**LAB Manual**

**PART A**

(PART A: TO BE REFERRED BY STUDENTS)

**Experiment No.02**

**A.1 Aim:**

Implementation of all dimension tables and fact tables related to the case study mentioned in the first experiment.

**A.2 Prerequisite:**

Refer to the DBMS manual for SQL Commands and ER diagram.

**A.3 Outcome:**

After successful completion of this experiment students will be able to design a data warehouse with dimension modelling.

**A.4 Theory:**

Dimensions are categories by which summarized data can be viewed. E.g. a profit summary in a fact table can be viewed by a Time dimension (profit by month, quarter, year), Region dimension (profit by country, state, city), Product dimension (profit for product1, product2).

A fact table is a table that contains summarized numerical and historical data (facts) and a multipart index composed of foreign keys from the primary keys of related dimension tables.

In data warehousing, a dimension is a collection of reference information about a measurable event. These events are known as facts and are stored in a fact table. Dimensions categorize and describe data warehouse facts and measures in ways that support meaningful answers to business questions. They form the very core of dimensional modelling.

Dimension tables are referenced by fact tables using keys. When creating a dimension table in a data warehouse, a system-generated key is used to uniquely identify a row in the dimension. This key is also known as a surrogate key. The surrogate key is used as the primary key in the dimension table. The surrogate key is placed in the fact table and a foreign key is defined between the two tables. When the data is joined, it does so just as any other join within the database.

**Algorithm:**

CREATION OF OLTP TABLES

1. **Customer table**



The CUSTOMER table is populated using the following sample DML statement



1. **Item table**



The ITEM table is populated using the sample DML statement



1. **Trans Table**

The TRANS table is populated using the following sample DML statement



1. **Item\_sold Table**

The ITEM\_SOLD table is populated using the following DML statement



1. **Branch Table**

The BRANCH table is populated using the following DML statement



**IMPLEMENTATION OF ENTERPRISE DATAMART**

CREATION OF DIMENSION TABLES

1. **Customer Dimension Table**
2. **Item dimension table**



1. **Branch dimension table**



1. **Time dimension table**



CREATION OF FACT TABLE

1. **Sales\_fact table**



**PART B**

(PART B: TO BE COMPLETED BY STUDENTS)

***(Students must submit the soft copy as per the following segments within two hours of the practical. The soft copy must be uploaded on the Blackboard or emailed to the concerned lab in charge faculties at the end of the practical in case there is no Blackboard access available)***

|  |  |
| --- | --- |
| **Roll No.** 50 | **Name:** AMEY THAKUR |
| **Class:** Comps TE B | **Batch:** B3 |
| **Date of Experiment:** 02/02/2021 | **Date of Submission:** 02/02/2021 |
| **Grade:** |  |

**B.1 Software Code written by a student:**

***(Paste your problem statement related to your case study completed during the 2 hours of practice in the lab here)***

**Dimension Table**

**CUSTOMER**

CREATE TABLE CUSTOMER

( CUSTOMER\_ID VARCHAR2(10) PRIMARY KEY,

NAME VARCHAR2(40) NOT NULL,

GENDER VARCHAR(10) );

INSERT INTO CUSTOMER VALUES ( 1, 'Archit Konde', 'MALE');

