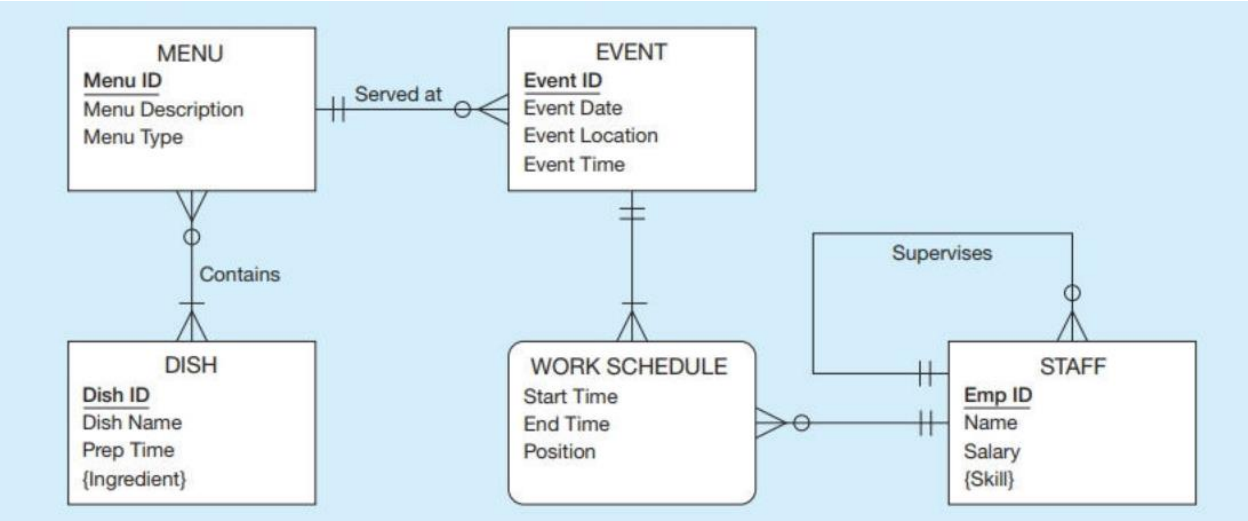
Homework 4

Your Name: Naga Satya Silpa Annadevara

Student ID: A20517818

Undergraduate (Yes/No): NO

Q1. Adding data types to the attributes in the figure below. Note, you do not need to draw ERD, you can simply give a list with table name and attribute names, along with the data types in Oracle. [10]



ANSWER:

MENU:

ATTRIBUTE NAME	DATA TYPE
Menu ID	INT or NUMBER **(Not Null)**
Menu Description	VARCHAR2 (100)
Menu Type	VARCHAR2 (50)

EVENT:

ATTRIBUTE NAME	DATA TYPE
Event ID	INT or NUMBER (Not Null)
Event Date	DATE
Event Location	VARCHAR2 (100)
Event Time	TIMESTAMP

DISH:

ATTRIBUTE NAME	DATA TYPE
Dish ID	INT or NUMBER (Not Null)
Dish Name	VARCHAR2 (50)
Prep Time	TIMESTAMP
Ingredient	VARCHAR2 (50)

WORK SCHEDULE:

ATTRIBUTE NAME	DATA TYPE
Start Time	TIMESTAMP
End Time	TIMESTAMP
Position	VARCHAR2 (20)

STAFF:

ATTRIBUTE NAME	DATA TYPE
Emp ID	NUMBER (Not Null)
Emp Name	VARCHAR2 (40)
Salary	DECIMAL or FLOAT or NUMBER
Skill	VARCHAR2 (50)

Q2. Below is the design for the case study of hotel-rooms-booking discussed in the class. [40]

System/Application Requirements

- We want to build hotel room booking system.
- Assume there are several hotels in Chicago.
- Each hotel only has two types of the rooms – Standard Room with 2 beds, Standard Room with 1 big bed
- Customer can book rooms from our website. But the availability may vary from hotels to hotels

Our design:

- Hotel (HID, Name, Address, Cell, …)
- RoomType (TypeID, NumBeds, RoomSize, …)
- HotelRooms (RoomID, HID, TypeID, Quantity)
- Customer (CID, Name, Address, Email, …)
- Order (OrderID, CID, CheckinDate, CheckoutDate)
- RoomsInOrder (OrderID, RoomID, RoomQuantity)

a). Identify the PK and FK in each entity [10]

ANSWER:

1. Hotel (HID, Name, Address, Cell, …)

- Primary key: HID
- Foreign key: None

2. Room Type (TypeID, NumBeds, RoomSize, …)

- Primary key: Type ID
- Foreign key: None

3. HotelRooms (RoomID, HID, TypeID, Quantity)

- Primary key: Room ID
- Foreign key: HID, Type ID

4. Customer (CID, Name, Address, Email, …)

- Primary key: CID
- Foreign key: None

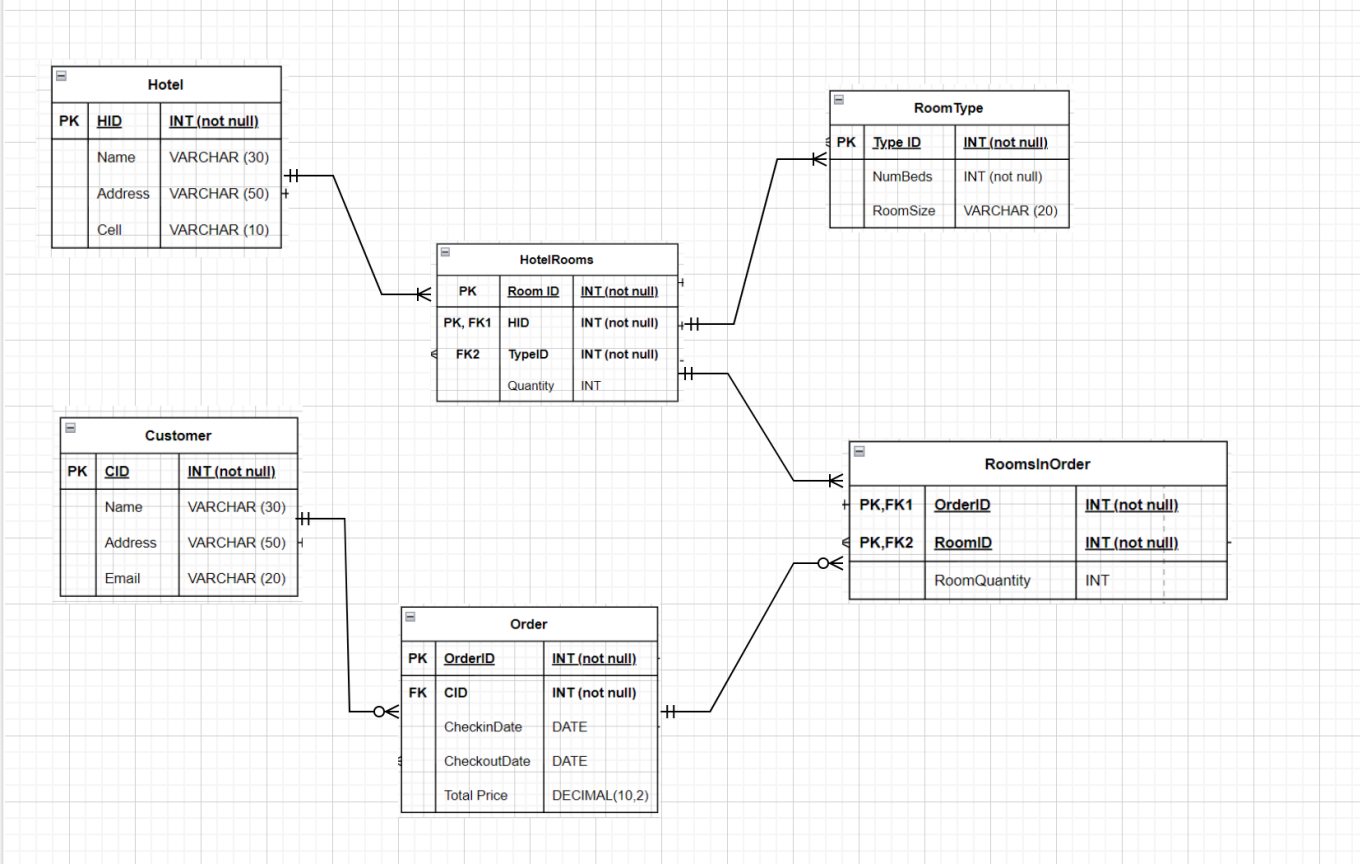
5. Order (OrderID, CID, CheckinDate, CheckoutDate)

