**The British College**

**KATHMANDU**

**Coursework Submission Coversheet**(individual coursework only)

**Faculty of Arts, Environment and Technology LBU Student Id:**

77261075

**For checking by the student:**

Please ensure all information is complete and correct and attach this form securely to the front of your work before posting it in a coursework collection box.

Award name: Bsc Hons in Computing

Module code: [15781 - AUT - 202210](https://my.leedsbeckett.ac.uk/webapps/blackboard/execute/courseMain?course_id=_157454_1)

Module name: Advance Database System (ADS)

Module run: 2022

Coursework title: Data warehouse Design and Development

Due Date:

Module leader: (In LBU): Jackie Campbell, Sanela Lazarevski

Module tutor: (In TBC): Dibya Tara Shakya

**TURNITIN** Checked: YES NO ***(please circle)***

Submission date& time: Date: Time: Before noon

**Total Word Count: Total Number of Pages (including this front sheet):**

**In submitting this form with your assignment, you make the following declaration:**  
I declare, that the coursework submitted is my own work and has not (either in whole or part) been submitted towards the award of any other qualification either at LBU or elsewhere. I have fully attributed/referenced all sources of information used during the completion of my assignment, and I am aware that failure to do so constitutes an assessment offence.

Signed: Mavira Bhattarai Date: <Submitted Date>

**You are strongly advised to retain a second copy of your work in case of any query about the assignment.**

**For completion by the faculty:**

**This mark is provisional and subject to moderation and approval by the relevant examining board**

**Teacher's Feedback**

**Teacher's Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Table of Content

Overview of the Project

Scope of the Project

Overview of Data Warehouse

Advantages of Data Warehouse

Limitations of Data Warehouse

Data Warehouse Architectures

## Inmon’s Corporate Information Factory (CIF)

## Kimball’s Business Dimensional Lifecycle:

Online Analytical Processing (OLAP)

Online Transaction Processing (OLTP)

Difference Between OLAP and OLTP

Data Warehouse Schemas

Star Schema

Snowflake Schema

Starflake/Hybrid Schema

Difference Between Star Schema and Snowflake Schema

Task 1: Data Warehouse (DW) Star Schema Design

Task 2: Set up the Data Source and Data Warehouse Environment

Task 3: Extract, Transform and Load (ETL) Script to Populate the Star Schema (DW) with Data

**Overview of the Project**

Our objective for this assignment is a police case study in which we will assume the position of an analyst or developer and produce a data mart (DM). Our main responsibilities are to assist the DM design, analyze the data, and gather data from numerous sources relevant to this case study. The project is concerned with crimes that take place in England and is broken down into three different KPIs (Key Performance Indicators), including "Reduce Crime," "Close Crimes," and "Identify Areas with Crime Hotspots." Each KPI has a distinct goal.

The Police Force oversees a number of stations around England. Every station is located in a distinct area, such as "Yorkshire" or "Lancashire." When a crime is reported in a particular location, the local station is promptly informed of the circumstances. A crime can be classified into three different types, Open, Closed, and Escalated, and its duration can vary. While offenses with a "Closed" status have previously been evaluated, those with a "Open" status are currently being investigated. Additionally, "Escalated" crimes are ones that have increased in severity. In cases where only one officer is available, the offenses are assigned to the stations based on their region and are handled by a lead police officer.

We were given a variety of data sources/files to use in our analysis for this work, including information regarding crimes and test results. Spreadsheets and SQL Scripts were the two types of files that were present.

## Project KPIs:

The project focused on three different KPIs (Key Performance Indicator) that are listed below:

1. **Reduce Crime**: This KPI deals with reducing the crimes.
2. **Close Crime**: This KPI is about making sure that crimes are solved and complete.
3. **Identify Areas with Crime Hotspots**: This KPI deals with the crime patterns and aids in understanding them.

The objectives mentioned for each KPIs are done and analyzed using several reports and for this assignment, the KPI that I will be working on is “**Close Crime**”.

**Scope of the Project**

The goal of this project is to build a useful data warehouse using the existing sources as the basis for the ETL process. The development of a star schema database model must be done in tandem with the construction of numerous reports. Focusing on extracting data from the original sources, editing them as needed, and loading the data into the data warehouse is the main duty.

**Overview of Data Warehouse**

A data warehouse (DW) is a method for gathering and controlling data from many sources to produce valuable business insights. To link and analyze corporate data from many sources, a data warehouse is often employed. The BI system, which is designed for data analysis and reporting, is centered around the data warehouse.

An OLAP engine, customer analysis tools, an extraction, transportation, and loading (ETL) solution, as well as additional applications that handle data collection and delivery to business users are all included in the data warehouse environment.

**Advantages of Data Warehouse**

Following are the advantages of Data Warehouse:

* Provides Historical Perspective
* Improves data quality and conformity
* Enhances Efficiency
* Boost the effectiveness and speed of data analytics
* Produce a High ROI
* Scalability
* Data Security
* Much better insight and query performance

**Limitation of Data Warehouse**

The limitation of data warehouse are as follows:

* Underestimation of ETL processing time
* Unreported issues with the Source
* Unable to collect the necessary data\
* Increasing user demands
* Long-term project

# Data Warehouse Architectures:

The architecture for a data warehouse can be developed or created using one of two methods. Two influential figures in the field of data warehousing, **Bill Inmon** and **Ralph Kimball**, proposed them. Both of them employed the ETL method to load the data into the DW and regarded it as the enterprise's primary data storage system. The way the data structures are modeled and loaded into the DW, however, is where they diverge most from one another. The methods are briefly detailed below:

## Inmon’s Corporate Information Factory (CIF) methodology:

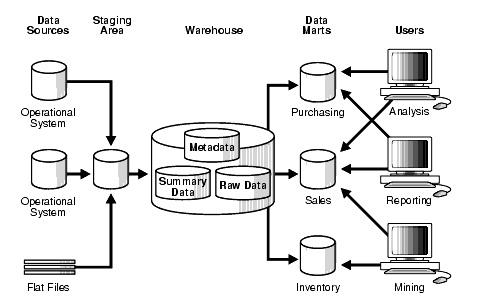
The DW architecture is built using a "Top-Down" methodology according to Inmon's method. This model outlines the major subject areas and, more crucially, the business-operated and cared-for entities, such as customers, products, etc. The principal entities are then given a thorough logical model that includes their keys, attributes, relationships, etc. The most important thing to keep in mind is that everything has been done in a normalized form. Due to the importance of having distinct company conceptions, data redundancy is minimized. The actual implementation is likewise carried out in a standard manner. Data loading was made simpler by the normalization idea, but query structure use was made challenging by the multiple table joins. The creation of Data Marts for certain departments, such as

Figure 1: Inmon's Top-down Architecture for DW, Source: Stanford, 2003, “Data Warehousing Concepts” (Accessed 11/22/2021)

### Advantages of Inmon’s Approach:

1. Since it is the single source for the data marts and the data is incorporated in the DW, the DW totally covers and serves the entire organization.
2. Because normalization helps to prevent data update anomalies, the ETL process becomes more efficient.
3. The logical model's display of specific entities helps one comprehend business operations better.
4. It can handle a range of reporting requirements across the company.

### Disadvantages of Inmon’s Approach:

1. As the number of tables and joins rises, the implementation of such a model gets increasingly challenging.
2. For it to operate properly, expensive and skilled human resources are needed.
3. It requires more ETL works because the data marts are built from the DW.
4. The initial set-up and delivery of the model takes a lot of time.

## Kimball’s Business Dimensional Lifecycle:

The DW architecture is built using a "Bottom-Up" strategy according to Kimball's method. The primary business operations and queries that a DW needs to be able to answer are initially identified via this model. Following that, information from different operational systems is examined and recorded in the staging area. The de-normalized version of the dimensional model is then constructed. The "Star Schema," which consists of a fact table surrounded by numerous dimension tables, is the basic idea behind this methodology. The dimension tables are totally de-normalized, whilst the fact table contains all the measurements based on the topic area, allowing the user to efficiently access data without the need to join numerous tables. Various Star Schemas can be merged based on the needs of the system. 2021 (Zentut)

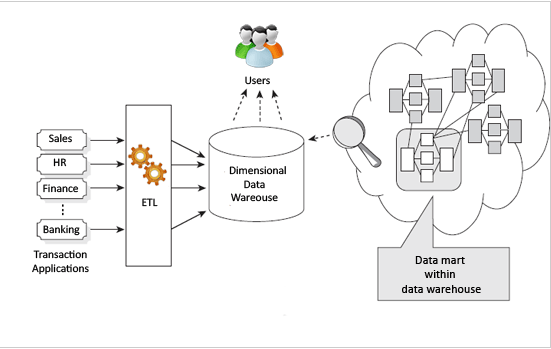


Figure 2: Kimball's Bottom-Up Approach Architecture for DW, Source: Zentut, 2021, “Ralph Kimball Data Warehouse Architecture” (Accessed 11/22/2021)

### Advantages of Kimball’s Approach:

1. This model is quick and simple to construct, and the DW will be sent quickly as well.
2. The Star Schema employed in this model is relatively straightforward, which facilitates and expedites reporting tasks.
3. The DW may be managed and operated well with a tiny human resource personnel.

### Disadvantages of Kimball’s Approach:

1. Data update anomalies might be seen with time due to data redundancy.
2. The dimensional model might become hard to change with respect to the business requirements.
3. The entire enterprise reporting cannot be handled because this model is oriented more over business processes.

## Key Differences between the Two DW Approaches:

|  |  |  |
| --- | --- | --- |
| **Characteristics** | **For Inmon’s Approach** | **For Kimball’s Approach** |
| **Approach** | Top-Down | Bottom-Up |
| **Business decision support** | Strategic | Tactical |
| **Data integration requirements** | Enterprise-wide integration | Individual business requirements |
| **Structure of data** | Data that meet multiple and varied information needs and non-metric data | KPI, business performance, measures, scorecards… |
| **Persistence of data in source** | Source systems have a high rate of change | Source systems are quite stable |
| **Skill set required** | Bigger team of specialists | A small team of generalists |
| **Time constraint** | Longer time is allowed to meet business needs | Urgent needs for the first data warehouse |
| **Cost of build** | High start-up costs | Low start-up cost |

Table 1: Key Differences between Inmon and Kimball Approaches (Zentut, 2021)

**Online Analytical Processing (OLAP)**

The multidimensional data model is the foundation of the Online Analytical Processing Server (OLAP). Through quick, consistent, and interactive access to information, it enables managers and analysts to gain an understanding of the information.

