

Auto Insurance Company CRM Database Design for Policy Management Project Report



University of Connecticut
Business Analytics & Project Management
OPIM 5272 - Data Management & Business Process Modeling

Team UDBP

Himadri Chobisa
Kamal Krishnan
Kashif Iqbal
Namita Singh

Table of Contents

1. Introduction	3
2. Project Overview	3
3. Business Rules and User Requirement	3
3.1 Business Rules	3
3.2 User Requirements	4
4. EER Diagram	5
5. Relational Schema	6
6. Data Dictionary	8
7. Output of DDL SQL Statements	14
8. Output of DML SQL Statements	26
9. Additional Features using PL/SQL	32
9.1 Calculate and Print Revenue using “Cursor” and “IF..THEN” statements	32
9.2 Audit Log using “Trigger” Command	34

1. Introduction

This document aims to describe the relational database management of the customer relationship (CRM) for policy management in an auto insurance company. The following sections provide an overview and scope of the project, along with sections that provide business rules and user requirements. It is followed by ERD (entity relationship diagram), schema which represents a visual form of relational database followed by data dictionary (which lists all the entities, their attributes, and the relationships between different entities), SQL statements (DDL) written to CREATE TABLE, INSERT data and advanced SQL statements (DML) using JOINS. The last section describes a few additional features developed using the cursor, IF..THEN and trigger features in PL/SQL.

2. Project Overview

This project is to create a database for Customer Relationship Management (CRM) of an Auto insurance company to drive insightful analysis about policy management. This database will consist of key entities like Policy coverage, Vehicle details, Driver details, Payment details, Incident records, etc. We intend to reduce the redundancy in the data and make it more efficient for use on websites by the policyholders of the insurance company.

3. Business Rules and User Requirement

3.1 Business Rules

An auto insurance company wants to build a database to manage details of policies, quotes, customers, insured vehicles, billing procedure, incidents, claims, etc.

1. Customers are identified by unique customer id and the company wants to store the first name, middle name, last name, address, phone no, email, and gender of all the customers. Each customer will have one policy only.
2. A single policy can insure multiple family members. Each policy will have a unique policy number with amendment options. The company offers a default term of 6 months with an option to extend for as long as 12 months and further renewals or amendments. Policy's effective date and expiry date will be recorded. The company offers 3 distinct product types (basic, silver and gold) with predefined liabilities and coverage details of bodily injury, property damage, comprehensive, collision and emergency bundled in them. A customer can choose any of the offered 3 packages over a period of time.
3. Vehicle is identified by its unique VIN number and other details like make, model, year, color, miles, usage type (commercial/personal), registration number etc. collected from customers. A vehicle may have one or more loans on it, which are stored as Lien details with the holder's name and record date, in the database. Each policy will have one insured vehicle. A vehicle may be insured under different policies over the period of time.

4. Multiple drivers of a vehicle may be insured under the same policy. License details of all drivers will be stored in the database. License details like first issue date, latest issue date, expiry date, issue state will be tracked. Date of birth of a customer will also be stored from the license. Records of all licenses (old & new, US & international) issued to a customer will be stored by the company.
5. Users/potential customers can apply for quotes (valid for 14 days) in office / phone. A customer can ask for multiple quotes for the same vehicle before finalizing the policy purchase. Discounts are offered during quote generation. One agent will be associated with one quote generation at a time with chosen product type, quote issue and expiry date.
6. Bill details of each policy will be stored in the database with amount, cycle, issue date, due date, deductible, and status. A policy will have bills as per the cycle selected. A bill will have a distinct bill number with one insurance policy. Receipts will be generated every time a bill is paid with a paid amount and date stored with it.
7. Company accepts payment by both credit card and bank account. Company needs to store card details like card number, card type, cvc, issue bank and expiry details. Bank account no, bank name and routing number are also stored in the company database as a bill can be paid by any type of payment methods over a period of time.
8. Company will keep track of all the claims (with unique claim number) associated with a policy with date of claim, type of damage, amount of claim. There is no upper limit on the number of claims attached to a policy. Settled claims will be stored separately with amount and date of settlement.

3.2 User Requirements

1. Users can choose to pay bills (premium and/or claims) either from credit card or bank account in single or recurring bill cycles.
2. Users can apply for claims for incidents like parking lot damage, accidents, roadside assistance, prop damage etc. and track it with a unique claim number.
3. Users can apply for auto insurance with either a US or international driving license.
4. Users can select an agent (office based) based on agent ratings.
5. A user can easily switch between product types as a unique amendment id is generated for each change. Policy renewals are also easily done.
6. Users can switch banks for loan and update the information with the unique lien id.
7. A user can benefit from the company's safe driver policy and earn a considerable discount if they meet the criteria.

5. Relational Schema

The relational schema for all tables is in 3 NF form. It is shown below:

Customer	<u>Cust_Id</u>	First_Nm	Middle_Nm	Last_Nm	Street_Name	Apt_No	City	State	Zip	Mobile_Phone	Home_Phone	Email	Gender	SSN
Insu_Policy	<u>Policy_No</u>	Agreement	Effective_Date	Expiration_Date	Premium_Amount	Policy_Term	Safe_Drive (Y/N)							
Policy_Amend	<u>Amned_Id</u>	Amend_date												
Ammend_type	<u>Policy_no</u>	Addition	Deletion	Renewal										
Policy_Coverage	<u>Policy_No</u>	<u>Prod_type_Id</u>	Policy_Status	Policy_start_date	Policy_End_Date									
Product_Type	<u>Prod_Type_Id</u>	Bodily_Injury	Prop_Damage	Comprehensive	Uninsured	Collison	Emergency							
Product_Type_Name	<u>Prod_Type_Id</u>	Basic	Silver	Gold										
Quote	<u>Quote_Id</u>	Amount	Issue_Date	Expiry_Date	Discount									
Insu_Agent	<u>Agent_Id</u>	Agent_Nm	Ratings	Off_Location	Salary									
Vehicle	<u>VIN_no</u>	Car_Make	Car_Model	Car_year	Vehicle_type	Miles	Airbags	Color						
Vehicle_driver	<u>License_No</u>	<u>VIN_No</u>	Registration_No	DMV_Plate_No	Issue_Date	Expiry_Date	Purchase_Date							
Vehicle_Lien	<u>Lien_Id</u>	<u>VIN_no</u>	Lien_holder	Record_date										
License_Informantion	<u>License_No</u>	Issue_Date	Expiry_Date	No_of_Years	Date_of_Birth	State_of_Issue	US_License	International_License						
Violation_History	<u>License_No</u>	Active_Violation	Expired_Violation											

Auto Insurance Company CRM Database Design for Policy Management



6. Data Dictionary

This section describes the details of the Entities and Attributes of the logical database design.

OBJECT NAME	OBJECT TYPE	OBJECT DEFINITION
Insu_Policy	Entity	Contains information on auto insurance policy and its related terms
Policy_no (Primary Key)	Attribute	Unique identification no. for entity Insu_policy
Effective_date	Attribute	First Date of policy coming into effect
Exp_date	Attribute	Policy expiration date
Policy_term	Attribute	Policy duration-Derived value from policy issue date and expiration date
Safe_drive	Attribute	Does the driver qualify for a discount as a safe drive. It is applicable if policyholder subscribed for SafeDrive option
Agreement	Attribute	Legal policy document with contract b/w customer and insurance company.
Premium_amount	Attribute	Amount to be paid by the customer for insurance policy
Insu_Policy -- > PolicyAmend	Relationship	One policy may have no or many amendments (renewal, change, cancellation) during the policy term. One policy amendment is always referring to one and one policy only.
Insu_Policy -- > PolicyCoverage	Relationship	One policy may be one or many Product types selected by Policyholder during policy term. It is linked to associate entity PolicyCoverage. A given PolicyCoverage is linked to one and only one policy.
Insu_Policy -- > Vehicle	Relationship	One policy is linked to one and one vehicle only. The insured vehicle may have zero or many different policies over the period of time.
Insu_Policy -- > Customer	Relationship	One policy has one primary insured person and may have more drivers added. The primary insured person to have one policy only.
Insu_Policy -- > Bill	Relationship	One policy can have 1 or many Bills. One bill is linked to one and one policy only.
Insu_Policy -- > Claim	Relationship	One policy may have zero or many Claims. One Claim is linked to one and one policy only.
Insu_Policy -- > Payment Method	Relationship	One policy may have zero or many Payment Methods associated. One PaymentMethod is linked to one and one policy only.
Customer	Entity	Contains client details on demographics
Cust_Id (Primary Key)	Attribute	The customer entity is uniquely identified by CustomerID
Nm(First,middle, last)	Attribute	First, Middle and last name of customer soliciting the policy

Auto Insurance Company CRM Database Design for Policy Management

Address	Attribute	Complete postal address of customer (First line, Second line, City, State, Zip)
Phone	Attribute	Customer's phone number (Home, Mobile, Office)
Email	Attribute	Customer's email id
SSN	Attribute	Customer's SSN no- saved safely in database
Gender	Attribute	Customer's gender
Customer ---> License	Relationship	One customer may have one or many US driving license information over a period of time. But for a given policy only one active US driving license information will be used. Each License information is associated with one and only Customer.
Customer---> Quote	Relationship	1:M Relationship. One quote is specific to one customer and one customer can get multiple quotes over time.
Product_type	Entity	Contains details about the different Insurance Product Types with varying coverage limits.
Product_type_ID (Primary Key)	Attribute	The Product Type entity is uniquely identified by Product Type ID.
Product_type_name	Attribute	Name of the 3 product types (pre-bundled) offered by the company - Basic, Silver, Gold
Bodily Injury	Attribute	Bodily injury coverage limit for different product type are: <ul style="list-style-type: none"> ● Basic- limit of max \$25,000 each person/\$50,000 each accident; ● Silver - limit of \$25,000 each person/\$50,000 each accident ● Gold- limit of \$50,000 each person/\$100,000 each accident.
Prop_Damage	Attribute	Property damage liability coverage; <ul style="list-style-type: none"> ● Basic- limit of \$25,000 each accident; ● Silver- limit of \$25,000 each accident; ● Gold- limit of \$25,000 each accident
Uninsured	Attribute	<ul style="list-style-type: none"> ● Basic- limit of \$25,000 each person/\$50,000 each accident ● Silver- limit of \$25,000 each accident ● Gold- limit of \$50,000 each person/\$100,000 each accident
Comprehensive	Attribute	Comprehensive coverage ; <ul style="list-style-type: none"> ● Not offered in basic plan; ● Silver- limit of Actual cash value with deductible of \$500; ● Gold- Actual cash value with deductible of \$100
Collison	Attribute	Collision coverage; <ul style="list-style-type: none"> ● Not offered in basic plan; ● Silver- limit of Actual cash value with deductible of \$500 ● Gold- Actual cash value with deductible of \$300
Emergency	Attribute	Emergency; <ul style="list-style-type: none"> ● Not offered in basic;