- Primary key: Order ID
- Foreign key: CID

6. RoomsInOrder (OrderID, RoomID, RoomQuantity)

- Primary key: Order ID + Room ID
  Foreign key: Order ID + Room ID

b). Draw ERD for physical DB design [10]



NOTE: The ID's Can use the datatype NUMBER as well

c). Now, we want to add "price". Note that there would be unit price for the rooms, as well as the price in the orders. Special notes: hotels may release difference prices for different time periods. Complete your design in BCNF by showing your physical design by ERD [20]

ANSWER:

As the price is introduced into the design, we have to add 3 more NEW entities with attributes as follows:

- HotelRoom Price (PriceID(pk) , RoomID(pk&fk) , Unit Price)
- Ordered Price (RoomsInOrderID(pk,fk) ,PriceID(pk,fk), Total Price)
- Periodical Price (PeriodID (pk) ,TypeID (fk), RoomID(fk), Current Price)

  (As mentioned in the question that hotels may release difference prices for different time periods)

In addition to the above entities, we need to add attributes to the old entities according to the question as follows:

- adding TotalPrice in Order Entity.

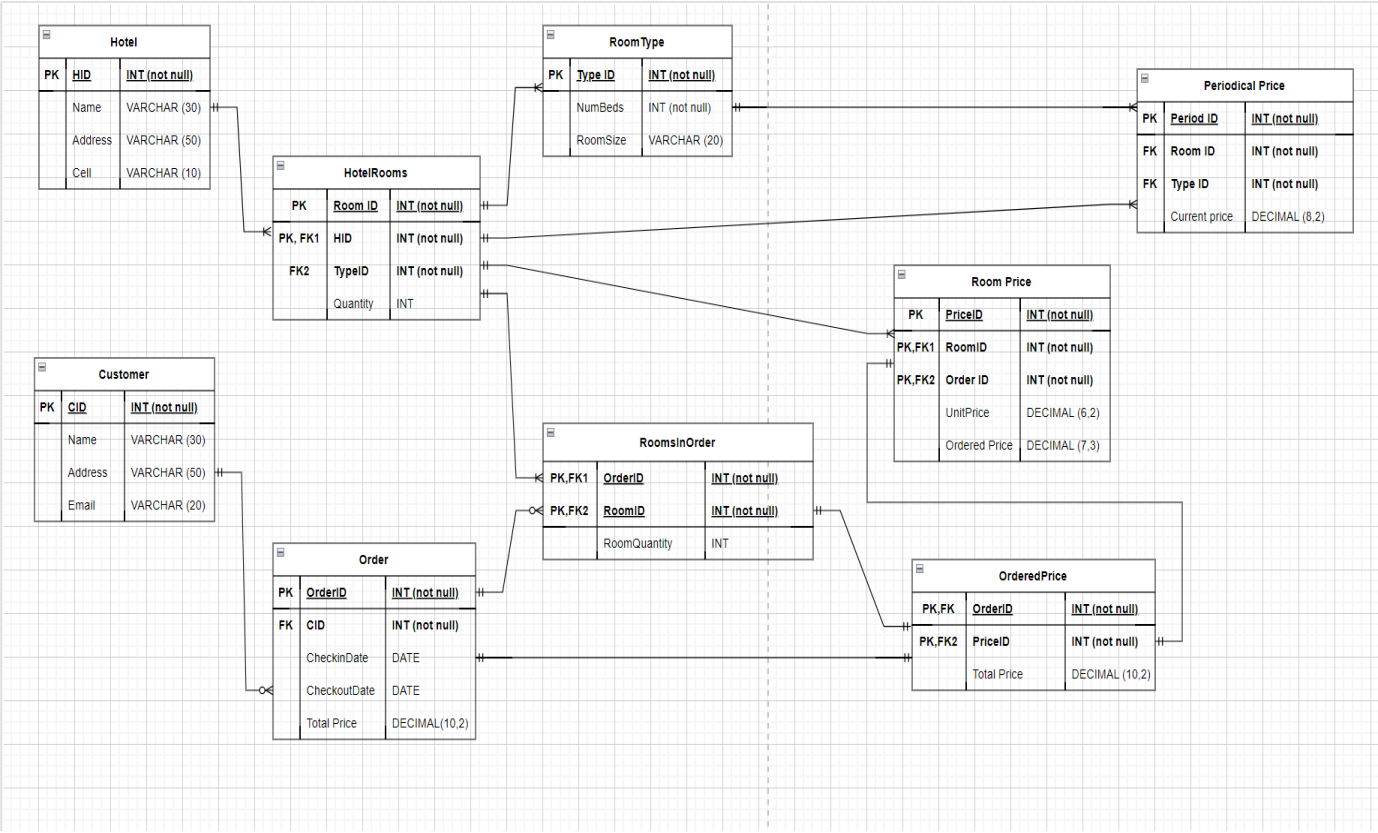
- Unit Price to the HotelRooms entity (this attribute will be moved to the new entity called HotelroomsPrice after normalization as shown in the figure below)

Hence, The final list of entities we have after adding price to the design and after normalization to BCNF are :

1. Hotel
2. HotelRooms
3. Room Type
4. Customer
5. Order
6. RoomsInOrder
7. HotelRoom Price
8. Ordered Price

   Periodical Price (As mentioned in the question that hotels may release difference prices for different time periods)

BCNF in Physical design by ERD:

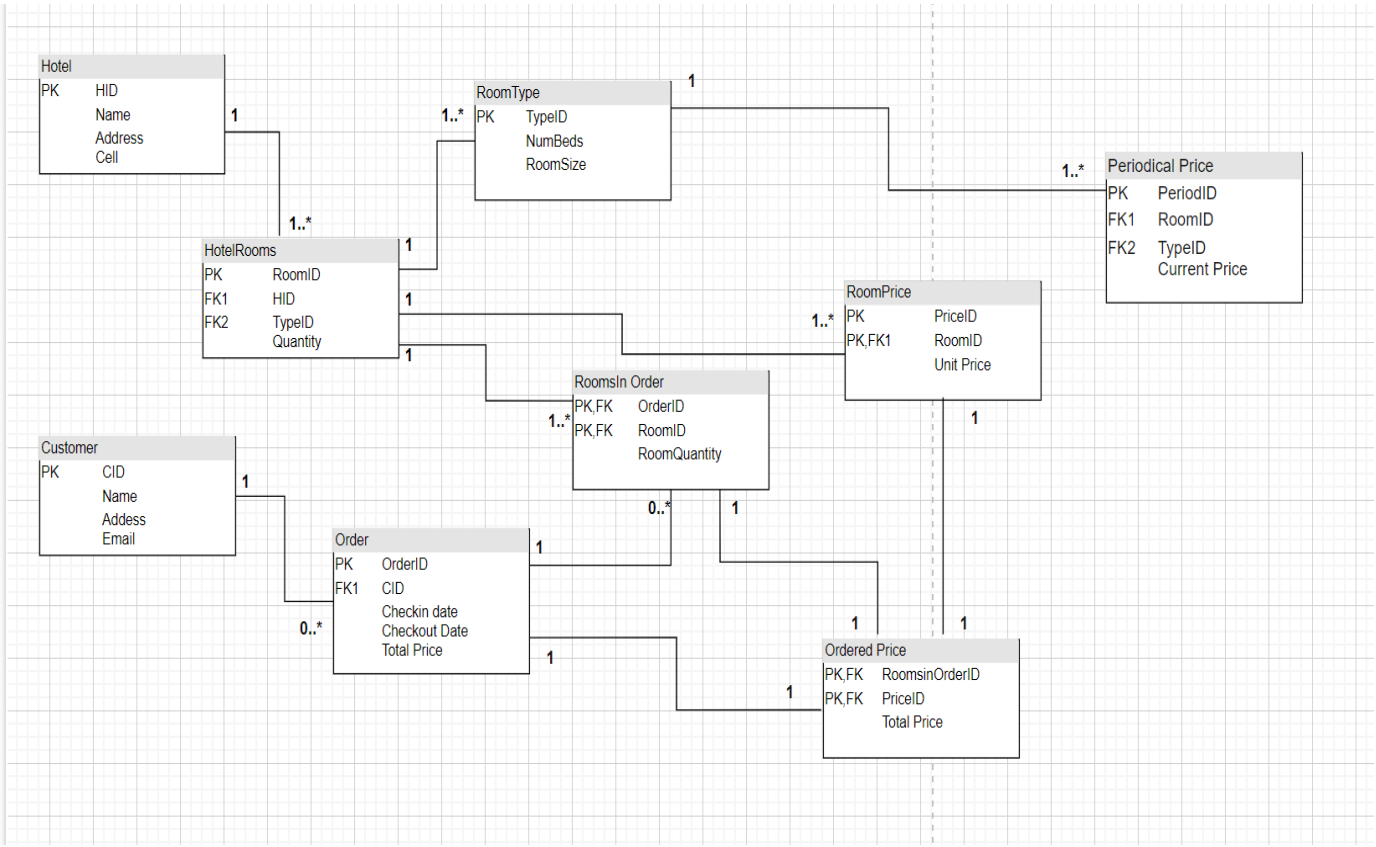


NOTE: The ID's Can use the datatype NUMBER as well

Q3. [10] Draw the final design in Q3 by using UML.

ANSWER:

Final design in UML:



Q4. [40] Using DDL and DML for the following.

Download the W5_MovieRatings.zip file, unzip it. You will find four csv files, each of which is an entity/table. Therefore, there are 4 tables: Users, Movies, Ratings, Tags.

[4] Identify PK and FK in these tables. Use * to indicate PK, and ___ to tell FK. For example, Student (StudentID*, DeptID, Firstname, Lastname)

[8] Use create statements to create these tables in Oracle, show your SQL coding

[8] Use insert statements to insert values in these tables, show your SQL coding

[4] Add a column "TagDate" in the table Tags. After that, insert date values for the current entries in table Tags. Note, you can use any date values, show your SQL coding.

[8] Drop table movies, show your SQL coding.

[4] In table Ratings, change the data type of ratings to Number (2, 1) , show your SQL coding

[4] Assume that users can only give ratings, like, 1, 2, 3, 4, 5. In table Ratings, add a check constraint to guarantee that the input values for the column rating must be a value from (1, 2, 3, 4, 5)

ANSWER:

1. Users Table (UserID, Email, BirthYear, City)

- Primary key: UserID
- Foreign key: None

2. Movies Table (MovieID, Title)

- Primary key: Movie ID
- Foreign key: None

3. Ratings (UserID, MovieID, Rating)

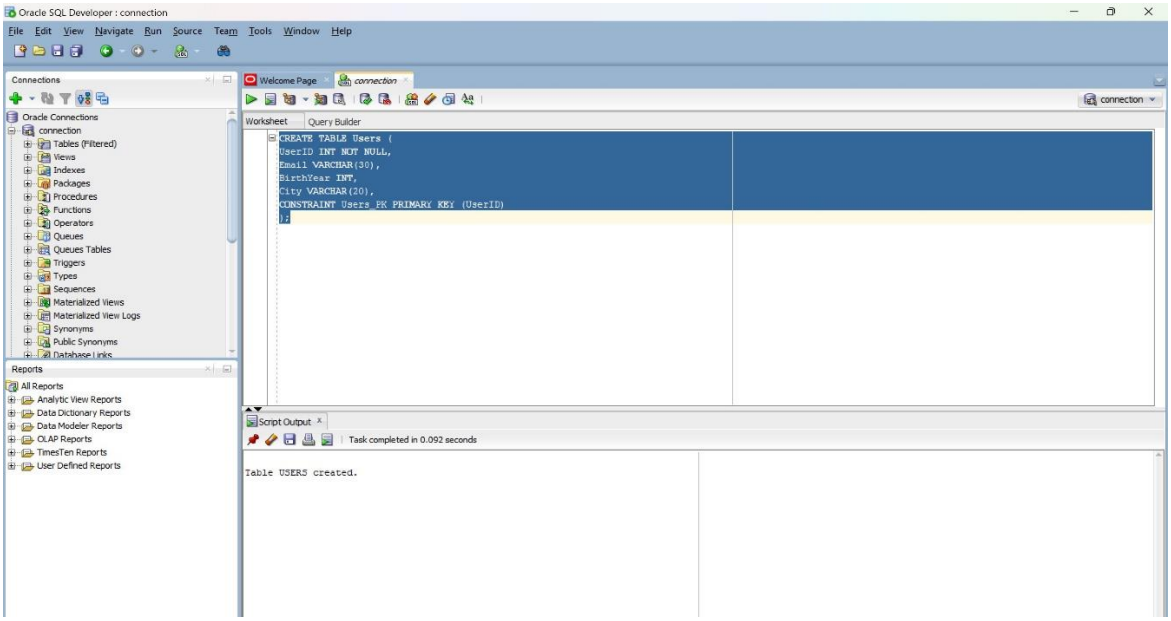
- Primary key: ratings + Movie ID +UserID
- Foreign key: UserID, Movie ID

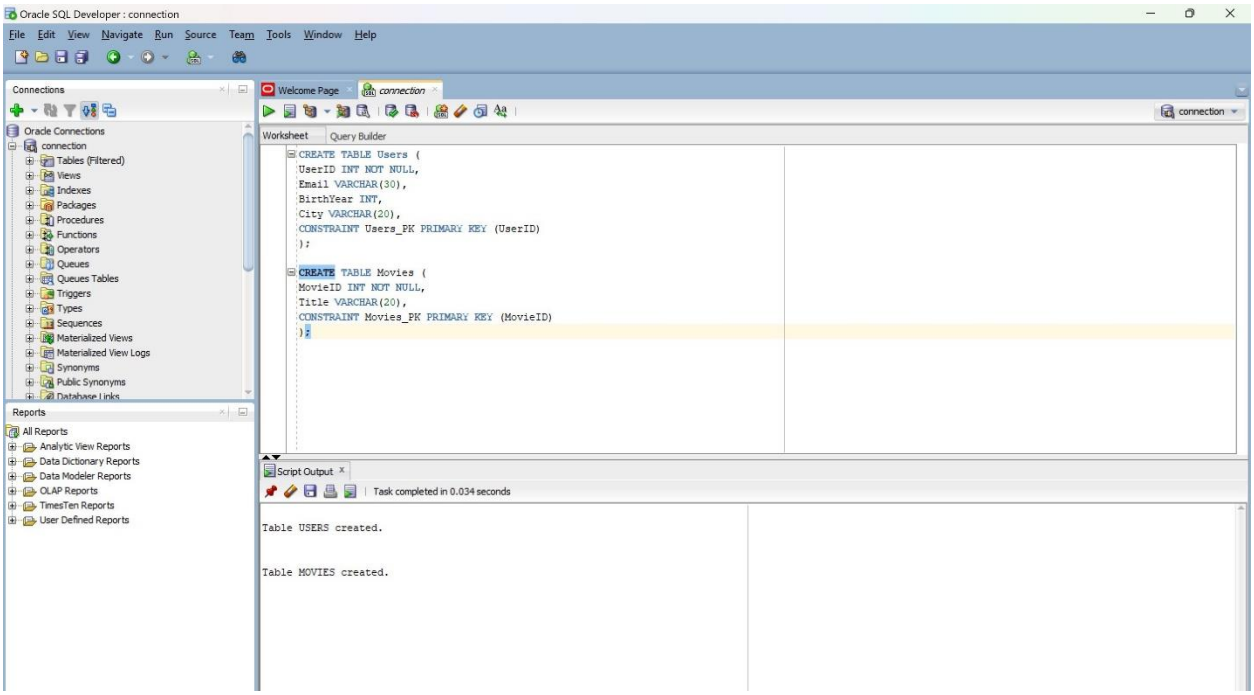
4. Tags (UserID, MovieID, Tag)