It is quickly improving the fundamental framework for intelligent solutions, which includes business performance management, planning, budgeting, forecasting, financial documentation, analysis, simulation-models, knowledge discovery, and reporting from data warehouses. End users can do ad hoc analyses of records in various dimensions using OLAP, which gives them the knowledge and insights they need to make better decisions.

**Online Transaction Processing (OLTP)**

Online transaction processing, often known as OLTP, is a type of data processing that entails carrying out several simultaneous transactions. These activities can include text message sending, online purchasing, online banking, and order entering, for example.

Economic or financial transactions are what these transactions are typically known as, and they are documented and safeguarded so that a company can access the information whenever needed for accounting or reporting purposes.

**Difference Between OLAP and OLTP**

|  |  |
| --- | --- |
| OLAP | OLTP |
| Processing of information from the past. | Involves processing done on a daily basis. |
| Knowledge professionals including executives, managers, and analysts use OLAP systems. | DBAs, clerks, and database specialists all use OLTP systems. |
| Useful for business analysis. | Helpful for managing the business |
| It emphasizes information out. | The emphasis is on Data in. |
| Based on Star Schema, Snowflake, Schema and Fact Constellation Schema. | Based on Entity Relationship Model. |
| Number or users is in hundreds. | Number of users is in thousands. |
| Database size is from 100 GB to 1 TB | Database size is from 100 MB to 1 GB. |

**Data Warehouse schemas**

Schemas or data models for data warehouses are specialized tools made for modeling data warehouses. They are used to analyze enormous databases by attending to the particular needs of the system. They provide a logical summary of the whole database.

There are three different types of data warehouse schemas and they are mentioned below:

1. Star Schema
2. Snowflake Schema
3. Hybrid Schema

## 1) Star Schema:

A fact table is present in the center of a data model called a "Star Schema" or "Star Join Schema," which is surrounded by dimension tables. Because a complete model of this schema resembles a "Star," it is known as the "Star Schema." It is the most extensively used data modeling framework for creating a data warehouse's table structure. While the dimension tables surrounding the fact table contain descriptive or attribute values, the fact table itself is composed of transactional values, or measurements.  (Taylor, 2021)

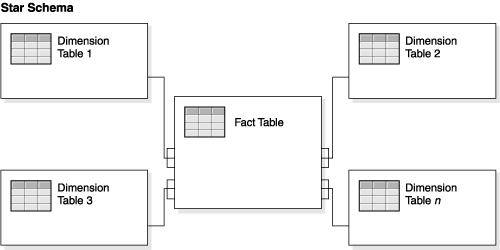


Figure 4: Star Schema, Source: IBM, 2021. “Designing and modeling” (Accessed 11/24/2021)

### Characteristics of Star Schema:

* Typically, the dimension tables are de-normalized while the fact table is normalized.
* In this structure, each dimension is represented by a separate table, which is connected to the fact table via a foreign key.
* The fact table has the keys and measures and the dimension tables can never have any kind of relationship with each other.
* This schema is easy to understand, so it is widely accepted by BI tools and it also provides optimal usage of storage.

## 2) Snowflake Schema:

A data model called a "Snowflake Schema" is one in which the fact and dimension tables are organized in a way that resembles a snowflake. It is derived from a "Star Schema." Because the dimension tables in a Snowflake Schema are additionally normalized, the data is divided into additional tables, which is the main distinction between a Snowflake Schema and a Star Schema. (Taylor, 2021)

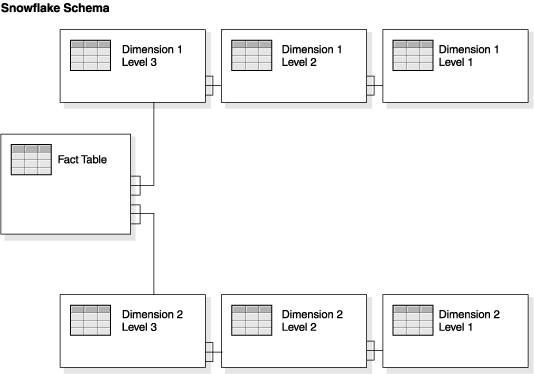


Figure 5: Snowflake Schema, Source: IBM, 2021. “Designing and modeling” (Accessed 11/24/2021)

### Characteristics of Snowflake Schema:

* The dimension tables and the fact tables are both in normalized form.
* The main benefit of this model is that it uses very low disk space and it becomes easier to apply a new dimension in the schema.
* Because of the de-normalized dimension tables, there occurs several table joins that reduces the query performance.
* This schema will require more maintenance to function properly due to more number of dimension lookup tables.

## 3) Starflake/Hybrid Schema:

The hybridized form of a star and snowflake schema is referred to as a starflake schema or a hybrid schema. They mostly fall under the category of Snowflake schemas that only have some of their dimension tables de-normalized. The major reason for developing such a schema is to take advantage of both the Star and Snowflake schemas' advantages.

### Characteristics of Hybrid Schema:

* In this type, the branches or hierarchies of a star schema are de-normalized while the hierarchies from a snowflake schema are in normalized state.
* Such schemas are normalized for removing any data duplication or redundancies present in the dimensions.

## Differences between Star and Snowflake Schema:

|  |  |
| --- | --- |
| **Star Schema** | **Snowflake Schema** |
| It consists of the fact and dimension tables only. | It consists of the fact, dimension and sub-dimension tables. |
| It is based on top-down approach of data modeling. | It is based on bottom-up approach of data modeling. |
| It consumes more disk storage. | It consumes less disk storage. |
| The query execution time is low for this model. | The query execution time is higher than in Star Schema. |
| Normalization is not done in this schema type. | Both normalization and de-normalization can be found in this schema. |
| It has simple design. | The design of this schema can be complex. |
| It is very easy to understand. | It can be hard to understand the overall process. |
| This type of schema has high data redundancy. | Data redundancy is very low in this schema type. |

Table 3: Differences between Star and Snowflake Schemas (GeeksforGeeks, 2020)

**Task 1: Data Warehouse (DW) Star Schema Design**

## Reports that are supported by the Star Schema:

* Name of the police officer and station where the crime has closed the most from 2018 to 2022
* Crime type where time duration between registering crime and closing crime of the solved crimes is more than 2 years
* Number of crimes and crime type that has been closed the most and least in 2019
* Type of crime that has been closed this year
* Year and month where crime has been closed the most and the least

## Star Schema Model:



## Data Dictionary for the star schema:

A data dictionary is a tool used to centralize information in tabular form by defining the information content, format, and structure of a database with appropriate relationships between its pieces. This tool is used to control who has access to and may manipulate the database. It can also be seen as a collection of metadata. In our Star Schema, there are a total of five tables: one fact table and the other four dimension tables. They are available below with all the necessary details regarding the columns and tables. The source of the column data and the necessary transformations are also included.

Data Dictionary for the star schema

Dimension Tables

**DIM\_TIME**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column | Data type | Column type (SK, PK, FK, | Definition, could be used a help text | Example data  To clarify the content of the attribute esp for flags, codes | Source element and transformation | Transformation rule. May include pseudo-sql |
| TIME\_ID | INTEGER | PK | The unique identifier for the time period | ‘1’ | A sequence dim\_time\_seq | Create a sequence named dim\_time\_seq to generate the primary key |
| DAY | INTEGER |  | The day the data relates to | ‘04’ | Day part taken from reported\_date | Will need to extract the year from the date |
| MONTH | INTEGER |  | The month the data relates to | ‘10’ | Month part taken from reported\_date | Will need to extract the month from the date |
| YEAR | INTEGER |  | The year the data relates to | ‘2015’ | Year part taken from reported\_date | Will need to extract the day from the date |

**DIM\_OFFICER**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column | Data type | Column type (SK, PK, FK, | Definition, could be used a help text | Example data  To clarify the content of the attribute esp for flags, codes | Source element and transformation | Transformation rule. May include pseudo-sql |
| OFFICER\_NO | INTEGER | SK, PK | The unique identifier for the officer | ‘1’ | A sequence dim\_officer\_seq | Create a sequence named dim\_officer\_seq to generate the primary key |
| OFFICER\_ID | INTEGER |  | The officer id kept at two source tables | ‘1’ | Officer\_id taken from Officer and pl\_police\_employee |  |
| OFFICER\_NAME | VARCHAR(50) |  | Full name of the lead police officer | ‘James Owen” | Officer\_name taken from Officer and pl\_police\_employee  Inconsistency check:   * The firstname, middlename and surname is not combined in a single column for PS\_Wale data source | Will need to combine first name, middle name and last name as full name |
| RANK | INTEGER |  | Rank of the officer | ‘1’ | Rank taken from Officer and pl\_police\_employee |  |
| DB\_SOURCE | VARCHAR(20) |  | Source of the data | ‘PRCS’ | Self- written to determine the source of the data |  |

**DIM\_REGISTER**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column | Data type | Column type (SK, PK, FK, | Definition, could be used a help text | Example data  To clarify the content of the attribute esp for flags, codes | Source element and transformation | Transformation rule. May include pseudo-sql |
| REGISTER\_NO | INTEGER | SK,PK | The unique identifier for the crime register | ‘2’ | A sequence dim\_register\_seq | Create a sequence named dim\_register\_seq to generate the primary key |
| REGISTER\_ID | INTEGER |  | The register id kept at two source tables | ‘2’ | Register\_id taken from PL\_REPORTED\_CRIME and CRIME\_REGISTER |  |
| CRIME\_NAME | VARCHAR(25) |  | Name/ Type of the crime | ‘DRUG VIOLENCE’ | Crime\_name taken from CRIME\_REGISTER and CRIME\_TYPE |  |
| CRIME\_STATUS | VARCHAR(25) |  | Status of crime | ‘OPEN’ | Crime\_status taken from PL\_REPORTED\_CRIME and CRIME\_REGISTER |  |
| REGISTER\_DATE | DATE |  | Crime registered date | ‘2013-02-13’ | REGISTER\_DATE taken from PL\_REPORTED\_CRIME and CRIME\_REGISTER |  |
| CLOSE\_DATE | DATE |  | Crime closed date | ‘2015-03-11’ | REGISTER\_DATE taken from PL\_REPORTED\_CRIME and CRIME\_REGISTER |  |
| OFFICER\_ID | INTEGER |  | The officer appointed for the crime | ‘1’ | OFFICER\_ID taken from PL\_WORK\_ALLOCATION and CRIME\_REGISTER |  |
| WORK\_START\_DATE | DATE |  | The work starting date of officer | ‘2015-12-22’ | WORK\_START\_DATE taken from PL\_WORK\_ALLOCATION |  |
| WORK\_END\_DATE | DATE |  | The work end date of officer | ‘2013-07-13’ | WORK\_END\_DATE taken from PL\_WORK\_ALLOCATION |  |
| STATION\_ID | INTEGER |  | The station where the crime is assigned | ‘3’ | STATION\_ID taken from CRIME\_REGISTER |  |
| DB\_SOURCE | VARCHAR(20) |  | Source of the data | ‘PL\_WALES’ | Self- written to determine the source of the data |  |

