**VEHICLE INSURANCE REPORT**

**Data Retrieval & Joins**

**1.Write a SQL query to retrieve all models associated with a given make (e.g., "Toyota").**

SELECT

m.model\_desc

FROM

model m

JOIN make mk ON m.make\_id = mk.make\_id

WHERE

mk.make\_desc = 'Toyota';

A screenshot of a computer

Description automatically generated

**2.List all brokers and their associated sales agents:**

Use a JOIN clause to retrieve the Broker\_name and Broker\_org\_name from the Broker table and the First\_name, Last\_name of associated sales agents from the Personal\_Information and Login\_User tables.

SELECT

b.broker\_name,

b.broker\_org\_name,

p.first\_name,

p.last\_name

FROM

broker b

JOIN login\_user lu ON lu.lead\_id = b.broker\_id

JOIN personal\_information p ON lu.user\_id = p.user\_id

WHERE

lu.user\_type = 'Sales Agent';

A screenshot of a computer

Description automatically generated

**3.Find all users in a specific state:**

Retrieve user information (e.g., First\_name, Last\_name, email,state) from Personal\_Information for users residing in a particular state.

SELECT

p.first\_name,

p.last\_name,

p.email,

p.state

FROM

personal\_information p

WHERE

p.state = 'TX';

A screenshot of a computer

Description automatically generated

**4.Get contact information for all brokers:**

Extract the Broker\_name, Broker\_org\_name, and contact\_info from the Broker table.

SELECT

broker\_name,

broker\_org\_name,

contact\_info

FROM

broker;

A screenshot of a computer

Description automatically generated

**Data Updates & Modifications**

**1.Update the status of a model to 'deactive'.**

UPDATE model

SET

status = 'deactive'

WHERE

model\_id = 25;

A screenshot of a computer

Description automatically generated

**2.Change the contact information for a specific broker.**

UPDATE broker

SET

contact\_info = 'New Address, Suite 500, New City, CA 90210

Phone: 555-9876

Email: new\_email@example.com'

WHERE

broker\_id = 1;

A screenshot of a computer

Description automatically generated

**3.Deactivate the login of a user.**

UPDATE login\_user

SET

status = 'DEACTIVATED'

WHERE

login\_id = 9;

A screenshot of a computer

Description automatically generated

**Data Analysis & Reporting**

**1.Count the number of active and inactive models for each make.**

SELECT mk.make\_desc,

COUNT(CASE WHEN m.status = 'active' THEN 1 END) AS active\_models,

COUNT(CASE WHEN m.status = 'deactive' THEN 1 END) AS inactive\_models

FROM Make mk

JOIN Model m ON mk.make\_id = m.make\_id

GROUP BY mk.make\_desc;

A screenshot of a computer

Description automatically generated

**2.Determine the total number of brokers, sales agents, and users.**

SELECT

(SELECT COUNT(\*) FROM Login\_User WHERE User\_Type = 'Broker') AS Broker\_Count,

(SELECT COUNT(\*) FROM Login\_User WHERE User\_Type = 'Sales Agent') AS Sales\_Agent\_Count,

(SELECT COUNT(\*) FROM Login\_User WHERE User\_Type = 'User') AS User\_Count

FROM DUAL;

A screenshot of a computer

Description automatically generated

**3.Find the number of users in each city.**

SELECT p.state, COUNT(\*) AS User\_Count

FROM Personal\_Information p

GROUP BY p.state;

A screenshot of a computer

Description automatically generated

**4.Generate a report of all brokers and their associated sales agents.**

SELECT b.Broker\_name, b.Broker\_org\_name, p.First\_name, p.Last\_name

FROM Broker b

JOIN Login\_User lu ON b.Broker\_id = lu.Lead\_Id

JOIN Personal\_Information p ON lu.User\_Id = p.User\_id

WHERE lu.User\_Type = 'Sales Agent'

ORDER BY b.Broker\_name;

A screenshot of a computer

Description automatically generated

**Advanced Queries**

**1.Retrieve the top 5 states with the highest number of users.**

SELECT p.state, COUNT(\*) AS User\_Count,

RANK() OVER (ORDER BY COUNT(\*) DESC) AS User\_Count\_Rank

FROM Personal\_Information p

GROUP BY p.state;

