Project - Technical Report

BUAN 6320.504

Project Group 5

JSOM - UTD

**Introduction**

This database design document outlines the creation and implementation of a database that will house user information for a fantasy sports application called “Dream 11”. With the dawn of the new age interactive applications also brought in the concept of fantasy sport, where participants assemble virtual teams that mimic real players of a professional sport. These teams compete based on their performance of those players in actual games. The outcome of this performance is converted into reward points to a session selected by the individual fantasy teams. The application tracks the points and actual results of the professional sport.

**DREAM 11 : Overview**

Dream 11 is a fantasy sport application that was founded by Harsh Jain and Bhavit Sheth in the year 2008 which allows its users to create imaginary teams. The application key highlight was that the users can earn cash based on actual match performance of the selected players on their teams. Each good move of the players earns cash rewards to the users. Moreover, Dream 11 provided its users to play games like cricket, football, basketball, baseball, hockey to even handball.

It has successfully registered more than 100 million users in 2020, which has grown to 140 million in 2021. The total number of users registered with the application was only 2 million in 2016. Sports is a passion for many and this platform lets sports lovers earn rewards with their knowledge and love for the game. The platform currently boasts of over 13 Crore users.

**Design Layout**

**Key Factors Influencing Design:**

The initial design of the database was established through the layout of the application process, this helped in the creation of the initial entities that will be involved in the game model (Customer, Games, Sports and Fantasy Team). This was then realized with an entity-relationship model with six entities to form the layout of the game

The independent entities were Customer, Player and Game Details. The Customer entity represents end users of the application, a Player entity represents the set of players of the sport that were linked with the real time games and the Game Details entity would draw in details of real time games. The Fantasy team entity will represent the customized team chosen by the customer for the game and it will be dependent on the Player and Game Details entity . The Transaction entity will be recording the rewards, bonus and subscription payments that will be done through the application. The Winner list will compile the outcome of the Dream 11 fantasy game.

**Assumptions**

Each customer will be allowed to play only one game at a session contained to an individual sport. This will allow the user to focus on the performance of his team with the real-time game to maximize his probability of winning the game.

**Constraints**

Transaction entities are restricted to the rewards that will be provided to the customer. The Game details and player score are static and wouldn't be changed dynamically with the live game play .

**Project Scope**

**Requirement**

The scope of this project is to design and implement the database. In-scope work will include documenting the requirements, modeling the database in an entity-relationship diagram, writing data definition language (DDL) SQL scripts to define and implement the database, and writing example data manipulation language (DML) and Structured Query Language (SQL) scripts to demonstrated the intended use of the database.

**In-Scope Work**

* Project requirements documentation

* Entity-relationship model

* DDL Scripts

* Example DML Scripts

* Example SQL Scripts

* Comprehensive Report

**Database Goals, Expectations, and Deliverables**

Upon the completion of this project, the database shall mimic the fantasy application and help in implementing the gameplay. Deliverables include this statement of work document, an entity-relationship diagram defining the structure of the database, DDL scripts for creating the database, example DML and SQL scripts that demonstrate proper usage of the database, and a final report on the project as a whole.

**Database Benefits**

The primary benefit of the database will be to enable players to browse and search the game’s library and players to create their team and compete with the competitive real time sport.  
 The Transaction entity helps to compile the rewards of the customers. Triggers have been used to automate the game points of players and sequences to reduce the identification of the customer Transaction and game entities.

**Project Hardware and Software Tools**

**Diagram Tool**

Microsoft VISIO, running on Windows 11

**Office Productivity Tools**

Google Drive, Google Suite (Docs, Presentation)  
Microsoft Word 2016

**Database**

Oracle Database

Oracle APEX

**General Format**

* Use consistent and descriptive identifiers and names.

* Use white space and indentation to make code easier to read.

* Store time and date formation in ISO-8601 format (YYYY-MM-DD

HH:MM:SS.SSSSS).

* Avoid redundant SQL, such as unnecessary quoting or parentheses or WHERE

clauses that can be derived.

* Use C-style comments with opening /\* and closing /\* digraphs whenever possible; otherwise, precede comments with -- and finish them with a new line.
* For the sake of quick readability, prefer snake\_case over CamelCase.

* Avoid Hungarian notation and other descriptive prefixes.

* Favor collective nouns over plurals, such as using staff instead of employees.

* When using quoted identifiers, use SQL92 double quotes to preserve portability.

* Avoid applying object-oriented design principles to SQL or database structures.

**Naming Conventions**

* Names must begin with a letter and may not end with an underscore.
* Ensure that all names are unique and do not conflict with reserved keywords. Keep name length to 30 bytes; this usually means 30 characters, unless the name uses a multi-byte character set.

* Names may contain only letters, numbers, and underscores.

* Multiple consecutive underscores are not allowed.

* Use underscores to represent spaces in names, e.g. “first name” becomes

first\_name.

* Avoid abbreviations; if it is necessary to use them, ensure they are commonly

known and understood.

* Prefer collective nouns for table names.

* Tables and columns should never share the same name.

* Avoid concatenating the names of two tables when naming their relationship table.
* When naming columns, always prefer singular nouns.

* Avoid the name id for primary keys.
* Use lowercase in column names whenever reasonable.

* Use commonly-known suffixes to indicate the purpose of a column: \_id, \_name,

\_size, \_addr, etc.

**Requirements Definition Document**

**Business Rules**

A Customer can make no or many Transactions

A Transaction must be made by only one Customer

A Customer must make only one Fantasy Team

A Fantasy Team will be made by only one Customer

A Fantasy team must have only one Game Detail

Game Detail has one or many Fantasy Teams

Fantasy Team will have one or many Players

Either one or many Players are in a Fantasy team

Game Details will have many Winners

Winner list will be assigned to a single Game

Winner list will have one or many transactions.

One Transaction will be made only to one winner

**Entity and Attribute Description.**

***Entities***

Entity Name: CUSTOMER

Entity Description: The primary end-user of the software.

Main Attributes of CUSTOMER:

cust\_Id: (Primary Key) A unique identifier for the customer’s ID

cust\_FName: A character record of the customer’s first name

cust\_LName: A character record of the customer’s last name

cust\_Age: A numerical record of the customer’s age

cust\_Email: A character record of the customer’s email address

cust\_Phone: A numeric record of the customer’s phone number

cust\_AccountNo: An alphanumeric record of the customer’s account number used for Transactions.

cust\_CardNo: A numeric record of the customer’s card number used for Transactions.

cust\_WalletPoints: A numeric record of the customer’s wallet points earned in joining the application.

cust\_subscription: The application subscription type of the customer.

Entity Name: PLAYER

Entity Description: The list of players who will be chosen for the fantasy team.

Main Attributes of PLAYER:

player\_Id: (Primary Key) A unique identifier for the player’s ID

player\_FName: A character record of the player’s first name

player\_LName: A character record of the player’s last name

player\_Age: A numerical record of the player’s age

player\_Points: A numerical record of the player’s performance points

player\_Type: A character record of the player’s position type

player\_Nationality: A character record of the player’s nation / team he represents.

Entity Name: TRANSACTION

Entity Description: This entity consists information regarding the Transaction

Main Attributes of TRANSACTION:

trans\_Id: (Primary Key) A unique identifier for the winner's ID

cust\_Id: (Foreign Key) A unique identifier for the customer’s ID

win\_Id:(Foreign Key) A unique identifier for the winner ID

trans\_Type: A character record of the transaction type

trans\_Amt: A numerical record of the transaction amount

trans\_currency: Type of currency the transaction has been made in

trans\_Status: A character record of the status of transaction

trans\_timestamp: A numerical record of the transaction time

confirm\_Id: A unique identifier for the transaction confirmation

Entity Name: FANTASY TEAM

Entity Description: This entity consists of the players chosen by the customer to form their fantasy team.