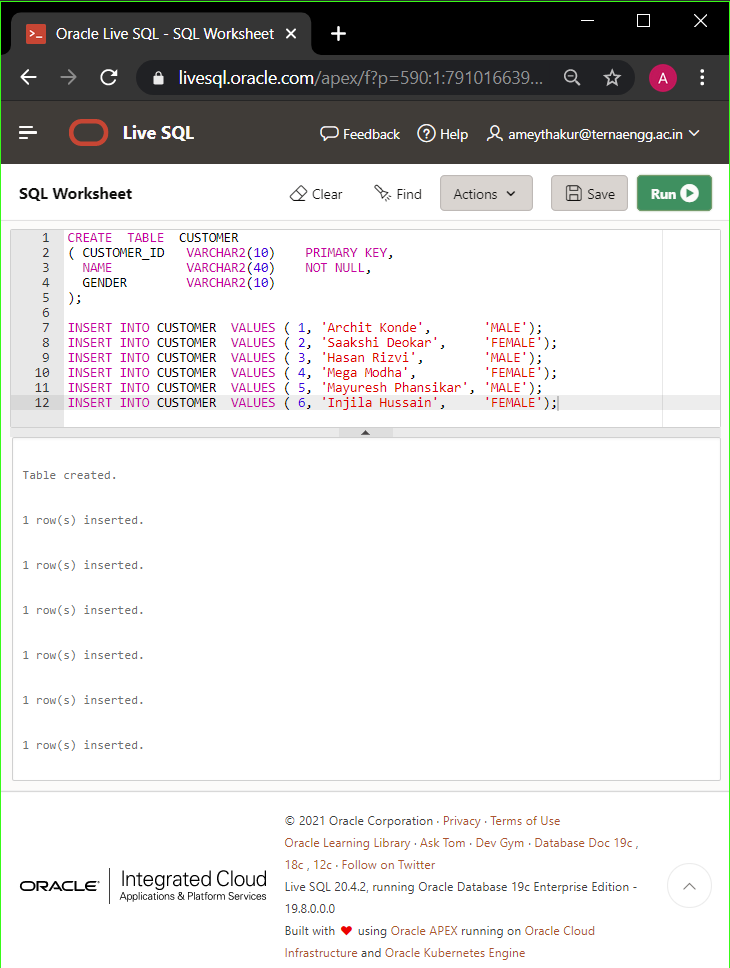
INSERT INTO CUSTOMER VALUES ( 2, 'Saakshi Deokar', 'FEMALE');

INSERT INTO CUSTOMER VALUES ( 3, 'Hasan Rizvi', 'MALE');

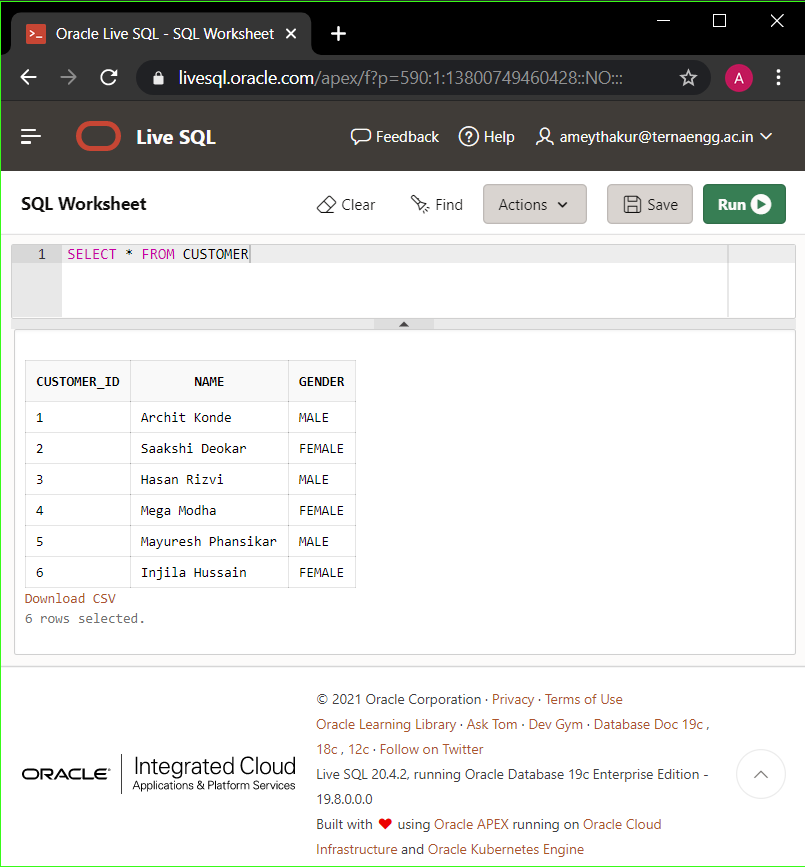
INSERT INTO CUSTOMER VALUES ( 4, 'Mega Modha', 'FEMALE');

