Chapter 7 Questions

1. Create a view named MAINE\_TRIPS. It consists of the trip ID, trip name, start location, distance, maximum group size, type, and season for every trip located in Maine (ME).
   1. Write and execute the CREATE VIEW command to create the MAINE\_TRIPS view.
      1. A screenshot of a computer

         Description automatically generatedCREATE VIEW MAINE\_TRIPS AS SELECT TRIP\_ID, TRIP\_NAME, START\_LOCATION, DISTANCE, MAX\_GRP\_SIZE, TYPE, SEASON FROM TRIP WHERE STATE='ME';
   2. Write and execute the command to retrieve the trip ID, trip name, and distance for every Biking trip.
      1. A screenshot of a computer

         Description automatically generatedSELECT TRIP\_ID, TRIP\_NAME, DISTANCE FROM MAINE\_TRIPS WHERE TYPE='Biking';
   3. Write and execute the query that the DBMS actually executes.
      1. A screenshot of a computer

         Description automatically generatedSELECT TRIP\_ID, TRIP\_NAME, DISTANCE FROM TRIP WHERE STATE='ME' AND TYPE='Biking';
   4. Does updating the database through this view create any problems? If so, what are they? If not, why not?
      1. Updating the database through this view does not create and problems. The reason for this being is because the view is a row and column subset that has a primary key enabled in each of the tables which helps in updating the database using the views.
2. Create a view named RESERVATION\_CUSTOMER. It consists of the reservation ID, trip ID, trip date, customer number, customer last name, customer first name, and phone number.
   1. Write and execute the CREATE VIEW command to create the RESERVATION\_ CUSTOMER view.
      1. A screenshot of a computer

         Description automatically generatedCREATE VIEW RESERVATION\_CUSTOMER AS SELECT RESERVATION\_ID, TRIP\_ID, TRIP\_DATE, RESERVATION.CUSTOMER\_NUM, LAST\_NAME, FIRST\_NAME, PHONE FROM RESERVATION, CUSTOMER WHERE RESERVATION.CUSTOMER\_NUM = CUSTOMER.CUSTOMER\_NUM;
   2. Write and execute the command to retrieve the reservation ID, trip ID, trip date, and customer last name for every reservation in the RESERVATION\_CUSTOMER view with a trip date of September 11, 2016.
      1. A screenshot of a computer

         Description automatically generatedSELECT RESERVATION\_ID, TRIP\_ID, TRIP\_DATE, LAST\_NAME FROM RESERVATION\_CUSTOMER WHERE TRIP\_DATE='9/11/2016';
   3. Write and execute the query that the DBMS actually executes.
      1. SELECT RESERVATION\_ID, TRIP\_ID, TRIP\_DATE, LAST\_NAME FROM RESERVATION, CUSTOMER WHERE RESERVATION.CUSTOMER\_NUM = CUSTOMER.CUSTOMER\_NUM AND TRIP\_DATE = '9/11/2016';

A screenshot of a computer

Description automatically generated

* 1. Does updating the database through this view create any problems? If so, what are they? If not, why not?
     1. Updating the database through this view will create problems. The reason for this is because the view uses join other than primary key and updating the database isn't possible via this view.

1. Create a view named TRIP\_INVENTORY. It consists of the state and the total number of trips for each state. Use UNITS as the column name for the total number of trips for each state. Group and order the rows by state.
   1. Write and execute the CREATE VIEW command to create the TRIP\_INVENTORY view.
      1. A screenshot of a computer

         Description automatically generatedCREATE VIEW TRIP\_INVENTORY AS SELECT STATE, COUNT(\*) AS UNITS FROM TRIP GROUP BY STATE ORDER BY STATE;
   2. Write and execute the command to retrieve the state and units for each state having more than 10 trips.
      1. A screenshot of a computer

         Description automatically generatedSELECT \* FROM TRIP\_INVENTORY WHERE UNITS > 10;
   3. Write and execute the query that the DBMS actually executes.
      1. SELECT STATE, COUNT(\*) AS UNITS FROM TRIP GROUP BY STATE HAVING COUNT(\*) > 10 ORDER BY STATE;

A screenshot of a computer

Description automatically generated

* 1. Does updating the database through this view create any problems? If so, what are they? If not, why not?
     1. With updating the database through this view will create problems. It is because this view uses statistics other than the primary key and because of that, updating the database is not possible.