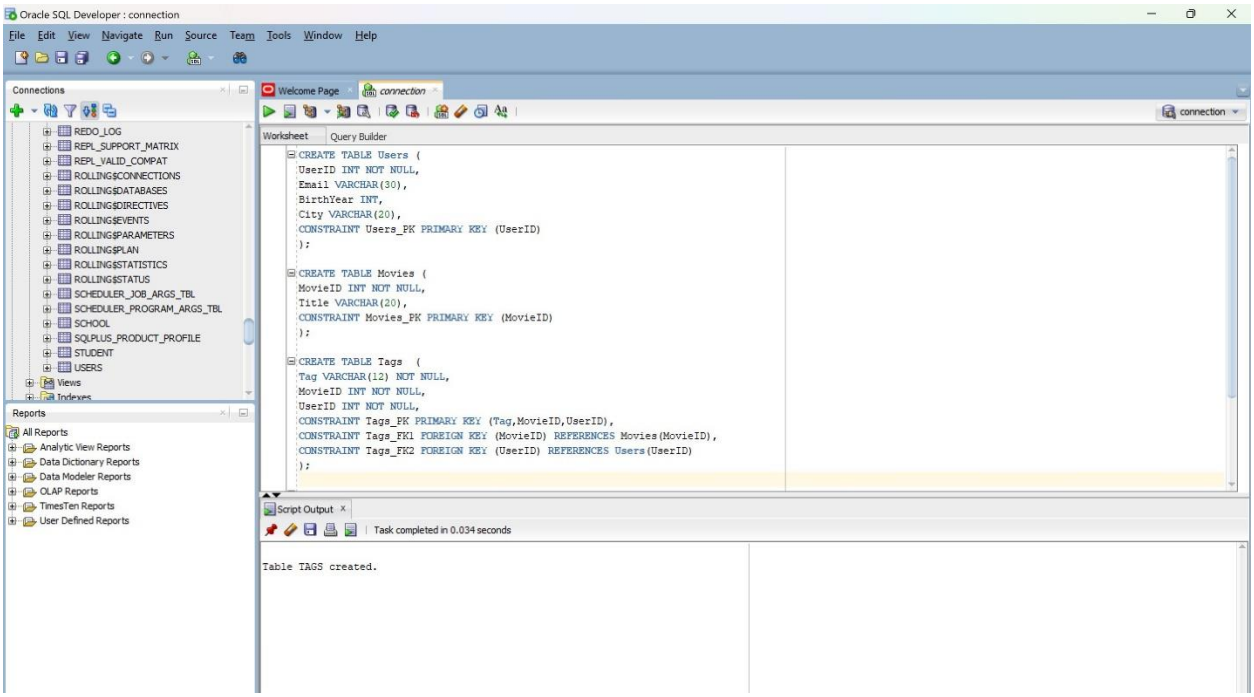
- Primary key: Tag + MovieID + UserID
- Foreign key: UserID, Movie ID

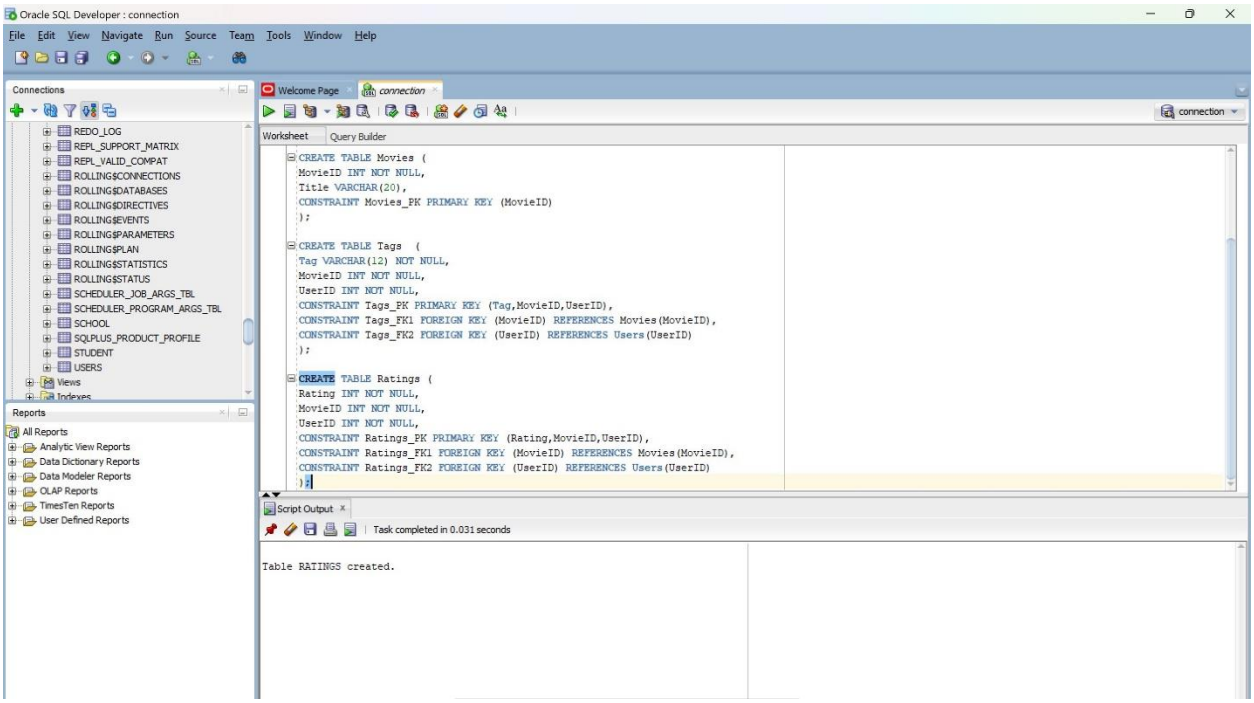
Oracle ScreenShots (SQL Coding):

Creating Tables (Users, Movies, Ratings, Tags) :