INSERT INTO CUSTOMER VALUES ( 5, 'Mayuresh Phansikar', 'MALE');

INSERT INTO CUSTOMER VALUES ( 6, 'Injila Hussain', 'FEMALE');



SELECT \* FROM CUSTOMER



**PRODUCT**

CREATE TABLE PRODUCT

( PRODUCT\_ID VARCHAR2(10) PRIMARY KEY,

PRODUCT\_NAME VARCHAR2(40) NOT NULL,

PRODUCT\_COST VARCHAR(10) );

INSERT INTO PRODUCT VALUES ( 1, 'Cadbury Dairy Milk Silk', '140');

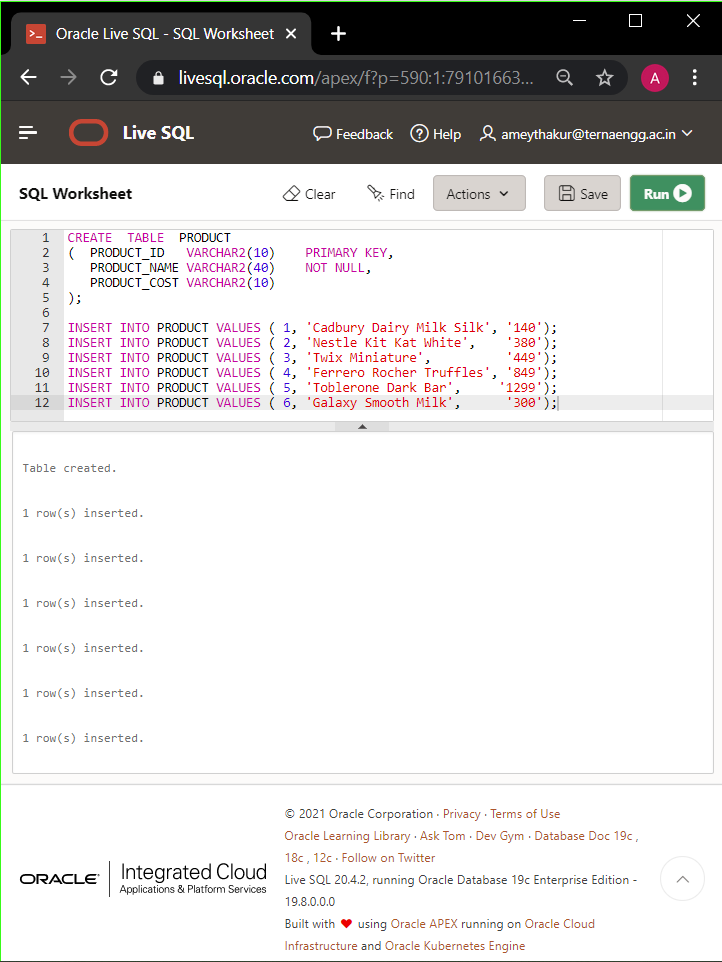
INSERT INTO PRODUCT VALUES ( 2, 'Nestle Kit Kat White', '380');

INSERT INTO PRODUCT VALUES ( 3, 'Twix Miniature', '449');

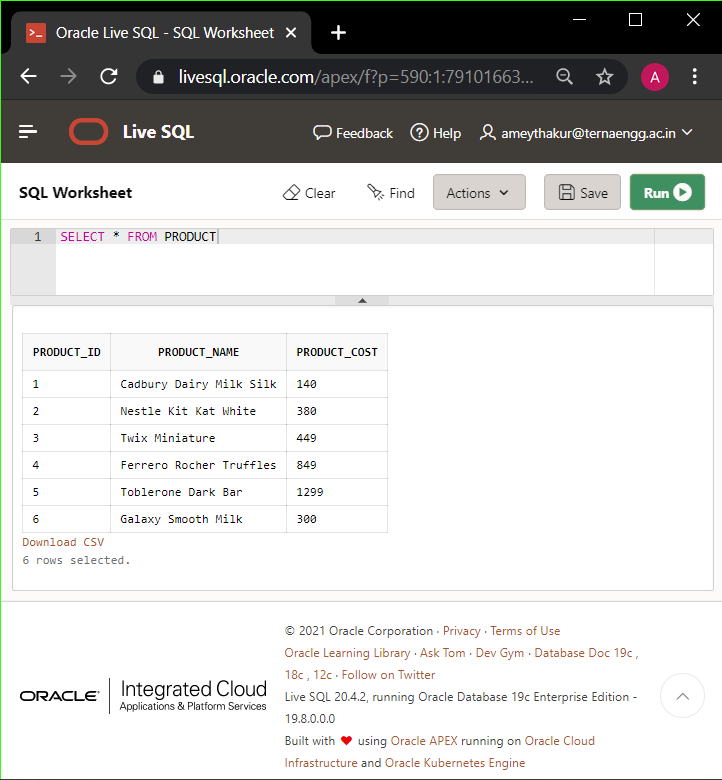
INSERT INTO PRODUCT VALUES ( 4, 'Ferrero Rocher Truffles', '849');