1. Write, but do not execute, the commands to grant the following privileges:
   1. User Rodriquez must be able to retrieve data from the TRIP table.
      1. GRANT SELECT ON TRIP TO RODRIGUEZ;
   2. Users Gomez and Liston must be able to add new reservations and customers to the database.
      1. GRANT INSERT ON RESERVATION TO GOMEZ, LISTON;
      2. GRANY INSERT ON CUSTOMER TO GOMEZ, LISTON;
   3. Users Andrews and Zimmer must be able to change the price of any trip.
      1. GRANT UPDATE (PRICE) ON RESERVATION TO ANDERS, ZIMMER;
   4. All users must be able to retrieve the trip name, start location, distance, and type for every trip.
      1. GRANT SELECT (TRIP\_NAME, START\_LOCATION, DISTANCE, TYPE) ON TRIP TO PUBLIC;
   5. User Golden must be able to add and delete guides.
      1. GRANT INSERT ON GUIDE TO GOLDEN;
   6. User Andrews must be able to create an index for the TRIP table.
      1. GRANT INDEX ON TRIP TO ANDREWS;
   7. Users Andrews and Golden must be able to change the structure of the CUSTOMER table.
      1. GRANT ALTER ON CUSTOMER TO GOLDEN, ANDREWS;
   8. User Golden must have all privileges on the TRIP, GUIDE, and TRIP\_GUIDES tables.
      1. GRANT ALL ON TRIP TO GOLDEN;
      2. GRANT ALL ON GUIDE TO GOLDEN;
      3. GRANT ALL ON TRIP\_GUIDES TO GOLDEN;
2. Write, but do not execute, the command to revoke all privileges from user Andrews.
   1. REVOKE ALL PROVILEGES FROM ANDREWS;
3. Create the following indexes:
   1. Create an index named TRIP\_INDEX1 on the TRIP\_NAME column in the TRIP table.
      1. CREATE INDEX TRIP\_INDEX1 ON TRIP (TRIP\_NAME);

A screenshot of a computer

Description automatically generated

* 1. Create an index named TRIP\_INDEX2 on the TYPE column in the TRIP table.
     1. A screenshot of a computer

        Description automatically generatedCREATE INDEX TRIP\_INDEX2 ON TRIP(TYPE);
  2. Create an index named TRIP\_INDEX3 on the LAST\_NAME and FIRST\_NAME columns in the CUSTOMER table.
     1. A screenshot of a computer

        Description automatically generatedCREATE INDEX TRIP\_INDEX3 ON CUSTOMER(LAST\_NAME, FIRST\_NAME);

1. Delete the index named TRIP\_INDEX3.
   1. A screenshot of a computer

      Description automatically generatedDROP INDEX TRIP\_INDEX3;
2. Write the commands to obtain the following information from the system catalog. Do not execute these commands unless your instructor asks you to do so.
   1. List every column in the GUIDE table and its associated data type.
      1. A screenshot of a computer

         Description automatically generated with medium confidenceSELECT COLUMN\_NAME, DATA\_TYPE FROM ALL\_TAB\_COLUMNS WHERE TABLE\_NAME = 'GUIDE';
   2. List every table that contains a column named TRIP\_ID.
      1. A screenshot of a computer

         Description automatically generatedSELECT TABLE\_NAME FROM ALL\_TAB\_COLUMNS WHERE COLUMN\_NAME = 'TRIP\_ID';
   3. List the table name, column name, and data type for the columns named TRIP\_ID, TRIP\_NAME, and TYPE. Order the results by table name within column name. (That is, column name is the major sort key and table name is the minor sort key.)
      1. A screenshot of a computer

         Description automatically generated with medium confidenceSELECT TABLE\_NAME, COLUMN\_NAME, DATA\_TYPE FROM ALL\_TAB\_COLUMNS WHERE COLUMN\_NAME = 'TRIP\_ID' OR COLUMN\_NAME ='TRIP\_NAME' OR COLUMN\_NAME = 'TYPE' ORDER BY COLUMN\_NAME, TABLE\_NAME;
3. Add the CUSTOMER\_NUM column as a foreign key in the RESERVATION table.
   1. ALTER TABLE RESERVATION ADD FOREIGN KEY (CUSTOMER\_NUM) REFERENCES CUSTOMER(CUSTOMER\_NUM);

A screenshot of a computer

Description automatically generated

1. Ensure that the only legal values for the TYPE column in the TRIP table are Biking, Hiking, or Paddling.
   1. ALTER TABLE TRIP ADD CHECK(TYPE IN('Biking', 'Hiking', 'Paddling'));

A screenshot of a computer

Description automatically generated

1. The maximum group size for a trip must be greater than one. Use the Internet to research the CHECK clause, determine how to add this constraint to the TRIP table, and then write the SQL command to make this change to the TRIP table. Be sure to cite your references.
   1. A screenshot of a computer

      Description automatically generatedALTER TABLE TRIP ADD CHECK (MAX\_GRP\_SIZE > 1);
   2. W3Schools. SQL CHECK Constraint. https://www.w3schools.com/sql/sql\_check.asp