Auto Insurance Company CRM Database Design for Policy Management

		<ul style="list-style-type: none"> Silver- Emergency road assistance; Gold- Emergency road assistance
Product Type -- > Policy Coverage	Relationship	One product type may be associated with many Policy Coverage. For a particular Policy Coverage at any given time will have only one Product Type.
Product type--->Quote	Relationship	One product type can have multiple quotes and one specific quote is linked to a particular product type
Policy_Coverage	Associative Entity	Associative entity between Policy and product type- Details of the product type and coverage selected by customer
Policy_no (PK, FK)	Composite Key	Used as composite primary key
Product_type_ID (PK, FK)	Composite Key	Used as composite primary key
Policy_status	Attribute	Status of policy (Issued/expired)
Policy_start_date	Attribute	Initiation date of the current policy purchased
Policy_end_date	Attribute	Expiration date of the current policy purchased
Policy_Amend	Entity	Contains information on changes in policy
Amend_ID (Primary key)	Attribute	Unique policy ids
Ammend_date	Attribute	Date amendment effective from
Ammend_type	Attribute	Type of amendment chosen - Renew/ Addition/ Deletion
License	Entity	Lists down information and history about the License details of the driver.
License_no. (Primary Key)	Attribute	Unique License number
First_Issue_date	Attribute	First issue date of License
Latest_Issue_date	Attribute	Most recent issue date of License
Expiry date	Attribute	Date of expiry of License
Experience_years	Attribute	Tot no. of years of driving experience
DOB	Attribute	DOB of the license holder
Violation_hist	Attribute	Record of all violations associated with the License Holder
Issue_State	Attribute	Name of the state where License was issued
Issue_Country	Attribute	Country of License issuance , whether domestic in US or international
License ----> Vehicle driver	Relationship	1:M from license to vehicle driver which is an associative entity between license and vehicle
License ---> Incident record	Relationship	Mandatory one to optional many from license info to incident record; that is there may be none or many incidences on one license id but a particular incident (incident no) will be associated with one license id only
Incident_Record	Entity	Gives detailed information about any incident/violation related to the policy and License holder.
Incident_no (Primary key)	Attribute	Unique Incident number for each incident
Incident_Location	Attribute	The actual area/location where the incident occurred.
Incident_Date	Attribute	Date on which the incident occurred

Auto Insurance Company CRM Database Design for Policy Management

Incident_Time	Attribute	Time at which the incident took place
Incident_type (Speeding, Parking, Accident, Roadside, Property Damage)	Attribute	Lists the major categories of incidents and violations that can be associated with a particular event for which claim is raised
Incident_Record--->Vehicle	Relationship	One Incident record can only be associated with only one Vehicle and one vehicle can have one or more records over the time
Vehicle	Entity	Contains information about insured vehicles.
VIN_no (Primary key)	Attribute	Unique identification number for a vehicle-Vehicle identification number (VIN) is a unique code that is assigned to every motor vehicle when it's manufactured.
Car_Make	Attribute	Manufacturer of the vehicle
Car_Model	Attribute	Name of the vehicle model
Car_Year	Attribute	Year of manufacturing of the vehicle
Color	Attribute	Color of the vehicle
Miles	Attribute	Total no. of miles accumulated on vehicle
Vehicle_type (Personal/Commercial)	Attribute	Type of vehicle used whether it's for personal use or of commercial use.
Airbags present/absent	Attribute	If safety feature in the form of airbag is present or not
Vehicle ---> Vehicle driver	Relationship	1:Many relation from vehicle to vehicle driver which is an associative entity between license information and vehicle
Vehicle ---> Vehicle_lien	Relationship	A vehicle can have one or more Liens registered over a period of time and one lien can be assigned to only one vehicle
Vehicle_Lien	Entity	Describes mortgage details of the vehicle to be insured
Lien_ID (Primary key)	Attribute	Uniquely identifies the lien records of the vehicle
Lien_holder	Attribute	Lists the specific bank/mortgage institution's information
Record_date	Attribute	Date of incorporation of lien
Vehicle_Driver	Associative entity	Associative entity b/w License information and driver
License_No. (PK,FK)	Composite key	Reference from License information table
VIN_no (PK,FK)	Composite key	Reference from Vehicle table
Registration_no	Attribute	Registration no. assigned to the vehicle
DMV_plate_no	Attribute	Plate no. issued by the state DMV
Issue_date	Attribute	First date of issue of registration no.
Expiry_date	Attribute	Date of expiry of registration no.
Purchase_date	Attribute	Date of purchase of vehicle
Insu_Agent	Entity	Information of agent soliciting policy to the client.
Agent_ID (Primary key)	Attribute	Uniquely identifies the Entity agent

Auto Insurance Company CRM Database Design for Policy Management

Agent_nm	Attribute	Name of the employee/agent soliciting the policy to the client
Off_location	Attribute	Which work location is the agent operating from, either Online or from Office
Ratings	Attribute	What is the rating of the agent based on customer reviews
Salary	Attribute	Fixed Salary of the agent
Insu_Agent--->Quote	Relationship	1:M Relationship. One agent can generate multiple quotes and one quote will be specific to an agent
Quote	Entity	Contains information on different quote amounts for different packages.
Quote_ID (Primary key)	Attribute	Uniquely identified by Quote ID
Amount	Attribute	Quotes for different type of pre-bundled packages eg- Basic, Silver and Gold
Issue_date	Attribute	First issue date of quote generated
Expiry_date	Attribute	Expiry date of issued quote
Discount	Attribute	Types of discount given to the customer either flat rate discount or via bundled packages
Payment_method	Entity (Supertype)	Contains the details of various payment methods provided by the customer.
Pay_method_ID (Primary key)	Attribute	Uniquely identified by Payment Method ID which lists method of payment towards policy.
Payment_Method - --> Payment_Receipt	Relationship	1:Many from Payment Method to Payment Receipt; that is a Payment Method (like credit card) can be used to pay for one or many bills and one or many receipts are generated. One generated Payment Receipt shall be linked to one and only PaymentMethod.
Credit_card	Entity(Subtype)	Contains credit card records used for policy payments
Pay_method_id (PK,FK)	Attribute	Reference from payment_method
Card_no	Attribute	16 digit unique identification no.of card
Holder_name	Attribute	Name of the person holding the credit card.
Holder_address	Attribute	Address of the person holding the credit card.
Card_type	Attribute	Type of card (Master, Visa, Amex, Discover, Diner)
CVV	Attribute	The 3 digit CVV number of the credit card
Issue_bank	Attribute	Name of bank issuing credit card
Exp_detail	Attribute	Card expiry in mmyy format. Month (2 digits) and Year (2 digits)
Bank_account	Entity(Subtype)	Contains issuing bank account information
Pay_method_id (PK,FK)	Attribute	Reference from payment_method
Account_No	Attribute	Account number of Checking account
Holder_nm	Attribute	Name of the Checking Account holder.
Bank_nm	Attribute	Name of bank where account holder have checking account
Routing_no	Attribute	Routing number of the Checking account

Auto Insurance Company CRM Database Design for Policy Management

Bill	Entity	Each policy will have many bills generated.
Bill_no (Primary key)	Attribute	These bills are identified by a unique Bill number.
Bill_amount	Attribute	Amount in the generated bill
Bill_cycle	Attribute	Cycle of bill generation(Monthly/quarterly/annualy etc)
Bill_date	Attribute	Date of the Bill generated
Due_date	Attribute	Final date of payment of the bill
Deductible	Attribute	Amount of deductible as per the policy
Status	Attribute	Status of payment (pending / paid partial / paid full)
Bill --- > Payment Receipt	Relationship	1:Many from Bill to Payment Receipt; that is a Bill can have one or many Payment Receipts. One generated Payment Receipt shall be linked to one and only one Bill.
Payment_receipt	Entity	Associative entity between payment method and bill and lists down details of receipt of payment
Pay_rcpt_id (Primary key)	Attribute	This is uniquely to identify receipt.
Paid_amount	Attribute	Amount paid by policyholder
Pay_date	Attribute	Date the payment is done
Claim	Entity	Contains information on the type of claims generated on this policy.
Claim_No (Primary key)	Attribute	This is uniquely identified by Claim Number.
Claim_date	Attribute	Date of claim
Amount	Attribute	Amount of claim
Status	Attribute	Status of claim (Approved / Rejected)
Claim - -- > Settlement	Relationship	One claim may have zero or many settlements. One settlement will be related to one claim only.
Settlement	Entity	Lists the settlement record associated with a particular claim.
Settlement_id (Primary key)	Attribute	This is uniquely identified by Settlement ID.
Settle_date	Attribute	Date of settlement
Amount	Attribute	Amount of settlement
Status	Attribute	Status of settlement (Pending / Paid)

7. Output of DDL SQL Statements

In this section, the details of the various entities are shown. The output of the DESCRIBE and SELECT * FROM <entity> are shown.

Entity 1: Customer

Note: Total of 14 customers information was stored in this table. As there are many attributes, the output of SELECT is shown in 2 split tables.

