Project Report

Reasons and motivations:

1. In this Internet era, credit card fraud is one of the most common economic crimes. Banks need to identify the customers carefully and roundly to lower risks, which is a tremendous task.
2. From this analysis, we can find some characteristics with which people are less likely to miss payments. Therefore, banks can divide customers into different groups and quantify the risks.
3. This project is in a real business scenario and can help us better understand banking and financial services. We can apply what we find and learn in our future careers when working in the finance industry.

Research:

Some findings reflect that smaller banks and credit unions (with less than $10 billion in assets) and digital lenders have been hit harder by small business lending fraud, which has increased at a higher rate since the beginning of 2020. As a percentage of revenue, they experienced an average fraud loss of 6.9%, compared to 5.9% for large banks ($10 billion in assets). And they are likely to expect fraud levels to rise in the year ahead.

Study results also show that organizations that reported that they're more effective at identifying small business lending fraud actually are more effective. They've not only largely prevented fraud increases over the past 24 months but experienced a lower level of fraud – 3.0% of annual revenues (vs. 5.0% among those that are reported as only somewhat effective at identifying small business lending fraud).

The initial plan of action:

1. Apply Data wrangling processes to ensure every data are suitable for operation.
2. Deal with missing values by dropping the rows, forcing a constant, imputing data with a package like mice, or inferring the value.
3. Find relationships between customers' characteristics and their possibility of missing payments.
4. Divide the customers into different groups based on the relationships and evaluate the level of reliability to give them loans.
5. Summarize.

Improvement in decision-making:

Our project can help banks better understand the driving factors behind loan defaults. We can quantify the risk of loaning to each customer according to their specific information. The results can also be applied to loan portfolios and credit assessments.

Other Benefits:
Effectively identifying SMB fraud before it makes its way into your loan portfolio is critical to protecting market share and avoiding revenue loss and operational delays caused by fraud. The right tools and layered techniques support proactive strategies to create an advantage in identifying and preventing SMB fraud. Comprehensive fraud solutions and proven analytics provide the risk intelligence the banks need to get a comprehensive view of the business and its partners.

Group Contract:
Leader: Weiyi Ni
Group Member Responsibilities:
All members should participate in the team meetings. All tasks are equally distributed after the previous meeting.

Task No.1
Group Contract: Due by **4/2/2022** and Completed and Approved by all group members on **3/31/2022.**

Task No.2
Due by 4/9/2022
Checkpoint No.3: Clean Dataset by using Python, Discuss normalization, Submit the Database Schema (ER diagram).

- Data Wrangling and Create Tables - Zhenwei Wei and Weiyi Ni
- Build 3NF - Mengyi Lyu and Shiyuan Ru
- Create ER diagram - Jianan Liu and Zhuojing Xie

Task No.3
Due by 4/16/2022
Checkpoint No.4: Develop Python Script.

- Table Gender, Property, Address_Match - Zhuojing Xie
- Table External_Source, AMT_REQ_Credit_Bureau, Region - Jianan Liu
- Table Building_Info, Observation, Income - Mengyi Lyu
- Table Document, Occupation, Contact - Shiyuan Ru
- Table Suite, Contract, Target - Zhenwei Wei
- TableFamily, Organization, Loan_Applicant - Weiyi Ni

Task No.4
Due by 4/23/2022

Checkpoint No.5: Design a Customer Interaction Plan.

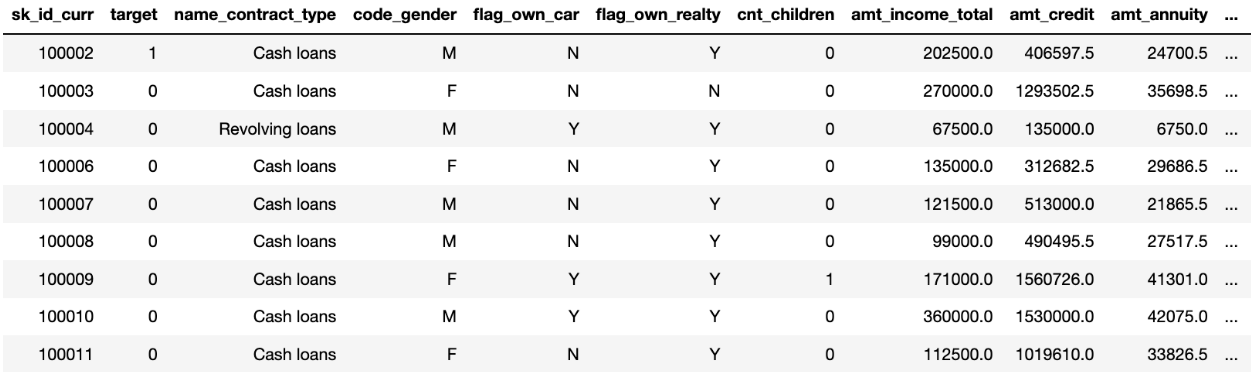
- What will you implement for analysts (direct querying) and for "C" level officers (reports)? - Zhenwei Wei and Weiyi Ni
- What tools are you using? - Mengyi Lyu and Shiyuan Ru
- Did you plan for redundancy and performance? - Jianan Liu and Zhuojing Xie

Task No.5

Due by 4/29/2022

Final Deliverable: Project Presentation and Report.

Sample of data:



sk_id_curr	target	name_contract_type	code_gender	flag_own_car	flag_own_realty	cnt_children	amt_income_total	amt_credit	amt_annuity	...
100002	1	Cash loans	M	N	Y	0	202500.0	406597.5	24700.5	...
100003	0	Cash loans	F	N	N	0	270000.0	1293502.5	35698.5	...
100004	0	Revolving loans	M	Y	Y	0	67500.0	135000.0	6750.0	...
100006	0	Cash loans	F	N	Y	0	135000.0	312682.5	29686.5	...
100007	0	Cash loans	M	N	Y	0	121500.0	513000.0	21865.5	...
100008	0	Cash loans	M	N	Y	0	99000.0	490495.5	27517.5	...
100009	0	Cash loans	F	Y	Y	1	171000.0	1560726.0	41301.0	...
100010	0	Cash loans	M	Y	Y	0	360000.0	1530000.0	42075.0	...
100011	0	Cash loans	F	N	Y	0	112500.0	1019610.0	33826.5	...

Our group decided to focus on credit card fraud based on the Kaggle dataset from https://www.kaggle.com/datasets/mishra5001/credit-card.

This dataset contains 122 variables, such as flag_own_car, cnt_children, and amt_income_total, including specific and comprehensive information about each loan applicant. There is also a previous dataset that we can use to make some comparisons. We can use some correlation coefficients to examine the strength and direction of the relationship between two continuous variables to detect which applicants are less likely to miss payments.

Normalization

We use ER Studio Data Architect 19.1 to create the ER Diagrams.

1NF:

Domains of all table attributes must be atomic and there cannot be repeating attributes.