Main Attributes of FANTASY TEAM:

t\_Id: (Primary Key) The unique identifier of the Team

game\_Id: (Foreign Key) The unique identifier of the Game

cust\_Id: (Foreign Key) A unique identifier for the customer’s ID

player\_Id: (Foreign Key) A unique identifier for the player’s ID

players\_role\_game: The role of player in the game

team\_name: A character record of the Team's name

players\_origin\_team: the original team that the player belongs to

points\_per\_team: A numerical record of the Team's performance points for the game

points\_per\_player: A numerical record of the player’s performance points for the game

Entity Name: GAME\_DETAILS

Entity Description: This entity consists of relevant details of the game .

Main Attributes of GAME\_DETAILS:

game\_Id: (Primary Key) A character record of the game

game\_Team\_1: A character record of the Team 1 in the game

game\_Team\_2: A character record of the Team 2 in the game

game\_Winner: The winner of the game

game\_Venue: The location details of where the game took place

game\_Notes: Description on the game statistics

Entity Name: WINNER LIST

Entity Description: This entity consists list of winners who attained the highest scores in the game through their fantasy team

Main Attributes of WINNER LIST:

win\_Id: The unique identifier of the winner

cust\_Id: (Foreign Key) A unique identifier for the customer’s ID

game\_Id: (Foreign Key) The unique identifier of the Game

t\_Id:(Foreign Key) The unique identifier of the Team

win\_Position: A character record of the winner's position in the game

win\_Points: A numerical record of the winner’s performance points for the game

contest\_Name: A character record of the contest’s name

contest\_PricePool: the total reward points placed by contest

**Relationship and Cardinality Description.**

Relationship ‘Makes’ between CUSTOMER and TRANSACTION.   
Cardinality: 1:M between CUSTOMER and TRANSACTION.   
Business rule: A Customer can makes either zero or many Transactions; A Transaction must be made by only one Customer

Relationship ‘Makes’ between CUSTOMER and FANTASY TEAM.   
Cardinality: 1:1 between CUSTOMER and FANTASY TEAM.   
Business rule: A Customer must mandatorily choose a Fantasy Team ; A Fantasy Team will be chosen only by one Customer

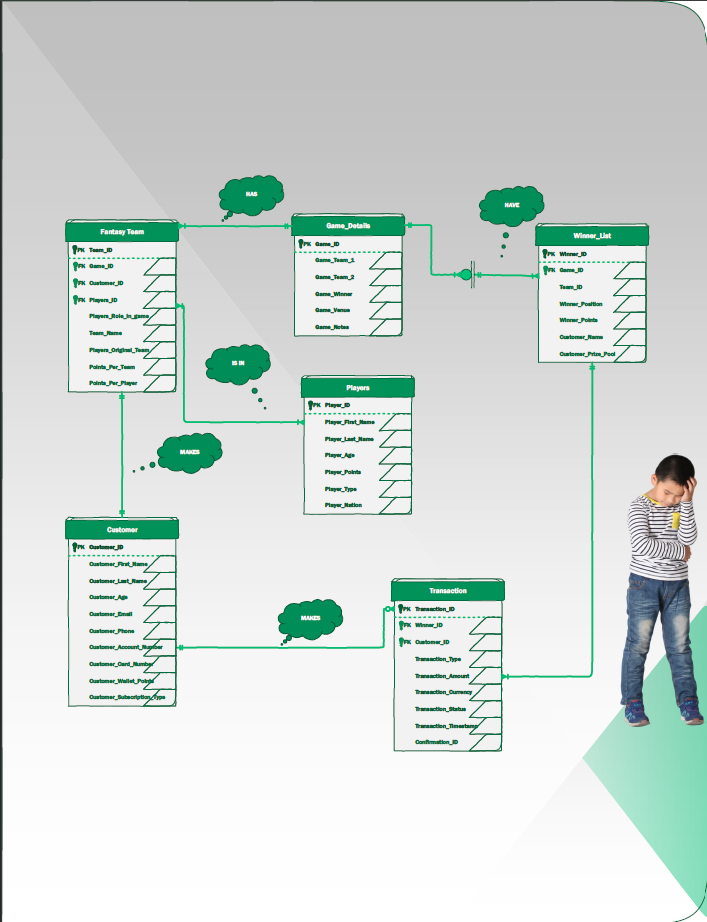
Relationship ‘Has’ between FANTASY TEAM and GAME DETAILS.   
Cardinality: M:1 between FANTASY TEAM and GAME DETAILS.   
Business rule: A Fantasy team must have only one Game Detail; A Game Detail will be having at least one Fantasy Team.

Relationship ‘Is in’ between PLAYER and FANTASY TEAM.   
Cardinality: M:M between FANTASY TEAM and PLAYER.   
Business rule: Fantasy Team will have one or many Players; Either one or many Players are in a Fantasy team

Relationship ‘Have’ between GAME DETAILS and WINNER LIST.   
Cardinality: 1:M between GAME DETAILS and WINNER LIST.   
Business rule: Game Details will have at least one to many Winners; Winner list will be assigned to only one single Game.

Relationship between WINNER LIST and TRANSACTION  
Cardinality: 1:M between GAME DETAILS and WINNER LIST.   
Business rule: Winner list will have one or many transactions; One Transaction will be made only to one winner

**Entity – Relationship Diagram**

****

**DDL Source Code**

/\*

Group 5 Project BAUN 6320 - UTD

\*/

/\* Create tables based on entities \*/

-- Triggers

DROP TRIGGER TRG\_Customer;

DROP TRIGGER TRG\_WinnerList;

-- Sequences

DROP SEQUENCE SEQ\_Player\_player\_Id;

DROP SEQUENCE SEQ\_Customer\_cust\_Id;

DROP SEQUENCE SEQ\_Game\_game\_Id;

DROP SEQUENCE SEQ\_WinnerList\_win\_Id;

DROP SEQUENCE SEQ\_Transcation\_trans\_Id;

DROP SEQUENCE SEQ\_FantasyTeam\_t\_Id;

--TABLES

DROP TABLE Transcation;

DROP TABLE WinnerList;

DROP TABLE FantasyTeam;

DROP TABLE Player;

DROP TABLE Customer;

DROP TABLE Game;

-- Views

Drop view Transcation\_PerCustomer\_Details;

/\*Player Entity \*/

CREATE TABLE Player(

player\_Id INTEGER NOT NULL,

player\_Fname VARCHAR(20) NOT NULL,

player\_Lname VARCHAR(20) NOT NULL,

player\_Age INTEGER,

player\_Points INTEGER,

player\_Type VARCHAR(20),

player\_Nationality VARCHAR(20),

CONSTRAINT PK\_Player PRIMARY KEY (player\_id)

);

/\*Customer Entity \*/

CREATE TABLE Customer(

cust\_Id INTEGER NOT NULL,

cust\_Fname VARCHAR(20) NOT NULL,

cust\_Lname VARCHAR(20) NOT NULL,

cust\_Age INTEGER,

cust\_Email VARCHAR(30),

cust\_Phone INTEGER,

cust\_AccountNo INTEGER ,

cust\_CardNo INTEGER,

cust\_WalletPoints INTEGER,

cust\_subscription VARCHAR(10),

CONSTRAINT PK\_Customer PRIMARY KEY (cust\_Id)

);

/\* Game Details Entity \*/

CREATE TABLE Game(

game\_Id INTEGER NOT NULL,

game\_Team1 VARCHAR(20) NOT NULL,

game\_Team2 VARCHAR(20) NOT NULL,

game\_Winner VARCHAR(20),

game\_Venue VARCHAR(25),

game\_Notes VARCHAR(50),

CONSTRAINT PK\_Game PRIMARY KEY (game\_Id)

);

/\* FantasyTeam Entity \*/

CREATE TABLE FantasyTeam(

t\_Id INTEGER NOT NULL,

team\_Name VARCHAR(20),

game\_Id INTEGER NOT NULL,

cust\_Id INTEGER NOT NULL,

player\_Id INTEGER NOT NULL,

player\_role\_game VARCHAR(20),

player\_origin\_team VARCHAR(20),

points\_per\_team INTEGER,

points\_per\_player INTEGER,

CONSTRAINT PK\_FantasyTeam PRIMARY KEY(t\_Id),

CONSTRAINT FK\_FantasyTeam\_Game FOREIGN KEY(game\_Id) REFERENCES GAME,

CONSTRAINT FK\_FantasyTeam\_Customer FOREIGN KEY(cust\_Id) REFERENCES Customer,

CONSTRAINT FK\_FantasyTeam\_Player FOREIGN KEY(player\_Id) REFERENCES Player

);

