

Name: Alison Zeng

Mini Project SQL

Creating tables in Insurance Database:

[Query Editor](#) [Query History](#)

```
1 CREATE TABLE claim_table (  
2     Claim_ID_Number serial PRIMARY KEY,  
3     Claim_Receipt_Date TIMESTAMP,  
4     Claim_Description VARCHAR,  
5     Claimant_Birth_Date TIMESTAMP,  
6     Law_Suit_Indicator VARCHAR,  
7     Claimant_First_Name VARCHAR,  
8     Claimant_Last_Name VARCHAR,  
9     Policy_number int,  
10    Claimant_Occupatoin VARCHAR,  
11    Loss_Date TIMESTAMP,  
12    Claim_Paid_Amount int,  
13    Claim_Paid_Date Timestamp);
```

[Data Output](#) [Messages](#) [Explain](#) [Notifications](#)

CREATE TABLE

Query returned successfully in 112 msec.

[Query Editor](#) [Query History](#)

```
1 CREATE TABLE customer_table (  
2     Customer_ID_Number int,  
3     Customer_First_Name VARCHAR,  
4     Customer_Last_Name VARCHAR,  
5     Customer_Type VARCHAR,  
6     Business_Name VARCHAR,  
7     Address_Street_Name_Line_One VARCHAR,  
8     Address_Street_Name_Line_Two VARCHAR,  
9     Address_City VARCHAR,  
10    Address_State VARCHAR,  
11    Address_Postal_Code int,  
12    Address_Type VARCHAR,  
13    Customer_Birth_Date TIMESTAMP,  
14    Risk_Score int);
```

[Data Output](#) [Messages](#) [Explain](#) [Notifications](#)

CREATE TABLE

Query returned successfully in 56 msec.

[Query Editor](#) [Query History](#)

```
1 CREATE TABLE policy_detail (  
2     Customer_ID_Number int PRIMARY KEY,  
3     Policy_Number int,  
4     Policy_Effective_Date TIMESTAMP,  
5     Policy_Expiration_Date TIMESTAMP,  
6     Policy_Limit VARCHAR,  
7     Policy_Type VARCHAR,  
8     Currency_Type VARCHAR,  
9     Premium_Cost VARCHAR);
```

[Data Output](#) [Messages](#) [Explain](#) [Notifications](#)

CREATE TABLE

Query returned successfully in 48 msec.

[Query Editor](#) [Query History](#)

```
1 CREATE TABLE reserves (  
2     Customer_ID_Number int PRIMARY KEY,  
3     Policy_Number int,  
4     Reserved_Amount VARCHAR,  
5     Reserved_Date TIMESTAMP);  
6  
7
```

[Data Output](#) [Messages](#) [Explain](#) [Notifications](#)

CREATE TABLE

Query returned successfully in 64 msec.

Inputing data into tables:

Insurance/postgres@PostgreSQL 13

Query EditorQuery History

```
1 copy claim_table (  
2     Claim_ID_Number,  
3     Claim_Receipt_Date,  
4     Claim_Description,  
5     Claimant_Birth_Date,  
6     Law_Suit_Indicator,  
7     Claimant_First_Name,  
8     Claimant_Last_Name,  
9     Policy_Number,  
10    Claimant_Occupation,  
11    Loss_Date,  
12    Claim_Amount,  
13    Claim_Paid_Amount,  
14    Claim_Paid_Date)  
15 --set the path for file location of student_marks.csv  
16 from '/Users/Shared/Insurance_Sample Data.csv'  
17 delimiter ',' CSV header
```

Data OutputMessagesExplainNotifications

COPY 34

Query returned successfully in 98 msec.

Query EditorQuery History

```
1 CREATE TABLE reserves (  
2 Customer_ID_Number VARCHAR,  
3 Policy_Number VARCHAR,  
4 Reserved_Amount VARCHAR,  
5 Reserved_Date VARCHAR);  
6  
7 Copy reserves (Customer_ID_Number,  
8 Policy_Number,  
9 Reserved_Amount,  
10 Reserved_Date)  
11 FROM '/Users/Shared/Insurance_Reserves.csv'  
12 delimiter ',' CSV header
```

Data OutputMessagesExplainNotifications

COPY 6

Query returned successfully in 2 secs 627 msec.

Results:

Claim table:

Query EditorQuery History

```
1 SELECT * FROM public.claims  
2
```

Data OutputMessagesExplainNotifications

	claim_id_number character varying	claim_receipt_date character varying	claim_description character varying	claimant_birth_date character varying	law_suit_indicator character varying	claimant_first_name character varying	claimant_last_name character varying	policy_number character varying	claimant_occupation character varying
1	Claim_ID_Number	Claim_Receipt_Date	Claim Description	Claimant_Birth_Date	Law_Suit_Indicator	Claimant_First_Name	Claimant_Last_Name	Policy_Number	Claimant_Occupation
2	C123450000	12/5/2020	Automobile accident	8/1/1975	N	Joe	Smith	P987654321	Physician
3	C123441000	8/5/2019	Automobile accident	8/1/1975	N	Joe	Smith	P987654321	Physician
4	C123440000	10/15/2020	Robbery	3/20/1986	N	Mary	Roberts	P987654111	Not provided
5	C345478900	7/13/2020	Property	5/27/1990	N	Aditya	Reddy	P987654222	Business Owner