Previous columns	Columns after 1NF
flag_own_car flag_own_realty	property_id property_name

flag_mobil … flag_email	contact_id contact_type contact_provided
region_rating_client region_rating_client_w_city	region_id region_name region_rating
reg_region_not_live_region … live_region_not_work_region	address_match_id address_match_type address_matched
external_source_1 … external_source_3	external_source_id external_source_value
apartments_avg … totalarea_mode	building_info_id building_info_type building_info_value
obs_30_cnt_social_circle … def_60_cnt_social_circle	observation_id observation_type observation_value
flag_document_2 … flag_document_21	document_id document_value

2NF:

Must be in 1NF and every non-key attribute must be *fully* dependent on the key (aka non-key columns cannot be dependent on a *part* of the primary key).

In 2NF, we divided the original table into 18 tables according to their categories.

Table	Attributes
Loan_Applicant	Sk_id_curr (PK) Own_car_age Days_registration Days_id_published Days_birth

	Gender_id (FK) Income_id (FK) Amt_req_credit_bureau_id (FK)
External_source	External_source_id (PK) Sk_id_curr (PK, FK) External_source_value
Observation	Observation_id (PK) Sk_id_curr (PK, FK) Observation_type Observation_value
Region	Region_id (PK) Sk_id_curr (PK, FK) Region_name Region_rating
Building_info	Building_info_id (PK) Sk_id_curr (PK, FK) Building_info_type Building_info_value
Document	Document_id (PK) Sk_id_curr (PK, FK) Document_provided
Amt_req_credit_bureau	Amt_req_credit_bureau_id (PK) Amt_req_credit_bureau_hour Amt_req_credit_bureau_day Amt_req_credit_bureau_week Amt_req_credit_bureau_mon Amt_req_credit_bureau_qrt Amt_req_credit_bureau_year
Income	Income_id (PK) Income_type Annual_income_total
Gender	Gender_id (PK)

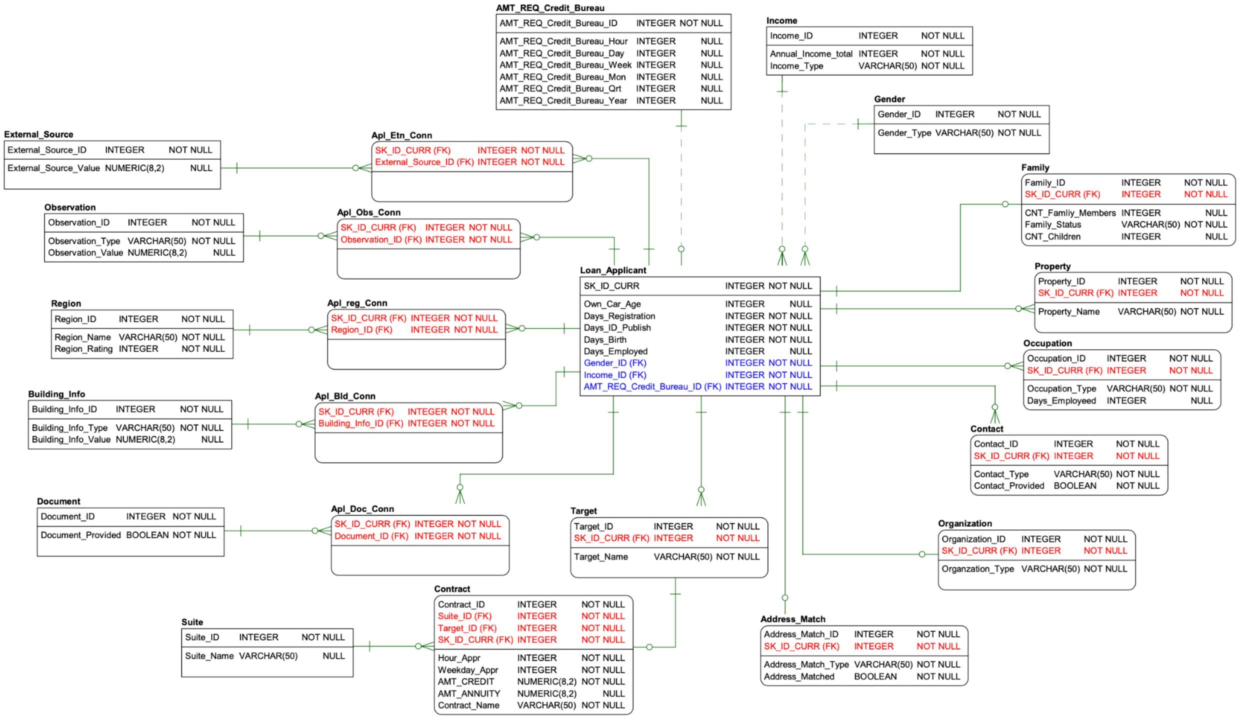
	Gender_type
Family	Family_id (PK) Sk_id_curr (PK, FK) Cnt_family_members Family_status cnt_children
Property	Property_id (PK) Sk_id_curr (PK, FK) Property_name
Occupation	Occupation_id (PK) Sk_id_curr (PK, FK) Occupation_type Days_employeed
Contact	Contact_id (PK) Sk_id_curr (PK, FK) Contact_type Contact_provided
Organization	Organization_id (PK) Sk_id_curr (PK, FK) Organzation_type
Address_match	Address_match_id (PK) Sk_id_curr (PK, FK) Address_match_type Address_matched
Target	Target_id (PK) Sk_id_curr (PK, FK) Target_name
Suite	Suite_id (PK) Suite_name
Contract	Contract_id (PK) Sk_id_curr (PK, FK) Target_id (PK, FK)

	Suite_id (PK, FK) Hour_appr Weekday_appr Amt_credit Amt_annuity contract_name

3NF:
Must be in 2NF and every non-key attribute must be non-transitively dependent on the key (aka all attributes must only depend on the key).
The relationship between table loan_applicant and external_source, observation, region, building_info, document is many to many. Hence, we can relatively create 5 bridge table to connect these tables.

Bridge table	Attributes
Apl_etn_conn	Sk_id_curr (PK, FK) External_source_id (PK, FK)
Apl_obs_conn	Sk_id_curr (PK, FK) Observation_id (PK, FK)
Apl_reg_conn	Sk_id_curr (PK, FK) Region_id (PK, FK)
Apl_bld_conn	Sk_id_curr (PK, FK) Building_info_id (PK, FK)
Apl_doc_conn	Sk_id_curr (PK, FK) Document_id (PK, FK)

Here is our final ER Diagram.


