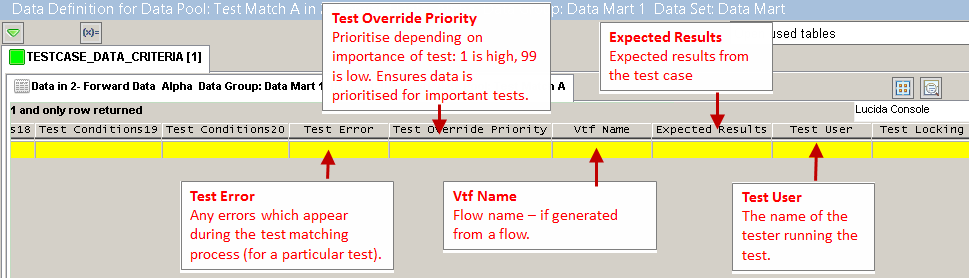
JOB ID ---> SAVE

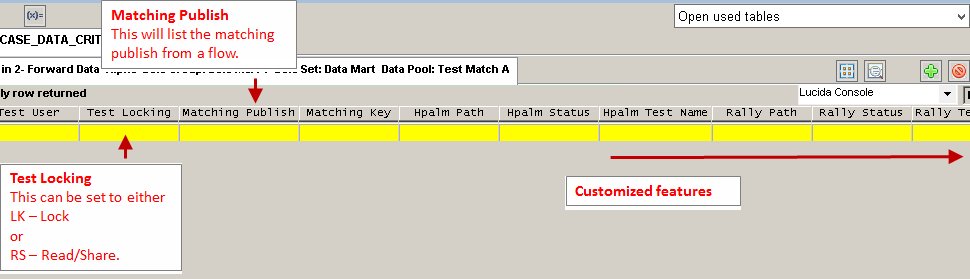


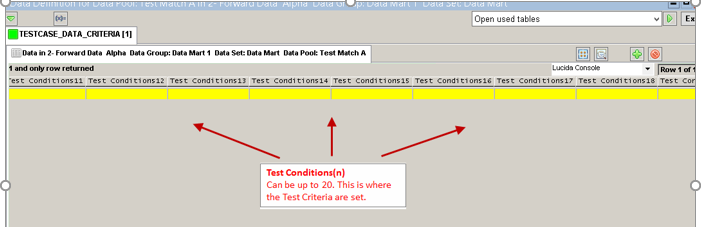


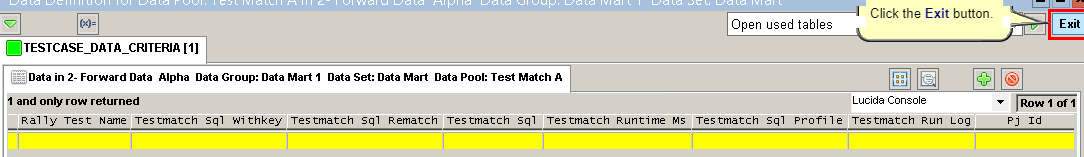
(ii) Right click on Test Match1 and select edit Data











(iii) Right click on Test Match1 and select edit Data and then setup the data as mentioned below.

Test Name – Credit Card

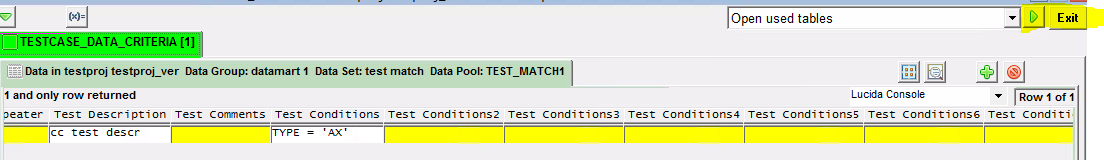
Test Active – Y

Test Description – Credit Card Test

Test Condition – “TYPE” = ‘AX’