INSERT INTO PRODUCT VALUES ( 5, 'Toblerone Dark Bar', '1299');

INSERT INTO PRODUCT VALUES ( 6, 'Galaxy Smooth Milk', '300');



SELECT \* FROM PRODUCT



**STORE**

CREATE TABLE STORE

( STORE\_ID VARCHAR2(10) PRIMARY KEY,

STORE\_NAME VARCHAR2(40) NOT NULL,

STORE\_LOCATION VARCHAR(40),

STORE\_CITY VARCHAR2(20),

STORE\_STATE varchar2(20) );

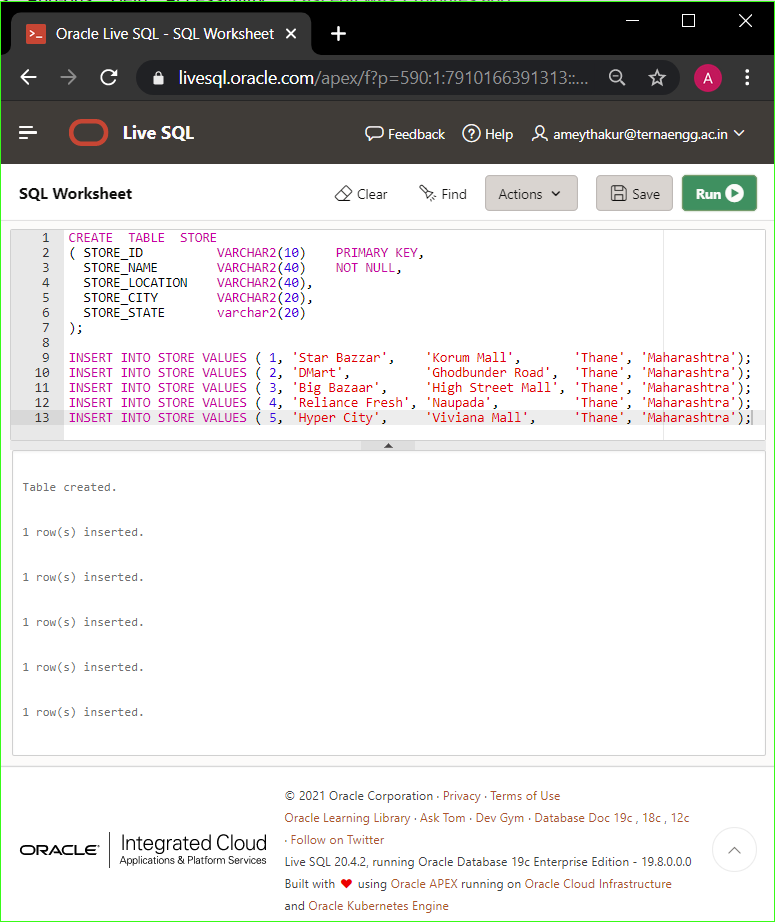
INSERT INTO STORE VALUES ( 1, 'Star Bazaar, 'Korum Mall', 'Thane', 'Maharashtra');

INSERT INTO STORE VALUES ( 2, 'DMart', 'Ghodbunder Road', 'Thane', 'Maharashtra');