**DIM\_STATION**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column | Data type | Column type (SK, PK, FK, | Definition, could be used a help text | Example data  To clarify the content of the attribute esp for flags, codes | Source element and transformation | Transformation rule. May include pseudo-sql |
| STATION\_NO | INTEGER | SK, PK | The unique identifier for the station | ‘5’ | A sequence dim\_station\_seq | Create a sequence named dim\_station\_seq to generate the primary key |
| STATION\_ID | INTEGER |  | Station\_id kept in two data sources | ‘5’ | STATION\_ID taken from PL\_STATION |  |
| STATION\_NAME | VARCHAR(20) |  | The name of the station | ‘CROSS GATES’ | STATION\_NAME taken from PL\_STATION |  |
| DB\_SOURCE | VARCHAR(20) |  | Source of the data | ‘PRCS’ | Self- written to determine the source of the data |  |

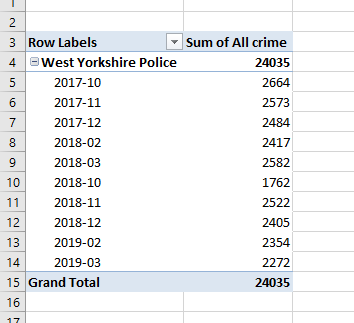
Fact table

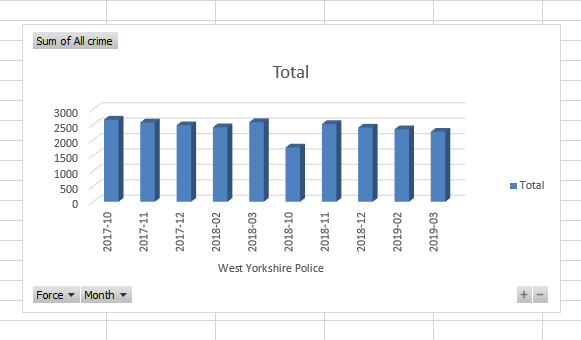
|  |  |
| --- | --- |
| Process, description of fact table and measures | The fact table supports the process – close crimes  And the KPI – ‘close’ crimes. The reports it supports are:   * Name of the police officer AND station where the crime has closed the most from 2018 to 2022 * Crime where time duration between registering crime and closing crime of the solved crimes is more than 2 years * Crime that has been closed the most and least in 2019 * Crime that has been closed this year * Year and month where crime has been closed the most and the least |
| Grain, exact level of data for each row of data | The data for the crime is held by crime\_name  The data for the station is held by station\_name  The data for the officer is held by officer\_name  The data for the time s held by year, month and day |
| PK’s | Crime\_id – surrogate PK from a sequence |
| FK’s | Time\_id, register\_id, officer\_id,station\_id |
| List facts (measures) | No\_of\_years |
| List dimensions that are used with this fact | DIM\_TIME, DIM\_REGISTER, DIM\_OFFICER, DIM\_STATION |

## Expected data for a selected report:

### Report selected:

* Year and month where crime has been closed the most and the least



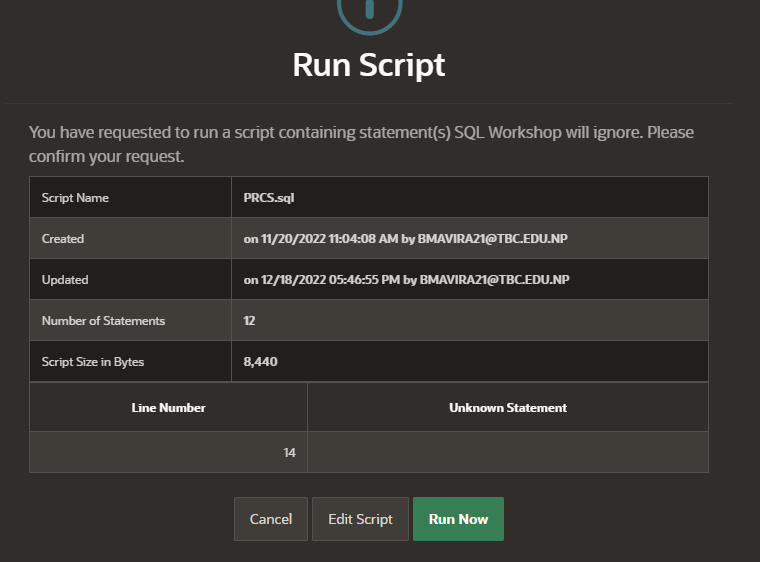


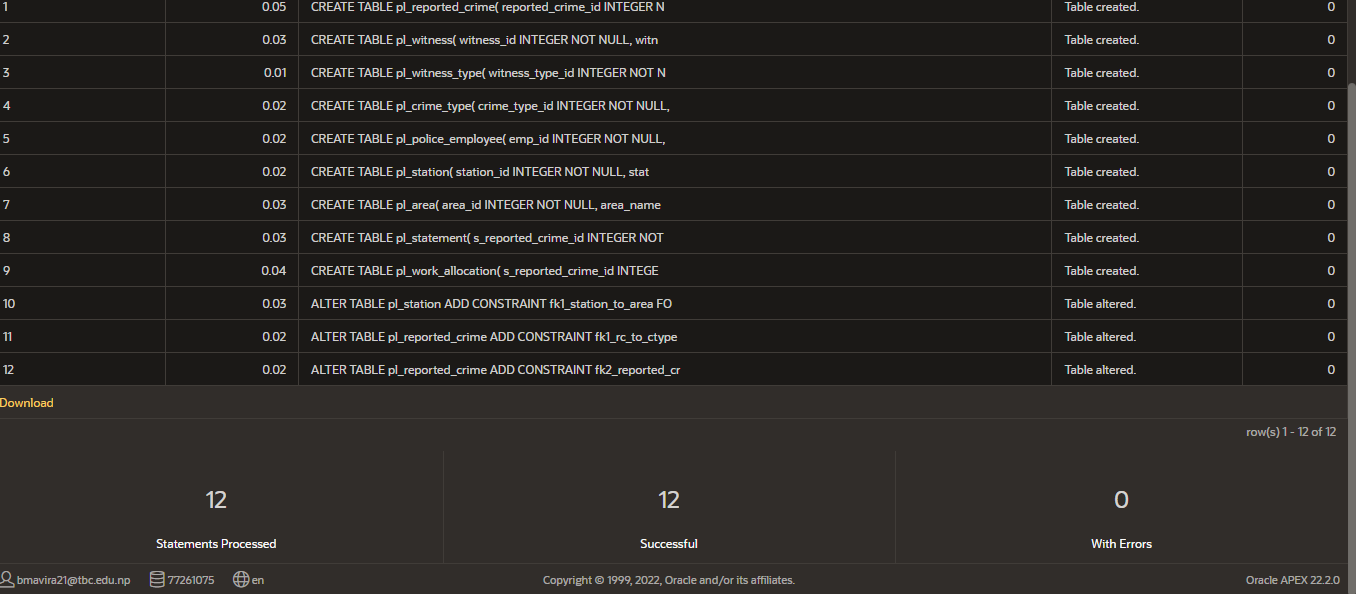
# Task 2: Star Schema set up (DM environment)

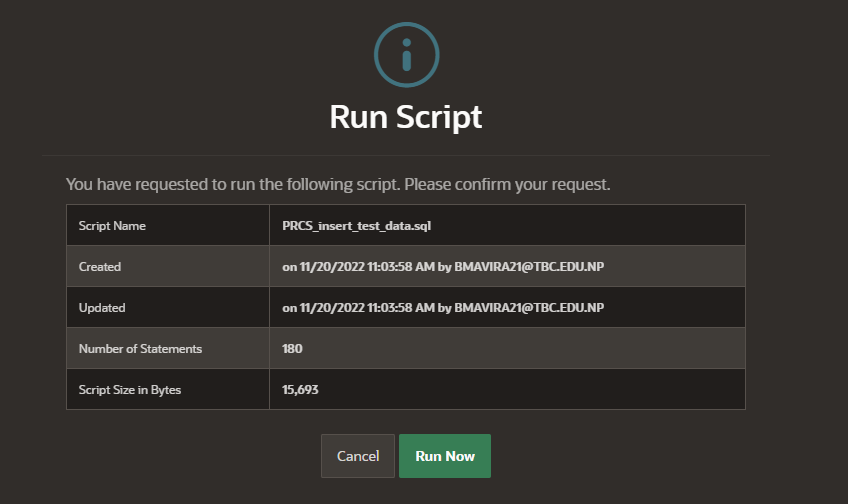
## Uploading data source script:

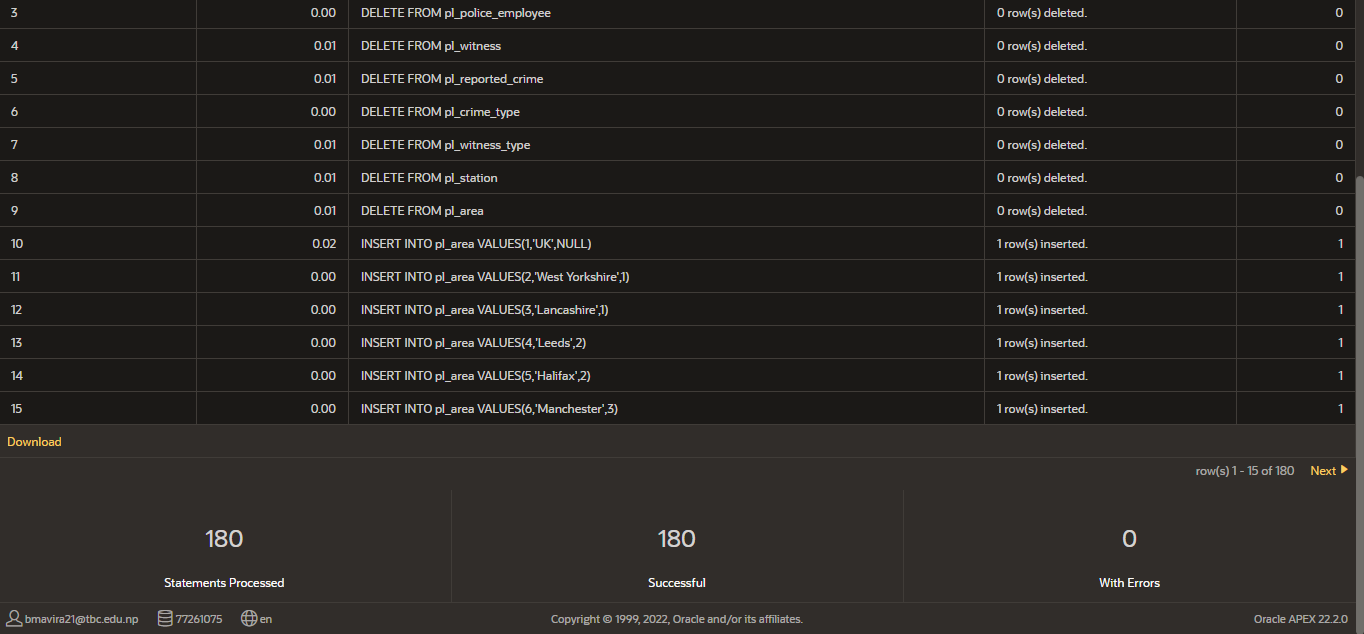
We were given access to three different data sources for this assignment, two of which were SQL scripts and one of which was an excel file. The screenshots below demonstrate how the data sources were successfully uploaded and used in the Oracle workspace.

### Running the PRCS script files:

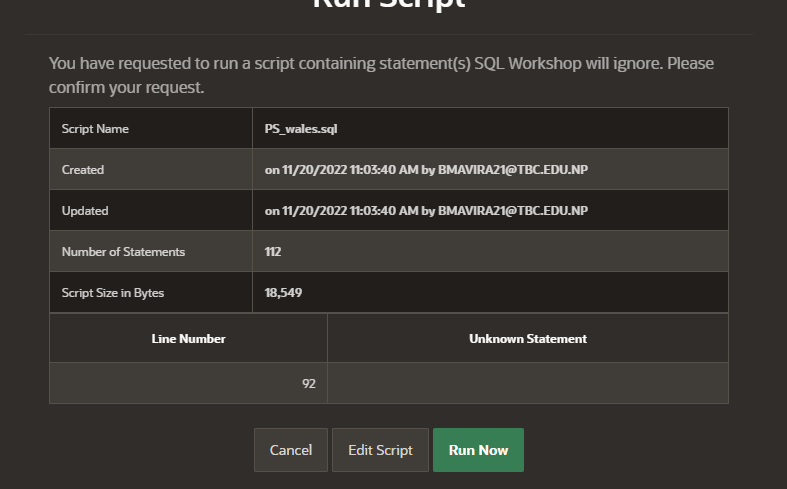


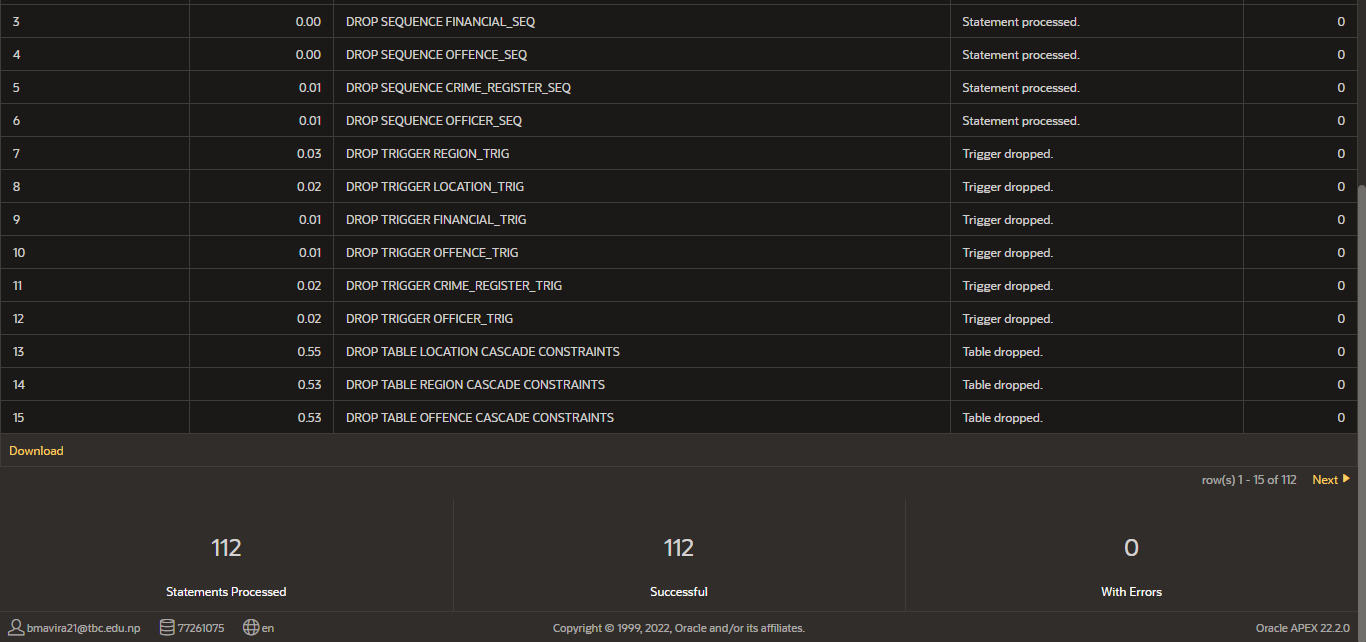


**Running PRCS\_insert \_test\_data script file**



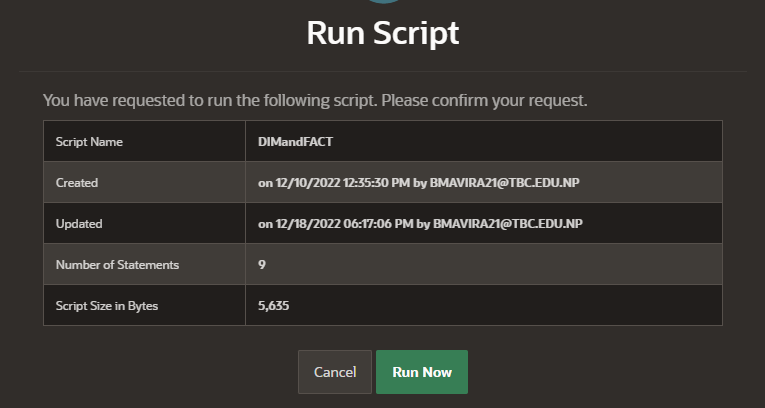
### Running the PS\_Wales script file:

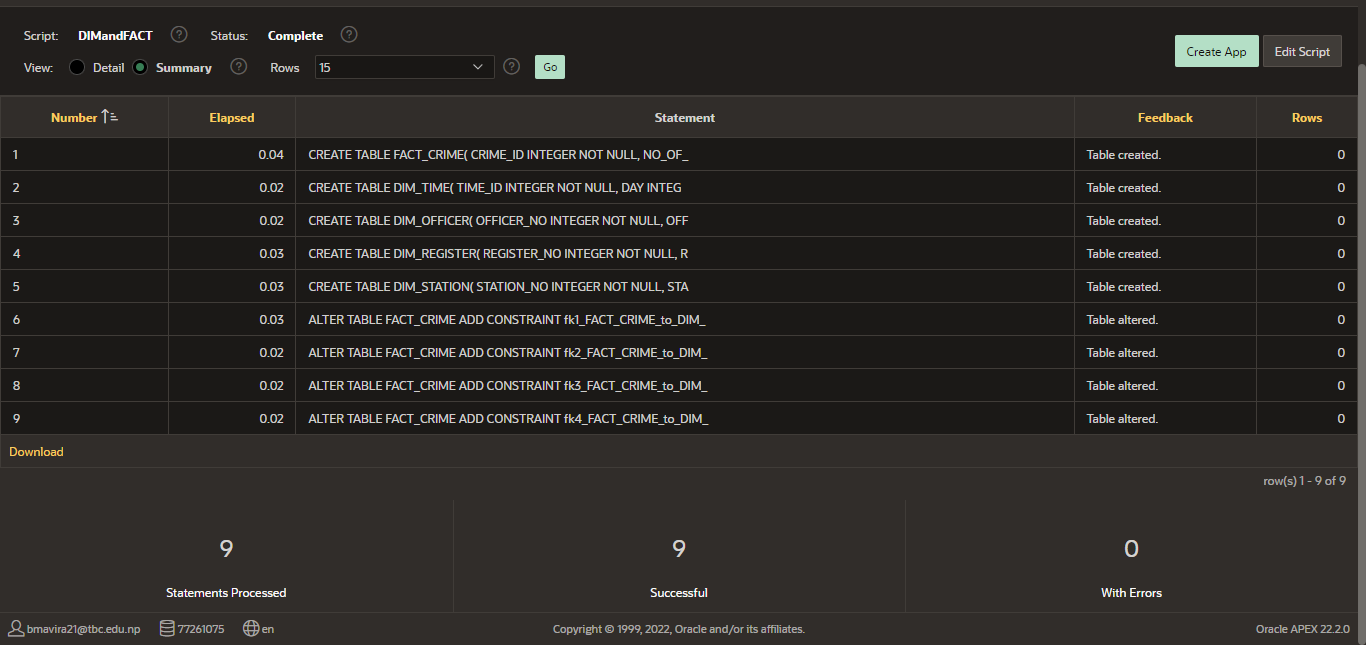




## Creating the Data Mart tables/Dimension and Fact tables:

The dimension tables were further created with the help of QSEE with appropriate edits. The evidence for the successful table creation is provided below:





The workspace was then used to generate the star schema model. The Oracle workspace had all the required files and data sources loaded. This checkpoint signaled the end of job 2 and opened the door for task 3, or the ETL process, to begin. From here, complete the remaining tasks for the assignment.