/\* Winner Entity \*/

CREATE TABLE WinnerList(

win\_Id INTEGER NOT NULL,

cust\_Id INTEGER NOT NULL,

game\_Id INTEGER NOT NULL,

t\_Id INTEGER NOT NULL,

winner\_position VARCHAR(10),

winner\_point INTEGER,

contest\_Name VARCHAR(20),

contest\_PrizePool INTEGER,

CONSTRAINT PK\_WinnerList PRIMARY KEY (win\_Id),

CONSTRAINT FK\_Winner\_Customer FOREIGN KEY (cust\_id) REFERENCES Customer,

CONSTRAINT FK\_Winner\_Game FOREIGN KEY (game\_id) REFERENCES Game,

CONSTRAINT FK\_Winner\_Team FOREIGN KEY (t\_Id) REFERENCES FantasyTeam

);

/\* Transcation Entity \*/

CREATE TABLE Transcation(

trans\_id INTEGER NOT NULL,

cust\_Id INTEGER NOT NULL,

win\_Id INTEGER NOT NULL,

trans\_Type VARCHAR(10),

trans\_Amt INTEGER,

trans\_currency VARCHAR(10),

trans\_Status VARCHAR(10),

trans\_timestamp TIMESTAMP,

confrim\_Id INTEGER,

CONSTRAINT PK\_Transcation PRIMARY KEY (trans\_Id),

CONSTRAINT FK\_Trans\_Customer FOREIGN KEY (cust\_Id) REFERENCES Customer,

CONSTRAINT FK\_Trans\_Winner FOREIGN KEY (win\_Id) REFERENCES WinnerList

);

-- Business purpose: The trigger TRG\_Customer adds 10 points to each newly inserted row or updated record in the customer table.

CREATE OR REPLACE TRIGGER TRG\_Customer

BEFORE INSERT OR UPDATE ON Customer

FOR EACH ROW

BEGIN

IF INSERTING THEN

IF :NEW.cust\_WalletPoints IS NULL THEN

:NEW.cust\_WalletPoints := 10;

END IF;

END IF;

END;

/

-- Business purpose: The trigger TRG\_WinnerList adds points to each Winner in the WinnerList table.

-- 1st position : 100 points

-- 2nd position : 50 points

-- 3rd position : 25points

CREATE OR REPLACE TRIGGER TRG\_WinnerList

BEFORE INSERT OR UPDATE ON WinnerList

FOR EACH ROW

BEGIN

IF INSERTING THEN

IF :NEW.winner\_position = 1 THEN

:NEW.winner\_point := 100;

END IF;

IF :NEW.winner\_position = 2 THEN

:NEW.winner\_point := 50;

END IF;

IF :NEW.winner\_position = 3 THEN

:NEW.winner\_point := 25;

END IF;

END IF;

END;

/

/\* Create Sequences \*/

CREATE SEQUENCE SEQ\_Player\_player\_Id

INCREMENT BY 1

START WITH 455

NOMAXVALUE

MINVALUE 455

NOCACHE;

CREATE SEQUENCE SEQ\_Customer\_cust\_Id

INCREMENT BY 1

START WITH 100

NOMAXVALUE

MINVALUE 100

NOCACHE;

CREATE SEQUENCE SEQ\_Game\_game\_Id

INCREMENT BY 1

START WITH 11001

NOMAXVALUE

MINVALUE 11001

NOCACHE;

CREATE SEQUENCE SEQ\_WinnerList\_win\_Id

INCREMENT BY 1

START WITH 111

NOMAXVALUE

MINVALUE 111

NOCACHE;

CREATE SEQUENCE SEQ\_Transcation\_trans\_Id

INCREMENT BY 1

START WITH 10000000001

NOMAXVALUE

MINVALUE 10000000001

NOCACHE;

CREATE SEQUENCE SEQ\_FantasyTeam\_t\_Id

INCREMENT BY 1

START WITH 1

NOMAXVALUE

MINVALUE 1

NOCACHE;

CREATE OR REPLACE VIEW Transcation\_PerCustomer\_Details AS

Select cust\_Fname,cust\_Lname,cust\_Age,Trans\_type, trans\_amt, trans\_currency, trans\_status, trans\_Timestamp from customer c

Inner join transcation t

on c.cust\_Id = t.cust\_Id;

COMMIT

**DML Source Code**

/\*

Project BUAN 6320

\*/

/\* Populate all tables \*/

-- Player table

--Players : INDIA

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Virat','Kholi',34,80,'Batsman','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Rohit','Sharma',35,73,'Batsman','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Jasprit','Bumrah',28,85,'Bowler','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Ravichandran','Ashwin',36,75,'Bowler','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Ravindra','Jadeja',33,77,'All rounder','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Cheteshwar','Pujara',34,68,'Batsman','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Ajinkya','Rahane',34,80,'Batsman','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Shikhar','Dhawan',36,81,'Batsman','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'KL','Rahul',30,85,'Batsman','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Mohammad','Shami',32,68,'Bowler','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Ishant','Sharma',34,64,'Bowler','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Rishab','Pant',25,78,'Wicket-keeper','India');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Hardik','Pandya',29,84,'Batsman','India');

--Players : Australia

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Aron','Finch',35,67,'Batsman','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'David','Warner',36,80,'Batsman','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Matthew','Wade',34,79,'Wicket-keeper','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Steve','Smith',33,59,'Batsman','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Tim','David',26,69,'Batsman','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Ashton','Agar',29,74,'All rounder','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Mitchell','Marsh',31,78,'All rounder','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Glenn','Maxwell',34,81,'All rounder','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Marcus','Stonis',33,77,'All rounder','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Pat','Cummins',29,85,'Bowler','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Josh','Hazelwood',31,88,'Bowler','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Kane','Richardson',31,68,'Bowler','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Adam','Zampa',30,65,'Bowler','Australia');

INSERT INTO PLAYER(player\_Id,player\_Fname,player\_Lname,player\_Age,player\_Points,player\_Type,player\_Nationality)

VALUES (SEQ\_Player\_player\_Id.nextval,'Mitchell','Starc',32,84,'Bowler','Australia');

--Customer table

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Sofia','Clark',20,'sofia@gmail.com',9346666324,22346,885368427951,NULL,'Premium');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Chad','Bose',25,'c.bose@gmail.com',5869349283,984345,75764838938,NULL,'regular');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Ruslan','Sum',18,'Rsum@gmail.com',3736543286,404852,839057187121 ,NULL,'regular');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Ramazan',' Eldar',29,'Eldar.r@gmail.com',6997747028,115001 ,462317834596 ,NULL,'Premium');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Beauregard','Vassily',22,'Bea.vass24@gmail.com',1592978019 ,139490 ,582247624112 ,NULL,'regular');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Aleksand','Cleitus',39,'alex.cl@gmail.com',2265965089,325590 ,569921028910 ,NULL,'regular');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Tapio','Adilet',44,'tapoi.ad@gmail.com',0937938695 ,245540 ,339281938260 ,NULL,'Premium');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Tullio',' Kartik',49,'Tullio.karthik@gmail.com',0680681271 ,659911 ,061352568797 ,NULL,'regular');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Valerian','Janaka ',35,'Janaka356@gmail.com',3945533827 ,998933 ,855241646409 ,NULL,'regular');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Leif','Alaric ',28,'lief267@gmail.com',6812312274 ,841579 ,318080545446 ,NULL,'Premium');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Kweku','Berahthraban',19,'Kweku795@gmail.com',9818469920 ,100106 ,603016679386 ,NULL,'regular');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Sigmund','Leudagar',56,'Sig.Leu@gmail.com',0029192242 ,184011 ,981673679699 ,NULL,'Premium');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Alon','Zaman',27,'Alan.zaman631@gmail.com',4690642658 ,340217 ,151749932192 ,NULL,'regular');