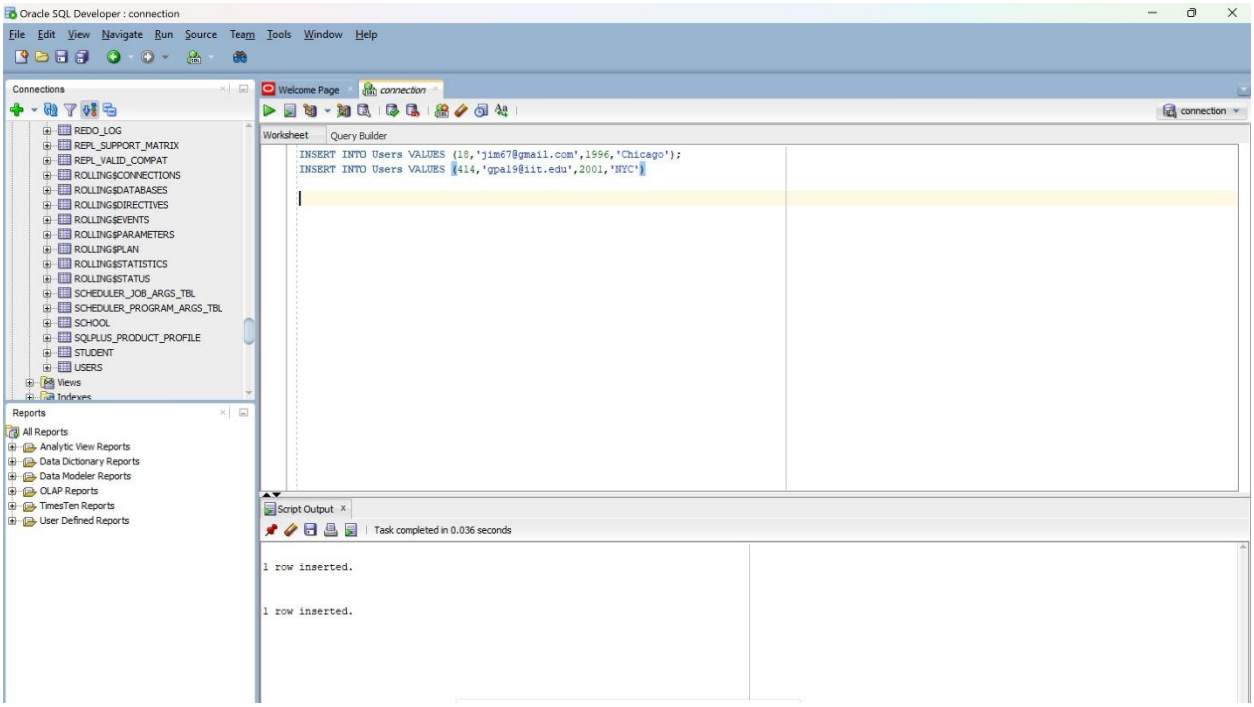


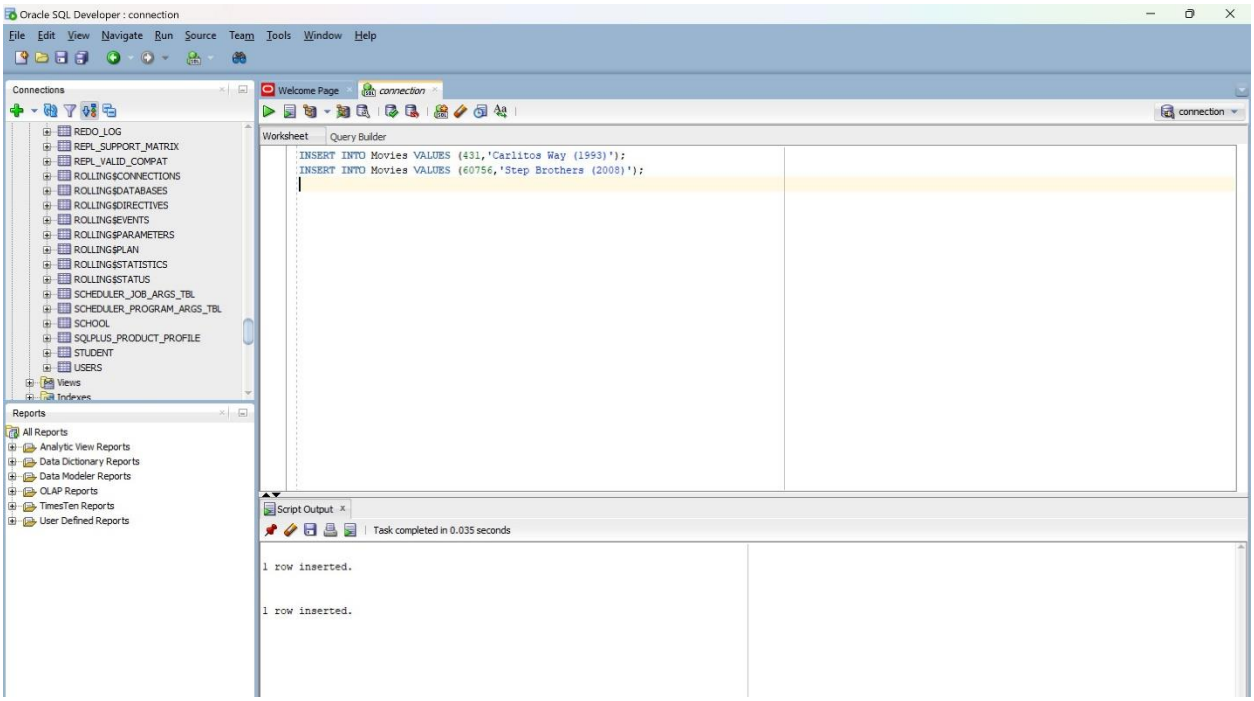

Oracle SQL Developer : connection
File Edit View Navigate Run Source Team Tools Window Help
Connections
Oracle Connections
connection
Tables (Filtered)
Views
Indexes
Packages
Procedures
Functions
Operators
Queues
Queues Tables
Triggers
Types
Sequences
Materialized Views
Materialized View Logs
Synonyms
Public Synonyms
Database Links
Reports
All Reports
Analytic View Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Welcome Page
connection
Worksheet
Query Builder
CREATE TABLE Users (
UserID INT NOT NULL,
Email VARCHAR(30),
BirthYear INT,
City VARCHAR(20),
CONSTRAINT Users_PK PRIMARY KEY (UserID)
);
CREATE TABLE Movies (
MovieID INT NOT NULL,
Title VARCHAR(20),
CONSTRAINT Movies_PK PRIMARY KEY (MovieID)
);
Script Output
Task completed in 0.034 seconds
Table USERS created.
Table MOVIES created.