# Task 3: Extract, Transform and Load (ETL) script to populate the Star Schema (DM) with data

## ETL Process:

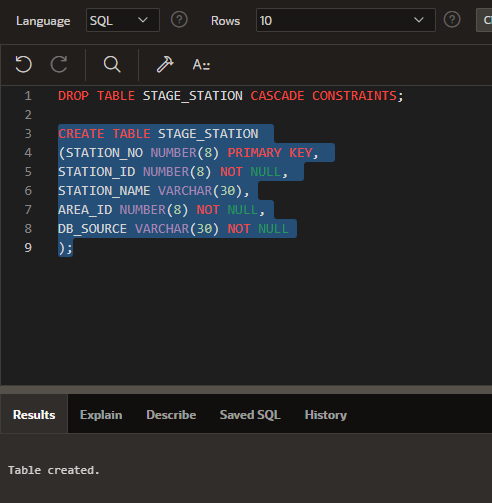
A data warehouse can be filled with information by using the ETL (Extraction, Transform, and Load) process, which is merely a way of copying data from one database to another. Data is extracted from one or more OLTP systems, transformed in accordance with the DW schesma, and then loaded into the DW database in this procedure. It is best to approach the ETL for a DW as a process as opposed to a physical implementation. Because fresh data can be added to the system on a weekly, daily, or even hourly basis, it is not a one-time occurrence. It is a process that requires a lot of time and skill and for it to be effective, it must be:

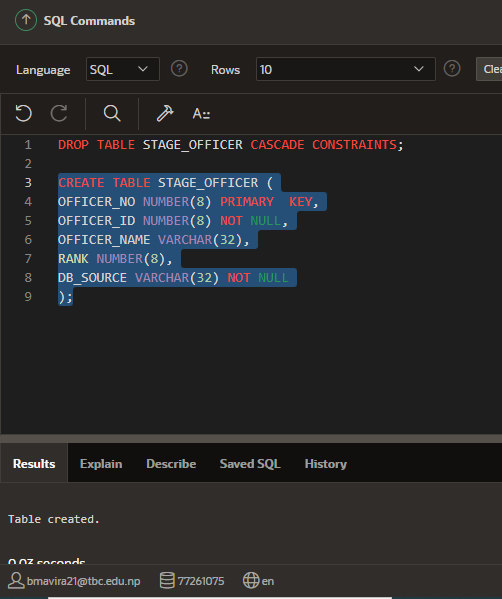
* Automated
* Easily changeable
* Well documented

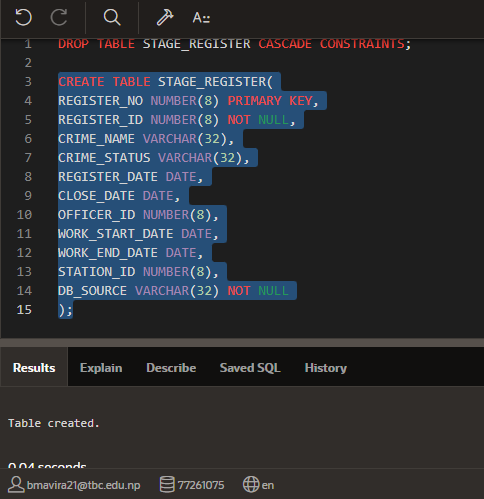
## The Staging area:

A staging area with the requisite tables, sequences, and triggers to hold the required data from the source system was constructed before data was extracted into the staging tables from the source system. The essential queries are successfully run below to complete this process.

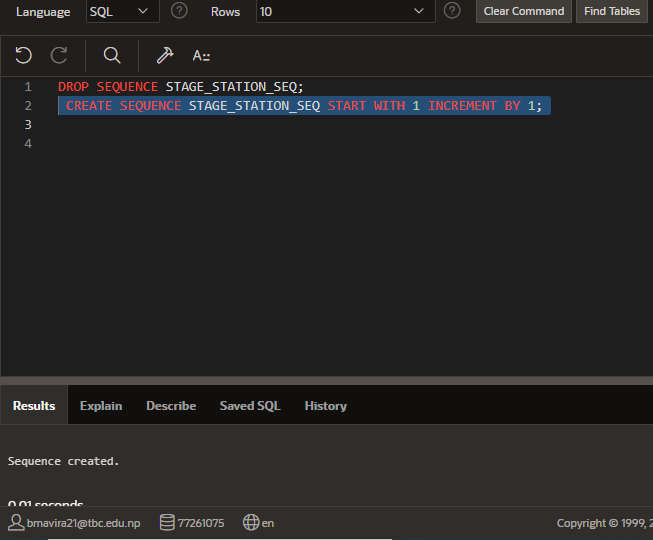
### Creating the staging tables:

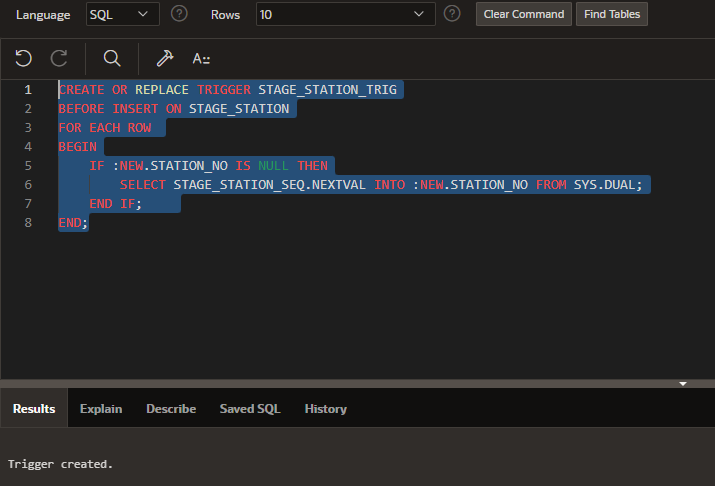


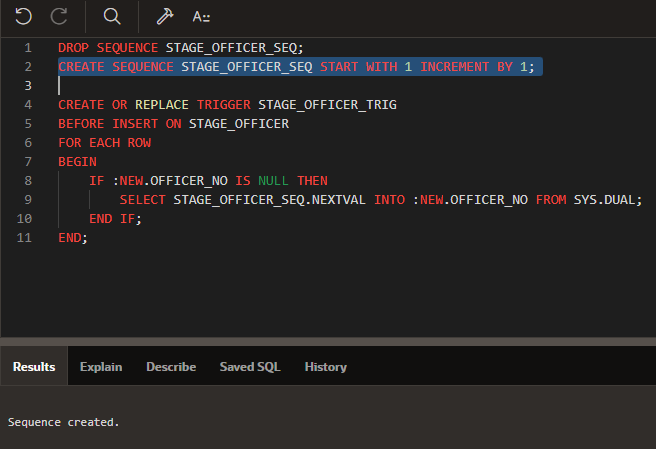


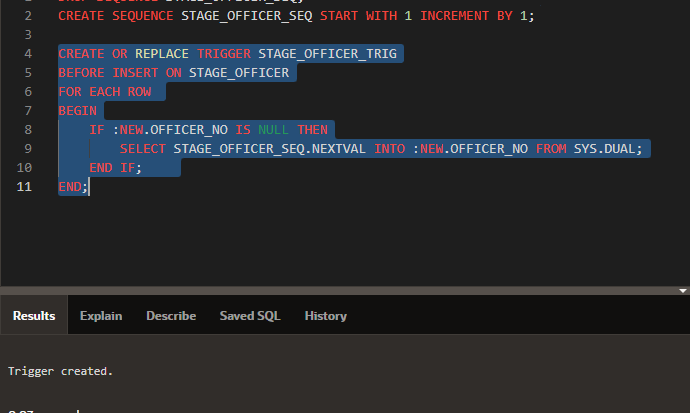


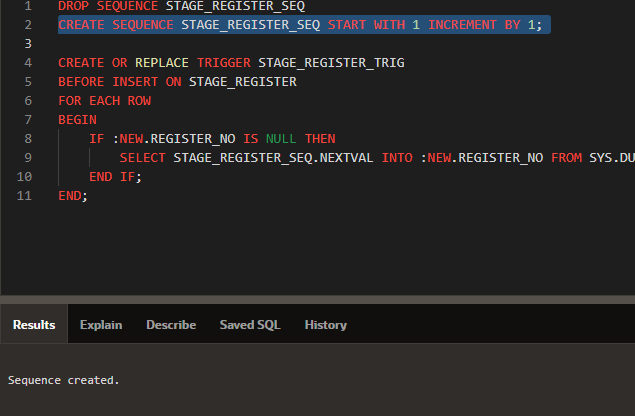
### Creating the sequences and triggers for staging tables:

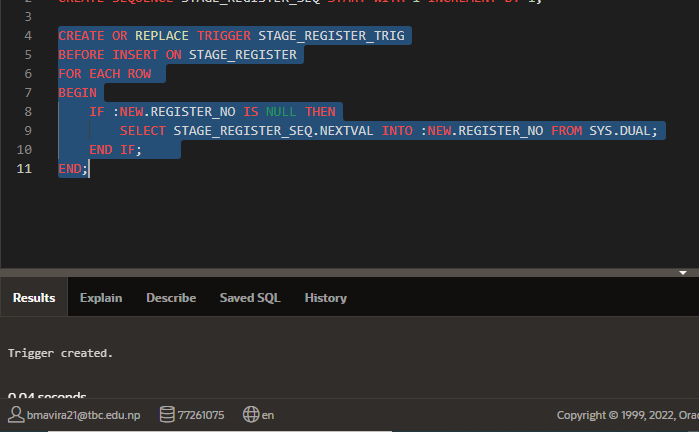










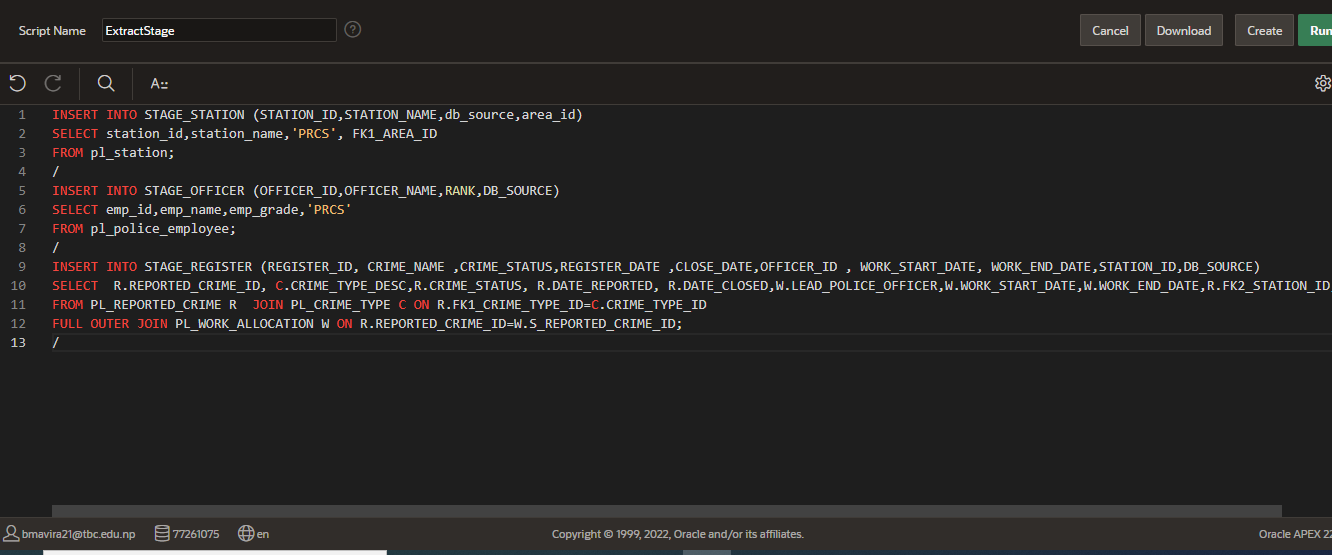


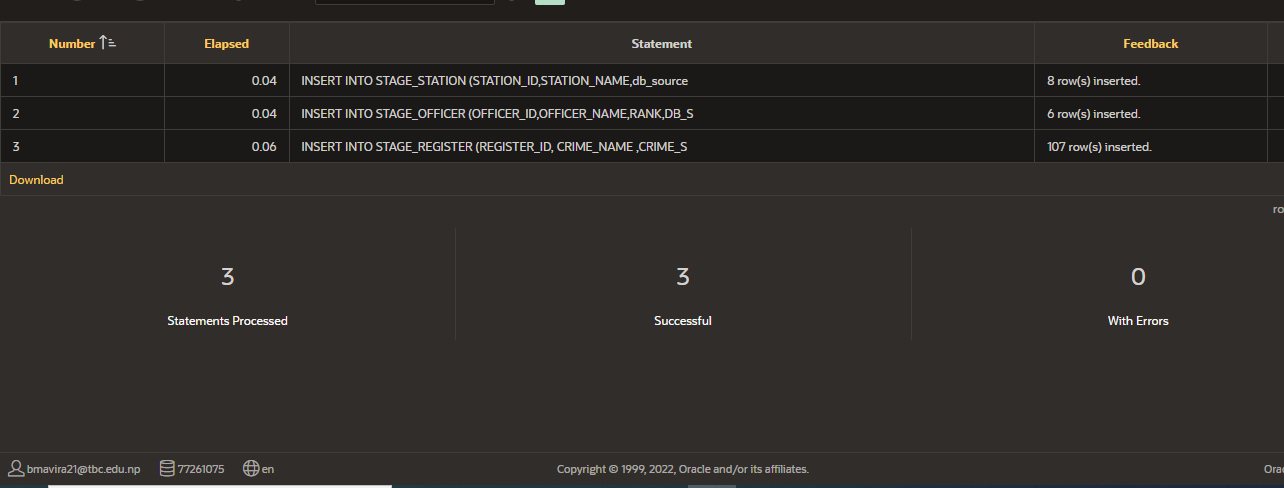
### Extraction:

The data is extracted from the available sources in this phase of the ETL process and then sent to the staging area. For efficient data extraction, the many data sources must be handled and merged with one another during the ETL process. The physical data should be transformed using a logical data map (similar to a data dictionary).The major validations that are done during extraction:

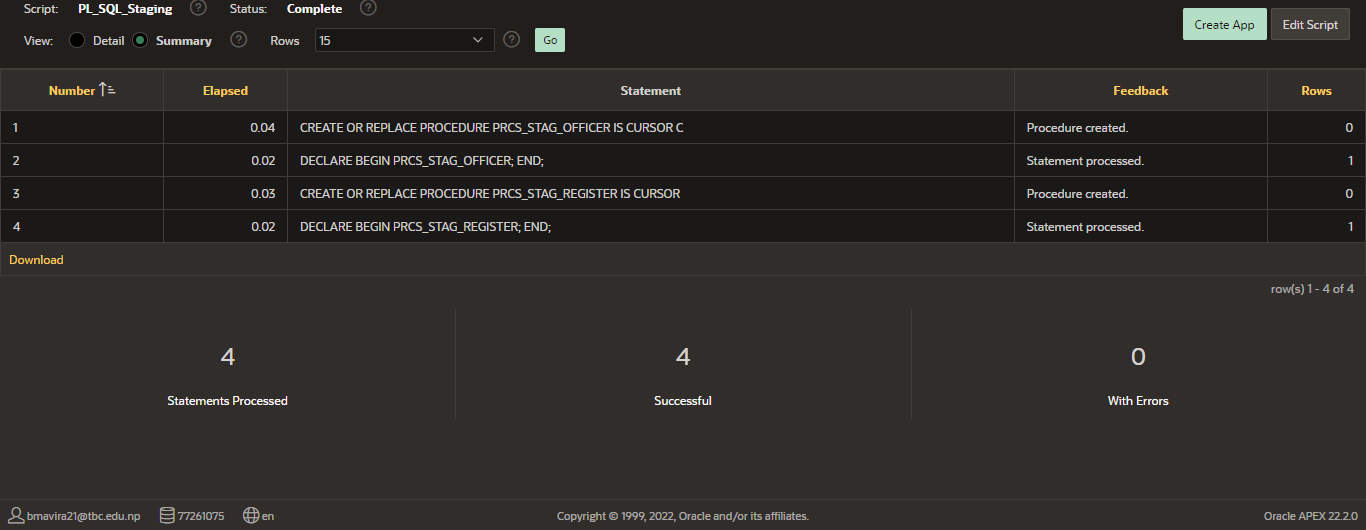
* Verifying the data's data type and ensuring that no undesired data has been loaded
* Eliminating all duplicate data and examining the keys' condition

### Extracting the source data into the staging tables:









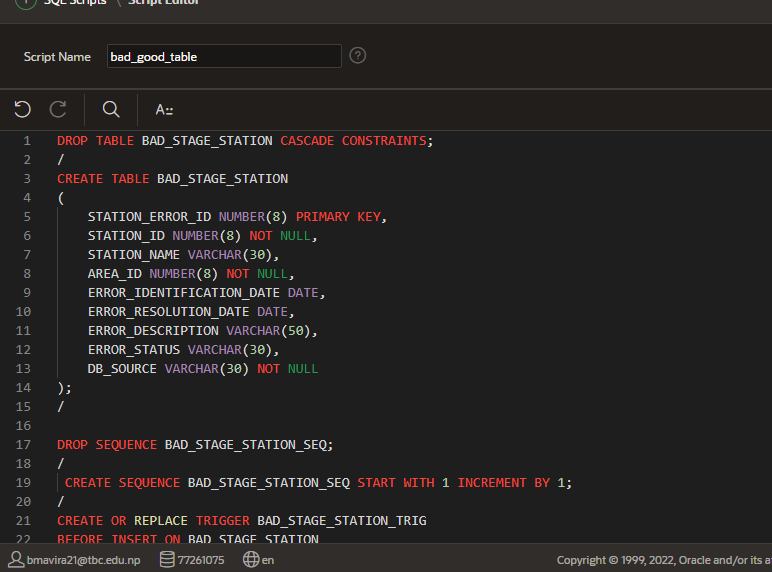
### Generating bad data:

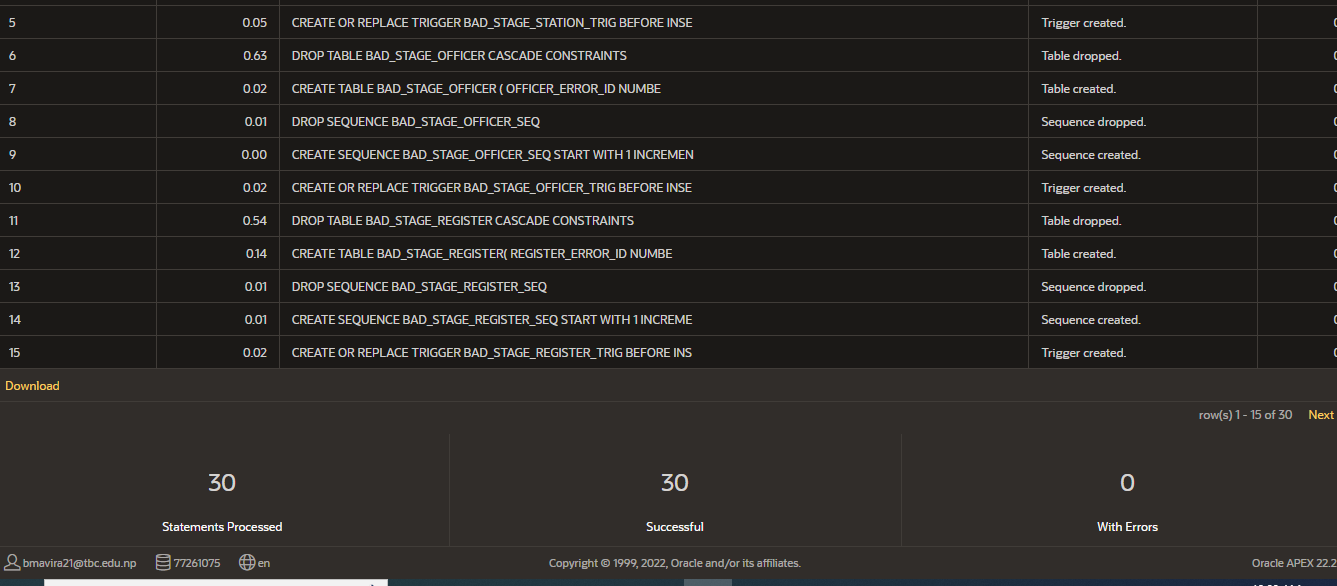
For cleansing and transformation process to ensure data quality, few data were made bad. The following updates were carried out for a few records in the staging table to generate bad data.