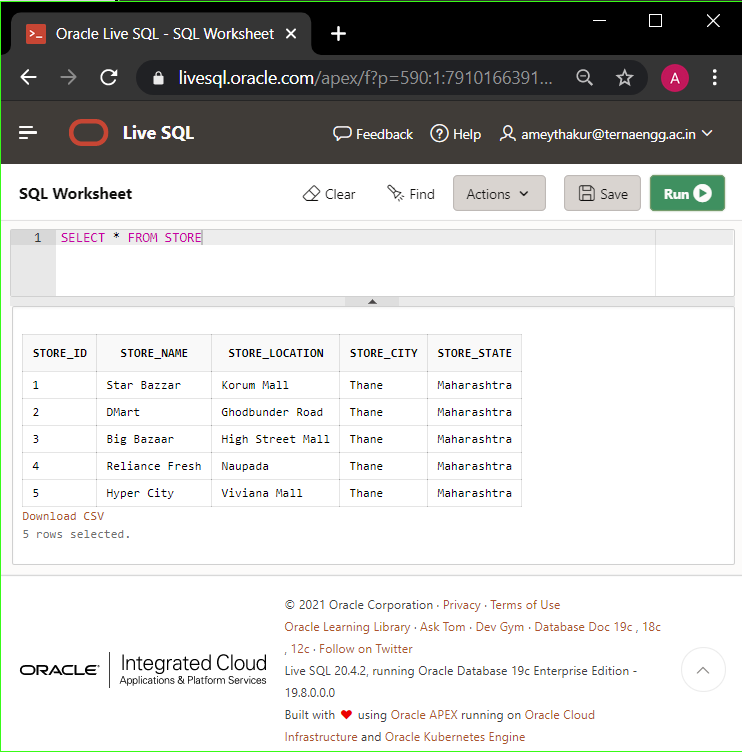
INSERT INTO STORE VALUES ( 3, 'Big Bazaar', 'High Street Mall', 'Thane', 'Maharashtra');

INSERT INTO STORE VALUES ( 4, 'Reliance Fresh', 'Naupada', 'Thane', 'Maharashtra');

INSERT INTO STORE VALUES ( 5,'Hyper City', 'Viviana Mall', 'Thane', 'Maharashtra');



SELECT \* FROM STORE



**SALESPERSON**

CREATE TABLE SALESPERSON

( SALESPERSON\_ID VARCHAR2(10) PRIMARY KEY,

SALESPERSON\_NAME VARCHAR2(40) NOT NULL,

STORE\_ID VARCHAR2(10),

LOCATION VARCHAR(20),

CITY VARCHAR2(10),

STATE VARCHAR2(20) );

INSERT INTO SALESPERSON VALUES ( 1 , 'Supreet', 1, 'Korum Mall', 'Thane', 'Maharahstra');

INSERT INTO SALESPERSON VALUES ( 2 , 'Filly', 1, 'Korum Mall', 'Thane', 'Maharahstra');