Name	Null?	Type
-----	-----	-----
CUST_ID	NOT NULL	NUMBER (38)
POLICY_NO		VARCHAR2 (200)
FIRST_NM		VARCHAR2 (200)
MIDDLE_NM		VARCHAR2 (200)
LAST_NM		VARCHAR2 (200)
ADDRESS		VARCHAR2 (200)
APT_NO		VARCHAR2 (20)
CITY		VARCHAR2 (200)
STATE_NM		VARCHAR2 (200)
ZIP		VARCHAR2 (20)
PHONE		VARCHAR2 (20)
MOBILE		VARCHAR2 (20)
EMAIL		VARCHAR2 (200)
SSN		VARCHAR2 (20)
GENDER		VARCHAR2 (200)

	CUST_ID	POLICY_NO	FIRST_NM	MIDDLE_NM	LAST_NM	ADDRESS	APT_NO	CITY	STATE_NM
1	150189001		Tom	Mathew	Cruise	47 W. Pulaski Rd	(null)	Joliet	IL
2	150289002		Angelia	Rachel	Green	2111 Barnes Street	(null)	Orlando	FL
3	1503 (null)		Amanda	Flower	Fisher	4172 Brooklyn Street	1E	Berryville	VA
4	160489003		Edward	(null)	Mills	1030 Douglas Dairy Road	1567	Gate City	VA
5	1705 (null)		Grace	Gabrielle	Garcia	1224 Blue Spruce Lane	(null)	Odenton	MD
6	1706 (null)		Evelyn	Emily	Sanders	2494 Caldwell Road	7E	Rochester	NY
7	1707 (null)		Robert	Lawrence	Gross	2545 Maple Lane	8G	Huntsville	AL
8	170889004		Heather	Gale	Urso	3803 Oak Street	(null)	Syracuse	NY
9	180989005		Robert	Smith	Bryson	3141 Coppel Street	2B	New York	NY
10	181089005		Madelyn	Jacob	Bryson	3142 Coppel Street	2B	New York	NY
11	1811 (null)		Duane	Biggs	Lewis	3615 Sugar Camp Road	(null)	Owatonna	MN
12	1912 (null)		Curt	Jake	Brown	49 Heliport Loop	(null)	Bloomington	IN
13	191389006		Julie	Harris	Wright	4374 Bagwell Avenue	(null)	Morrisville	PA
14	191489006		Sam	Hansen	Wright	4374 Bagwell Avenue	(null)	Morrisville	PA

Auto Insurance Company CRM Database Design for Policy Management

ZIP	PHONE	MOBILE	EMAIL	SSN	GENDER
60435	481-629-2343	466-647-4324	tcruise@hotmail.com	499-05-0979	Male
32810	407-287-7033	407-376-1649	agreen@yahoo.com	771-18-5685	Female
22611	540-955-3503	757-662-5541	amanda@gmail.com	225-86-9885	Female
24251	276-452-0623	276-594-9430	emills@hotmail.com	230-92-3262	Male
21113	410-305-6002	443-277-5871	grace20@yahoo.com	216-37-0024	Female
14608	585-309-7538	516-263-7391	sanders@outlook.com	061-30-7132	Female
35816	256-508-9037	256-337-8382	grossg@alabama.com	423-53-8790	Male
13202	315-333-9223	315-440-9802	yoe4mpunki8@hotmail.com	093-62-0995	Female
11361	718-819-7591	347-319-9576	bry03@gmail.com	111-09-1984	Male
11361	718-819-7591	347-310-6336	madell19@gmail.com	552-15-4537	Female
55060	507-623-8123	507-475-5746	dlewis@yahoo.com	471-10-8455	Male
47404	812-652-2696	812-688-0402	curt.b@hotmail.com	333-64-7419	Male
19067	352-572-8362	215-932-9418	brown4418@yahoo.com	261-19-5471	Female
19067	352-572-8362	619-992-9425	sam11@gmail.com	623-04-6990	Male

Entity 2: License

Note: For each of 14 customers, their License information is stored in this License table.

Name	Null?	Type
-----	-----	-----
LICENSE_NO	NOT NULL	NUMBER(38)
CUST_ID		NUMBER(38)
FIRST_ISSUE_DATE		DATE
LATEST_ISSUE_DATE		DATE
EXPIRY_DATE		DATE
DOB		DATE
VIOLATION_HIST		NUMBER(38)
ISSUE_STATE		VARCHAR2(10)
ISSUE_COUNTRY		VARCHAR2(14)

	LICENSE_NO	CUST_ID	FIRST_ISSUE_DATE	LATEST_ISSUE_DATE	EXPIRY_DATE	DOB	VIOLATION_HIST	ISSUE_STATE	ISSUE_COUNTRY
1	872904534	1501	23-SEP-00	31-JUL-19	31-JUL-24	11-JUN-82	0	IL	USA
2	430982367	1502	08-SEP-89	01-MAR-18	25-FEB-23	01-JAN-71	3	FL	USA
3	615471827	1503	23-FEB-91	06-MAY-20	30-OCT-22	16-DEC-72	0	VA	USA
4	639164064	1604	08-JAN-11	25-OCT-19	23-OCT-24	25-SEP-92	2	VA	USA
5	120934874	1705	24-JUL-83	27-MAR-18	31-JAN-24	27-JAN-65	0	France	International
6	763290487	1706	11-DEC-98	11-SEP-18	10-AUG-21	12-JUL-80	1	NY	USA
7	912376536	1707	08-APR-88	24-NOV-19	20-SEP-23	25-DEC-69	0	Costa Rica	International
8	518221360	1708	13-JAN-98	08-AUG-18	12-JAN-22	01-OCT-78	1	NY	USA
9	619852978	1809	20-NOV-07	21-NOV-18	30-NOV-22	19-APR-88	0	NY	USA
10	852978431	1810	17-MAR-12	28-NOV-19	16-FEB-23	04-DEC-91	0	NY	USA
11	875465824	1811	09-JAN-98	28-JAN-19	25-JUL-23	27-SEP-78	3	MN	USA
12	621567963	1912	25-OCT-77	25-MAR-18	17-SEP-02	13-JUL-59	2	IN	USA
13	447885933	1913	08-DEC-01	15-JAN-18	21-SEP-23	19-JAN-83	0	PA	USA
14	135697465	1914	04-DEC-86	23-MAR-18	31-MAY-22	18-MAY-68	0	PA	USA

Entity 3: Vehicle

Note: There are 12 vehicles for which quotation was requested. Two of the vehicles are driven by 2 customers). Thereby for 14 customers, there is 12 vehicles only.

VIN_NO	NOT NULL	VARCHAR2 (200)
POLICY_NO		VARCHAR2 (200)
CAR_MAKE		VARCHAR2 (200)
CAR_MODEL		VARCHAR2 (200)
CAR_YEAR		NUMBER (38)
VEHICLE_TYPE		VARCHAR2 (200)
AIRBAGS		NUMBER (38)
COLOR		VARCHAR2 (200)
MILES		NUMBER (38)

⚡ VIN_NO	⚡ POLICY_NO	⚡ CAR_MAKE	⚡ CAR_MODEL	⚡ CAR_YEAR	⚡ VEHICLE_TYPE	⚡ AIRBAGS	⚡ COLOR	⚡ MILES
1 SAJWA1C78D8V38055	89001	Jaguar	XJ	2013	Sedan	6	Black	23014
2 JH4DB8580SS001230	89002	Acura	Integra	1995	Sedan	2	Red	125671
3 JTHFF2C26B2515141	(null)	Lexus	IS	2011	Sedan	6	White	56743
4 WMWRC33474TC49530	89003	MINI	Cooper	2004	Sedan	6	Green	76423
5 WAUDK84AXRN029130	(null)	Audi	Q5	2016	SUV	6	Grey	23505
6 5TFUW5F13CX228552	(null)	Toyota	Tundra	2010	Pickup	2	Brown	96435
7 4A3AA46G13E081883	89004	Honda	Accord	2012	Sedan	2	Black	122345
8 1GKER23767J144063	89005	Subaru	Legacy	2015	Sedan	2	Blue	88907
9 1FTRW07L01KB09635	(null)	Ford	F-450	2013	Pickup	3	Grey	105662
10 1J8GR48K78C206923	(null)	BMW	550i	2018	Coupe	4	White	26735
11 WBXPA93426WG80137	89006	Toyota	Camry	2014	Sedan	2	Tan	77875
12 1FTRW08L82KB90741	(null)	Jeep	Sahara	2016	SUV	4	Black	66526

Entity 4: Vehicle driver

Note: This is associate entity between the Vehicle and License. Thereby there is 14 rows.

Name	Null?	Type
-----	-----	-----
LICENSE_NO	NOT NULL	NUMBER (38)
VIN_NO		VARCHAR2 (200)
REGISTRATION_NO		NUMBER (38)
DMV_PLATE_NO		VARCHAR2 (200)
ISSUE_DATE		DATE
EXPIRY_DATE		DATE
PURCHASE_DATE		DATE

Auto Insurance Company CRM Database Design for Policy Management

	LICENSE_NO	VIN_NO	REGISTRATION_NO	DMV_PLATE_NO	ISSUE_DATE	EXPIRY_DATE	PURCHASE_DATE
1	872904534	SAJWA1C78D8V38055	534210	LRX2716	23-DEC-19	21-DEC-21	23-DEC-13
2	430982367	JH4DB8580SS001230	842084	FEE3094	21-NOV-19	15-NOV-20	21-NOV-95
3	615471827	JTHFF2C26B2515141	845123	6UXM244	02-OCT-19	01-OCT-21	02-OCT-19
4	639164064	WMWRC33474TC49530	132786	AT00339	04-JUN-19	20-MAY-21	04-JUN-11
5	120934874	WAUDK84AXRN029130	3092341	UTE4561	23-MAR-20	15-MAR-21	23-MAR-17
6	763290487	5TFUW5F13CX228552	945213	CA51023	02-NOV-20	25-OCT-21	02-NOV-11
7	912376536	4A3AA46G13E081883	654684	KAS8733	04-AUG-19	06-FEB-21	04-AUG-18
8	518221360	1FTRW07L01KB09635	787876	IUWY66	10-DEC-19	04-DEC-23	10-DEC-15
9	619852978	1GKER23767J144063	5436541	YWH723	05-NOV-19	12-DEC-22	05-NOV-18
10	852978431	1GKER23767J144063	5436541	YWH723	05-NOV-19	12-DEC-22	05-NOV-18
11	875465824	1J8GR48K78C206923	164654	ALSJ79	16-JUN-20	15-NOV-23	16-JUN-10
12	621567963	WBXPA93426WG80137	6878563	YGV6235	22-OCT-19	04-SEP-23	22-OCT-14
13	447885933	1FTRW08L82KB90741	855159	7YUS982	20-APR-19	15-APR-21	20-APR-14
14	135697465	1FTRW08L82KB90741	855159	7YUS982	20-APR-19	15-APR-21	20-APR-14

Entity 5: Vehicle Lien

Note: Out of 12 vehicles to which quotation was given, 6 of the vehicles are having lien. The bank which offered loan is stored in this table.