AMT_REQ_Credit_Bureau
AMT_REQ_Credit_Bureau_ID INTEGER NOT NULL
AMT_REQ_Credit_Bureau_Hour INTEGER NULL
AMT_REQ_Credit_Bureau_Day INTEGER NULL
AMT_REQ_Credit_Bureau_Week INTEGER NULL
AMT_REQ_Credit_Bureau_Mon INTEGER NULL
AMT_REQ_Credit_Bureau_Qrt INTEGER NULL
AMT_REQ_Credit_Bureau_Year INTEGER NULL
Income
Income_ID INTEGER NOT NULL
Annual_Income_total INTEGER NOT NULL
Income_Type VARCHAR(50) NOT NULL
Gender
Gender_ID INTEGER NOT NULL
Gender_Type VARCHAR(50) NOT NULL
External_Source
External_Source_ID INTEGER NOT NULL
External_Source_Value NUMERIC(8,2) NULL
Apl_Etn_Conn
SK_ID_CURR (FK) INTEGER NOT NULL
External_Source_ID (FK) INTEGER NOT NULL
Family
Family_ID INTEGER NOT NULL
SK_ID_CURR (FK) INTEGER NOT NULL
CNT_Famliy_Members INTEGER NULL
Family_Status VARCHAR(50) NOT NULL
CNT_Children INTEGER NULL
Observation
Observation_ID INTEGER NOT NULL
Observation_Type VARCHAR(50) NOT NULL
Observation_Value NUMERIC(8,2) NULL
Apl_Obs_Conn
SK_ID_CURR (FK) INTEGER NOT NULL
Observation_ID (FK) INTEGER NOT NULL
Loan_Applicant
SK_ID_CURR INTEGER NOT NULL
Own_Car_Age INTEGER NULL
Days_Registration INTEGER NOT NULL
Days_ID_Publish INTEGER NOT NULL
Days_Birth INTEGER NOT NULL
Days_Employed INTEGER NULL
Gender_ID (FK) INTEGER NOT NULL
Income_ID (FK) INTEGER NOT NULL
AMT_REQ_Credit_Bureau_ID (FK) INTEGER NOT NULL
Property
Property_ID INTEGER NOT NULL
SK_ID_CURR (FK) INTEGER NOT NULL
Property_Name VARCHAR(50) NOT NULL
Region
Region_ID INTEGER NOT NULL
Region_Name VARCHAR(50) NOT NULL
Region_Rating INTEGER NOT NULL
Apl_reg_Conn
SK_ID_CURR (FK) INTEGER NOT NULL
Region_ID (FK) INTEGER NOT NULL
Occupation
Occupation_ID INTEGER NOT NULL
SK_ID_CURR (FK) INTEGER NOT NULL
Occupation_Type VARCHAR(50) NOT NULL
Days_Employeed INTEGER NULL
Building_Info
Building_Info_ID INTEGER NOT NULL
Building_Info_Type VARCHAR(50) NOT NULL
Building_Info_Value NUMERIC(8,2) NULL
Apl_Bld_Conn
SK_ID_CURR (FK) INTEGER NOT NULL
Building_Info_ID (FK) INTEGER NOT NULL
Contact
Contact_ID INTEGER NOT NULL
SK_ID_CURR (FK) INTEGER NOT NULL
Contact_Type VARCHAR(50) NOT NULL
Contact_Provided BOOLEAN NOT NULL
Document
Document_ID INTEGER NOT NULL
Document_Provided BOOLEAN NOT NULL
Apl_Doc_Conn
SK_ID_CURR (FK) INTEGER NOT NULL
Document_ID (FK) INTEGER NOT NULL
Target
Target_ID INTEGER NOT NULL
SK_ID_CURR (FK) INTEGER NOT NULL
Target_Name VARCHAR(50) NOT NULL
Organization
Organization_ID INTEGER NOT NULL
SK_ID_CURR (FK) INTEGER NOT NULL
Organzation_Type VARCHAR(50) NOT NULL
Contract
Contract_ID INTEGER NOT NULL
Suite_ID (FK) INTEGER NOT NULL
Target_ID (FK) INTEGER NOT NULL
SK_ID_CURR (FK) INTEGER NOT NULL
Hour_Appr INTEGER NOT NULL
Weekday_Appr INTEGER NOT NULL
AMT_CREDIT NUMERIC(8,2) NOT NULL
AMT_ANNUITY NUMERIC(8,2) NULL
Contract_Name VARCHAR(50) NOT NULL
Suite
Suite_ID INTEGER NOT NULL
Suite_Name VARCHAR(50) NULL
Address_Match
Address_Match_ID INTEGER NOT NULL
SK_ID_CURR (FK) INTEGER NOT NULL
Address_Match_Type VARCHAR(50) NOT NULL
Address_Matched BOOLEAN NOT NULL

ETL process:

This dataset has about 300,000 rows and 122 columns. The column that we should focus on is Target, which has 2 values: 1 - client with payment difficulties: he/she had late payment more than X days on at least one of the first Y installments of the loan in our sample, 0 - all other cases. We need to find the relationship between target with other variables to find which customers are less likely to miss payment.

Step 1:

Drop duplicates based on sk_id_curr (the primary key that identifies each customer)

Step 2:

Deal with the null values. This dataset has 67 columns with null values. In this case, we set the threshold to be 30%, i.e. if the percentage of null value in a column is larger than 30%, then this columns is considered to be useless and we can delete it. After this process, we have 72 columns left. To deal with the null values in these columns, we can take 3 methods.

- Force a Constant
- Use a Package like mice to impute
- Infer the Value

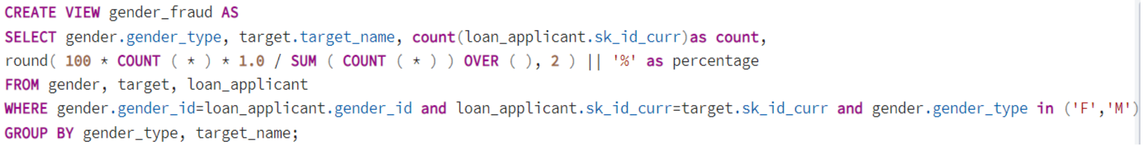
Step 3:

Unite a format. This dataset has some columns with many decimal digits, which is lack of consistency and readability. Hence, we set the decimal digits to be 2 to handle this problem. In addition, all columns are presented in upper case. To increase the convenience for reading, we transform them to lower case and change some abbrivations.

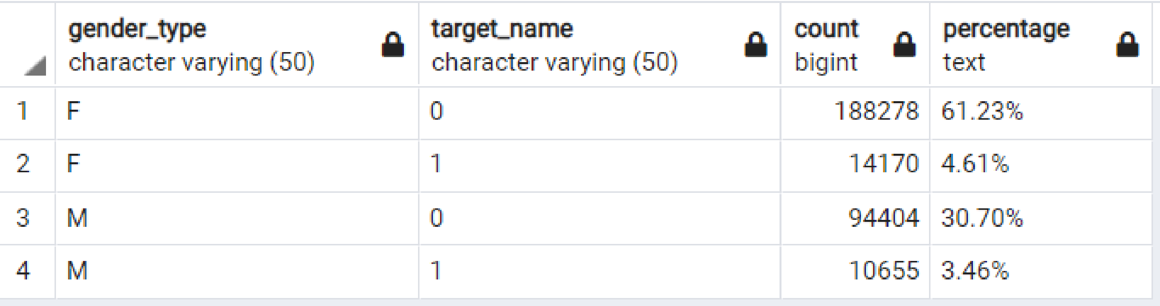
Step 4:
Modify data type. Some boolean data is represented by number 0 and 1, some date data is represented by number. We modify the data type to suitable formats.