INSERT INTO Customer (cust\_Id,cust\_Fname, cust\_Lname, cust\_Age, cust\_Email, cust\_Phone, cust\_AccountNo, cust\_CardNo,cust\_WalletPoints,cust\_subscription)

VALUES (SEQ\_Customer\_cust\_Id.nextval,'Jong','Ailbe',30,'Jong9852@gmail.com',1835784441,509462,569046183941 ,NULL,'regular');

--Game table

INSERT INTO Game (game\_Id,game\_Team1,game\_Team2,game\_Winner,game\_Venue,game\_Notes)

VALUES (SEQ\_Game\_game\_Id.nextval,'India','Australia','India','Melbourne','won by 2 wickets');

INSERT INTO Game (game\_Id,game\_Team1,game\_Team2,game\_Winner,game\_Venue,game\_Notes)

VALUES (SEQ\_Game\_game\_Id.nextval,'England','Sri Lanka','Sri Lanka','London','won by 1 wicket');

INSERT INTO Game (game\_Id,game\_Team1,game\_Team2,game\_Winner,game\_Venue,game\_Notes)

VALUES (SEQ\_Game\_game\_Id.nextval,'India','Australia','Australia','Sydney','won by 4 wicket');

INSERT INTO Game (game\_Id,game\_Team1,game\_Team2,game\_Winner,game\_Venue,game\_Notes)

VALUES (SEQ\_Game\_game\_Id.nextval,'Australia','India','India','New Delhi','won by 3 wicket');

INSERT INTO Game (game\_Id,game\_Team1,game\_Team2,game\_Winner,game\_Venue,game\_Notes)

VALUES (SEQ\_Game\_game\_Id.nextval,'Australia','India','India','Hyderabad','won by 6 wicket');

--FantasyTeam table

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'QUERTY',11001,100,456,'Batsman','India',1000,73);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'QUERTY',11001,100,463,'Batsman','India',1000,85);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'QUERTY',11001,100,468,'Batsman','Australia',1000,67);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'QUERTY',11001,100,472,'Batsman','Australia',1000,69);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'QUERTY',11001,100,460,'Batsman','India',1000,68);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Mash Heads',11001,109,457,'Bowler','India',800,85);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Mash Heads',11001,109,465,'Bowler','India',800,64);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Mash Heads',11001,109,480,'Bowler','Australia',800,65);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Mash Heads',11001,109,477,'Bowler','Australia',800,85);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Mash Heads',11001,109,459,'All rounder','India',800,77);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Purple Stomp',11001,106,470,'Wicket-keeper','Australia',1000,79);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Purple Stomp',11001,106,480,'Bowler','Australia',1000,65);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Purple Stomp',11001,106,466,'Wicket-keeper','India',1000,78);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Purple Stomp',11001,106,456,'Batsman','India',1000,73);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Purple Stomp',11001,106,463,'Batsman','India',1000,85);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Blank Crash',11005,102,476,'All rounder','Australia',1000,77);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Blank Crash',11005,102,461,'Batsman','India',1000,80);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Blank Crash',11005,102,468,'Batsman','Australia',1000,67);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Blank Crash',11005,102,477,'Bowler','Australia',1000,85);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Blank Crash',11005,102,481,'Bowler','Australia',1000,84);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Synergy Clan',11001,111,476,'All rounder','Australia',1000,77);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Synergy Clan',11001,111,457,'Bowler','India',1000,85);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Synergy Clan',11001,111,462,'Batsman','India',1000,81);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Synergy Clan',11001,111,466,'Wicket-keeper','India',1000,78);

INSERT INTO FantasyTeam (t\_Id,team\_Name,game\_Id,cust\_Id,player\_Id,player\_role\_game,player\_origin\_team,points\_per\_team,points\_per\_player)

VALUES (SEQ\_FantasyTeam\_t\_Id.nextval,'Synergy Clan',11001,111,480,'Bowler','Australia',1000,65);

--WinnerList

INSERT INTO WinnerList (win\_Id,cust\_Id,game\_Id,t\_Id,winner\_position,winner\_point,contest\_Name,contest\_PrizePool)

VALUES (SEQ\_WinnerList\_win\_Id.nextval,100,11001,1,1,NULL,'T20-Finals',10000);

INSERT INTO WinnerList (win\_Id,cust\_Id,game\_Id,t\_Id,winner\_position,winner\_point,contest\_Name,contest\_PrizePool)

VALUES (SEQ\_WinnerList\_win\_Id.nextval,109,11001,6,2,NULL,'T20-Finals',5000);

INSERT INTO WinnerList (win\_Id,cust\_Id,game\_Id,t\_Id,winner\_position,winner\_point,contest\_Name,contest\_PrizePool)

VALUES (SEQ\_WinnerList\_win\_Id.nextval,106,11001,11,3,NULL,'T20-Finals',2500);

INSERT INTO WinnerList (win\_Id,cust\_Id,game\_Id,t\_Id,winner\_position,winner\_point,contest\_Name,contest\_PrizePool)

VALUES (SEQ\_WinnerList\_win\_Id.nextval,111,11001,16,1,NULL,'WorldCup-Finals',1000);

INSERT INTO WinnerList (win\_Id,cust\_Id,game\_Id,t\_Id,winner\_position,winner\_point,contest\_Name,contest\_PrizePool)

VALUES (SEQ\_WinnerList\_win\_Id.nextval,102,11001,21,2,NULL,'WorldCup-Finals',500);

--Transaction list

INSERT INTO Transcation (trans\_id,cust\_Id,win\_Id,trans\_Type,trans\_Amt,trans\_currency,trans\_Status,trans\_timestamp,confrim\_Id)

VALUES (SEQ\_Transcation\_trans\_id.nextval,100,111,'Credit',10000,'USD','Pending',TO\_DATE('2020/11/03 21:02:44', 'yyyy/mm/dd hh24:mi:ss'),5001);

INSERT INTO Transcation (trans\_id,cust\_Id,win\_Id,trans\_Type,trans\_Amt,trans\_currency,trans\_Status,trans\_timestamp,confrim\_Id)

VALUES (SEQ\_Transcation\_trans\_id.nextval,100,111,'Credit',10000,'USD','Declined',TO\_DATE('2020/11/02 21:02:44', 'yyyy/mm/dd hh24:mi:ss'),5001);

INSERT INTO Transcation (trans\_id,cust\_Id,win\_Id,trans\_Type,trans\_Amt,trans\_currency,trans\_Status,trans\_timestamp,confrim\_Id)

VALUES (SEQ\_Transcation\_trans\_id.nextval,109,112,'Credit',5000,'USD','Sucess',TO\_DATE('2020/11/02 21:01:44', 'yyyy/mm/dd hh24:mi:ss'),5002);

INSERT INTO Transcation (trans\_id,cust\_Id,win\_Id,trans\_Type,trans\_Amt,trans\_currency,trans\_Status,trans\_timestamp,confrim\_Id)

VALUES (SEQ\_Transcation\_trans\_id.nextval,111,113,'Credit',2500,'USD','Pending',TO\_DATE('2020/11/03 22:02:44', 'yyyy/mm/dd hh24:mi:ss'),5001);

INSERT INTO Transcation (trans\_id,cust\_Id,win\_Id,trans\_Type,trans\_Amt,trans\_currency,trans\_Status,trans\_timestamp,confrim\_Id)

VALUES (SEQ\_Transcation\_trans\_id.nextval,102,114,'Credit',1000,'USD','Pending',TO\_DATE('2020/11/04 21:02:44', 'yyyy/mm/dd hh24:mi:ss'),5001);

INSERT INTO Transcation (trans\_id,cust\_Id,win\_Id,trans\_Type,trans\_Amt,trans\_currency,trans\_Status,trans\_timestamp,confrim\_Id)

VALUES (SEQ\_Transcation\_trans\_id.nextval,109,115,'Credit',500,'USD','Sucess',TO\_DATE('2020/11/01 21:02:44', 'yyyy/mm/dd hh24:mi:ss'),5001);

COMMIT

--query--

**DDL Output**

Trigger TRG\_Customer dropped

Trigger TRG\_WinnerList dropped

Sequence SEQ\_Player\_player\_Id dropped.