Name	Null?	Type
-----	-----	-----
LIEN_ID	NOT NULL	VARCHAR2 (200)
VIN_NO		VARCHAR2 (200)
LIEN_HOLDER		VARCHAR2 (200)
RECORD_DATE		DATE

	LIEN_ID	VIN_NO	LIEN_HOLDER	RECORD_DATE
1	LCT001	SAJWA1C78D8V38055	Bank Of America	23-OCT-15
2	LCT002	JH4DB8580SS001230	Peoples Bank	01-NOV-15
3	LCT003	WMWRC33474TC49530	Bank Of America	30-JAN-16
4	LCT004	1FTRW07L01KB09635	Wells Fargo	10-NOV-18
5	LCT005	1J8GR48K78C206923	Bank Of America	28-DEC-18
6	LCT006	WBXPA93426WG80137	TD Bank	04-OCT-19

Entity 6: Quote

Note: The quotations given to the 12 vehicles is stored in this table.

Name	Null?	Type
QUOTE_ID	NOT NULL	NUMBER(38)
AGENT_ID		VARCHAR2(200)
CUST_ID		NUMBER(38)
PROD_TYPE_ID		VARCHAR2(200)
AMOUNT		NUMBER(6,2)
ISSUE_DATE		DATE
EXPIRY_DATE		DATE
DISCOUNT		NUMBER(5,2)
Name	Null?	Type

	QUOTE_ID	AGENT_ID	CUST_ID	PROD_TYPE_ID	AMOUNT	ISSUE_DATE	EXPIRY_DATE	DISCOUNT
1	1250	4-59944	1501	P101B	1200	23-OCT-15	07-NOV-15	10
2	8037	9-14206	1502	P101B	1224	01-NOV-15	16-NOV-15	5.5
3	3209	(null)	1503	P102S	456	03-DEC-15	18-DEC-15	3
4	2734	9-14206	1604	P102S	960	30-JAN-16	14-FEB-16	0
5	5489	4-59944	1705	P103G	1345	03-FEB-17	18-FEB-17	5
6	4965	4-59944	1706	P102S	567	06-JUN-17	21-JUN-17	10
7	1252	(null)	1707	P101B	1272	18-OCT-17	02-NOV-17	3
8	8039	4-59944	1708	P103G	732	24-SEP-17	09-OCT-17	0
9	3229	9-14206	1809	P101B	956	10-NOV-18	25-NOV-18	5
10	2727	4-59944	1811	P101B	657	28-DEC-18	12-JAN-19	10
11	5492	9-14206	1912	P102S	540	04-OCT-19	19-OCT-19	3
12	4972	4-59944	1913	P102S	1224	27-NOV-19	12-DEC-19	0

Entity 7 – Agent

Note: The information of the insurance Agent who generated the quote is stored in this table.

Name	Null?	Type
AGENT_ID	NOT NULL	VARCHAR2(200)
AGENT_NM		VARCHAR2(200)
LOCATION		VARCHAR2(200)
RATINGS		NUMBER(38)
SALARY		VARCHAR2(200)

	AGENT_ID	AGENT_NM	LOCATION	RATINGS	SALARY
1	9-14206	George Robinson	MA	5	\$85,000
2	4-59944	Linda Kendrick	WI	4	\$90,000

Auto Insurance Company CRM Database Design for Policy Management

Entity 8 - Insu_Policy

Note: Out of the 12 quotes given for 12 vehicles, only 6 of the customers were interested in buying the policy. 6 more customers did not buy any policy. Thereby this table have the information of the 6 policies issued by insurance company.

Name	Null?	Type
POLICY_NO	NOT NULL	VARCHAR2(200)
VIN_NO		VARCHAR2(200)
EFFECTIVE_DATE		DATE
EXP_DATE		DATE
PREMIUM_AMOUNT		NUMBER(8,2)
AGREEMENT		VARCHAR2(200)
SAFE_DRIVER		VARCHAR2(200)
AMEND_ID		VARCHAR2(200)
AMEND_DATE		DATE
AMEND_TYPE		VARCHAR2(200)

	POLICY_NO	VIN_NO	EFFECTIVE_DATE	EXP_DATE	PREMIUM_AMOUNT	AGREEMENT	SAFE_DRIVER	AMEND_ID	AMEND_DATE	AMEND_TYPE
1	89001	SAJWA1C78D8V38055	23-OCT-19	22-OCT-20	925.97	1 year	Y	89001-05	17-OCT-19	Renewal
2	89002	JH4DB8580S001230	01-NOV-19	31-OCT-20	944.48	1 year	N	89002-05	26-OCT-19	Renewal
3	89003	WMWRC33474TC49530	28-JAN-20	27-JAN-21	938.31	1 year	N	89003-05	18-DEC-19	Renewal
4	89004	4A3AA46G13E081883	17-OCT-20	17-OCT-21	1033.18	1 year	N	89004-04	17-OCT-20	Renewal
5	89005	1GKER23767J144063	26-SEP-20	28-MAR-21	389.01	6 months	Y	89005-07	26-SEP-20	Renewal
6	89006	WBXPA93426WG80137	16-OCT-20	17-APR-21	1046.52	6 months	Y	89006-03	16-OCT-20	Renewal

Entity 9 - Policy_Coverage

Note: The details of the policy coverage are stored in this table.

Name	Null?	Type
POLICY_NO		VARCHAR2(200)
PROD_TYPE_ID		VARCHAR2(200)
POLICY_STATUS		VARCHAR2(200)
POLICY_START_DATE		DATE
POLICY_END_DATE		DATE

	POLICY_NO	PROD_TYPE_ID	POLICY_STATUS	POLICY_START_DATE	POLICY_END_DATE
1	89001	P101B	Active	23-OCT-19	22-OCT-20
2	89002	P101B	Expired	01-NOV-19	31-OCT-20
3	89003	P101B	Renewed	02-FEB-16	11-MAY-21
4	89003	P102S	Active	12-MAY-16	27-JAN-21
5	89004	P101B	Active	17-OCT-20	17-OCT-21
6	89005	P103G	Active	26-SEP-20	28-MAR-21
7	89006	P102S	Active	16-OCT-20	17-APR-21

Auto Insurance Company CRM Database Design for Policy Management

Entity 10 - Product Type

Note: The insurance company have 3 product offerings (Gold, Silver, Basic). The coverage / liability is different for them. The details of the coverage is stored in this table.

Name	Null?	Type
PROD_TYPE_ID	NOT NULL	VARCHAR2(200)
PROD_TYPE_NAME		VARCHAR2(100)
BODILY_INJURY		VARCHAR2(200)
PROP_DAMAGE		NUMBER(38)
COMPREHENSIVE		VARCHAR2(200)
UNINSURED		VARCHAR2(200)
COLLISION		VARCHAR2(200)
EMERGENCY		VARCHAR2(100)

PROD_TYPE_ID	PROD_TYPE_NAME	BODILY_INJURY	PROP_DAMAGE	COMPREHENSIVE	UNINSURED	COLLISION	EMERGENCY
1 P101B	Basic	25000/50000	25000	Not Covered	25000/50000	Not Covered	Not Covered
2 P102S	Silver	25000/50000	25000	Yes with Deductible 500	25000/50000	Yes upto Actual cash value with Deductible 500	Yes Covered
3 P103G	Gold	50000/100000	25000	Yes with Deductible 100	50000/100000	Yes upto Actual cash value with Deductible 100	Yes Covered

Entity 11- Policy Amend

Note: The various amendments made (renewal or the change of product coverage) to the policies are stored in this table.

Name	Null?	Type
AMEND_ID	NOT NULL	VARCHAR2(200)
POLICY_NO		VARCHAR2(200)
AMEND_DATE		DATE
AMEND_TYPE		VARCHAR2(200)

AMEND_ID	POLICY_NO	AMEND_DATE	AMEND_TYPE
1 89001-01 89001		(null)	(null)
2 89001-02 89001		27-OCT-16	Renewal
3 89001-03 89001		17-OCT-17	Renewal
4 89001-04 89001		17-OCT-18	Renewal
5 89001-05 89001		17-OCT-19	Renewal
6 89002-01 89002		(null)	(null)
7 89002-02 89002		05-NOV-16	Renewal
8 89002-03 89002		26-OCT-17	Renewal
9 89002-04 89002		26-OCT-18	Renewal
10 89002-05 89002		26-OCT-19	Renewal
11 89003-01 89003		(null)	(null)
12 89003-02 89003		12-MAY-16	Product change - Basic to Silver
13 89003-03 89003		17-JAN-17	Renewal
14 89003-04 89003		07-JAN-18	Renewal
15 89003-05 89003		28-DEC-18	Renewal
16 89003-06 89003		18-DEC-19	Renewal
17 89004-01 89004		(null)	(null)
18 89004-02 89004		20-OCT-18	Renewal
19 89004-03 89004		19-OCT-19	Renewal
20 89004-04 89004		17-OCT-20	Renewal
21 89005-01 89005		(null)	(null)
22 89005-02 89005		31-MAR-18	Renewal
23 89005-03 89005		29-SEP-18	Renewal
24 89005-04 89005		30-MAR-19	Renewal
25 89005-05 89005		28-SEP-19	Renewal
26 89005-06 89005		28-MAR-20	Renewal
27 89005-07 89005		26-SEP-20	Renewal
28 89006-01 89006		(null)	(null)
29 89006-02 89006		17-APR-20	Renewal
30 89006-03 89006		16-OCT-20	Renewal

Entity 12: Claim

Note: The various claims done to the policies are stored in this table

Name	Null?	Type
CLAIM_NO	NOT NULL	VARCHAR2 (200)
POLICY_NO		VARCHAR2 (200)
CLAIM_DATE		DATE
AMOUNT		NUMBER (10, 2)
STATUS		VARCHAR2 (200)

	CLAIM_NO	POLICY_NO	CLAIM_DATE	AMOUNT	STATUS
1	CL6101	89002	20-APR-16	8943	Settled
2	CL6102	89003	17-JUL-17	4152	Rejected
3	CL6103	89004	06-OCT-18	7532	Settled
4	CL6104	89004	06-OCT-19	9053	Rejected

Entity 13: Settlement

Note: If the claims are Approved, then it goes to settlement. The status of settlement is stored in this table.