* Adding special characters in station\_name
* Setting the station\_name as null
* Adding special characters in officer\_name
* Setting the officer\_name as null

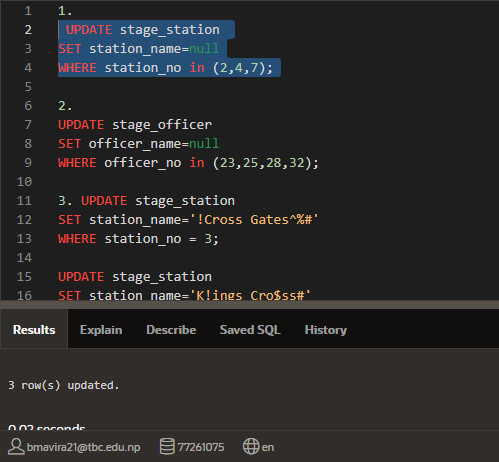
### Creating the bad and good tables:

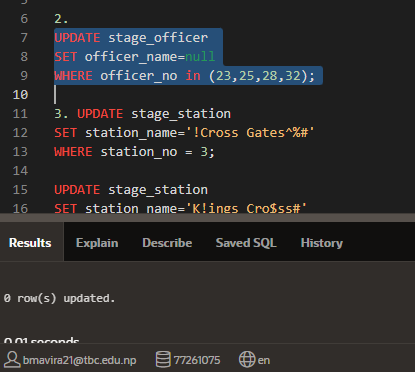
Prior to creating the bad data, good tables and bad tables were created for storing them. The necessary sequences and triggers were also created along with the tables.

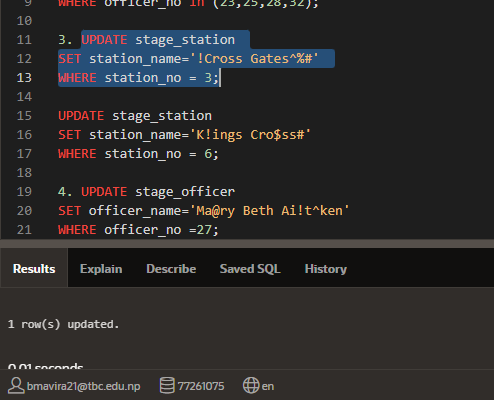


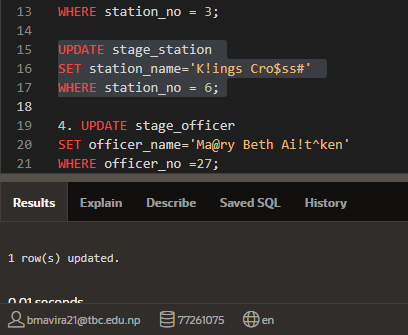


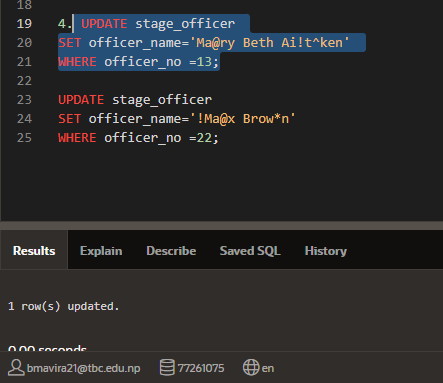
### Creating bad data in the staging tables:

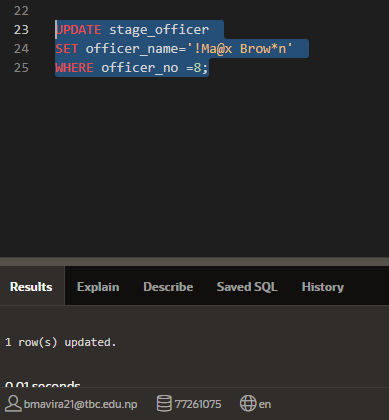




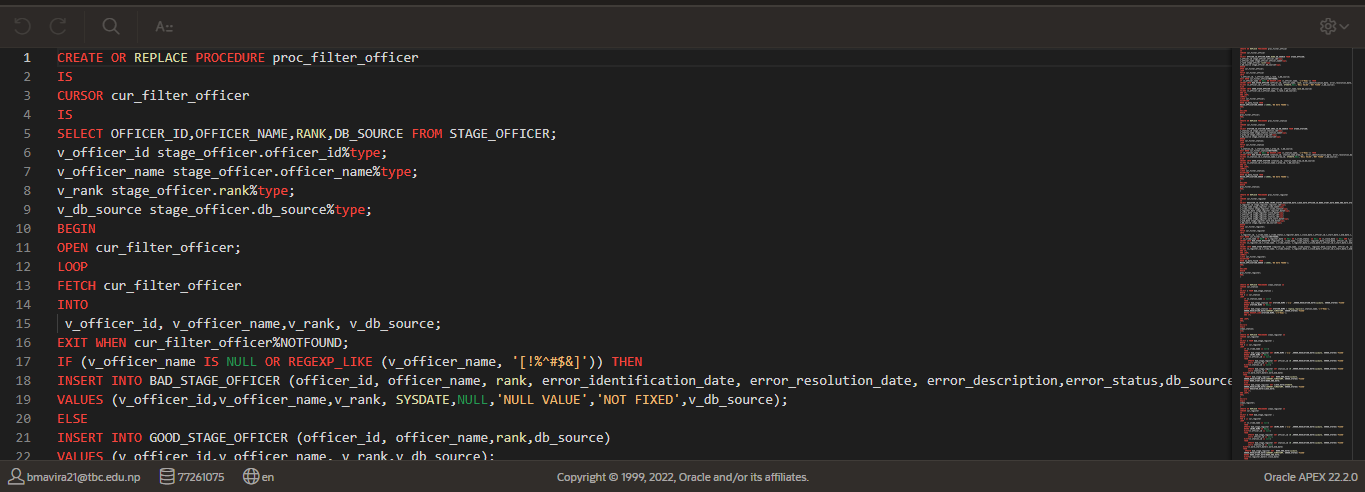


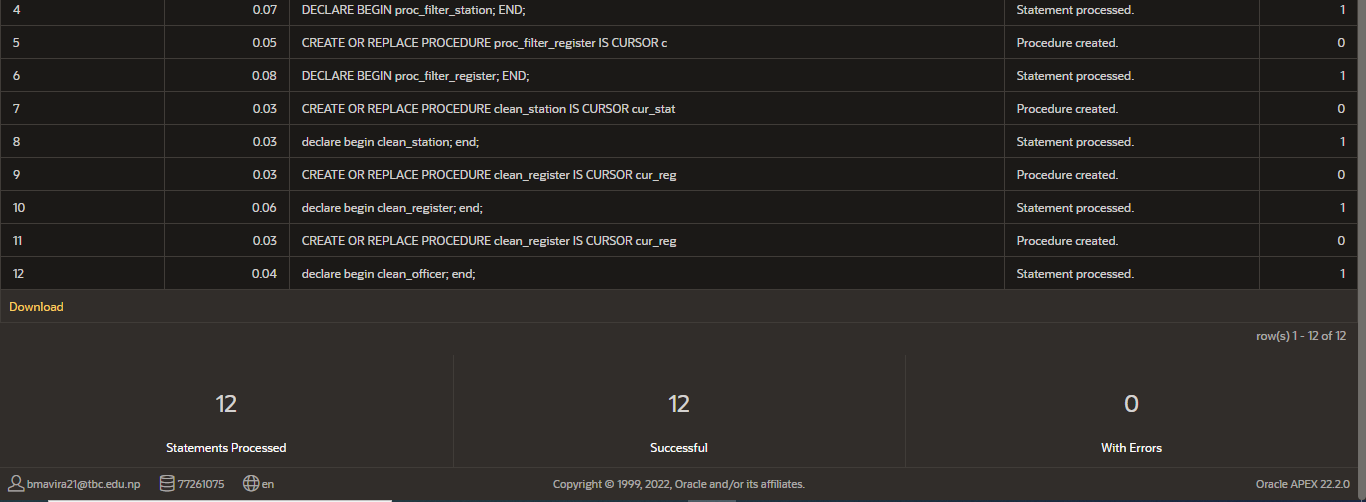






### Filtering the bad and good data:





### Transformation:

In the ETL process, this stage determines whether or not the data can be used for the intended purpose, adding value to the data. After the extraction process and after the cleansing and confirming procedure, the data is checked for quality in both locations. Applying certain functions to the retrieved data allows for the quality-checking procedure. "Pass Through data" refers to the extracted data that needs no modifications. In this step, the following validations are performed: (Guru, 2021)

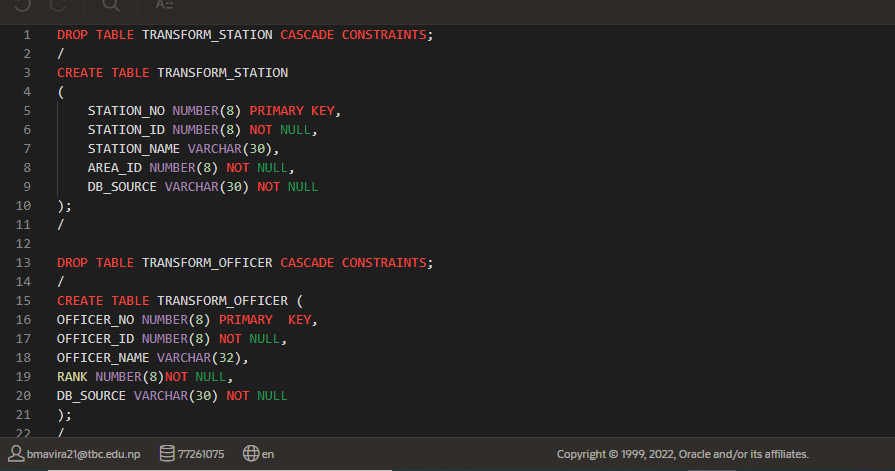
* Filtering certain columns for loading
* Applying rules and lookup tables for data adjustment
* Conversions like measurement units, date and time, currency, etc.
* Data length validation
* Checking blank required fields
* Cleaning the extracted data
* Merging data
* Splitting or merging table columns as required, etc.

