Exercise 3   Primary Key, Foreign Key & Normalization

Consider the Insurance database given below. The primary keys are underlined and the data types are specified:

  PERSON (driver – id #: String, name: string, address: string)

  CAR (regno: string, model: string, year: int)

  ACCIDENT (report-number: int, accd-date: date, location: string)

  OWNS (driver-id #:string, Regno:string)

  PARTICIPATED (driver-id: string, Regno:string, report-number:int, damage amount:int)

Demonstrate how you:

1. **Create the above tables by properly specifying the primary keys and the foreign keys.**

2. **Enter at least five tuples for each relation.**

3. **Update the damage amount to certain amount (say 25000) for the car with a specific Regno in the ACCIDENT table with certain report- number.**

4.   **Add a new accident to the database.**

5. **Find the total number of people who owned cars that were involved in accidents in certain years (say 2008).**

6. **Find the number of accidents in which cars belonging to a specific model were involved.**

7. **Display the owners and their car details.**

Objective:
The objective of this exercise is to enable you to understand SQL relational concepts – Primary Key and Foreign Key and the concept of normalization.

Procedure and description:
Primary Key is a column or a combination of columns that uniquely identifies each row in a table. Foreign Key is a column or a combination of columns whose values match a Primary Key in a different table. In the most common scenario the relationship between 2 tables matches the Primary Key in one of the tables with a Foreign Key in the second table.

There are 3 types of relations between tables – One-To-Many, Many-To-Many and One-To-One. In One-To-Many relation a row in one of the tables can have many matching rows in the second table, but a row the second table can match only one row in the first table. In the Many-To-Many relation, many rows from the first table can match many rows in the second and the other way around.

The process of removing redundant data by creating relations between tables is known as Normalization. Normalization process uses formal methods to design the database in interrelated tables.

Algorithm: The steps for this exercise are given below:

Step – 1:   Start

Step – 2:   Create Database, tables using CREATE commands with its essential attributes.

Step – 3:   Insert the values using INSERT INTO statements. (Insert the suitable values (tuples) so that queries are executed correctly.)

Step – 4:   Execute different Commands and extract information from the table. (Hint: use keywords like PRIMARY, FOREIGN, FROM, WHERE, DISTINCT, MINUS, GROUP BY, ORDERBY etc. You can use suitable operators like AND & OR for certain conditions to meet)

Step – 5:   Execute and check.

Expected Output: Creation of database, tables with the use of primary keys, foreign keys and its modification through SQL commands.