Name	Null?	Type
SETTLEMENT_ID	NOT NULL	VARCHAR2 (200)
CLAIM_NO		VARCHAR2 (200)
SETTLE_DATE		DATE
AMOUNT		NUMBER (10, 2)
STATUS		VARCHAR2 (200)

	SETTLEMENT_ID	CLAIM_NO	SETTLE_DATE	AMOUNT	STATUS
1	SE7101	CL6101	25-APR-16	8943	Invoice Paid
2	SE7102	CL6103	11-OCT-18	4152	Invoice Paid

Entity 14: Incident Record

Note: The details of any incident (like accident, parking violation, etc.) are stored in this table.

Name	Null?	Type
INCIDENT_NO	NOT NULL	NUMBER (38)
VIN_NO		VARCHAR2 (200)
LICENSE_NO		NUMBER (38)
INCIDENT_LOCATION		VARCHAR2 (200)
INCIDENT_DATE		DATE
INCIDENT_TIME		VARCHAR2 (200)
INCIDENT_TYPE		VARCHAR2 (200)

Auto Insurance Company CRM Database Design for Policy Management

	INCIDENT_NO	VIN_NO	LICENSE_NO	INCIDENT_LOCATION	INCIDENT_DATE	INCIDENT_TIME	INCIDENT_TYPE
1	982364	JH4DB8580SS001230	872904534	Stamford, CT	10-APR-16	10:11 AM	Speeding greater than 20miles limit
2	783021	WMWRC33474TC49530	430982367	Hoboken, NJ	07-JUL-17	8:30 AM	Parking in No Parking Zone
3	840268	4A3AA46G13E081883	639164064	AllenTown, PA	26-SEP-18	10:00 PM	Accident - Hit front vehicle
4	492817	4A3AA46G13E081883	639164064	Albany, NY	26-SEP-19	6:21 PM	PropertyDamage

Entity 15: Bill

Note: The details of the bill send from the company to the policy holder is stored in this table.

Name	Null?	Type
BILL_NO	NOT NULL	NUMBER (38)
POLICY_NO		VARCHAR2 (200)
BILL_AMOUNT		NUMBER (10, 2)
BILL_CYCLE		VARCHAR2 (100)
DEDUCTIBLE		NUMBER (10, 2)
BILL_DATE		DATE
DUE_DATE		DATE
STATUS		VARCHAR2 (200)

	BILL_NO	POLICY_NO	BILL_AMOUNT	BILL_CYCLE	DEDUCTIBLE	BILL_DATE	DUE_DATE	STATUS
1	1056789001		600	Half Yearly		028-OCT-15	27-NOV-15	Fully Paid
2	1056889001		600	Half Yearly		025-APR-16	25-MAY-16	Fully Paid
3	1056989001		540	Half Yearly		027-OCT-16	26-NOV-16	Fully Paid
4	1057089001		540	Half Yearly		025-APR-17	25-MAY-17	Fully Paid
5	1057189001		513	Half Yearly		027-OCT-17	26-NOV-17	Fully Paid
6	1057289001		513	Half Yearly		025-APR-18	25-MAY-18	Fully Paid
7	1057389001		487.35	Half Yearly		027-OCT-18	26-NOV-18	Fully Paid
8	1057489001		487.35	Half Yearly		025-APR-19	25-MAY-19	Fully Paid
9	1057589001		462.98	Half Yearly		027-OCT-19	26-NOV-19	Fully Paid
10	1057689001		462.98	Half Yearly		024-APR-20	24-MAY-20	Fully Paid
11	1057789001		462.98	Half Yearly		026-OCT-20	25-NOV-20	Fully Paid
12	2064589002		612	Half Yearly		006-NOV-15	06-DEC-15	Fully Paid
13	2064689002		612	Half Yearly	500	004-MAY-16	03-JUN-16	Fully Paid
14	2064789002		550.8	Half Yearly		005-NOV-16	05-DEC-16	Fully Paid
15	2064889002		550.8	Half Yearly		004-MAY-17	03-JUN-17	Fully Paid
16	2064989002		523.26	Half Yearly		005-NOV-17	05-DEC-17	Fully Paid
17	2065089002		523.26	Half Yearly		004-MAY-18	03-JUN-18	Fully Paid
18	2065189002		497.1	Half Yearly		005-NOV-18	05-DEC-18	Fully Paid
19	2065289002		497.1	Half Yearly		004-MAY-19	03-JUN-19	Fully Paid
20	2065389002		472.24	Half Yearly		005-NOV-19	05-DEC-19	Fully Paid
21	2065489002		472.24	Half Yearly		003-MAY-20	02-JUN-20	Fully Paid
22	3067389003		480	Half Yearly		002-FEB-16	03-MAR-16	Fully Paid
23	3067489003		672	Half Yearly		031-JUL-16	30-AUG-16	Fully Paid
24	3067589003		547.2	Half Yearly		001-FEB-17	03-MAR-17	Fully Paid
25	3067689003		547.2	Half Yearly	100	31-JUL-17	30-AUG-17	Fully Paid
26	3067789003		519.84	Half Yearly		001-FEB-18	03-MAR-18	Fully Paid
27	3067889003		519.84	Half Yearly		031-JUL-18	30-AUG-18	Fully Paid
28	3067989003		493.85	Half Yearly		001-FEB-19	03-MAR-19	Fully Paid

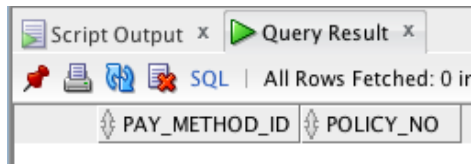
Auto Insurance Company CRM Database Design for Policy Management

29	3068089003	493.85	Half Yearly	031-JUL-19	30-AUG-19	Fully Paid
30	3068189003	469.16	Half Yearly	027-JAN-20	26-FEB-20	Fully Paid
31	3068289003	469.16	Half Yearly	025-JUL-20	24-AUG-20	Fully Paid
32	4014589004	1272	Yearly	021-OCT-17	20-NOV-17	Fully Paid
33	4015689004	1144.8	Yearly	50021-OCT-18	20-NOV-18	Fully Paid
34	4016789004	1087.56	Yearly	50021-OCT-19	20-NOV-19	Fully Paid
35	4017889004	1033.18	Yearly	020-OCT-20	19-NOV-20	Fully Paid
36	5023189005	732	Half Yearly	030-SEP-17	30-OCT-17	Fully Paid
37	5023289005	658.8	Half Yearly	001-APR-18	01-MAY-18	Fully Paid
38	5023389005	592.92	Half Yearly	001-OCT-18	31-OCT-18	Fully Paid
39	5023489005	533.63	Half Yearly	002-APR-19	02-MAY-19	Fully Paid
40	5023589005	480.27	Half Yearly	002-OCT-19	01-NOV-19	Fully Paid
41	5023689005	432.24	Half Yearly	002-APR-20	02-MAY-20	Fully Paid
42	5023789005	389.01	Half Yearly	002-OCT-20	01-NOV-20	Pending
43	6013289006	204	Monthly	018-OCT-19	17-NOV-19	Fully Paid
44	6013389006	204	Monthly	017-NOV-19	17-DEC-19	Fully Paid
45	6013489006	204	Monthly	017-DEC-19	16-JAN-20	Fully Paid
46	6013589006	204	Monthly	016-JAN-20	15-FEB-20	Fully Paid
47	6013689006	204	Monthly	015-FEB-20	16-MAR-20	Fully Paid
48	6013789006	204	Monthly	016-MAR-20	15-APR-20	Fully Paid
49	6013889006	1101.6	Monthly	020-APR-20	20-MAY-20	Fully Paid
50	6013989006	1101.6	Monthly	020-MAY-20	19-JUN-20	Fully Paid
51	6014089006	1101.6	Monthly	019-JUN-20	19-JUL-20	Fully Paid
52	6014189006	1101.6	Monthly	019-JUL-20	18-AUG-20	Fully Paid
53	6014289006	1101.6	Monthly	018-AUG-20	17-SEP-20	Fully Paid
54	6014389006	1101.6	Monthly	017-SEP-20	17-OCT-20	Fully Paid
55	6014489006	1046.52	Monthly	020-OCT-20	19-NOV-20	Fully Paid
56	6014589006	1046.52	Monthly	019-NOV-20	19-OCT-20	Pending

Entity 16: Payment_Method

Note: The payment method used by policy holder is either credit card or the bank account. This entity “Payment_Method” is a super type and thereby no information is stored in the table. The details of the credit card and bank account is stored in the “sub type” of this entity.

Name	Null?	Type
PAY_METHOD_ID		VARCHAR2(200)
POLICY_NO		VARCHAR2(200)




Entity 17: Credit Card

Note: This entity is “sub type” of the “Payment Method” super type. The details of the credit card are stored in the “sub type” entity.

Name	Null?	Type
PAY_METHOD_ID		VARCHAR2(200)
POLICY_NO		VARCHAR2(200)
CARD_NO		NUMBER(38)
HOLDER_NAME		VARCHAR2(200)
STREET_NM		VARCHAR2(200)
APT_NO		VARCHAR2(20)
CITY		VARCHAR2(200)
STATE_NM		VARCHAR2(20)
ZIP		VARCHAR2(20)
CARD_TYPE		VARCHAR2(20)
CVC		NUMBER
ISSUE_BANK		VARCHAR2(20)
EXP_DETAIL		VARCHAR2(20)

Script Output x Query Result x

 SQL | All Rows Fetched: 3 in 0.033 seconds


	PAY_METHOD_ID	POLICY_NO	CARD_NO	HOLDER_NAME	STREET_NM	APT_NO	CITY	STATE_NM	ZIP	CARD_TYPE	CVC	ISSUE_BANK	EXP_DETAIL
1	PM1001	89001	4916689811047971	Tom Cruise	47 W. Pulaski Rd	(null)	Joliet	IL	60435	VISA		156 Bank Of America	10/25
2	PM1002	89002	4916689811047971	Angelia Green	2111 Barnes Street	(null)	Orlando	FL	32810	Amex		972 Peoples United	11/23
3	PM1004	89004	6515846293546	Heather Gale	3803 Oak Street	(null)	Syracuse	NY	13202	MasterCard		966 Citi Bank	04/24

Entity 18: Bank Account

Note: This entity is “sub type” of the “Payment Method” super type. The details of the bank account are stored in the “sub type” entity.