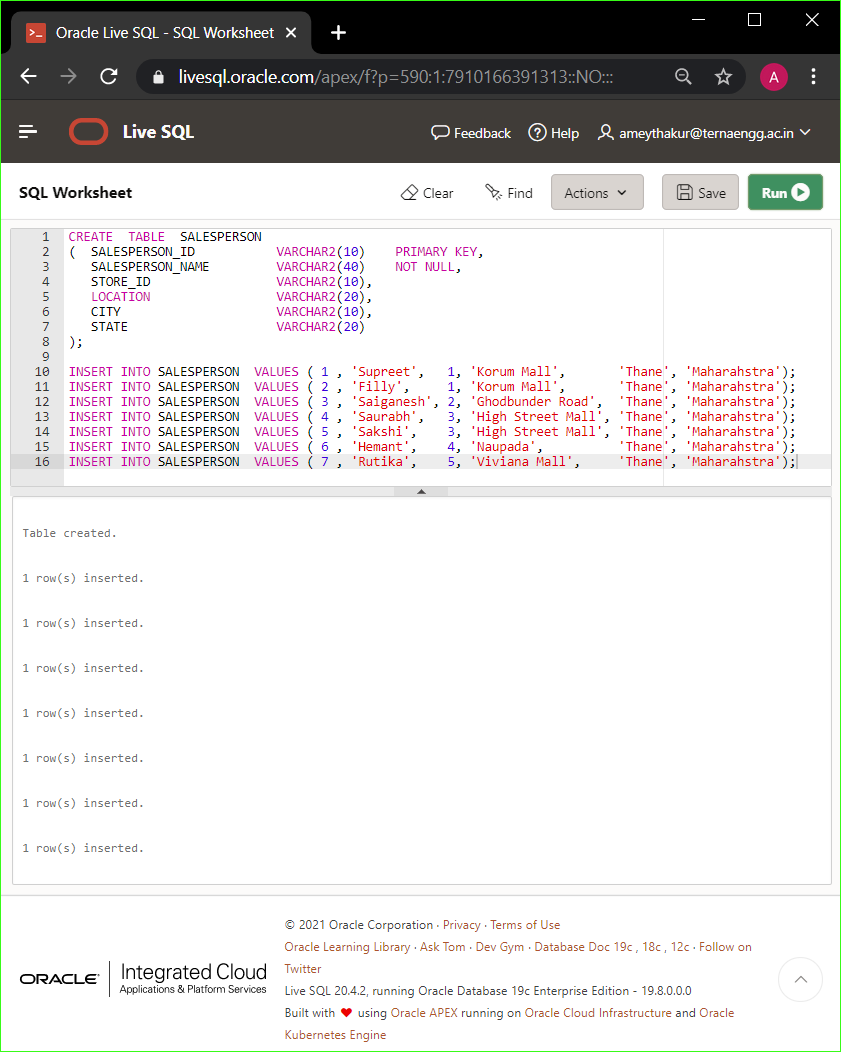
INSERT INTO SALESPERSON VALUES ( 3 , 'Saiganesh', 2, 'Ghodbunder Road', 'Thane', 'Maharahstra');

INSERT INTO SALESPERSON VALUES ( 4 , 'Saurabh', 3, 'High Street Mall', 'Thane', 'Maharahstra');

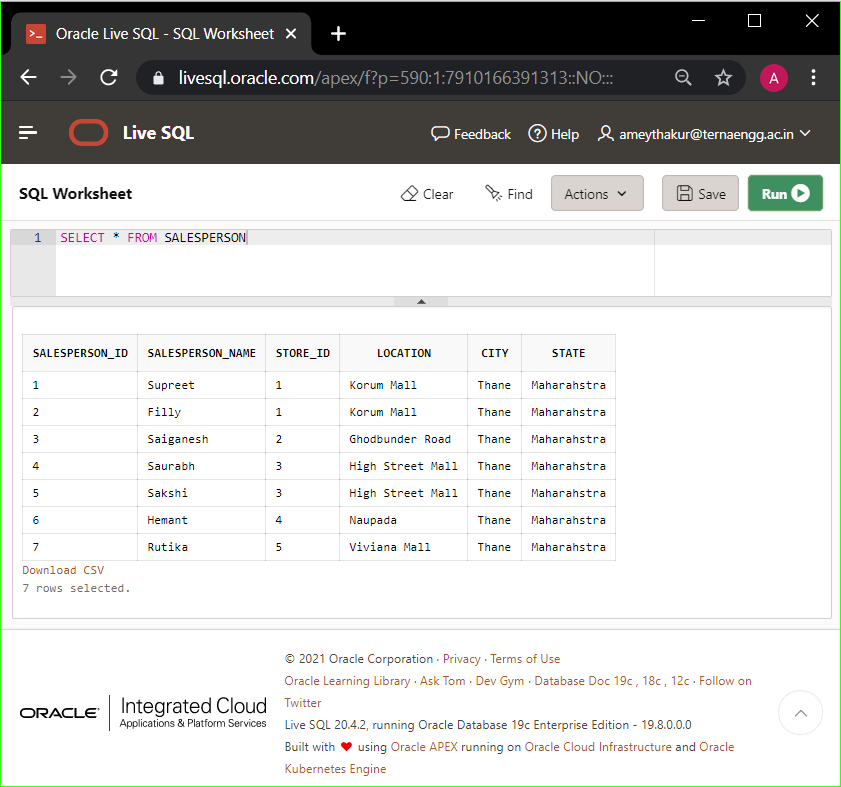
INSERT INTO SALESPERSON VALUES ( 5 , 'Sakshi', 3, 'High Street Mall', 'Thane', 'Maharahstra');