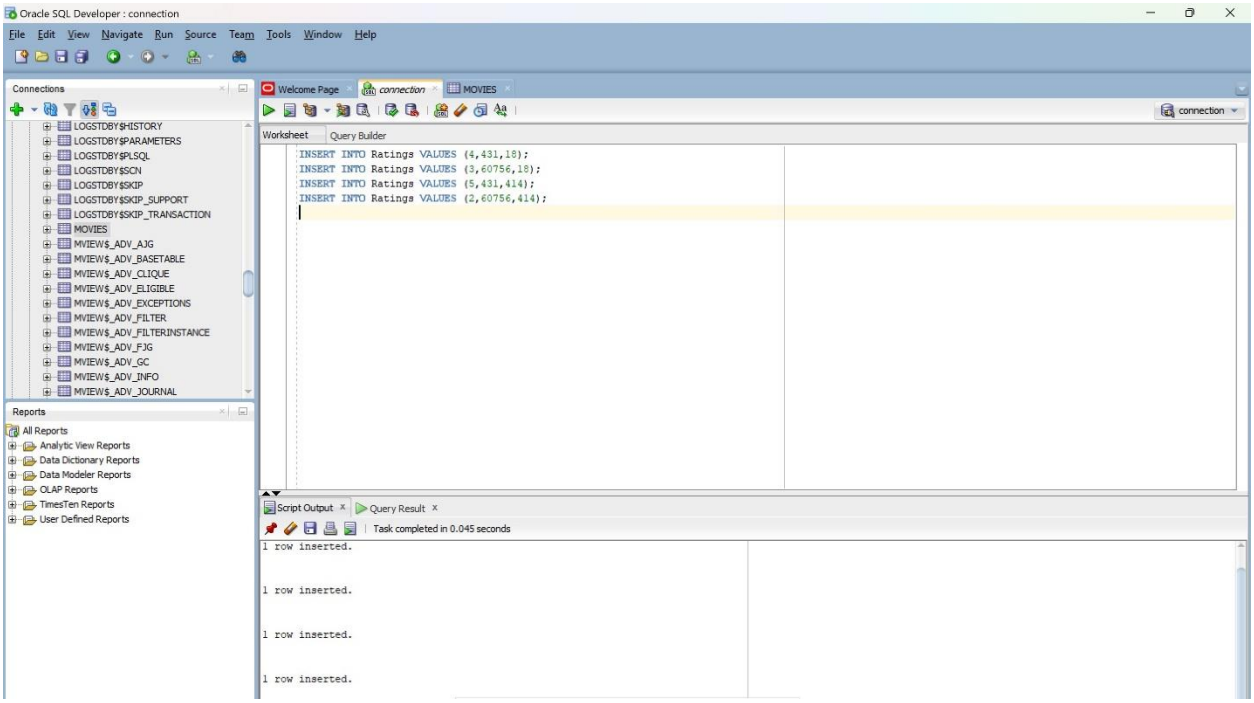

Oracle SQL Developer : connection
File Edit View Navigate Run Source Team Tools Window Help
Connections
REDO_LOG
REPL_SUPPORT_MATRIX
REPL_VALID_COMPAT
ROLLING$CONNECTIONS
ROLLING$DATABASES
ROLLING$DIRECTIVES
ROLLING$EVENTS
ROLLING$PARAMETERS
ROLLING$PLAN
ROLLING$STATISTICS
ROLLING$STATUS
SCHEDULER_JOB_ARGS_TBL
SCHEDULER_PROGRAM_ARGS_TBL
SCHOOL
SQLPLUS_PRODUCT_PROFILE
STUDENT
USERS
Views
Indexes
Reports
All Reports
Analytic View Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Welcome Page
connection
Worksheet
Query Builder
CREATE TABLE Users (
UserID INT NOT NULL,
Email VARCHAR(30),
BirthYear INT,
City VARCHAR(20),
CONSTRAINT Users_PK PRIMARY KEY (UserID)
);
CREATE TABLE Movies (
MovieID INT NOT NULL,
Title VARCHAR(20),
CONSTRAINT Movies_PK PRIMARY KEY (MovieID)
);
CREATE TABLE Tags (
Tag VARCHAR(12) NOT NULL,
MovieID INT NOT NULL,
UserID INT NOT NULL,
CONSTRAINT Tags_PK PRIMARY KEY (Tag,MovieID,UserID),
CONSTRAINT Tags_FK1 FOREIGN KEY (MovieID) REFERENCES Movies(MovieID),
CONSTRAINT Tags_FK2 FOREIGN KEY (UserID) REFERENCES Users(UserID)
);
Script Output
Task completed in 0.034 seconds
Table TAGS created.

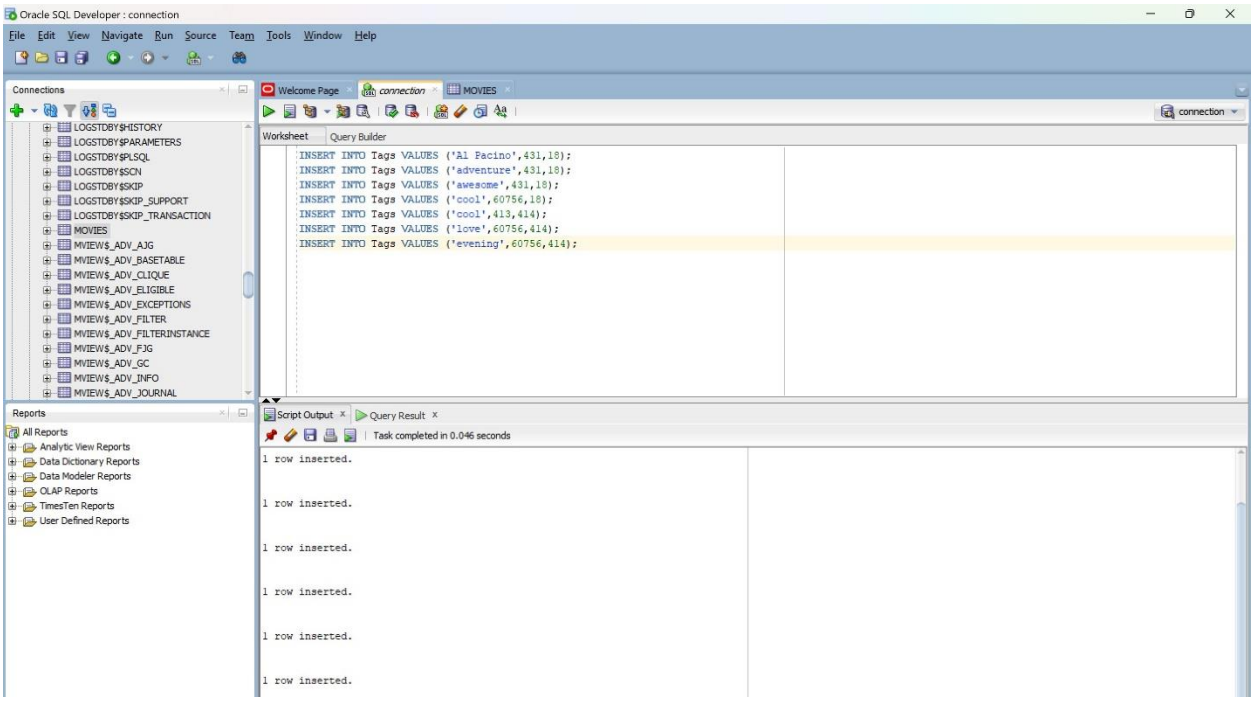


Inserting Values into the Tables:

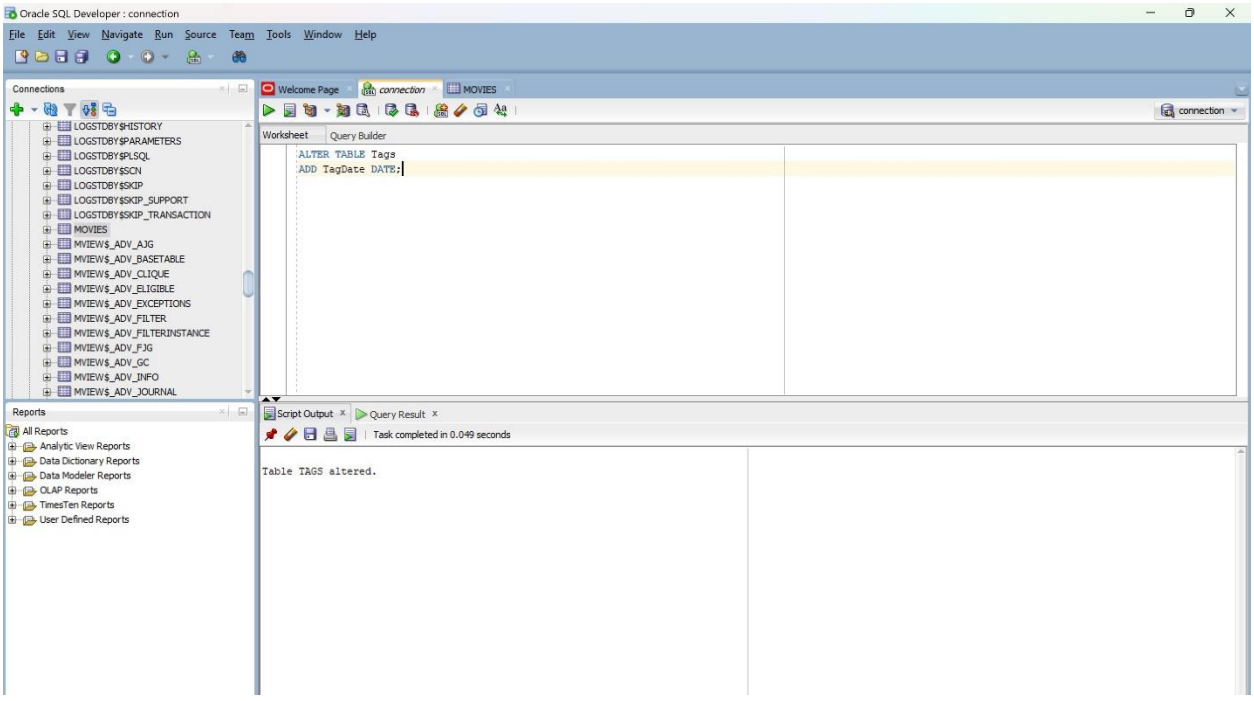


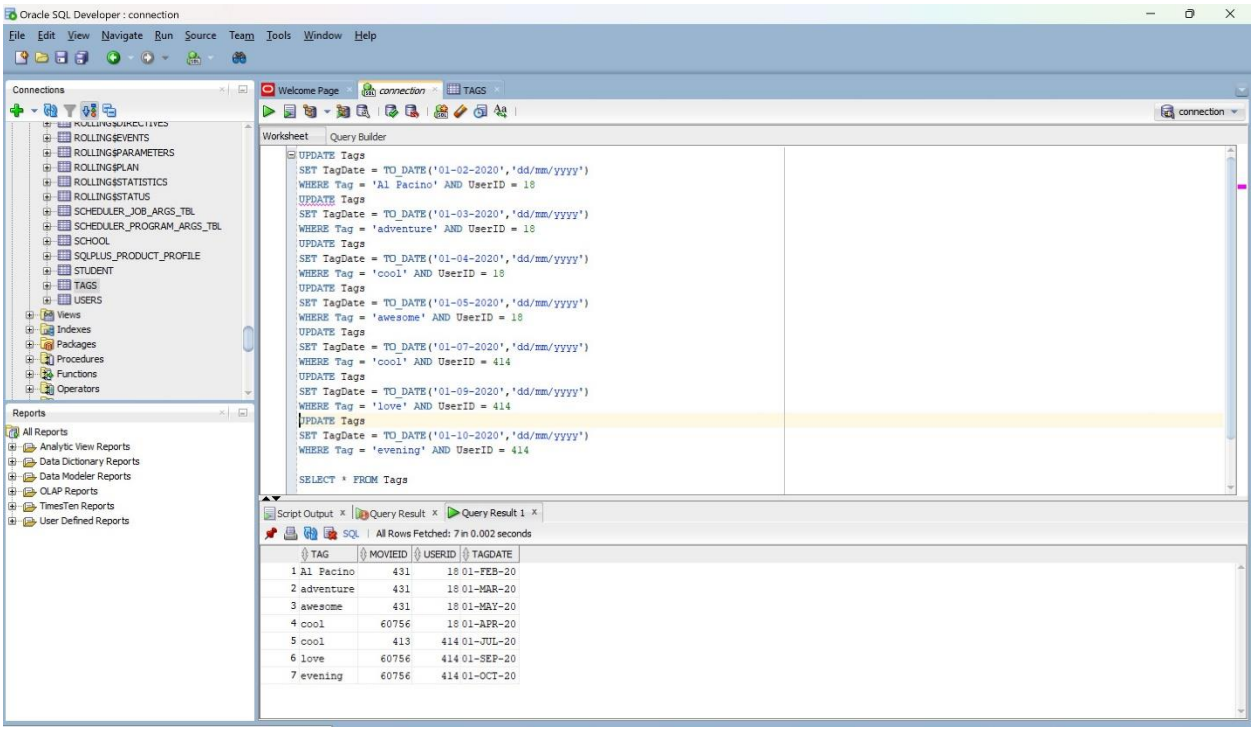

Oracle SQL Developer : connection
File Edit View Navigate Run Source Team Tools Window Help
Connections
REDO_LOG
REPL_SUPPORT_MATRIX
REPL_VALID_COMPAT
ROLLING$CONNECTIONS
ROLLING$DATABASES
ROLLING$DIRECTIVES
ROLLING$EVENTS
ROLLING$PARAMETERS
ROLLING$PLAN
ROLLING$STATISTICS
ROLLING$STATUS
SCHEDULER_JOB_ARGS_TBL
SCHEDULER_PROGRAM_ARGS_TBL
SCHOOL
SQLPLUS_PRODUCT_PROFILE
STUDENT
USERS
Views
Indexes
Reports
All Reports
Analytic View Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Welcome Page
connection
Worksheet
Query Builder
INSERT INTO Movies VALUES (431,'Carlitos Way (1993)');
INSERT INTO Movies VALUES (60756,'Step Brothers (2008)');
Script Output
Task completed in 0.035 seconds
1 row inserted.
1 row inserted.


Oracle SQL Developer : connection
File Edit View Navigate Run Source Team Tools Window Help
Connections
LOGSTDBY$HISTORY
LOGSTDBY$PARAMETERS
LOGSTDBY$PLSQL
LOGSTDBY$SCN
LOGSTDBY$SKIP
LOGSTDBY$SKIP_SUPPORT
LOGSTDBY$SKIP_TRANSACTION
MOVIES
MVIEW$_ADV_AJG
MVIEW$_ADV_BASETABLE
MVIEW$_ADV_CLIQUE
MVIEW$_ADV_ELIGIBLE
MVIEW$_ADV_EXCEPTIONS
MVIEW$_ADV_FILTER
MVIEW$_ADV_FILTERINSTANCE
MVIEW$_ADV_FJG
MVIEW$_ADV_GC
MVIEW$_ADV_INFO
MVIEW$_ADV_JOURNAL
Reports
All Reports
Analytic View Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Welcome Page
connection
MOVIES
Worksheet
Query Builder
INSERT INTO Ratings VALUES (4,431,18);
INSERT INTO Ratings VALUES (3,60756,18);
INSERT INTO Ratings VALUES (5,431,414);
INSERT INTO Ratings VALUES (2,60756,414);
Script Output
Query Result
Task completed in 0.045 seconds
1 row inserted.
1 row inserted.
1 row inserted.
1 row inserted.

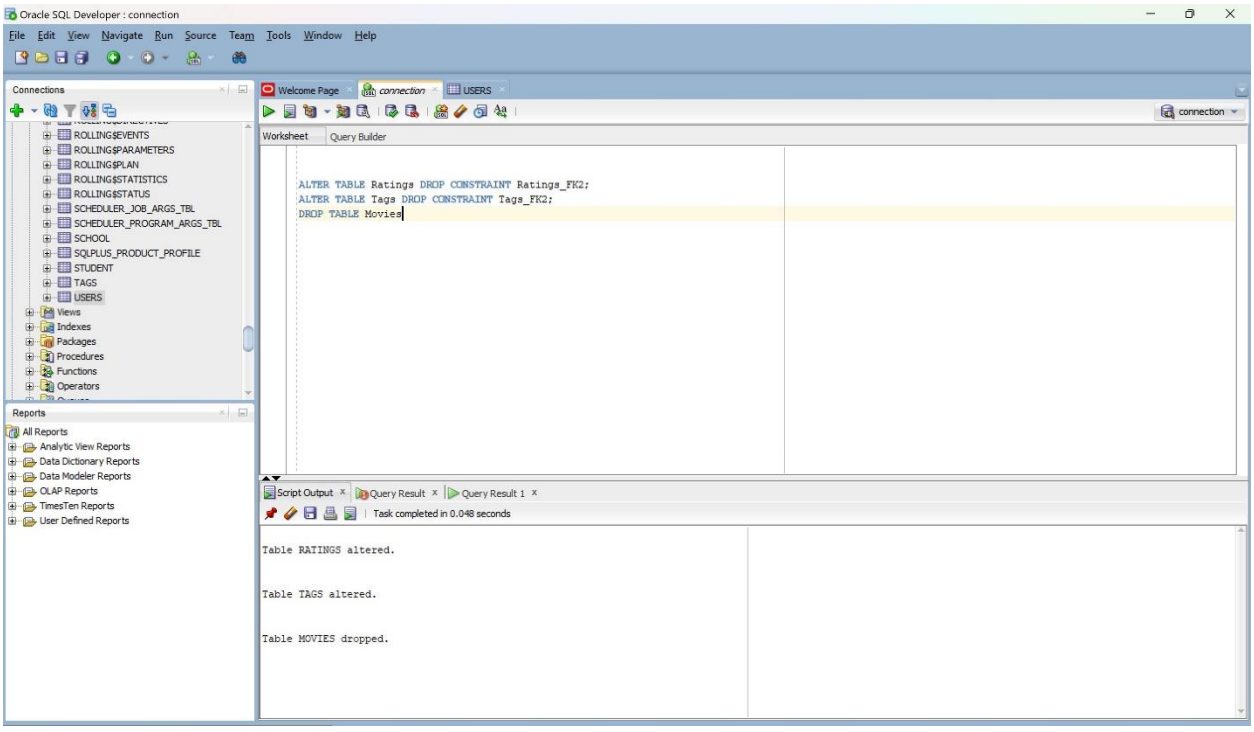


Add a column “TagDate” in the table Tags. After that, insert date values for the current entries in table Tags. Note, you can use any date values, show your SQL coding.