Name	Null?	Type
PAY_METHOD_ID		VARCHAR2(200)
POLICY_NO		VARCHAR2(200)
ACCOUNT_NO		VARCHAR2(200)
HOLDER_NM		VARCHAR2(20)
BANK_NAME		VARCHAR2(20)
ROUTING_NO		VARCHAR2(200)

Script Output x Query Result x

 All Rows Fetched: 4 in 0.03 seconds

	PAY_METHOD_ID	POLICY_NO	ACCOUNT_NO	HOLDER_NM	BANK_NAME	ROUTING_NO
1	PM1003	89003	456781914	Edward Mills	HSBC Bank	9845672
2	PM1004	89004	852325255	Heather Gale	Citi Bank	21256482
3	PM1005	89005	645511651	Robert Smith	HSBC Bank	55296632
4	PM1006	89006	35465436	Sam Hansen	Chase Bank	2455453

Entity 19: Payment Receipt

Note: When the payment is made to the bills sent, the receipts are generated. The details of the payment receipt generated is stored in this table.

Name	Null?	Type
PAY_RCPT_ID	NOT NULL	NUMBER(38)
PAY_METHOD_ID		VARCHAR2(200)
BILL_NO		NUMBER(38)
PAID_AMOUNT		NUMBER(10,2)
PAY_DATE		DATE
PAY_METHOD		VARCHAR2(200)

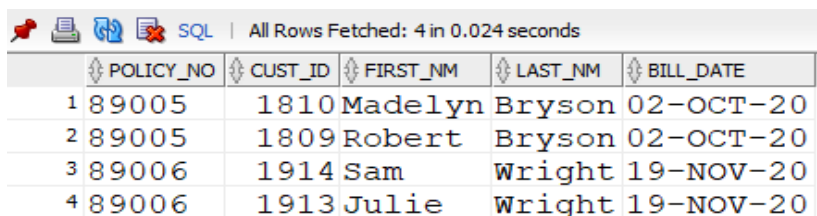
	PAY_RCPT_ID	PAY_METHOD_ID	BILL_NO	PAID_AMOUNT	PAY_DATE	PAY_METHOD
1	563354	PM1001	10568	600.28	28-APR-16	Credit Card
2	746333	PM1001	10569	540.30	30-OCT-16	Credit Card
3	462623	PM1001	10570	540.28	28-APR-17	Credit Card
4	452984	PM1001	10571	513.30	30-OCT-17	Credit Card
5	526294	PM1001	10572	513.28	28-APR-18	Credit Card
6	563355	PM1001	10573	487.35	30-OCT-18	Credit Card
7	746337	PM1001	10574	487.35	28-APR-19	Credit Card
8	462625	PM1001	10575	462.98	30-OCT-19	Credit Card
9	452987	PM1001	10576	462.98	27-APR-20	Credit Card
10	526298	PM1001	10577	462.98	29-OCT-20	Credit Card
11	563356	PM1002	20645	612.09	09-NOV-15	Credit Card
12	746341	PM1002	20646	1112.07	07-MAY-16	Credit Card
13	462627	PM1002	20647	550.80	08-NOV-16	Credit Card
14	452990	PM1002	20648	550.80	07-MAY-17	Credit Card
15	526302	PM1002	20649	523.26	08-NOV-17	Credit Card
16	563357	PM1002	20650	523.26	07-MAY-18	Credit Card
17	746345	PM1002	20651	497.10	08-NOV-18	Credit Card
18	462629	PM1002	20652	497.10	07-MAY-19	Credit Card
19	452993	PM1002	20653	472.24	08-NOV-19	Credit Card
20	526306	PM1002	20654	472.24	06-MAY-20	Credit Card
21	563358	PM1003	30673	480.05	05-FEB-16	Checking Account
22	746349	PM1003	30674	672.03	03-AUG-16	Checking Account
23	462631	PM1003	30675	547.20	04-FEB-17	Checking Account
24	452996	PM1003	30676	647.20	03-AUG-17	Checking Account
25	526310	PM1003	30677	519.84	04-FEB-18	Checking Account
26	563359	PM1003	30678	519.84	03-AUG-18	Checking Account
27	746353	PM1003	30679	493.85	04-FEB-19	Checking Account
28	462633	PM1003	30680	493.85	03-AUG-19	Checking Account
29	452999	PM1003	30681	469.16	30-JAN-20	Checking Account
30	526314	PM1003	30682	469.16	28-JUL-20	Checking Account
31	563360	PM1004	40145	1272.24	24-OCT-17	Credit Card
32	746357	PM1004	40156	1644.82	24-OCT-18	Credit Card
33	462635	PM1004	40167	1587.56	24-OCT-19	Credit Card
34	453002	PM1004	40178	1033.18	23-OCT-20	Credit Card
35	526318	PM1005	50231	732.03	03-OCT-17	Credit Card
36	563361	PM1005	50232	658.80	04-APR-18	Credit Card
37	746361	PM1005	50233	592.92	04-OCT-18	Credit Card
38	462637	PM1005	50234	533.63	05-APR-19	Credit Card
39	453005	PM1005	50235	480.27	05-OCT-19	Credit Card
40	526322	PM1005	50236	432.24	05-APR-20	Credit Card
41	746365	PM1006	60132	204.21	21-OCT-19	Checking Account
42	462639	PM1006	60133	204.20	20-NOV-19	Checking Account
43	453008	PM1006	60134	204.20	20-DEC-19	Checking Account
44	526326	PM1006	60135	204.19	19-JAN-20	Checking Account
45	563363	PM1006	60136	204.18	18-FEB-20	Checking Account
46	746369	PM1006	60137	204.19	19-MAR-20	Checking Account
47	462641	PM1006	60138	1101.62	23-APR-20	Checking Account
48	453011	PM1006	60139	1101.62	23-MAY-20	Checking Account
49	526330	PM1006	60140	1101.62	22-JUN-20	Checking Account
50	563364	PM1006	60141	1101.62	22-JUL-20	Checking Account
51	746373	PM1006	60142	1101.62	21-AUG-20	Checking Account
52	462643	PM1006	60143	1101.62	20-SEP-20	Checking Account
53	453014	PM1006	60144	1046.52	23-OCT-20	Checking Account

8. Output of DML SQL Statements

This section shows the Advanced SQL commands and outputs. There are 16 statements, out of which 10 are using the JOIN command.

Q1. What are the Policy Number, Customer ID, First and Last name, Bill Date for the current unpaid “Pending” bills ?

```
SELECT C.POLICY_NO, CUST_ID, FIRST_NM, LAST_NM, B.BILL_DATE
FROM CUSTOMER C INNER JOIN BILL B
ON C.POLICY_NO = B.POLICY_NO
WHERE B.STATUS = 'Pending';
```

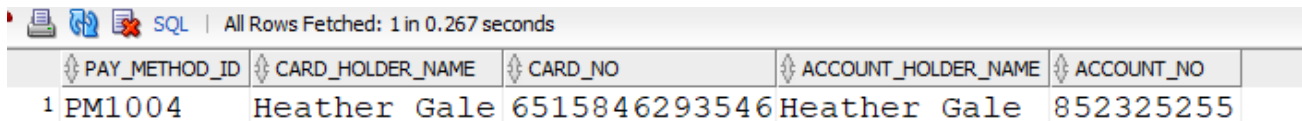


SQL | All Rows Fetched: 4 in 0.024 seconds

	POLICY_NO	CUST_ID	FIRST_NM	LAST_NM	BILL_DATE
1	89005	1810	Madelyn	Bryson	02-OCT-20
2	89005	1809	Robert	Bryson	02-OCT-20
3	89006	1914	Sam	Wright	19-NOV-20
4	89006	1913	Julie	Wright	19-NOV-20

Q2. Identify Customers who have both Credit Card and Bank Account as payment methods. Then list the Card holder name, Card number, Bank Account holder name and bank Account number.

```
SELECT C.PAY_METHOD_ID, C.HOLDER_NAME AS CARD_HOLDER_NAME, C.CARD_NO, B.HOLDER_NM AS
ACCOUNT_HOLDER_NAME, B.ACCOUNT_NO
FROM CREDIT_CARD C INNER JOIN BANK_ACCOUNT B
ON C.PAY_METHOD_ID = B.PAY_METHOD_ID;
```

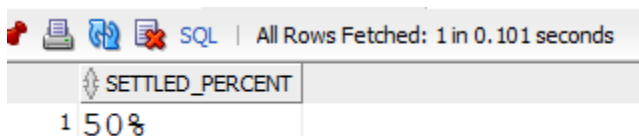


SQL | All Rows Fetched: 1 in 0.267 seconds

	PAY_METHOD_ID	CARD_HOLDER_NAME	CARD_NO	ACCOUNT_HOLDER_NAME	ACCOUNT_NO
1	PM1004	Heather Gale	6515846293546	Heather Gale	852325255

Q3. Out of all claims filed, what is the percentage that was settled?

```
SELECT COUNT(CASE WHEN STATUS = 'Settled' THEN 1 END)/COUNT(CLAIM_NO) *100 || '%' AS
SETTLED_PERCENT
FROM CLAIM;
```

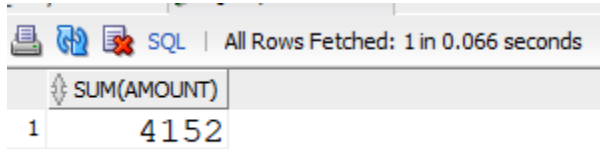


SQL | All Rows Fetched: 1 in 0.101 seconds

	SETTLED_PERCENT
1	50%

Q4. Calculate and display the Total Net settlement amount the company paid in the last 3 years?

```
SELECT SUM(AMOUNT)
FROM SETTLEMENT
WHERE (SYSDATE-SETTLE_DATE)/365 < 3.0;
```

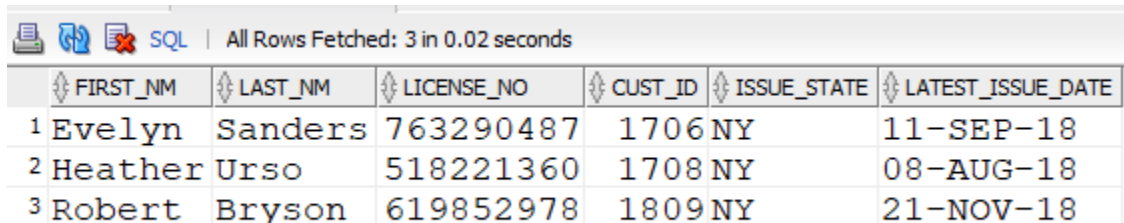


SQL | All Rows Fetched: 1 in 0.066 seconds

	SUM(AMOUNT)
1	4152

Q5. Identify all customers of NY state registered in 2018 with their first & last name, cust_id & license no.

```
SELECT customer.first_nm,
customer.last_nm,license.license_no,license.cust_id,issue_state,latest_issue_date
FROM Customer INNER JOIN License
ON license.cust_id = customer.cust_id
WHERE issue_state = 'NY'
AND EXTRACT(YEAR FROM latest_issue_date)='2018';
```