Customer Interaction Plan
Through the database system, we can further help companies reduce the number of fraud and help companies increase the number of users. Our plan is that the analyst can analyze from two aspects: the user side and the product side. By analyzing user characteristics, the analyst can understand the current customer type and discover which customers are high quality. By analyzing the characteristics of products, companies can realize the advantages and disadvantages of products and better improve them.
Firstly, for the user side, they can query customers' gender, age, income, occupation, and family status to determine which types of customers are prone to delinquency and which customers are willing to apply for credit cards.
Secondly, for the product side, analysts can realize which types of loans have low risks and good stability by inquiring about loan types and loan status because different loan types may relate to fraud rates.
From these two aspects, we came up with some related questions that analysts can use directly with our data in conjunction with the SQL Query tool:

1. **How many frauds are committed by women and men, respectively?**

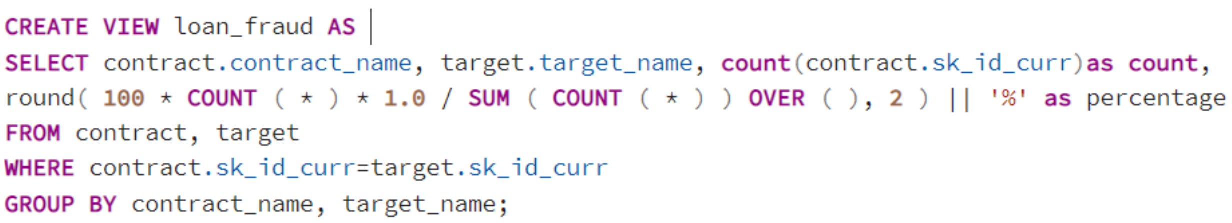


```
CREATE VIEW gender_fraud AS
SELECT gender.gender_type, target.target_name, count(loan_applicant.sk_id_curr)as count,
round( 100 * COUNT ( * ) * 1.0 / SUM ( COUNT ( * ) ) OVER ( ), 2 ) || '%' as percentage
FROM gender, target, loan_applicant
WHERE gender.gender_id=loan_applicant.gender_id and loan_applicant.sk_id_curr=target.sk_id_curr and gender.gender_type in ('F','M')
GROUP BY gender_type, target_name;
```

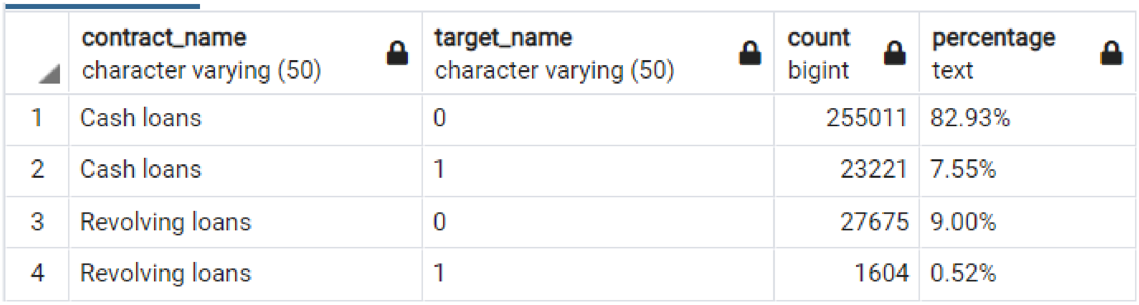


	gender_type character varying (50)	target_name character varying (50)	count bigint	percentage text
1	F	0	188278	61.23%
2	F	1	14170	4.61%
3	M	0	94404	30.70%
4	M	1	10655	3.46%

2. **What is the number of frauds and non-frauds in different loan types?**

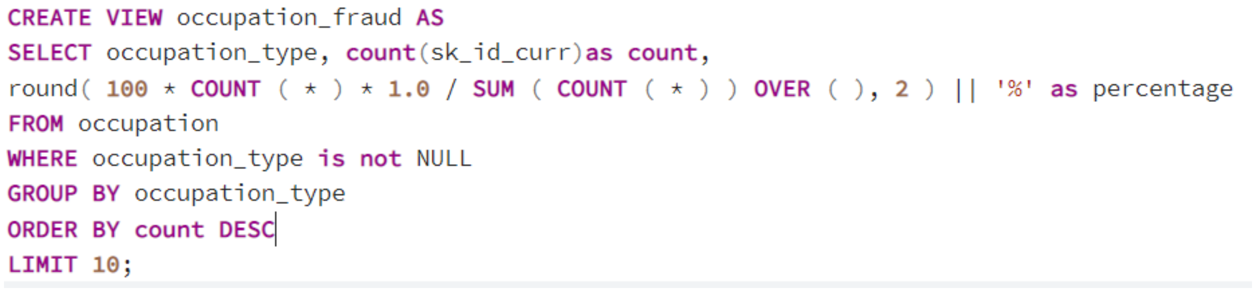


```sql
CREATE VIEW loan_fraud AS
SELECT contract.contract_name, target.target_name, count(contract.sk_id_curr)as count,
round( 100 * COUNT ( * ) * 1.0 / SUM ( COUNT ( * ) ) OVER ( ), 2 ) || '%' as percentage
FROM contract, target
WHERE contract.sk_id_curr=target.sk_id_curr
GROUP BY contract_name, target_name;
```

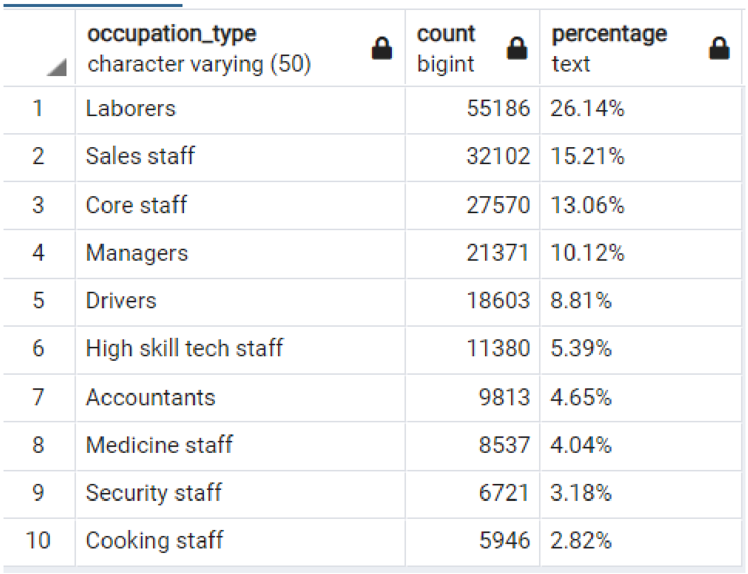


	contract_name character varying (50)	target_name character varying (50)	count bigint	percentage text
1	Cash loans	0	255011	82.93%
2	Cash loans	1	23221	7.55%
3	Revolving loans	0	27675	9.00%
4	Revolving loans	1	1604	0.52%

3. **Which occupation has the most credit card applications（top 10）?**

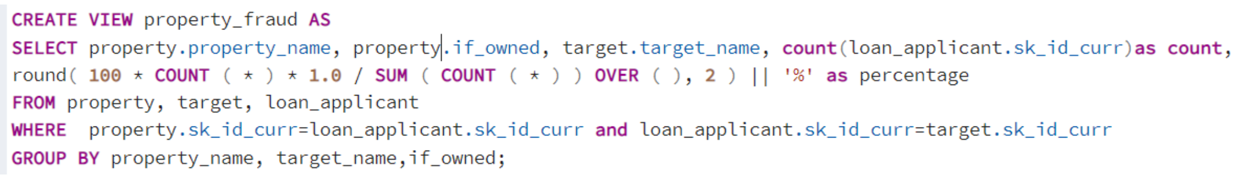


```sql
CREATE VIEW occupation_fraud AS
SELECT occupation_type, count(sk_id_curr)as count,
round( 100 * COUNT ( * ) * 1.0 / SUM ( COUNT ( * ) ) OVER ( ), 2 ) || '%' as percentage
FROM occupation
WHERE occupation_type is not NULL
GROUP BY occupation_type
ORDER BY count DESC
LIMIT 10;
```