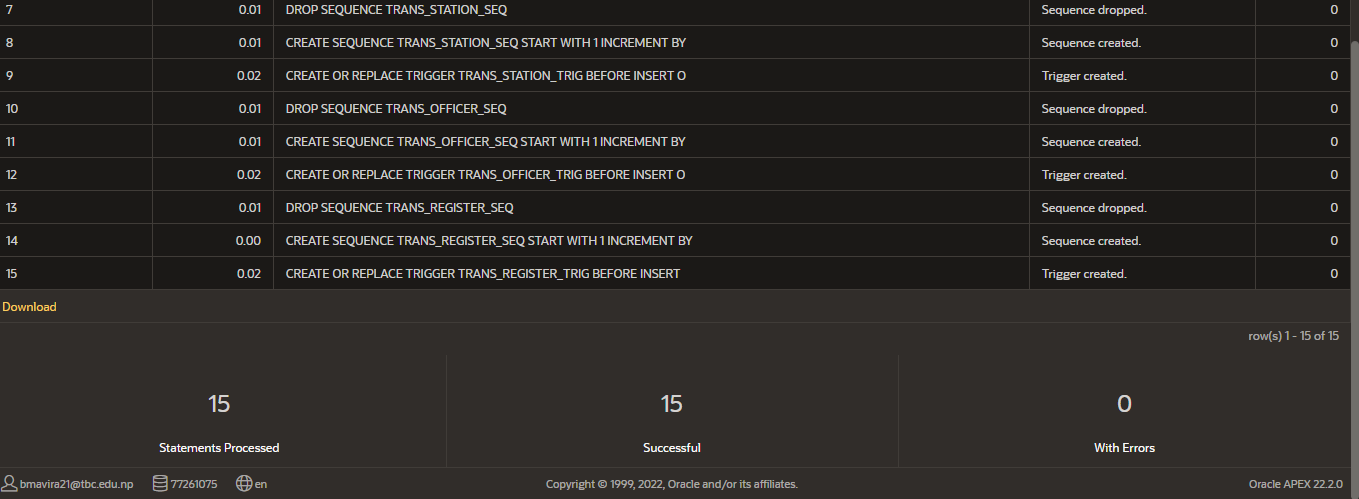
The data must be cleaned, mapped, and transformed into correct useable, relevant data because it may not be usable in its original form when it is taken from the sources and loaded into the corresponding staging tables. On the basis of the final data requirements, customized operations are carried out on the data in this step. The necessary records would be transformed after being imported into the appropriate transformation tables.

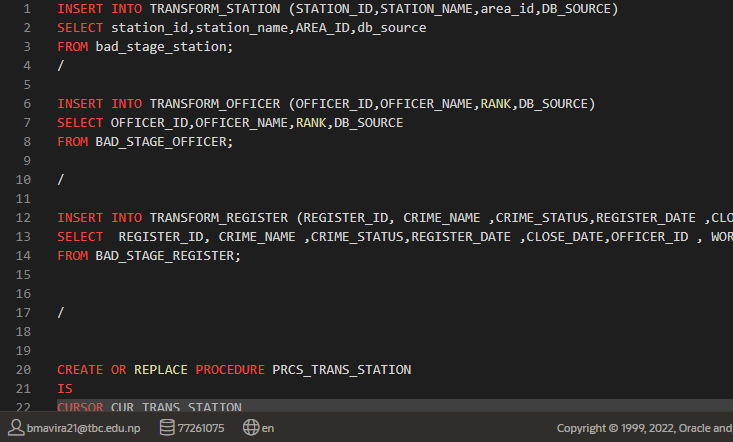
In this step, the script to generate the transformation tables and a package to transfer the clean data into the transformation tables were created. The necessary data transformations and cleaning were also completed.

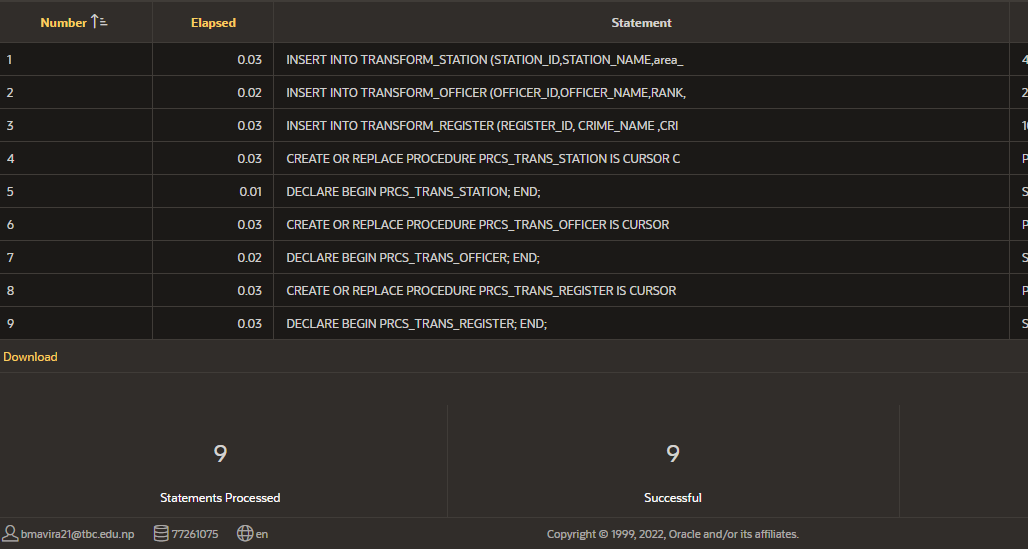
The script that generates the transformation tables (trns lead officer, trns station, and trns crime type) as well as the sequences and triggers to automatically generate the primary key upon the insertion of new data into the transformation tables are shown in the screenshots below. Additionally, a package called merge clean trns Pkg is created, including the instructions for merging or moving the data from the clean tables to the appropriate transformation tables.

### Creating transformation tables:



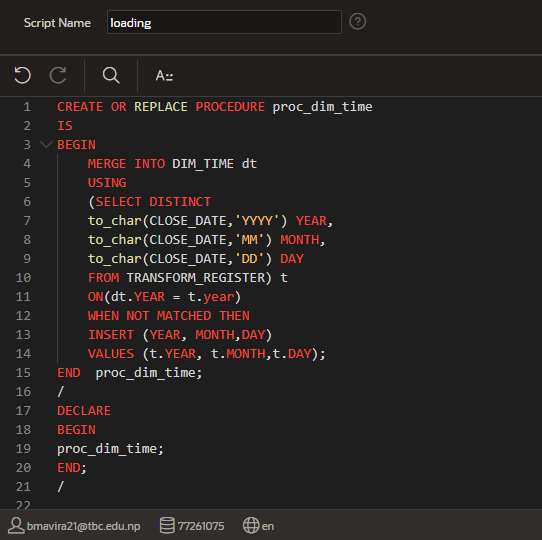


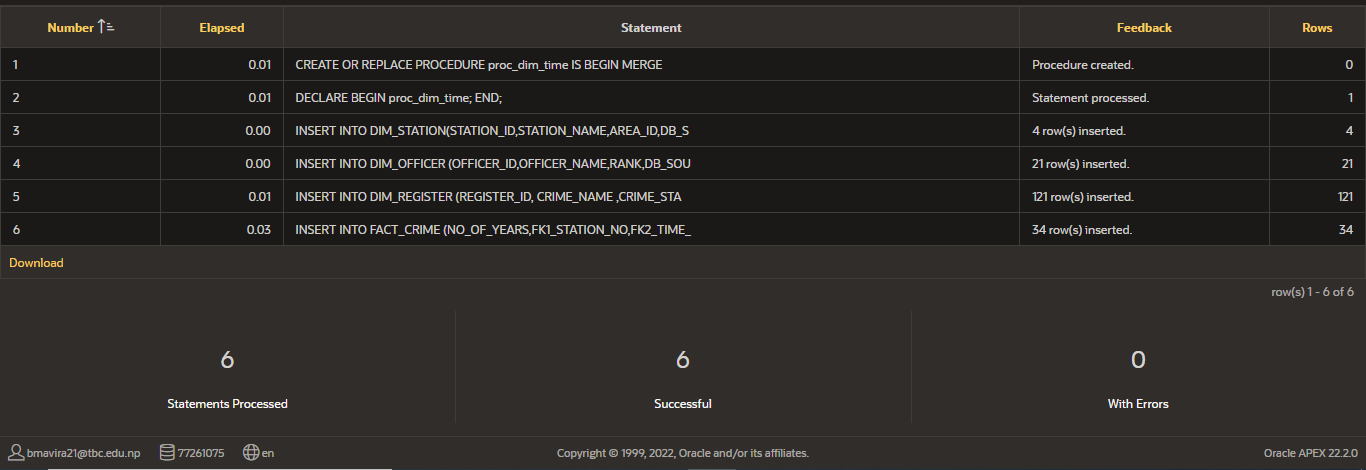




### Loading:

The targeted DW database is loaded with data from both the dimension and fact tables in this last step of the ETL process. The SCD (Slowly Changing Dimensions) are also taken care of in this step by using the appropriate PKs, Natural Keys, and final characteristics. This stage also includes the generation and assignment of the surrogate keys. Indexes and partitions are taken care of in this stage in preparation for loading the fact table. In the event that this step fails, the recovery measures should be applied without compromising the integrity of the data.





**SCD**

Type 1

Data is simply overwritten into the dimension tables in Type 1 SCD. When the record is imported into the dimension tables, there may be instances where we are unable to locate all of the data or where the data may change. In this project, the new value from the OLTP system replaces the old data warehouse value when the crime register record is initiated into the dimension table. Therefore, the crime status in the old data warehouse is replaced with the new data from the database when the crime status in the OLTP system is altered.

Type 2

In Type 2 SCD, rather than overriding updated data is added in a new row in dimension tables. In this project, when the police officer is appointed new work, the data is added in new row with previous data of the same officer existing.