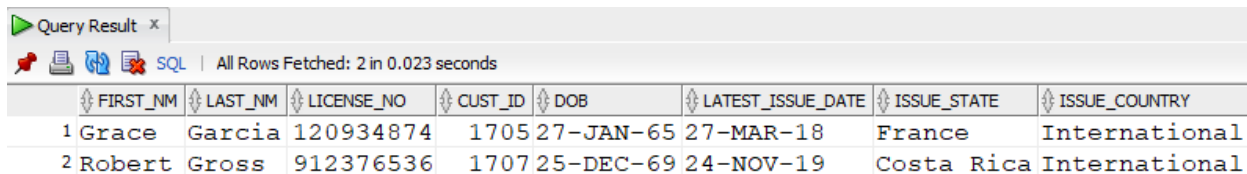


SQL | All Rows Fetched: 3 in 0.02 seconds

	FIRST_NM	LAST_NM	LICENSE_NO	CUST_ID	ISSUE_STATE	LATEST_ISSUE_DATE
1	Evelyn	Sanders	763290487	1706	NY	11-SEP-18
2	Heather	Urso	518221360	1708	NY	08-AUG-18
3	Robert	Bryson	619852978	1809	NY	21-NOV-18

Q6. Retrieve the customer first and last name with license no, cust_ID, DOB, latest issue date & issuing country with international licenses (using EQUI JOIN).

```
SELECT customer.First_nm, customer.Last_nm,
License.license_no,License.cust_id,License.DOB,License.latest_issue_date,License.Issue_state,License.Issue_
country
FROM customer,License
WHERE license.cust_id = customer.cust_id
AND issue_country <> 'USA';
```

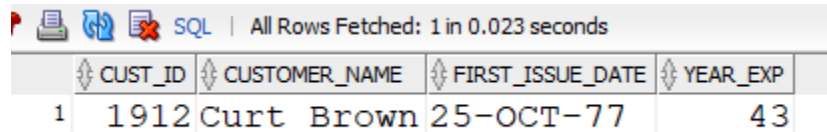


Query Result x
SQL | All Rows Fetched: 2 in 0.023 seconds

	FIRST_NM	LAST_NM	LICENSE_NO	CUST_ID	DOB	LATEST_ISSUE_DATE	ISSUE_STATE	ISSUE_COUNTRY
1	Grace	Garcia	120934874	1705	27-JAN-65	27-MAR-18	France	International
2	Robert	Gross	912376536	1707	25-DEC-69	24-NOV-19	Costa Rica	International

Q7. Identify the customer ID, customer full name in one column (first + last), first issue date and years of total experience for the customer with maximum driving experience (using EQUI JOIN).

```
SELECT customer.cust_id, customer.first_nm || ' ' || customer.last_nm AS Customer_Name,
license.First_issue_date, ROUND ((sysdate - license.First_issue_date)/365,0) AS Year_Exp
FROM customer,license
WHERE license.cust_id = customer.cust_id
AND first_issue_date IN (SELECT MIN(First_issue_date) FROM License);
```

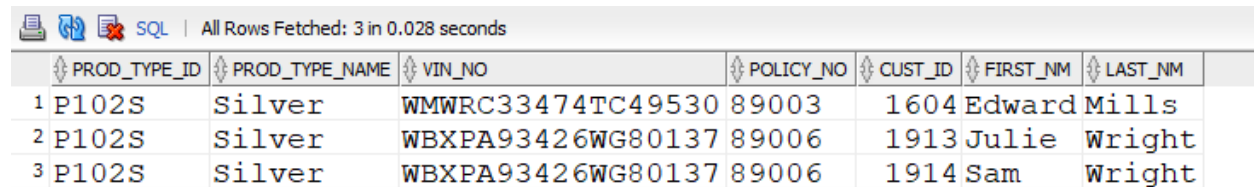


SQL | All Rows Fetched: 1 in 0.023 seconds

	CUST_ID	CUSTOMER_NAME	FIRST_ISSUE_DATE	YEAR_EXP
1	1912	Curt Brown	25-OCT-77	43

Q8. For all policies with Silver Coverage, determine the VIN number, policy number, customer ID, customer first name, customer last name:

```
SELECT PC.PROD_TYPE_ID, PT.PROD_TYPE_NAME, P.VIN_NO, P.POLICY_NO, C.CUST_ID, C.FIRST_NM,
C.LAST_NM
FROM INSU_POLICY P INNER JOIN CUSTOMER C
ON P.POLICY_NO = C.POLICY_NO
INNER JOIN POLICY_COVERAGE PC
ON P.POLICY_NO = PC.POLICY_NO
RIGHT JOIN PRODUCT_TYPE PT
ON PC.PROD_TYPE_ID = PT.PROD_TYPE_ID
WHERE PC.PROD_TYPE_ID = 'P102S';
```



SQL | All Rows Fetched: 3 in 0.028 seconds

	PROD_TYPE_ID	PROD_TYPE_NAME	VIN_NO	POLICY_NO	CUST_ID	FIRST_NM	LAST_NM
1	P102S	Silver	WMWRC33474TC49530	89003	1604	Edward	Mills
2	P102S	Silver	WBXPA93426WG80137	89006	1913	Julie	Wright
3	P102S	Silver	WBXPA93426WG80137	89006	1914	Sam	Wright

Q9. Identify all the VIN numbers that have a mortgage from Bank of America:

```
Select vin_no, lien_holder
from vehicle_lien
where lien_holder = 'Bank Of America';
```

	VIN_NO	LIEN_HOLDER
1	SAJWA1C78D8V38055	Bank Of America
2	WMWRC33474TC49530	Bank Of America
3	1J8GR48K78C206923	Bank Of America

Q10. List all the license numbers with expiration in 2022

```
Select license_no, expiry_date
from license
where EXTRACT(year from expiry_date) = '2022';
```

	LICENSE_NO	EXPIRY_DATE
1	615471827	30-OCT-22
2	518221360	12-JAN-22
3	619852978	30-NOV-22
4	135697465	31-MAY-22

Q11. Display first name, last name, email ids with aliases of all customers in PA state:

```
Select first_nm, last_nm, email, state_nm from customer
where state_nm = 'PA';
```

	FIRST_NM	LAST_NM	EMAIL	STATE_NM
1	Julie	Wright	brown4418@yahoo.com	PA
2	Sam	Wright	sam11@gmail.com	PA

Q12. Rank the order of the most common vehicle type to purchase policy with the company.

```
Select vehicle_type,count(vehicle_type) AS "NO.OF POLICY PURCHASED"
from vehicle
group by vehicle_type
order by count(vehicle_type) Desc;
```

	VEHICLE_TYPE	NO.OF POLICY PURCHASED
1	Sedan	7
2	SUV	2
3	Pickup	2
4	Coupe	1

Q13. Determine the Customer ID, name of top 3 customers on the basis of premium amount?

```
SELECT C.CUST_ID, IP.POLICY_NO, C.FIRST_NM, C.LAST_NM, PREMIUM_AMOUNT,
       RANK() OVER (ORDER BY PREMIUM_AMOUNT DESC) AS TOP_CUSTOMERS
FROM INSU_POLICY IP JOIN CUSTOMER C
ON IP.POLICY_NO = C.POLICY_NO
WHERE ROWNUM <4;
```

	CUST_ID	POLICY_NO	FIRST_NM	LAST_NM	PREMIUM_AMOUNT	TOP_CUSTOMERS
1	1502	89002	Angelia	Green	944.48	1
2	1604	89003	Edward	Mills	938.31	2
3	1501	89001	Tom	Cruise	925.97	3

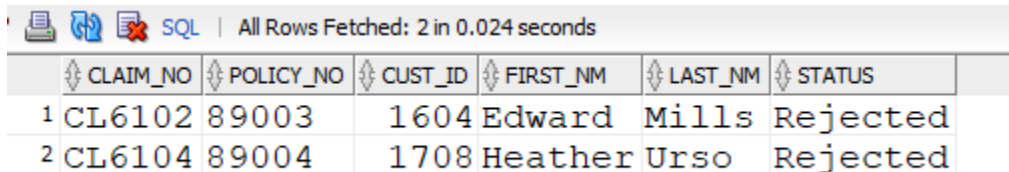
Q14. How many quotes (given to potential customers) were converted into policies successfully?

```
SELECT COUNT(QUOTE_ID) AS NO_OF_SUCCESS_POLICY
FROM CUSTOMER C JOIN INSU_POLICY IP
ON C.POLICY_NO = IP.POLICY_NO JOIN QUOTE Q
ON C.CUST_ID = Q.CUST_ID;
```

SQL All Rows Fetched: 1 in 0.023 seconds	
	NO_OF_SUCCESS_POLICY
1	6

Q15. Provide the customer id, customer first name and last name whose claim was “rejected”?

```
SELECT CL.CLAIM_NO, C.POLICY_NO, C.CUST_ID, C.first_nm, C.last_nm, CL.STATUS
FROM INSU_POLICY IP JOIN CLAIM CL
ON IP.POLICY_NO = CL.POLICY_NO JOIN CUSTOMER C
ON IP.POLICY_NO = C.POLICY_NO
WHERE CL.STATUS = 'Rejected';
```



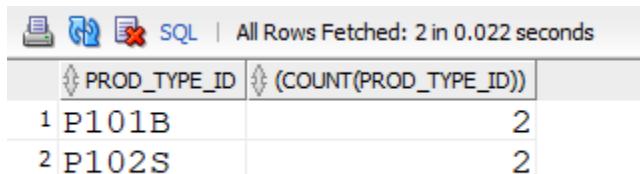
SQL | All Rows Fetched: 2 in 0.024 seconds

	CLAIM_NO	POLICY_NO	CUST_ID	FIRST_NM	LAST_NM	STATUS
1	CL6102	89003	1604	Edward	Mills	Rejected
2	CL6104	89004	1708	Heather	Urso	Rejected

Q16. Which is the most common product coverage with maximum subscription among all ACTIVE policies ?

```
SELECT PROD_TYPE_ID, (COUNT(PROD_TYPE_ID))
FROM POLICY_COVERAGE
WHERE POLICY_STATUS = 'Active'
GROUP BY PROD_TYPE_ID
HAVING (COUNT(PROD_TYPE_ID)) =
(
    SELECT MAX (COUNT(PROD_TYPE_ID))
    FROM POLICY_COVERAGE
    WHERE POLICY_STATUS = 'Active'
    GROUP BY PROD_TYPE_ID
);
```

Answer: There are 2 product types having the same maximum number of subscriptions.