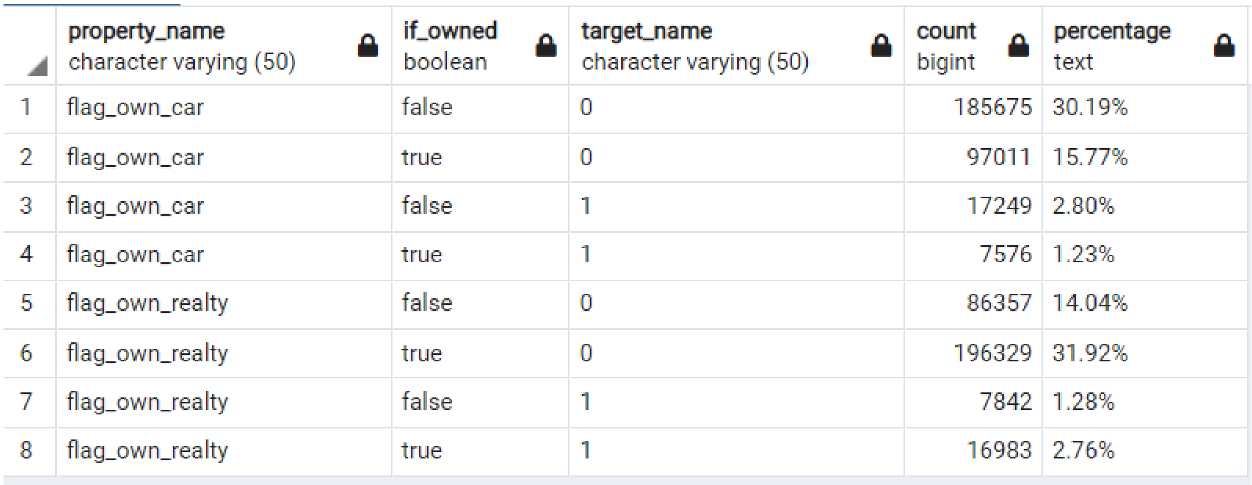


	occupation_type character varying (50)	count bigint	percentage text
1	Laborers	55186	26.14%
2	Sales staff	32102	15.21%
3	Core staff	27570	13.06%
4	Managers	21371	10.12%
5	Drivers	18603	8.81%
6	High skill tech staff	11380	5.39%
7	Accountants	9813	4.65%
8	Medicine staff	8537	4.04%
9	Security staff	6721	3.18%
10	Cooking staff	5946	2.82%

4. **How many frauds are committed by people who don’t own a car and realty versus those who own a car and a realty, respectively?**

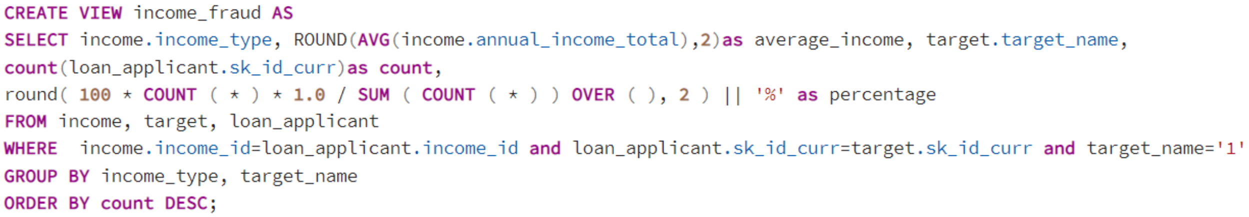


```
CREATE VIEW property_fraud AS
SELECT property.property_name, property.if_owned, target.target_name, count(loan_applicant.sk_id_curr)as count,
round( 100 * COUNT ( * ) * 1.0 / SUM ( COUNT ( * ) ) OVER ( ), 2 ) || '%' as percentage
FROM property, target, loan_applicant
WHERE  property.sk_id_curr=loan_applicant.sk_id_curr and loan_applicant.sk_id_curr=target.sk_id_curr
GROUP BY property_name, target_name,if_owned;
```

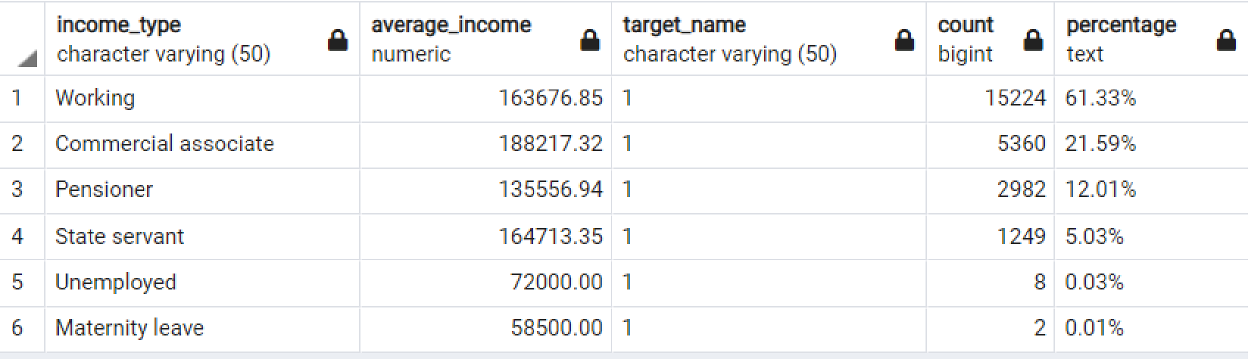


	property_name character varying (50)	if_owned boolean	target_name character varying (50)	count bigint	percentage text
1	flag_own_car	false	0	185675	30.19%
2	flag_own_car	true	0	97011	15.77%
3	flag_own_car	false	1	17249	2.80%
4	flag_own_car	true	1	7576	1.23%
5	flag_own_realty	false	0	86357	14.04%
6	flag_own_realty	true	0	196329	31.92%
7	flag_own_realty	false	1	7842	1.28%
8	flag_own_realty	true	1	16983	2.76%

5. Which income type has the highest number of frauds? What is the average salary for each income type?

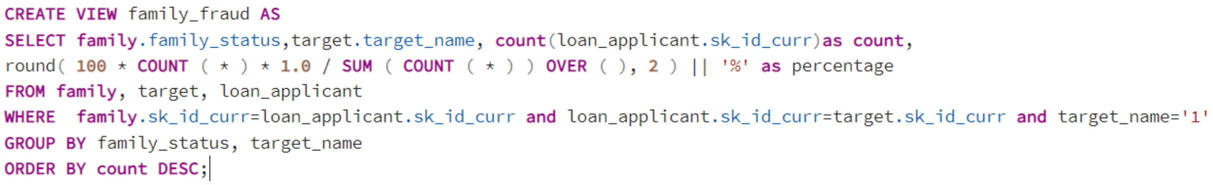


```
CREATE VIEW income_fraud AS
SELECT income.income_type, ROUND(AVG(income.annual_income_total),2)as average_income, target.target_name,
count(loan_applicant.sk_id_curr)as count,
round( 100 * COUNT ( * ) * 1.0 / SUM ( COUNT ( * ) ) OVER ( ), 2 ) || '%' as percentage
FROM income, target, loan_applicant
WHERE  income.income_id=loan_applicant.income_id and loan_applicant.sk_id_curr=target.sk_id_curr and target_name='1'
GROUP BY income_type, target_name
ORDER BY count DESC;
```