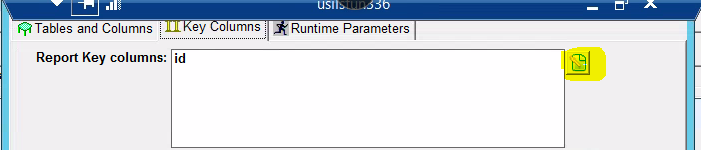
Test User - Sushanta

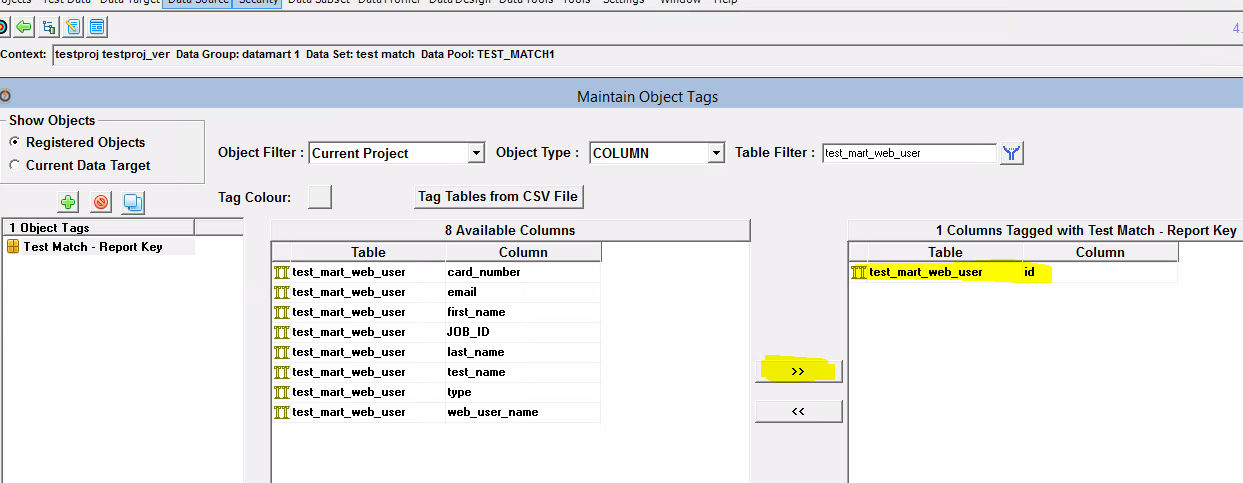
Then SAVE and EXIST.



**Step5 – Prepare and run test match**

(i) Go to the Project and right click on data pool Test Match1 -> select **Test Match** -> Go to the Key Column Tab- > click on the edit column -> SAVE -> Exit.

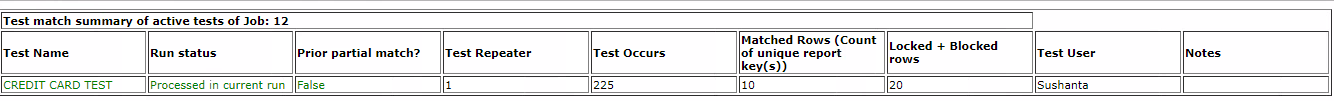


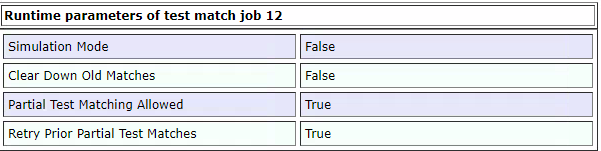


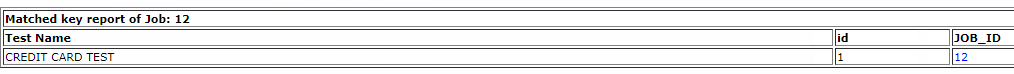
(ii) Click the Match button and submit Testmatch