SQL | All Rows Fetched: 2 in 0.022 seconds

	PROD_TYPE_ID	(COUNT(PROD_TYPE_ID))
1	P101B	2
2	P102S	2

9. Additional Features using PL/SQL

This section shows additional features using PL/SQL commands. For the query, used the “cursor”, “IF..THEN” and “trigger” feature in PL/SQL commands.

9.1 Calculate and Print Revenue using “Cursor” and “IF..THEN” statements

This feature helps to calculate the Total Revenue for the company from 2015 year to 2020 year. It also displays the output of the calculation in the SQL client.

-- Set to ON, in order to display in the SQL client

SET SERVEROUTPUT ON;

DECLARE

TOTAL_REVENUE NUMBER(10,2);

TOTAL2015 NUMBER(10,2);

TOTAL2016 NUMBER(10,2);

TOTAL2017 NUMBER(10,2);

TOTAL2018 NUMBER(10,2);

TOTAL2019 NUMBER(10,2);

TOTAL2020 NUMBER(10,2);

CURSOR C_BILL

IS

SELECT

BILL_NO, BILL_AMOUNT, BILL_DATE

FROM

BILL;

BEGIN

TOTAL_REVENUE := 0;

TOTAL2015 := 0;

TOTAL2016 := 0;

TOTAL2017 := 0;

TOTAL2018 := 0;

TOTAL2019 := 0;

TOTAL2020 := 0;

FOR BILL_ROW IN C_BILL

LOOP

if BILL_ROW.BILL_DATE > TO_DATE('01-01-2015', 'MM-DD-YYYY') AND

BILL_ROW.BILL_DATE < TO_DATE('12-31-2015', 'MM-DD-YYYY') then

TOTAL2015 := TOTAL2015 + BILL_ROW.BILL_AMOUNT;

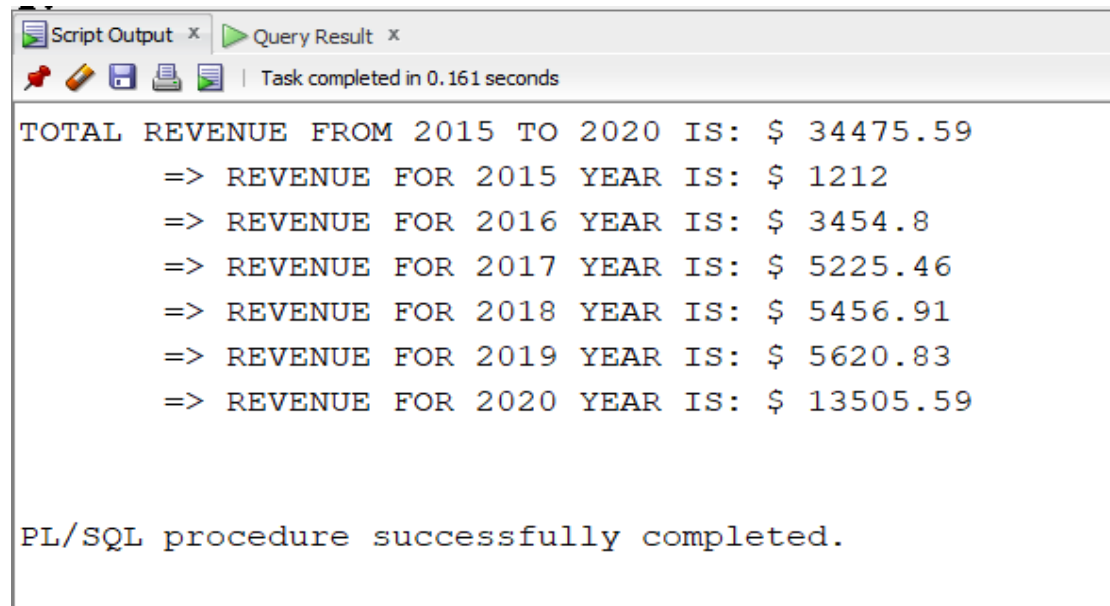
elsif BILL_ROW.BILL_DATE > TO_DATE('01-01-2016', 'MM-DD-YYYY') AND

BILL_ROW.BILL_DATE < TO_DATE('12-31-2016', 'MM-DD-YYYY') then


```

TOTAL2016 := TOTAL2016 + BILL_ROW.BILL_AMOUNT;
elsif BILL_ROW.BILL_DATE > TO_DATE('01-01-2017', 'MM-DD-YYYY') AND
BILL_ROW.BILL_DATE < TO_DATE('12-31-2017','MM-DD-YYYY')then
    TOTAL2017 := TOTAL2017 + BILL_ROW.BILL_AMOUNT;
elsif BILL_ROW.BILL_DATE > TO_DATE('01-01-2018', 'MM-DD-YYYY') AND
BILL_ROW.BILL_DATE < TO_DATE('12-31-2018','MM-DD-YYYY')then
    TOTAL2018 := TOTAL2018 + BILL_ROW.BILL_AMOUNT;
elsif BILL_ROW.BILL_DATE > TO_DATE('01-01-2019', 'MM-DD-YYYY') AND
BILL_ROW.BILL_DATE < TO_DATE('12-31-2019','MM-DD-YYYY')then
    TOTAL2019 := TOTAL2019 + BILL_ROW.BILL_AMOUNT;
else
    TOTAL2020 := TOTAL2020 + BILL_ROW.BILL_AMOUNT;
END IF;
TOTAL_REVENUE := TOTAL_REVENUE + BILL_ROW.BILL_AMOUNT;
END LOOP;
DBMS_OUTPUT.PUT_LINE('TOTAL REVENUE FROM 2015 TO 2020 IS: $ ' || TOTAL_REVENUE);
DBMS_OUTPUT.PUT_LINE('    => REVENUE FOR 2015 YEAR IS: $ ' || TOTAL2015);
DBMS_OUTPUT.PUT_LINE('    => REVENUE FOR 2016 YEAR IS: $ ' || TOTAL2016);
DBMS_OUTPUT.PUT_LINE('    => REVENUE FOR 2017 YEAR IS: $ ' || TOTAL2017);
DBMS_OUTPUT.PUT_LINE('    => REVENUE FOR 2018 YEAR IS: $ ' || TOTAL2018);
DBMS_OUTPUT.PUT_LINE('    => REVENUE FOR 2019 YEAR IS: $ ' || TOTAL2019);
DBMS_OUTPUT.PUT_LINE('    => REVENUE FOR 2020 YEAR IS: $ ' || TOTAL2020);
END;
```

The following is the output:



```

TOTAL REVENUE FROM 2015 TO 2020 IS: $ 34475.59
    => REVENUE FOR 2015 YEAR IS: $ 1212
    => REVENUE FOR 2016 YEAR IS: $ 3454.8
    => REVENUE FOR 2017 YEAR IS: $ 5225.46
    => REVENUE FOR 2018 YEAR IS: $ 5456.91
    => REVENUE FOR 2019 YEAR IS: $ 5620.83
    => REVENUE FOR 2020 YEAR IS: $ 13505.59

PL/SQL procedure successfully completed.
```

9.2 Audit Log using “Trigger” Command

This feature tracks any update or deletion done to “customer” table of the database. The “Trigger” function is used to implement this feature. Below code is written to track the changes to “customer table.

A new TABLE called “audits” is created and the changes are stored in it automatically when any changes are done to “customer” table.

```
DROP TABLE audits CASCADE CONSTRAINTS;
```

```
CREATE TABLE audits(  
    table_name    VARCHAR2(255),  
    transaction_name VARCHAR2(10),  
    by_user       VARCHAR2(30),  
    transaction_date DATE  
);
```

```
CREATE OR REPLACE TRIGGER customers_audit_trg  
    AFTER  
    UPDATE OR DELETE  
    ON customer  
    FOR EACH ROW
```

```
DECLARE
```

```
    l_transaction VARCHAR2(10);
```

```
BEGIN
```

```
    -- determine the transaction type
```

```
    l_transaction := CASE
```

```
        WHEN UPDATING THEN 'UPDATE'
```

```
        WHEN DELETING THEN 'DELETE'
```

```
    END;
```

```
    -- insert a row into the audit table
```

```
    INSERT INTO audits (table_name, transaction_name, by_user, transaction_date)
```

```
    VALUES('CUSTOMER', l_transaction, USER, SYSDATE);
```

```
END;
```

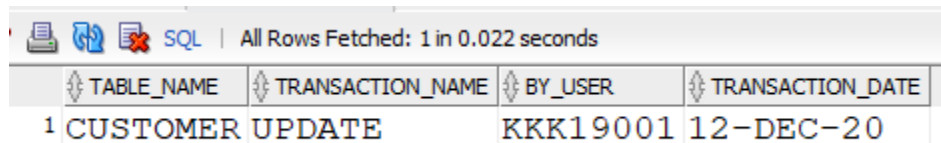
In order to demonstrate the feature, the email of customer in row with cust_id of 1501 is updated with below commands:

```
UPDATE
  customer
SET
  email = 'new@email.com'
WHERE
  cust_id = 1501;
```

Then when “audits” table is checked by using following command.

```
SELECT * FROM audits;
```

It can be seen in the output (shown below) that a new record is created automatically showing the user who did the update to “customer” table.



The screenshot shows a database query result window. At the top, there are icons for file operations and a status bar indicating 'All Rows Fetched: 1 in 0.022 seconds'. Below this is a table with four columns: TABLE_NAME, TRANSACTION_NAME, BY_USER, and TRANSACTION_DATE. The table contains one row with the following values: 1, CUSTOMER, UPDATE, KKK19001, and 12-DEC-20.

	TABLE_NAME	TRANSACTION_NAME	BY_USER	TRANSACTION_DATE
1	CUSTOMER	UPDATE	KKK19001	12-DEC-20