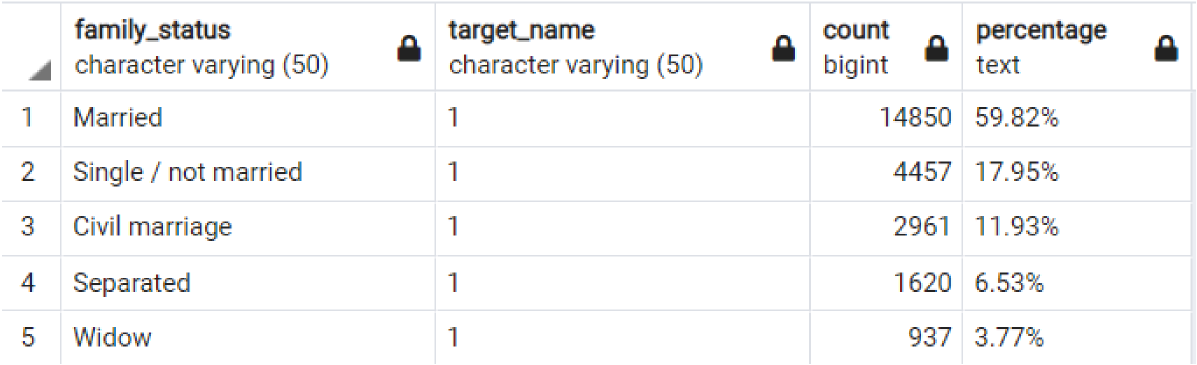


	income_type character varying (50)	average_income numeric	target_name character varying (50)	count bigint	percentage text
1	Working	163676.85	1	15224	61.33%
2	Commercial associate	188217.32	1	5360	21.59%
3	Pensioner	135556.94	1	2982	12.01%
4	State servant	164713.35	1	1249	5.03%
5	Unemployed	72000.00	1	8	0.03%
6	Maternity leave	58500.00	1	2	0.01%

6. What is the number of frauds for different family status (married/single)?

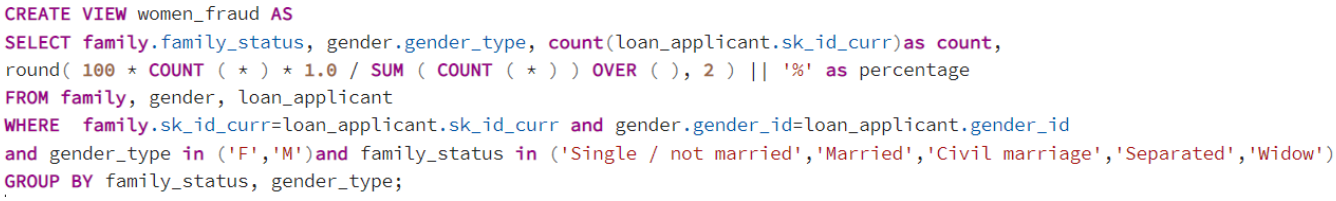


```
CREATE VIEW family_fraud AS
SELECT family.family_status,target.target_name, count(loan_applicant.sk_id_curr)as count,
round( 100 * COUNT ( * ) * 1.0 / SUM ( COUNT ( * ) ) OVER ( ), 2 ) || '%' as percentage
FROM family, target, loan_applicant
WHERE  family.sk_id_curr=loan_applicant.sk_id_curr and loan_applicant.sk_id_curr=target.sk_id_curr and target_name='1'
GROUP BY family_status, target_name
ORDER BY count DESC;
```

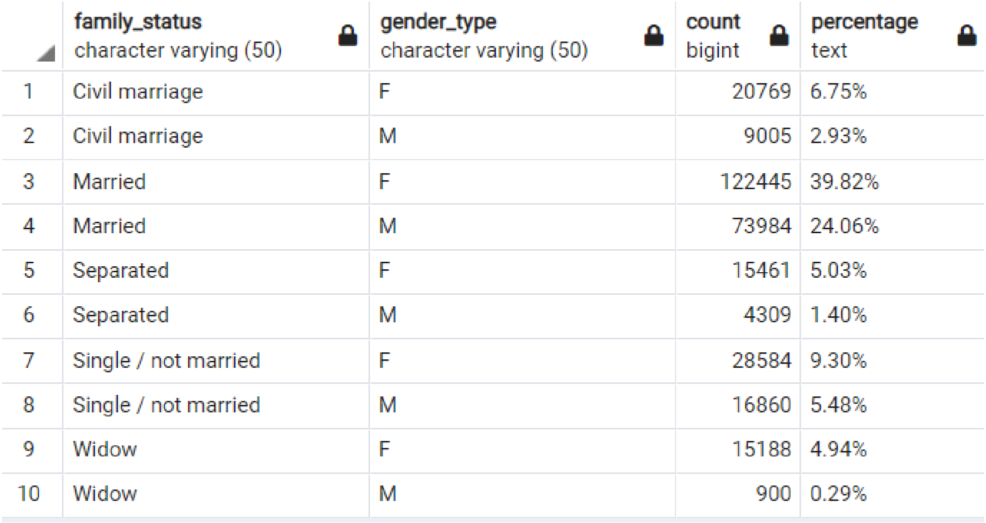


	family_status character varying (50)	target_name character varying (50)	count bigint	percentage text
1	Married	1	14850	59.82%
2	Single / not married	1	4457	17.95%
3	Civil marriage	1	2961	11.93%
4	Separated	1	1620	6.53%
5	Widow	1	937	3.77%

7. **What percentage of single and married women apply for credit cards, respectively?**

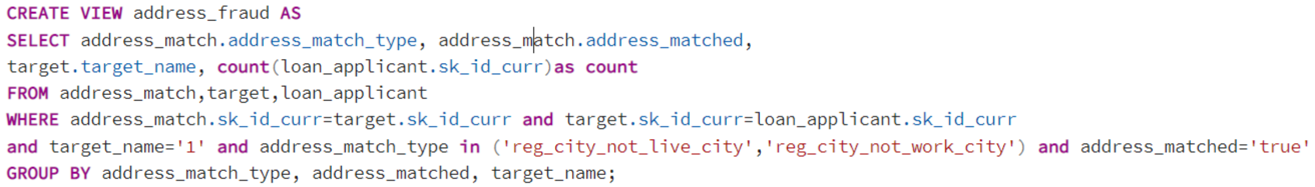


```
CREATE VIEW women_fraud AS
SELECT family.family_status, gender.gender_type, count(loan_applicant.sk_id_curr)as count,
round( 100 * COUNT ( * ) * 1.0 / SUM ( COUNT ( * ) ) OVER ( ), 2 ) || '%' as percentage
FROM family, gender, loan_applicant
WHERE  family.sk_id_curr=loan_applicant.sk_id_curr and gender.gender_id=loan_applicant.gender_id
and gender_type in ('F','M')and family_status in ('Single / not married','Married','Civil marriage','Separated','Widow')
GROUP BY family_status, gender_type;
```