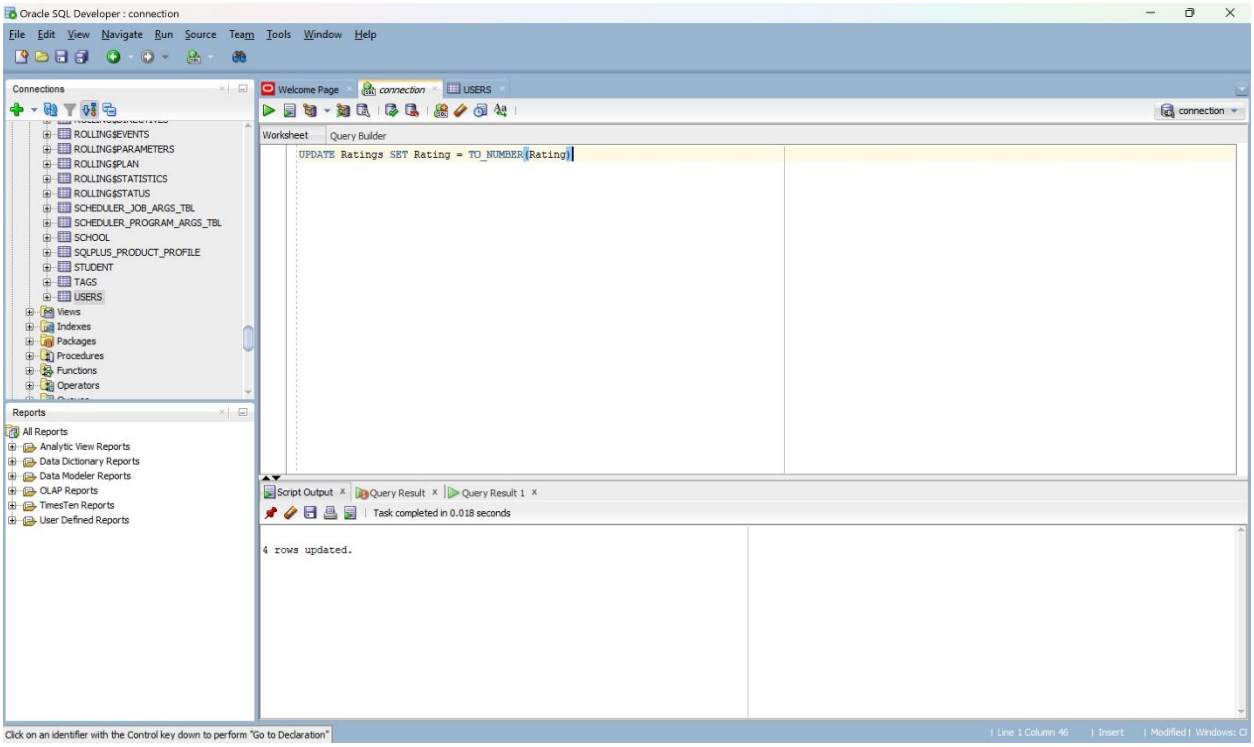


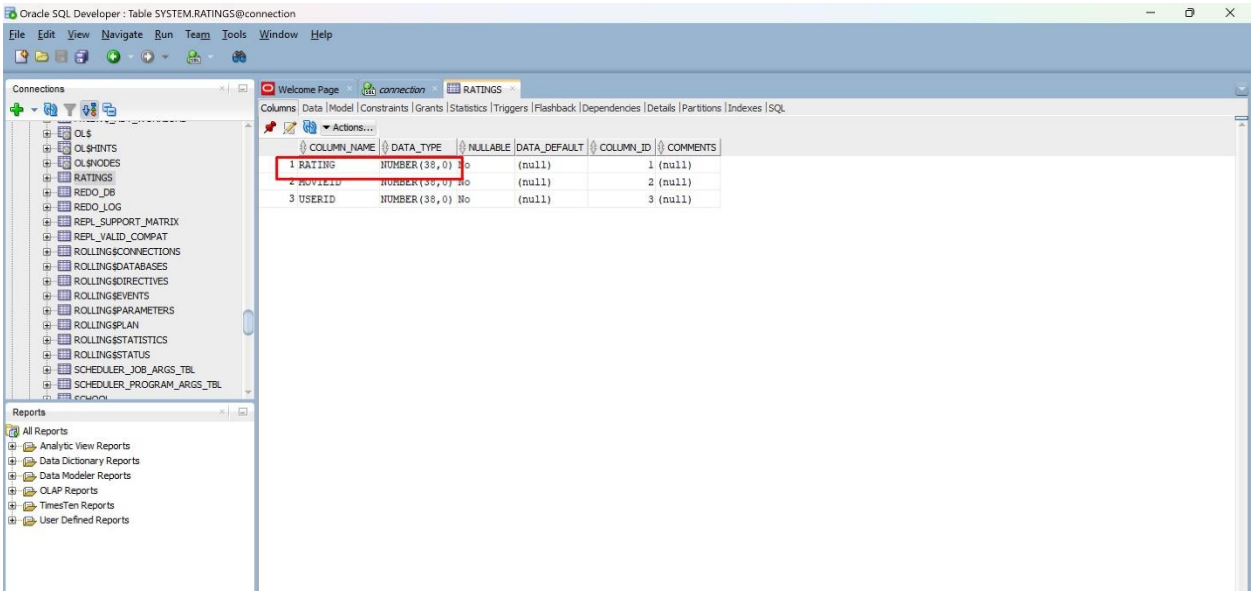


Drop table movies, show your SQL coding.

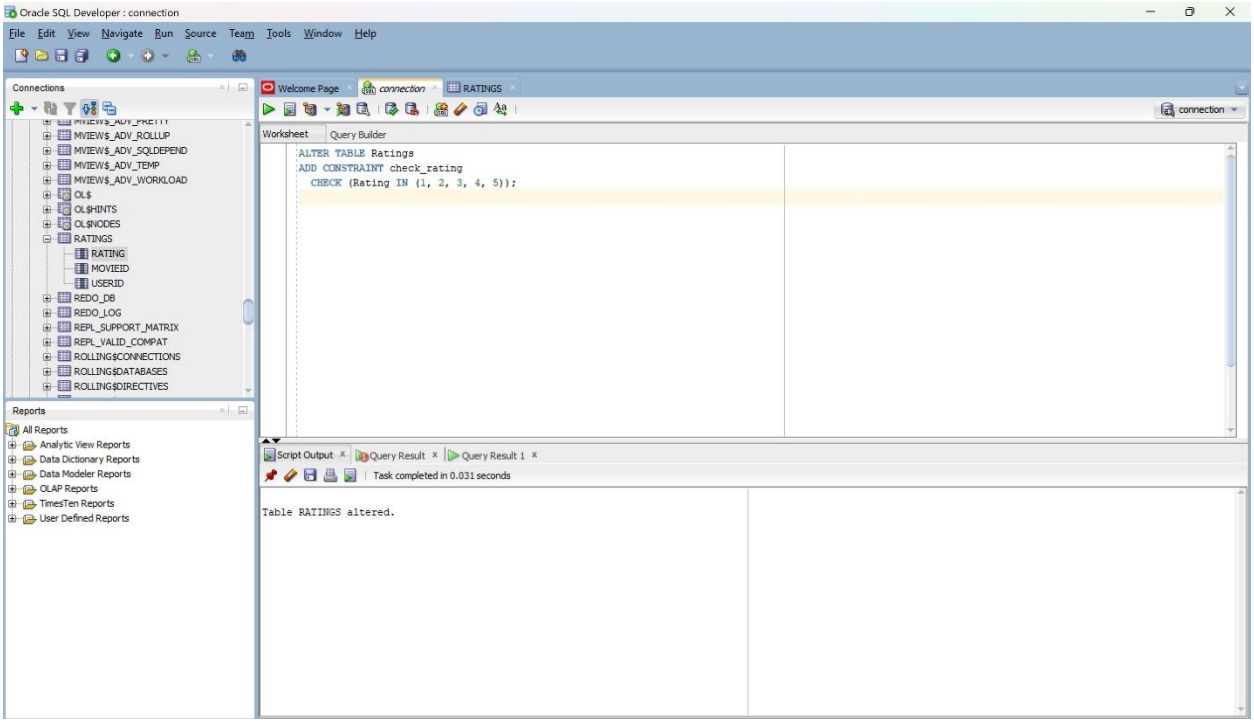


In table Ratings, change the data type of ratings to Number (2, 1) , show your SQL coding:





Assume that users can only give ratings, like, 1, 2, 3, 4, 5. In table Ratings, add a check constraint to guarantee that the input values for the column rating must be a value from (1, 2, 3, 4, 5)



FEEDBACK AND GRADE:

I got 100 **********