Query EditorQuery History

```
1 SELECT * FROM public.customer_table  
2
```

Data OutputMessagesExplainNotifications

	customer_first_name character varying	customer_last_name character varying	customer_type character varying	business_name character varying	address_street_name_line_one character varying	address_street_name_line_two character varying	address_city character varying	address_state character varying	address character varying
1	Customer_First_Name	Customer_Last_Name	Customer_Type	Business_Name	Address_Street_Name_Line_One	Address_Street_Name_Line_Two	Address_City	Address_State	Address
2	Mary	Roberts	Individual	[null]	123 Main Street	[null]	Chicago	Illinois	Residence
3	Joe	Smith	Individual	[null]	123 Park Avenue	[null]	New York	New York	Residence
4	Aditya	Reddy	Business	ABC Systems	222 Hyde Street	Floor 2	San Francisco	California	Business

Policy details table:

Query Editor

Query History

1

SELECT * FROM public.policy_detail

2

Data Output

Messages

Explain

Notifications

	customer_id_number character varying	policy_number character varying	policy_effective_date character varying	policy_expiration_date character varying	policy_limit character varying	policy_type character varying	currency_type character varying	premium_cost character varying
1	Customer_ID_Number	Policy_Number	Policy_Effective_Date	Policy_Expiration_Date	Policy_Limit	Policy_Type	Currency_Type	Premium_Cost
2	777543800	P987654321	2/11/2020	2/11/2021	500,000.00	Automobile	USD	2600.00
3	777543800	P987654321	2/11/2019	2/11/2020	500,000.00	Automobile	USD	2300.00
4	777543800	P987654321	2/11/2018	2/11/2019	500,000.00	Automobile	USD	2150.00
5	777543800	P987654321	2/11/2017	2/11/2018	500,000.00	Automobile	USD	2100.00
6	777543325	P987654111	3/15/2020	3/15/2021	5,000,000.00	Excess/Umbrella	USD	5500.00
7	776983401	P987654222	6/15/2020	6/15/2020	3,000,000.00	Property	USD	10000.00

Reserves table:

Query Editor

Query History

1

SELECT * FROM public.reserves

2

Data Output

Messages

Explain

Notifications

	customer_id_number character varying	policy_number character varying	reserved_amount character varying	reserved_date character varying
1	Customer_ID_Number	Policy_Number	Reserved_Amount	Reserved_Date
2	777543325	P987654111	5000.00	3/15/2020
3	776983401	P987654222	7000.00	6/15/2020
4	777543800	P987654321	1500.00	2/11/2017
5	777543800	P987654321	1500.00	2/11/2018
6	777543800	P987654321	1500.00	2/11/2019

1. Counting how many Automobile insurance policies there are

Query Editor		Query History
<pre>1 SELECT count(policy_type) FROM policy_detail 2 WHERE policy_type='Automobile';</pre>		
Data Output		Messages Explain Notifications
	count bigint	
1	4	

2. Using JOIN to find all rows from policy detail table with matching customer_id_number (if available) in reserves table

Query Editor

Query History

```
1 SELECT * FROM policy_detail
2 RIGHT JOIN reserves
3 On policy_detail.customer_id_number=reserves.customer_id_number;
```

Data Output


Messages

Explain

Notifications

	customer_id_number character varying	policy_number character varying	policy_effective_date character varying	policy_expiration_date character varying	policy_limit character varying	policy_type character varying	currency_type character varying	premium_cost character varying	customer_id_number character varying
1	Customer_ID_Number	Policy_Number	Policy_Effective_Date	Policy_Expiration_Date	Policy_Limit	Policy_Type	Currency_Type	Premium_Cost	Customer_ID_Number
2	777543325	P987654111	3/15/2020	3/15/2021	5,000,000.00	Excess/Umbrella	USD	5500.00	777543325
3	776983401	P987654222	6/15/2020	6/15/2020	3,000,000.00	Property	USD	10000.00	776983401
4	777543800	P987654321	2/11/2017	2/11/2018	500,000.00	Automobile	USD	2100.00	777543800
5	777543800	P987654321	2/11/2018	2/11/2019	500,000.00	Automobile	USD	2150.00	777543800
6	777543800	P987654321	2/11/2019	2/11/2020	500,000.00	Automobile	USD	2300.00	777543800
7	777543800	P987654321	2/11/2020	2/11/2021	500,000.00	Automobile	USD	2600.00	777543800
8	777543800	P987654321	2/11/2017	2/11/2018	500,000.00	Automobile	USD	2100.00	777543800
9	777543800	P987654321	2/11/2018	2/11/2019	500,000.00	Automobile	USD	2150.00	777543800
10	777543800	P987654321	2/11/2019	2/11/2020	500,000.00	Automobile	USD	2300.00	777543800
11	777543800	P987654321	2/11/2020	2/11/2021	500,000.00	Automobile	USD	2600.00	777543800
12	777543800	P987654321	2/11/2017	2/11/2018	500,000.00	Automobile	USD	2100.00	777543800
13	777543800	P987654321	2/11/2018	2/11/2019	500,000.00	Automobile	USD	2150.00	777543800
14	777543800	P987654321	2/11/2019	2/11/2020	500,000.00	Automobile	USD	2300.00	777543800
15	777543800	P987654321	2/11/2020	2/11/2021	500,000.00	Automobile	USD	2600.00	777543800

3. Which policy details contains customer_id_number with '00' anywhere in it



Insurance/postgres@PostgreSQL 13 ▾

Query Editor

Query History

1

SELECT * FROM reserves

2

WHERE customer_id_number LIKE '%00%';

Data Output

Messages

Explain

Notifications

	<div>customer_id_number</div> <div>character varying</div>	<div>policy_number</div> <div>character varying</div>	<div>reserved_amount</div> <div>character varying</div>	<div>reserved_date</div> <div>character varying</div>
1	777543800	P987654321	1500.00	2/11/2017
2	777543800	P987654321	1500.00	2/11/2018
3	777543800	P987654321	1500.00	2/11/2019