|  |  |  |
| --- | --- | --- |
|  | **>** |  |

(iii) Test match report generated

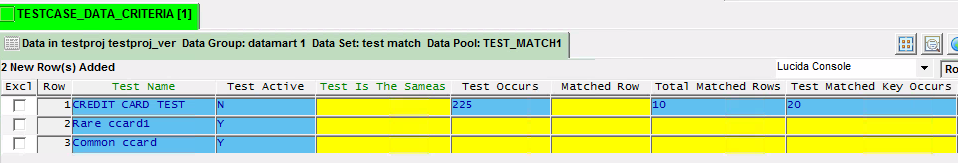


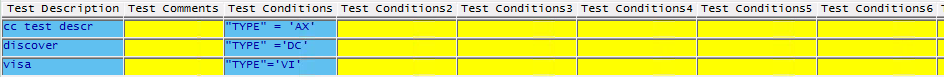


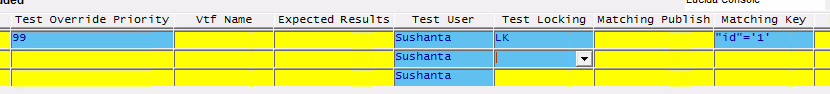


**Scenario 2**

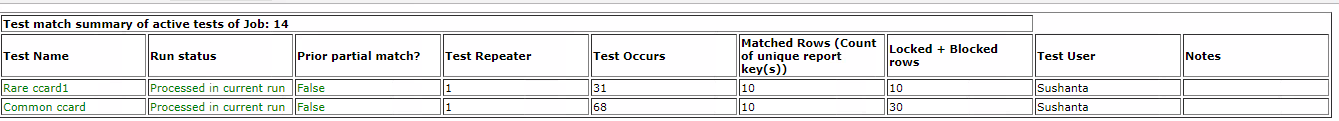
(i) Project -> right click on data poll(test match1) -> edit data -> Select used table “TESTCASE\_DATA\_CRITERIA” -> Add two more rows test cases-> set the test value -> SAVE -> Exit

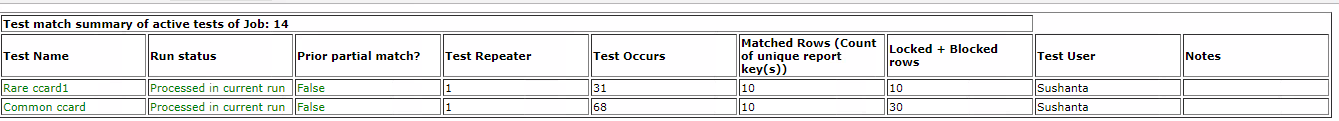


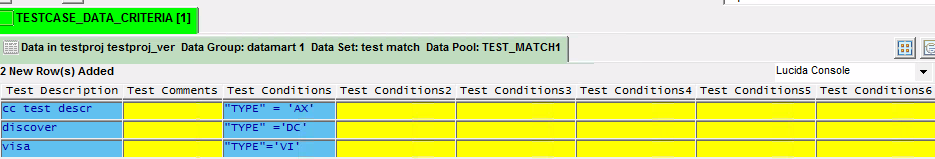


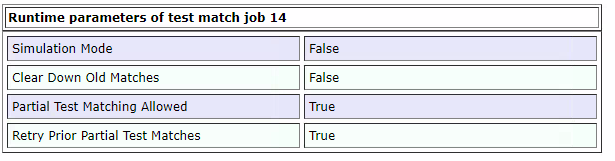


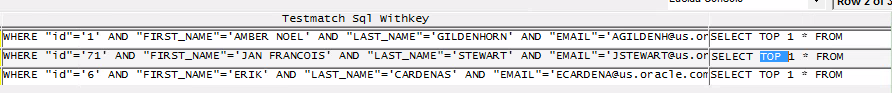
(ii) Project -> right click on data pool(test match1) -> test match -> submit test match to verify the report











# 5. DATA GENERATION (88TDM20170)