Sequence SEQ\_Customer\_cust\_Id dropped.

Sequence SEQ\_Game\_game\_Id dropped.

Sequence SEQ\_WinnerList\_win\_Id dropped.

Sequence SEQ\_Transaction\_trans\_Id dropped.

Sequence SEQ\_FantasyTeam\_t\_Id dropped.

Table Transaction dropped.

Table Winnerlist dropped.

Table FantasyTeam dropped.

Table Player dropped.

Table Customer dropped.

Table Game dropped.

View Transaction\_PerCustomer\_Details dropped.

Table Player( player\_Id INTEGER N created.

Table Customer( cust\_Id INTEGER NOT NULL created.

Table Game( game\_Id INTEGER NOT NULL, created.

Table FantasyTeam( t\_Id INTEGER NOT NU created.

Table WinnerList( win\_Id INTEGER created.

Table Transaction( trans\_id INTEGER created.

Trigger TRG\_Customer created.

Trigger TRG\_WinnerList created.

Sequence SEQ\_Player\_player\_Id created.

Sequence SEQ\_Customer\_cust\_Id created.

Sequence SEQ\_Game\_game\_Id created.

Sequence SEQ\_WinnerList\_win\_Id created.

Sequence SEQ\_Transaction\_trans\_Id created.

Sequence SEQ\_FantasyTeam\_t\_Id created.

View Transaction\_PerCustomer\_Details created.

Statement processed.

**DML Output**

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

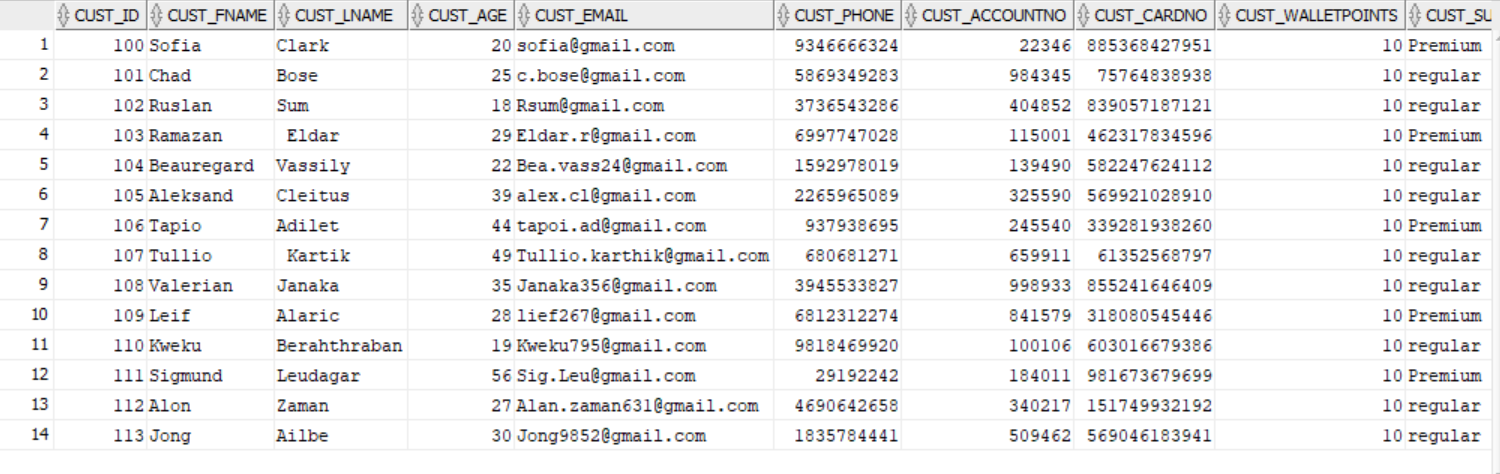
statement processed.

**Query Source Code and Output**

--query--

**-- Q1. Select all columns and all rows from one table**

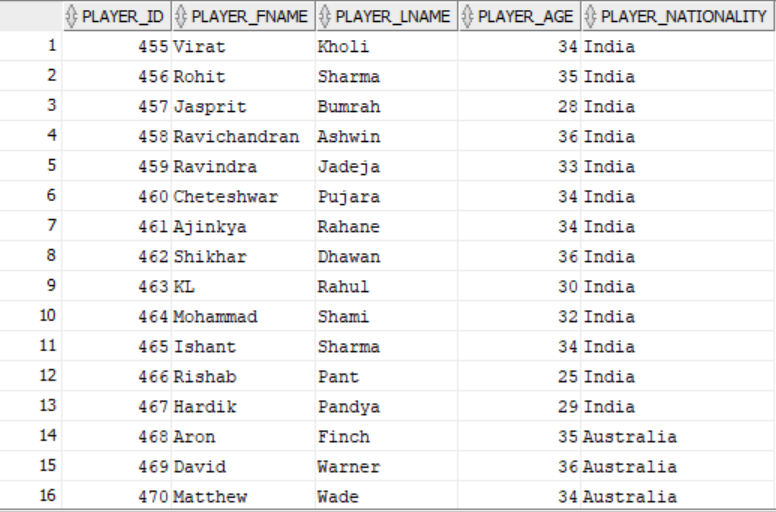
Select \* from Customer;



**-- Q2. Select five columns and all rows from one table**

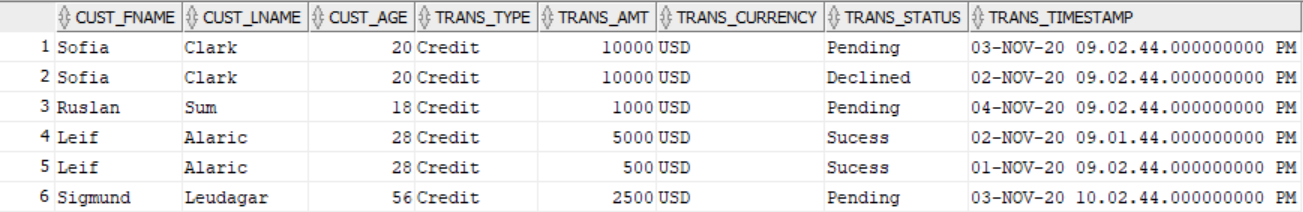
SELECT player\_Id, player\_Fname, player\_Lname, player\_Age, player\_Nationality

FROM Player;



**-- Q3. Select all columns from all rows from one view**

Select \* from Transaction\_PerCustomer\_Details;