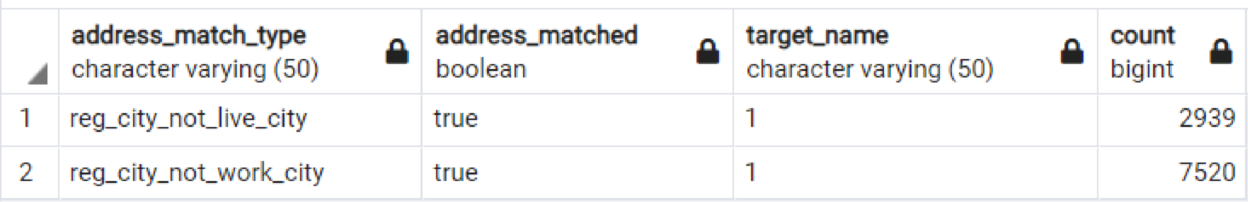


	family_status character varying (50)	gender_type character varying (50)	count bigint	percentage text
1	Civil marriage	F	20769	6.75%
2	Civil marriage	M	9005	2.93%
3	Married	F	122445	39.82%
4	Married	M	73984	24.06%
5	Separated	F	15461	5.03%
6	Separated	M	4309	1.40%
7	Single / not married	F	28584	9.30%
8	Single / not married	M	16860	5.48%
9	Widow	F	15188	4.94%
10	Widow	M	900	0.29%

8. **What number of frauds are committed by people whose registration city is not the same as live city or work city?**

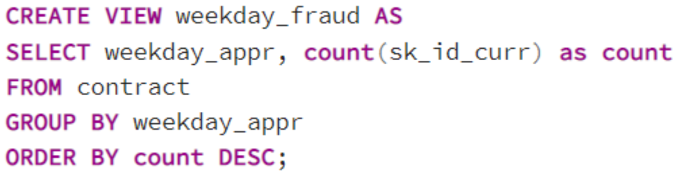


```
CREATE VIEW address_fraud AS
SELECT address_match.address_match_type, address_match.address_matched,
target.target_name, count(loan_applicant.sk_id_curr)as count
FROM address_match,target,loan_applicant
WHERE address_match.sk_id_curr=target.sk_id_curr and target.sk_id_curr=loan_applicant.sk_id_curr
and target_name='1' and address_match_type in ('reg_city_not_live_city','reg_city_not_work_city') and address_matched='true'
GROUP BY address_match_type, address_matched, target_name;
```

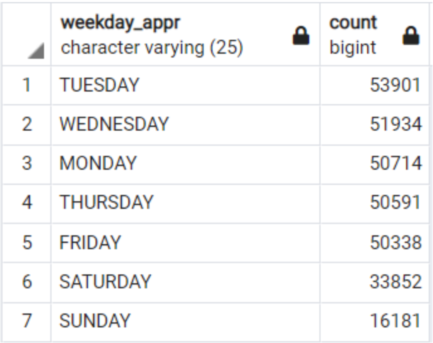


	address_match_type character varying (50)	address_matched boolean	target_name character varying (50)	count bigint
1	reg_city_not_live_city	true	1	2939
2	reg_city_not_work_city	true	1	7520

9. What day of the week do most people apply for credit cards?

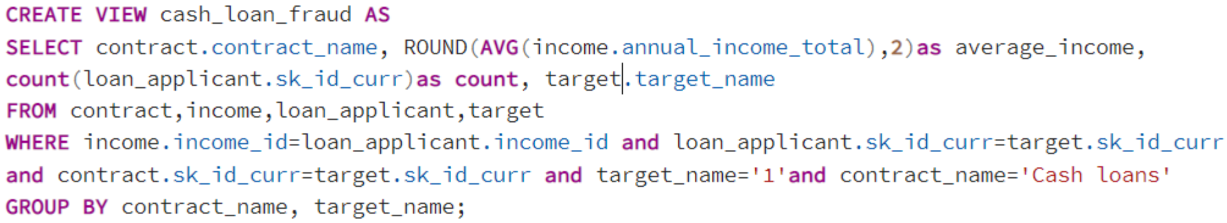


```
CREATE VIEW weekday_fraud AS
SELECT weekday_appr, count(sk_id_curr) as count
FROM contract
GROUP BY weekday_appr
ORDER BY count DESC;
```

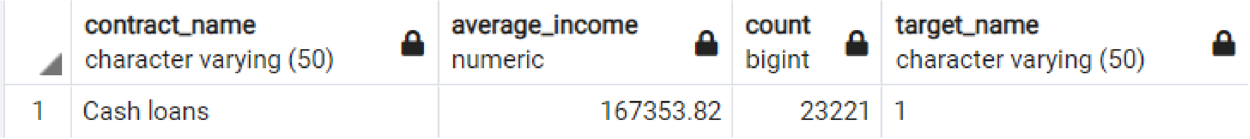


	weekday_appr character varying (25)	count bigint
1	TUESDAY	53901
2	WEDNESDAY	51934
3	MONDAY	50714
4	THURSDAY	50591
5	FRIDAY	50338
6	SATURDAY	33852
7	SUNDAY	16181

10. What is the average income of people who committed fraud on cash loans?

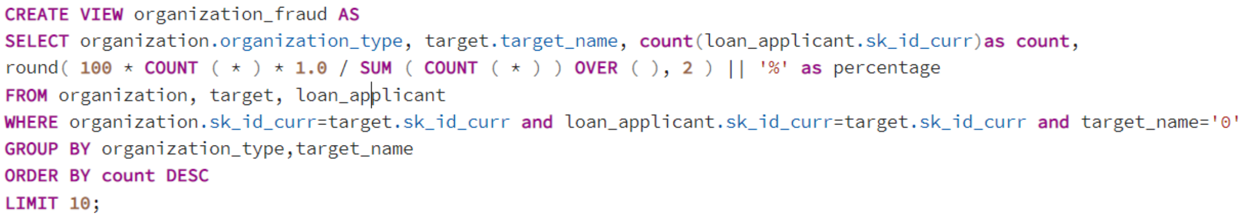


```
CREATE VIEW cash_loan_fraud AS
SELECT contract.contract_name, ROUND(AVG(income.annual_income_total),2)as average_income,
count(loan_applicant.sk_id_curr)as count, target.target_name
FROM contract,income,loan_applicant,target
WHERE income.income_id=loan_applicant.income_id and loan_applicant.sk_id_curr=target.sk_id_curr
and contract.sk_id_curr=target.sk_id_curr and target_name='1'and contract_name='Cash loans'
GROUP BY contract_name, target_name;
```