INSERT INTO SALESPERSON VALUES ( 6 , 'Hemant', 4, 'Naupada', 'Thane', 'Maharahstra');

INSERT INTO SALESPERSON VALUES ( 7 , 'Rutika', 5, 'Viviana Mall', 'Thane', 'Maharahstra');



SELECT \* FROM SALESPERSON



**Fact Table**

CREATE TABLE PRODUCTSALES

( TRANSACTION\_ID VARCHAR2(20) PRIMARY KEY,

INVOICE\_NO VARCHAR2(10),

TOTAL\_AMOUNT VARCHAR2(10),

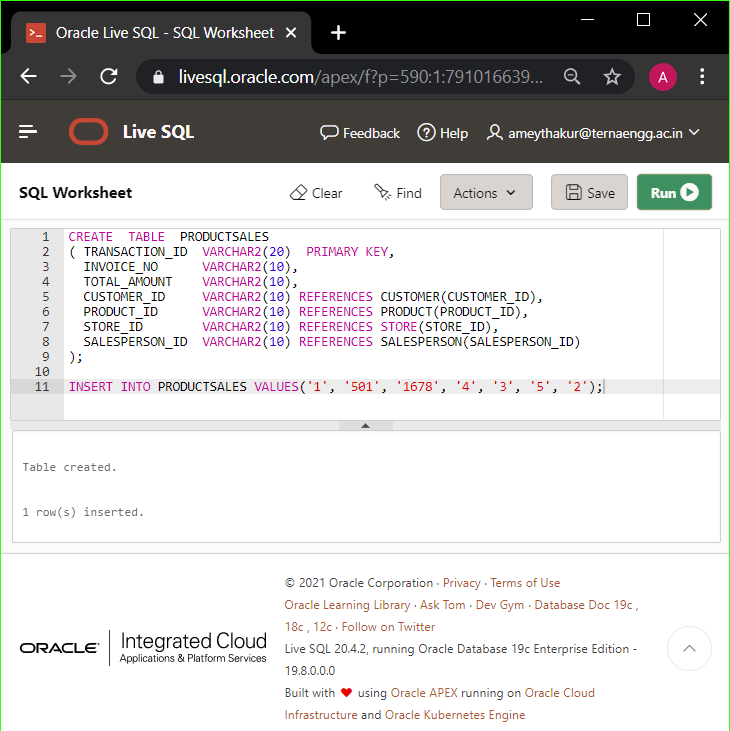
CUSTOMER\_ID VARCHAR2(10) REFERENCES CUSTOMER(CUSTOMER\_ID),

PRODUCT\_ID VARCHAR2(10) REFERENCES PRODUCT(PRODUCT\_ID),

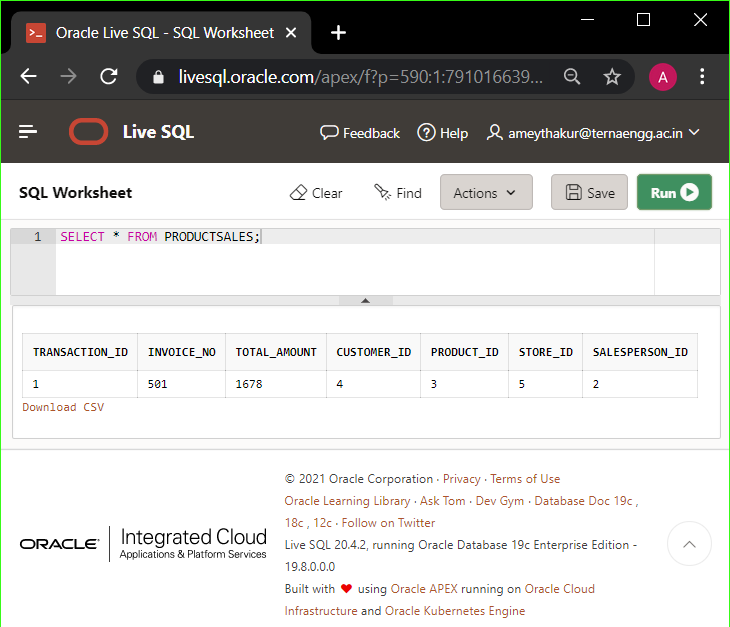
STORE\_ID VARCHAR2(10) REFERENCES STORE(STORE\_ID),