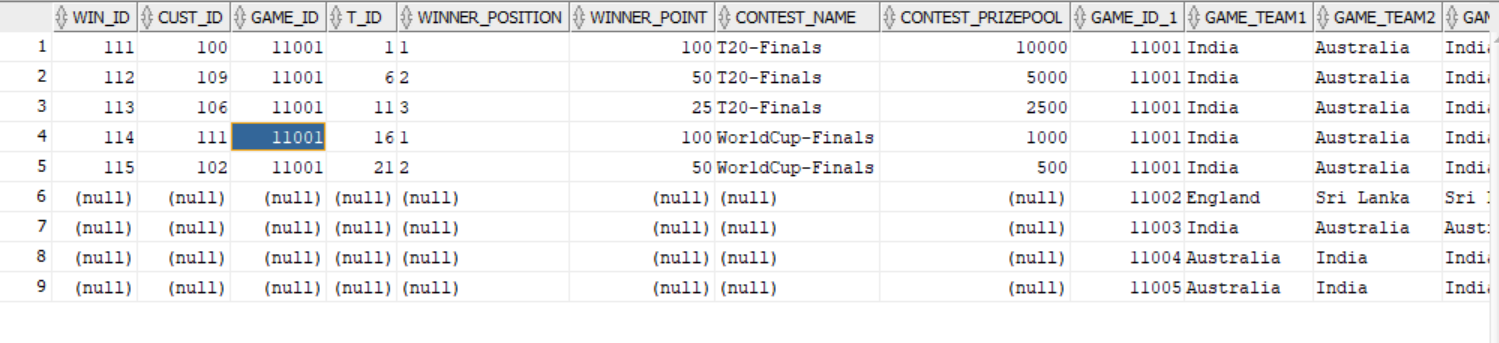


-- Q4. Using a join on 2 tables, select all columns and all rows

Select \* from Winnerlist w

FULL OUTER JOIN Game g

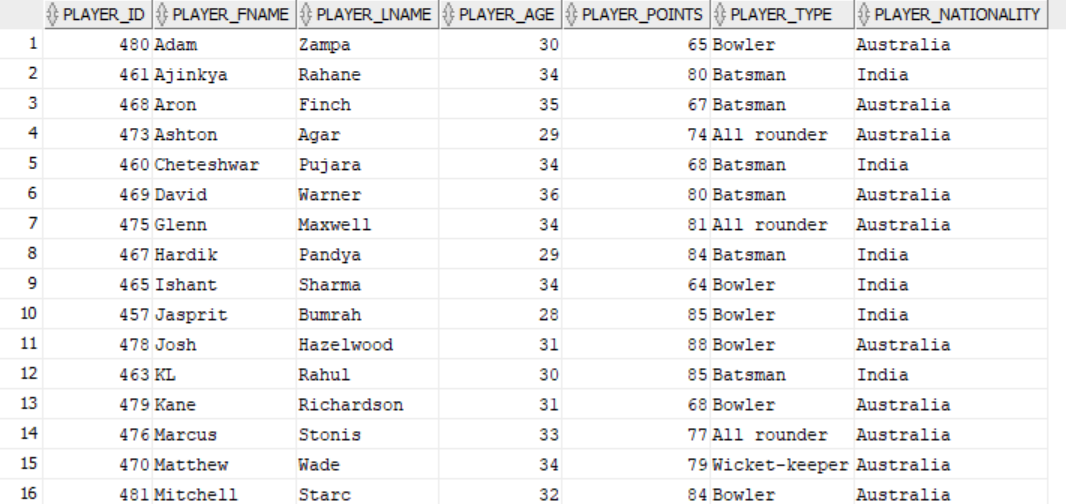
ON W.game\_Id = g.game\_Id;



**-- Q5. Select and order data retrieved from one table**

Select \* FROM Player

ORDER BY player\_Fname;



**-- Q6. Using a join on 3 tables, select 5 columns from the 3 tables.Use syntax that would limit the output to 10 rows**

Select c.cust\_Id,c.cust\_Fname,w.winner\_position,t.trans\_amt,t.trans\_status from Customer c

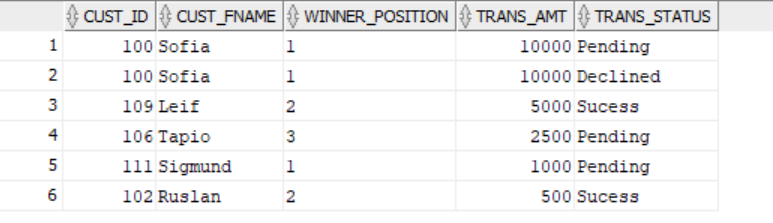
Inner Join Winnerlist w

ON c.cust\_Id = w.cust\_Id

Inner Join TRANSACTION t

on w.win\_Id = t.win\_Id

FETCH FIRST 10 ROWS ONLY;



**-- Q7: Select distinct rows using joins on 3 tables (5 points)**

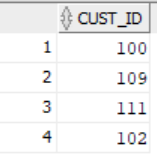
Select distinct (c.cust\_id) from TRANSACTION t

inner join customer c

on t.cust\_Id = c.cust\_Id

inner join winnerlist w

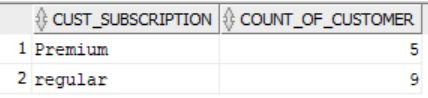
on w.win\_Id = t.win\_Id;



**-- Q8: Use GROUP BY and HAVING in a select statement using one or more tables**

Select Cust\_Subscription,count(\*) As Count\_Of\_Customer From Customer

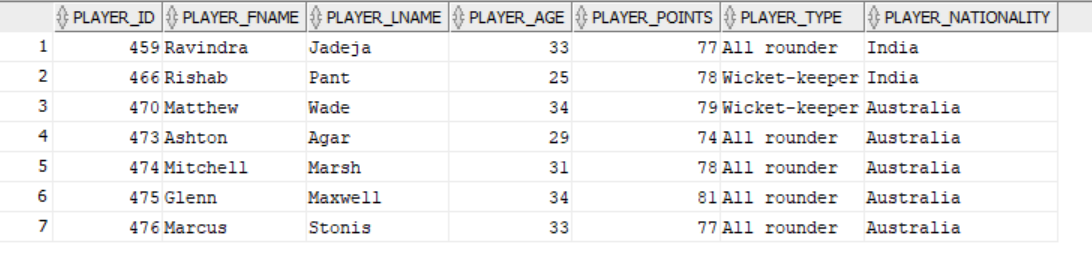
group by(Cust\_Subscription) Having count(\*)>4;



**-- Q9. Use IN clause to select data from one or more tables**

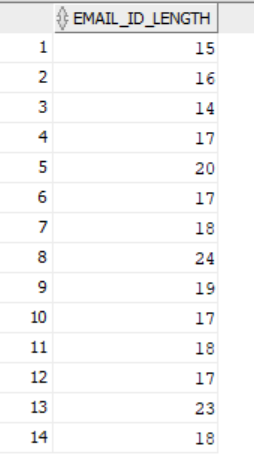
Select \* from player

Where player\_Type IN ('All rounder','Wicket-keeper');



**-- Q10. Select length of one column from one table (use LENGTH function)**

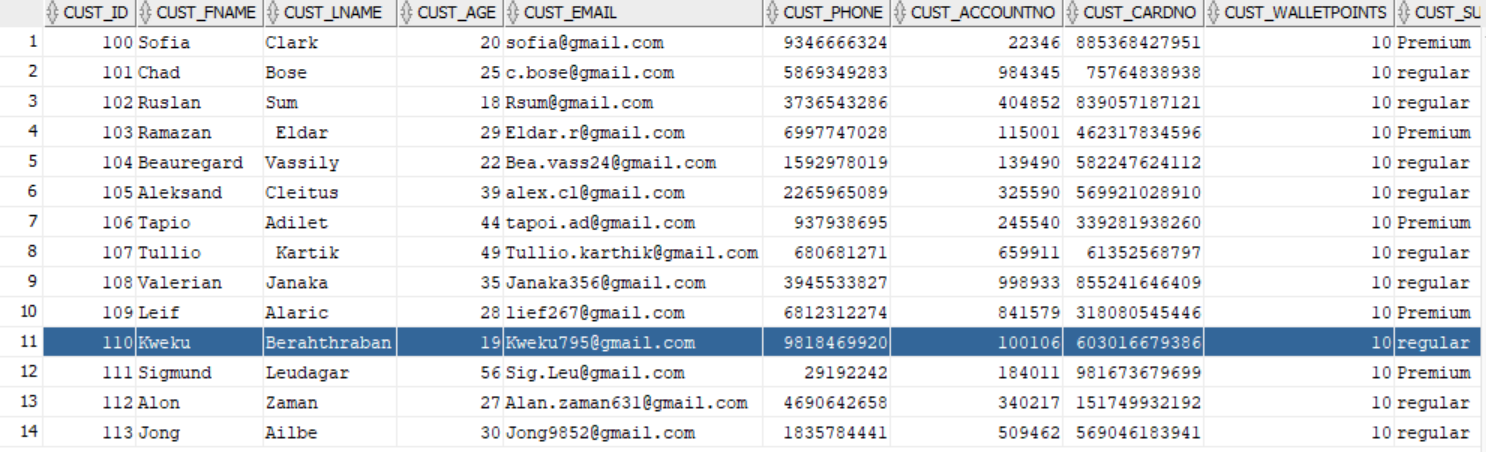
SELECT LENGTH(cust\_Email) As Email\_ID\_length FROM Customer;



