**SET-1**

Create the following relations with the given constraints:

Event (eventid, name, description,city)

Participant (playerid, name, eventid, gender, year)

Prizes (prizeid, prize-money, eventid, rank,year)

Winners (prizeid, playerid)

1. Choose appropriate primary keys for each table.

2. Choose foreign keys wherever needed.

3. Prize-money is of format xxxx.xx and maximum prize that can be awarded is

1500.00 and minimum prize awarded is 500.00

4. Names across all tables are inserted have their first character in capitals.

5. Playerid should contain at least one digit character.

6. Event names cannot be repeated.

7. Player name, event name, prize-money and dateofevent cannot be blank.

8. Rank can take any of these values only - 1, 2, 3.

9. Prevent users from giving bogus gender values

Create the following table with the given constraints:

Event (eventid, name, description,city)

Participant (playerid, name, eventid, gender, year)

Prizes (prizeid, prize-money, eventid, rank,year)

Winners (prizeid, playerid)

1. Choose appropriate primary keys for each table.

2. Choose foreign keys wherever needed.

3. Prize-money is of format xxxx.xx and maximum prize that can be awarded is 1500.00 and minimum prize awarded is 500.00

4. Names across all tables are inserted have their first character in capitals.

5. Playerid should contain at least one digit character.

6. Event names cannot be repeated.

7. Rank can take any of these values only - 1, 2, 3.

**QUESTIONS**

1. Retrieve the name of the person who has won the highest amount of prize.

2. Retrieve the name of the events which do not have 3 prize winners.

3. Retrieve the name of all 2nd prize winners along with the event name.

4. Retrieve the names of the persons who have won the highest number of 1st, 2nd and 3rd prizes.

5. Retrieve the name of events where all prize winners are females.

6. Retrieve the name of the person who has won 1st prize of an event for maximum number of successive years. Give the event name also along with the name of the player and period for which he has won it.

7. Create a view to retrieve the names of all participants who won 1st prizes along with their event names.

9. Write a PL/SQL procedure for to insert names of all prize winners with the event and the rank into the relation Result using cursor.

10. Write a trigger to make sure that for every new event created, 3 prizes are created in prizes table. (1st prize - 1500, 2nd - 1000, 3rd 500)

**SET-2**

Create the following tables for a movie database:

Movie (id, name, yrofrelease, status, Censored)

People (id, name, movie-id, role, DOB, gender)

Awards (id, name, peopleid, movieid, yr)

(For data entry into awards enter awards name as Oscar, film-fare or Emmy)

1. Choose appropriate primary keys for all the relations (foreign keys if necessary) by considering the following

a. Lot of movies can be released in the same year and two movies released in different years can have same names.

b. More than one person can be related to a movie and one person can be related to a lot of movies.

c. More than one person can get an award for the same movie or different movie and an award can be shared between different persons.

2. Movie status depicts whether a movie is a hit, average or flop.

3. Role can be actor, actress, director or crew.

4. Names (movie cast or award) are entered in first letter capitals format.

5. Prevent users from entering random gender values

6. Year of release and year of awards are only year values whereas dob in people

table is the proper date of birth

7. Censored is a Boolean field

6. Year of release and year of awards are only year values whereas dob in people table is the proper date of birth

7. Censored is a Boolean field

Write queries for the following:

1. Display the number of hit movies released in a three consecutive year time range(choose range as per your inserted data)

2. Display name and age for all people who won Oscar award when they were less than 30 yrs old.

3. Display movie name and award name for movies which have won Emmy for actor, actress and director in same year.

4. Display for each year the highest award winning movie name and number of awards won.

5. Display the name of directors who have won an award for a censored film.

6. Display award name and year for all awards that were shared between different movies.

7. Create a view to display the serial No., Movie name, year of release, director name. Serial No. is generated through a sequence. The display is should be in the increasing order of year of release.

8. Write a pl/sql procedure to display data in following order:

Actor / actress name Movie Name Director Name Awards Won

If no awards were won by the actor/actress, then leave the field blank. (You can use cursors for this)

9. Write a trigger to accomplish the following:

When a movie is deleted from movie table, all its references need to be deleted from people and awards table

Create the following tables for a movie database:

Movie (id, name, yrofrelease, status, Censored)

People (id, name, movie-id, role, DOB, gender)

Awards (id, name, peopleid, movieid, yr)

(For data entry into awards enter awards name as Oscar, film-fare or Emmy)

1. Choose appropriate primary keys for all the relations (foreign keys if

necessary) by considering the following

a. Lot of movies can be released in the same year and two movies

released in different years can have same names.

b. More than one person can be related to a movie and one person can be

related to a lot of movies.

c. More than one person can get an award for the same movie or different

movie and an award can be shared between different persons.

2. Movie status depicts whether a movie is a hit, average or flop.

3. Role can be actor, actress, director or crew.

4. Names (movie cast or award) are entered in first letter capitals format.

5. Prevent users from entering random gender values