SALESPERSON\_ID VARCHAR2(10) REFERENCES SALESPERSON(SALESPERSON\_ID) );

INSERT INTO PRODUCTSALES VALUES('1', '501', '1678', '4', '2,3,4', '5', '2');



SELECT \* FROM PRODUCTSALES;



**B.2 Input and Output:**

***(Paste your program input and output in the following format, If there is an error then paste the specific error in the output part. In case of an error with the due permission of the faculty, an extension can be given to submit the error-free code with output in due course of time. Students will be graded accordingly.)***

**Note:** Input and Output are mentioned in section B.1

**Input :**

**SQL commands/script which satisfies Two different outcomes mentioned in Problem statements**.

**Output:**

1. **Dimensional Tables created after firing the above SQL commands.**
2. **The output satisfies 2 different outcomes mentioned in Problem statements**.

**B.3 Observations and learning:**

***(Students are expected to comment on the output obtained with clear observations and learning for each task/ subpart assigned)***

A dimensional model is a data structure technique optimized for Data warehousing tools. The concept of Dimensional Modelling is comprised of "fact" and "dimension" tables. These dimensional and relational models have their unique way of data storage that has specific advantages.

**B.4 Conclusion:**

*(****Students must write the conclusion as per the attainment of individual outcome listed above and learning/observation noted in section B.3)***

The concept of a fact table and dimension table was implemented using dimension modelling.

**B.5 Question of Curiosity**

***(To be answered by the student based on the practical performed and learning/observations)***

Q1: What are the differences between the Dimension table and the fact table?

Ans:

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Fact Table** | **Dimension Table** |
| **Definition** | Measurements, metrics or facts about a business process. | The companion table to the fact table contains descriptive attributes to be used as query constraining. |
| **Characteristic** | Located at the centre of a star or snowflake schema and surrounded by dimensions. | Connected to the fact table and located at the edges of the star or snowflake schema |
| **Design** | Defined by their grain or its most atomic level. | Should be wordy, descriptive, complete, and quality assured. |
| **Task** | A fact table is a measurable event for which dimension table data is collected and is used for analysis and reporting. | Collection of reference information about a business. |
| **Type of Data** | Facts tables could contain information like sales against a set of dimensions like Product and Date. | Evert dimension table contains attributes that describe the details of the dimension. E.g., Product dimensions can contain Product ID, Product Category, etc. |
| **Key** | The primary Key in the fact table is mapped as foreign keys to Dimensions. | The dimension table has primary key columns that uniquely identifies each dimension. |
| **Storage** | Helps to store report labels and filter domain values in dimension tables. | Load detailed atomic data into dimensional structures. |
| **Hierarchy** | Does not contain Hierarchy | Contains Hierarchies. For example, Location could contain, country, pin code, state, city, etc. |

Q2: Explain Primary Keys, Surrogate Keys & Foreign Keys with an example.

Ans:

**Primary key:**

A column or group of columns in a table that helps us to uniquely identifies every row in that table is called a primary key. This DBMS can't be a duplicate. The same value can't appear more than once in the table.

**Surrogate key:**

An artificial key that aims to uniquely identify each record is called a surrogate key. These kinds of keys are unique because they are created when you don't have any natural primary key. They do not lend any meaning to the data in the table. The surrogate key is usually an integer.

**Foreign key:**

A foreign key is a column that is added to create a relationship with another table. Foreign keys help us to maintain data integrity and also allows navigation between two different instances of an entity. Every relationship in the model needs to be supported by a foreign key.