**-- Q11. Delete one record from one table. Use select statements to demonstrate the table contents before and after the DELETE statement.**

--Make sure you use ROLLBACK afterwards so that the data will not be physically removed

SELECT \* FROM Customer;

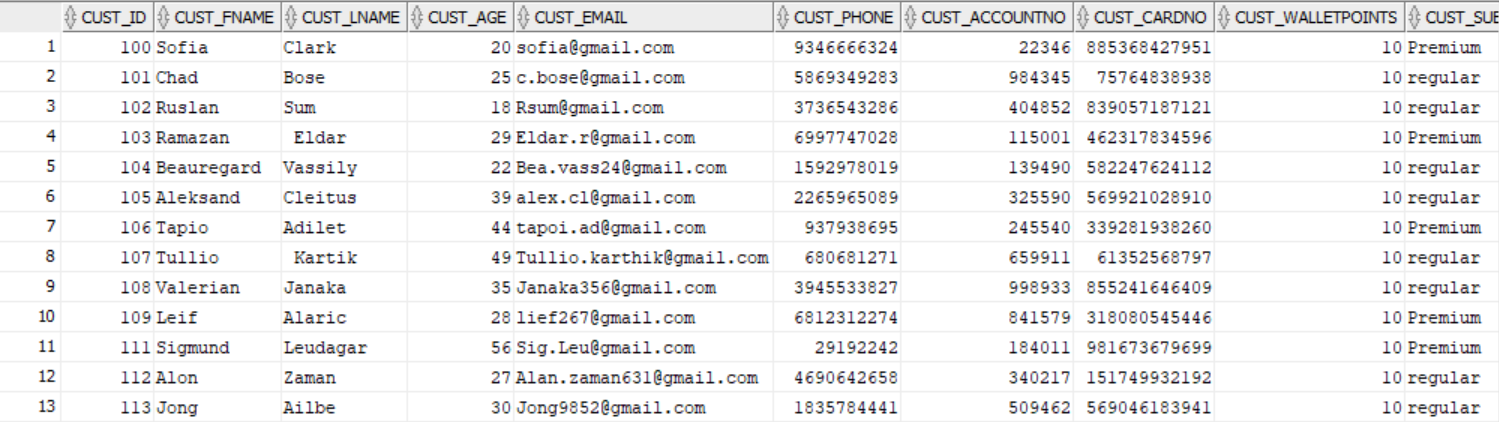


DELETE FROM customer

WHERE cust\_Id = 110;



SELECT \* FROM Customer;

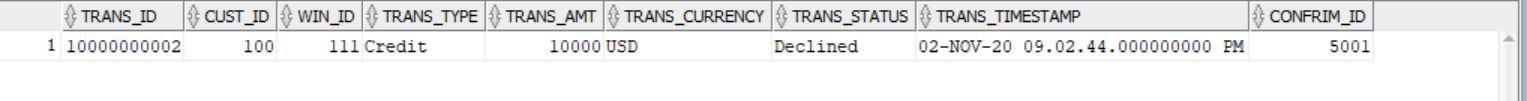


ROLLBACK;

**-- Q12.Update one record from one table. Use select statements to demonstrate the table contents before and after the UPDATE**

**--statement. Make sure you use ROLLBACK afterwards so that the data will not be physically removed**

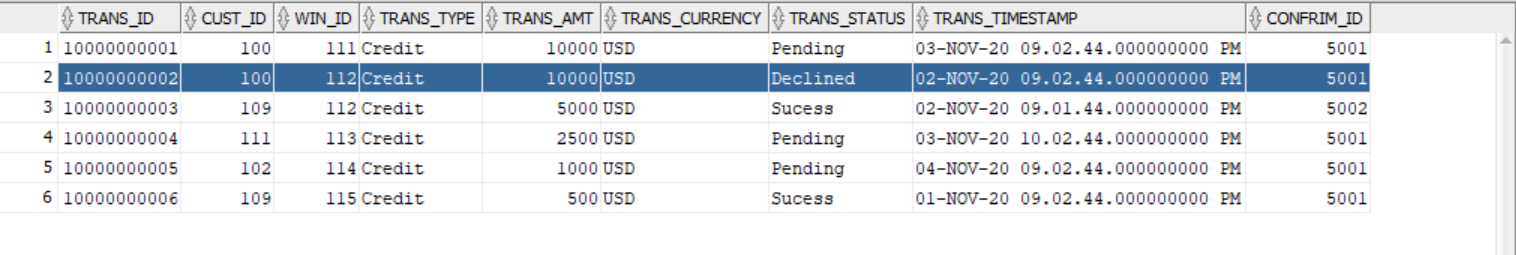
SELECT \* FROM TRANSACTION WHERE Trans\_Status = 'Declined';



UPDATE TRANSACTION SET win\_id = 112 WHERE Trans\_Status = 'Declined' ;

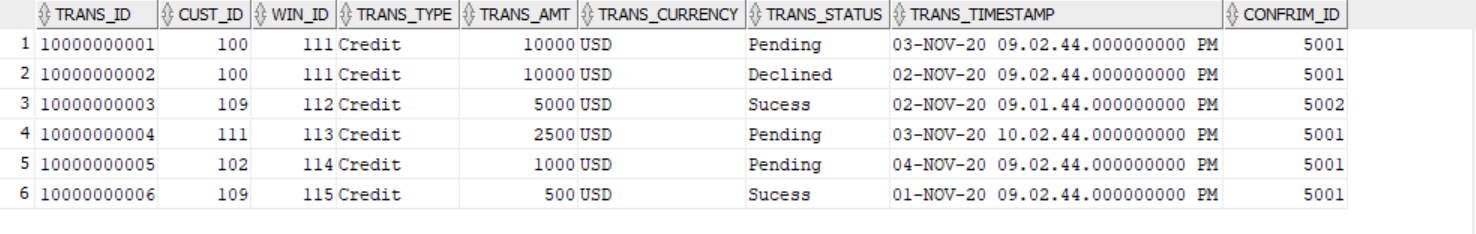


SELECT \* FROM TRANSACTION;



/\* updating the changed entry \*/

UPDATE TRANSACTION SET win\_id = 111 WHERE Trans\_Status = 'Declined' ;



ROLLBACK;

**--Q13: Query to display TRANSACTION status of a particular customer\_id and print details like customer name,age**

**--who are winner of a particular game**

select c.cust\_Id, c.cust\_fname, c.cust\_lname, c.cust\_Age,t.trans\_amt,t.trans\_status from customer c

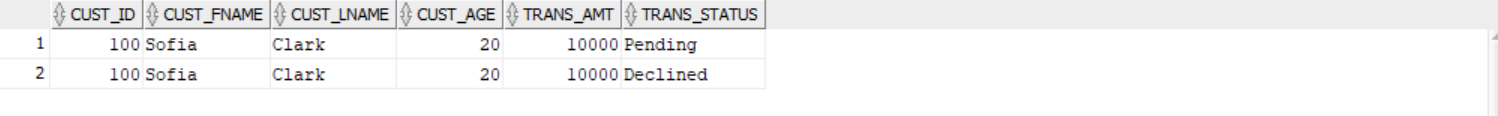
inner join winnerlist w

on c.cust\_Id = w.cust\_Id

inner join TRANSACTION t

on t.win\_Id = w.win\_Id

Where c.cust\_Id= 100;



**--Q14 : Query to display player\_id,firstname,lastname, Age,points,player type of particular with player age at least 35**

select p.player\_Id, p.player\_Fname, p.player\_Lname, p.player\_Age, p.player\_Points, p.player\_Type

from player p

left outer join fantasyteam f

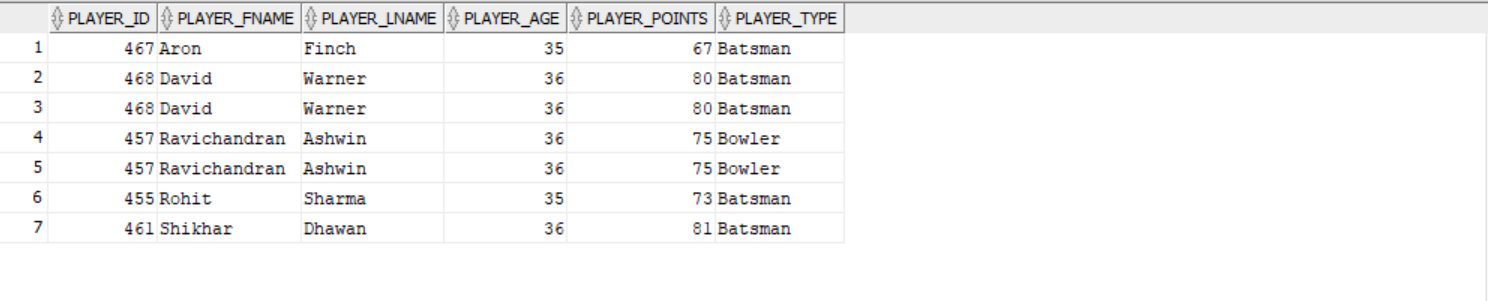
on p.player\_Id = f.player\_Id

left outer join game g

on f.game\_Id = g.game\_Id

Where p.player\_Age > = 35

order by(player\_Fname);



**-- Q15: Query to display the total points earned by batsmen and bowlers of every nationality**

select player\_nationality,sum(player\_points) as Total\_Point\_by\_Batsmen\_Bowlers from

(select player\_id,player\_points, player\_nationality, player\_type from player where player\_type='Batsman' or player\_type='Bowler')

group by player\_nationality;



**-- Q16: query to display count of player chosen by a particular customer for 1 game to create his own fantasy team**

Select c.cust\_Fname,c.cust\_Lname,count(game\_Id) as Total\_Players\_

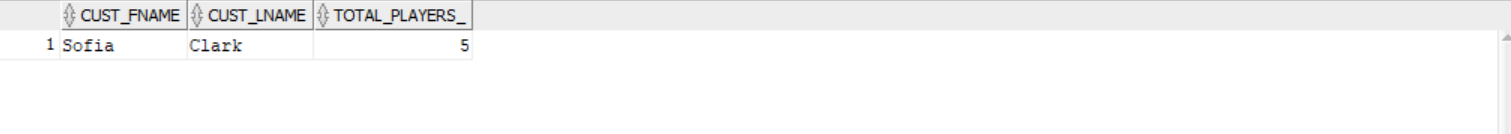
from fantasyteam f

inner join Customer c

on c.cust\_id = f.cust\_id

where f.cust\_Id = 100

group by f.game\_Id,c.cust\_Fname,c.cust\_Lname;



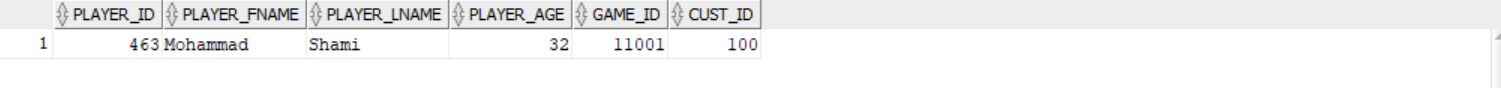
**--Q17: query to display player picked by a particular customer in his fantasy team whose first name starts with letter 'M'**

Select p.player\_Id, p.player\_Fname, p.player\_Lname, p.player\_Age, f.game\_Id, f.cust\_Id from player p

Inner join fantasyteam f

on p.player\_Id = f.player\_Id

where player\_fname like('M%') and cust\_Id = 102;



**-- Q18: Query to display the most chosen player by the customers in their fantasy team**

Select \* from player

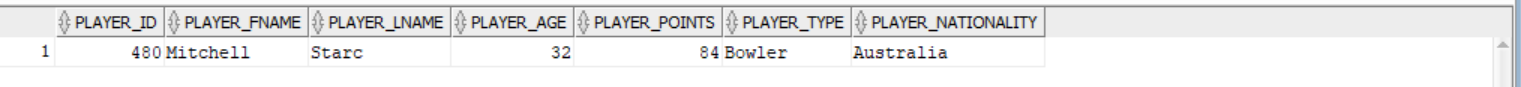
where player\_Id =(

select player\_Id from

(Select distinct player\_Id, count(player\_Id)as count from fantasyteam

group by player\_Id order by count desc

FETCH FIRST 1 ROWS ONLY));



**-- Q19 : Query to display the total points earned by the customer in a game. Display the team name and total points earned by that team**

Select c.cust\_Id,c.cust\_Fname ||' '|| c.cust\_Lname as Customer\_Name,f.team\_name,sum(points\_per\_player)as total\_Point

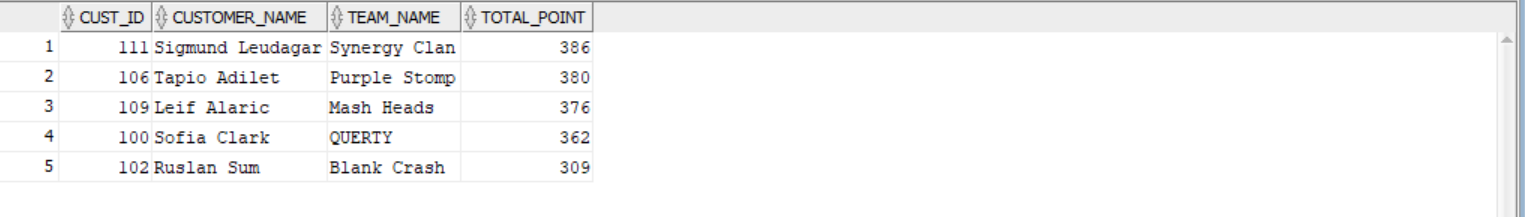
from fantasyteam f

Inner join Customer c

on c.cust\_Id = f.cust\_Id

group by c.cust\_Id,c.cust\_Fname,c.cust\_Lname,f.team\_name

order by total\_Point desc ;



**-- Q20:Query to display the details of customer firstname starting with 'S', and player selected by the customer whose lastname starts with 'S'. Also display customer TRANSACTION status and sort the result using customer's last name**

Select c.cust\_fname,c.cust\_lname,f.team\_name,p.player\_Fname,p.player\_Lname,p.player\_nationality

,t.trans\_status from customer c

Inner join fantasyteam f

on c.cust\_Id = f.cust\_Id

Inner join player p

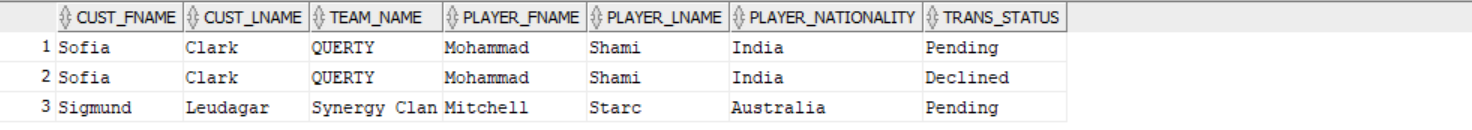
on p.player\_Id = f.player\_Id

Inner join TRANSACTION t

on t.cust\_Id = c.cust\_Id

Where c.cust\_Fname like ('S%') and p.player\_Lname like 'S%'

order by c.cust\_Lname



**References:**

Dream11 “Makes Sports Better” with 100 Million Users on AWS ([**https://aws.amazon.com/solutions/case-studies/dream11/**](https://aws.amazon.com/solutions/case-studies/dream11/) **)**

<https://www.dreamsports.group/>

https://www.dreamsports.group/dream-sports-foundation/

https://www.sqlstyle.guide/

Su, A. I. (2018). Dizeez: An online game for fans