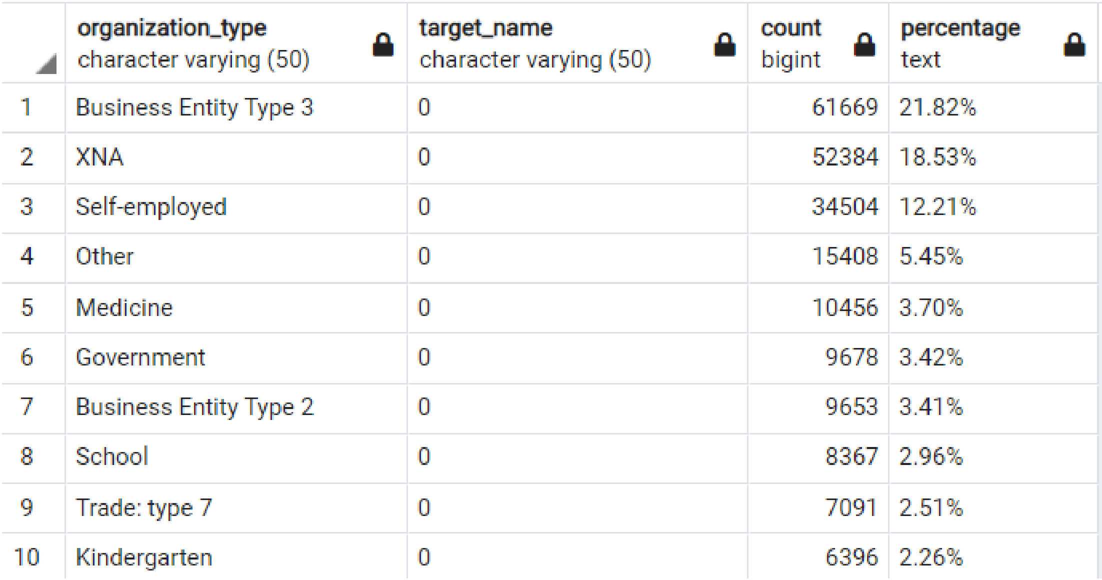


	contract_name character varying (50)	average_income numeric	count bigint	target_name character varying (50)
1	Cash loans	167353.82	23221	1

11. What organization is the most creditworthy user from（top 10）?

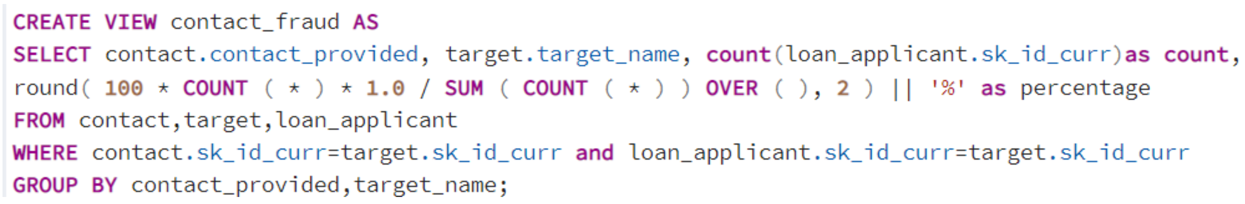


```
CREATE VIEW organization_fraud AS
SELECT organization.organization_type, target.target_name, count(loan_applicant.sk_id_curr)as count,
round( 100 * COUNT ( * ) * 1.0 / SUM ( COUNT ( * ) ) OVER ( ), 2 ) || '%' as percentage
FROM organization, target, loan_applicant
WHERE organization.sk_id_curr=target.sk_id_curr and loan_applicant.sk_id_curr=target.sk_id_curr and target_name='0'
GROUP BY organization_type,target_name
ORDER BY count DESC
LIMIT 10;
```

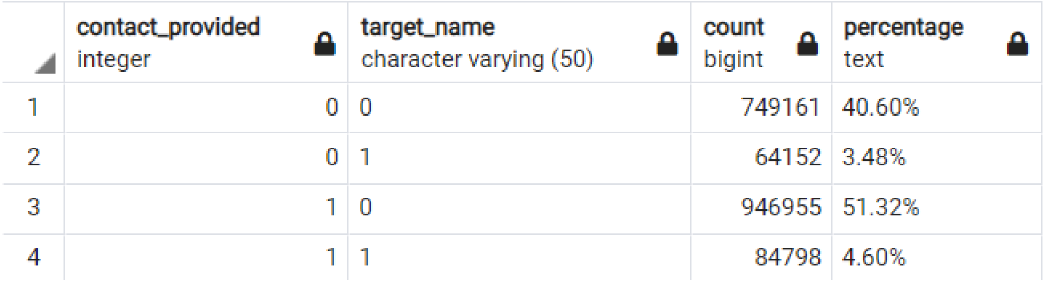


	organization_type character varying (50)	target_name character varying (50)	count bigint	percentage text
1	Business Entity Type 3	0	61669	21.82%
2	XNA	0	52384	18.53%
3	Self-employed	0	34504	12.21%
4	Other	0	15408	5.45%
5	Medicine	0	10456	3.70%
6	Government	0	9678	3.42%
7	Business Entity Type 2	0	9653	3.41%
8	School	0	8367	2.96%
9	Trade: type 7	0	7091	2.51%
10	Kindergarten	0	6396	2.26%

12. Is the customer who did not provide support documents more likely to have a late payment?



```
CREATE VIEW contact_fraud AS
SELECT contact.contact_provided, target.target_name, count(loan_applicant.sk_id_curr)as count,
round( 100 * COUNT ( * ) * 1.0 / SUM ( COUNT ( * ) ) OVER ( ), 2 ) || '%' as percentage
FROM contact,target,loan_applicant
WHERE contact.sk_id_curr=target.sk_id_curr and loan_applicant.sk_id_curr=target.sk_id_curr
GROUP BY contact_provided,target_name;
```



	contact_provided integer	target_name character varying (50)	count bigint	percentage text
1	0	0	749161	40.60%
2	0	1	64152	3.48%
3	1	0	946955	51.32%
4	1	1	84798	4.60%

By answering the above questions through SQL language, analysts can sufficiently understand the company's users and determine which customers the company should give credit priority to and which users need to be reasonably avoided. Using SQL queries, analysts can retrieve a large number of records from the database quickly and efficiently, while ensuring database consistency and integrity.

In addition, after the tables are obtained through the query, the analyst also needs to visualize the data to effectively report the results to the "C" level officers. Data visualization is an obvious way of communication. Using visualization tools such as Tableau, complex information can be reflected in simple graphs. For stakeholders, the

visualized data allows them to trace the causes of differences more quickly, thus helping them make operational decisions.
We conjecture that "C" level officers will be concerned about:

1. What is the age group of customers most prone to committing fraud？
2. How many users with fraud history are there in each income phase, and what are the percentages?
3. What is the number of loan applications received on working days? Which day has the most applications？

Therefore, the analyst needs to import the previously obtained tables into Metabase and draw a series of clear graphs to answer the questions posed by these officers. Finally, analysts need to use R or Python to observe the correlation between various variables and use machine learning methods to further analyze the driving factors behind loan frauds, that is, which variables are strong indicators of frauds.

To avoid data redundancy, firstly we dealt with the problems of repeating attributes, multi-value cells, and full functional dependency when we built ER-diagram to ensure that we follow the rules of 1nf and 2nf. Secondly, when we did data wrangling and ETL through python, we deleted the repeating data by deleting duplicate rows and combining the columns with similar meanings to one column. For example, columns like def_30_cnt_social_circle, def_60_cnt_social_circle, and def_90_cnt_social_circle have the same meaning with different requirements of quantity, so we combined these columns into one column, and created another column to record values.

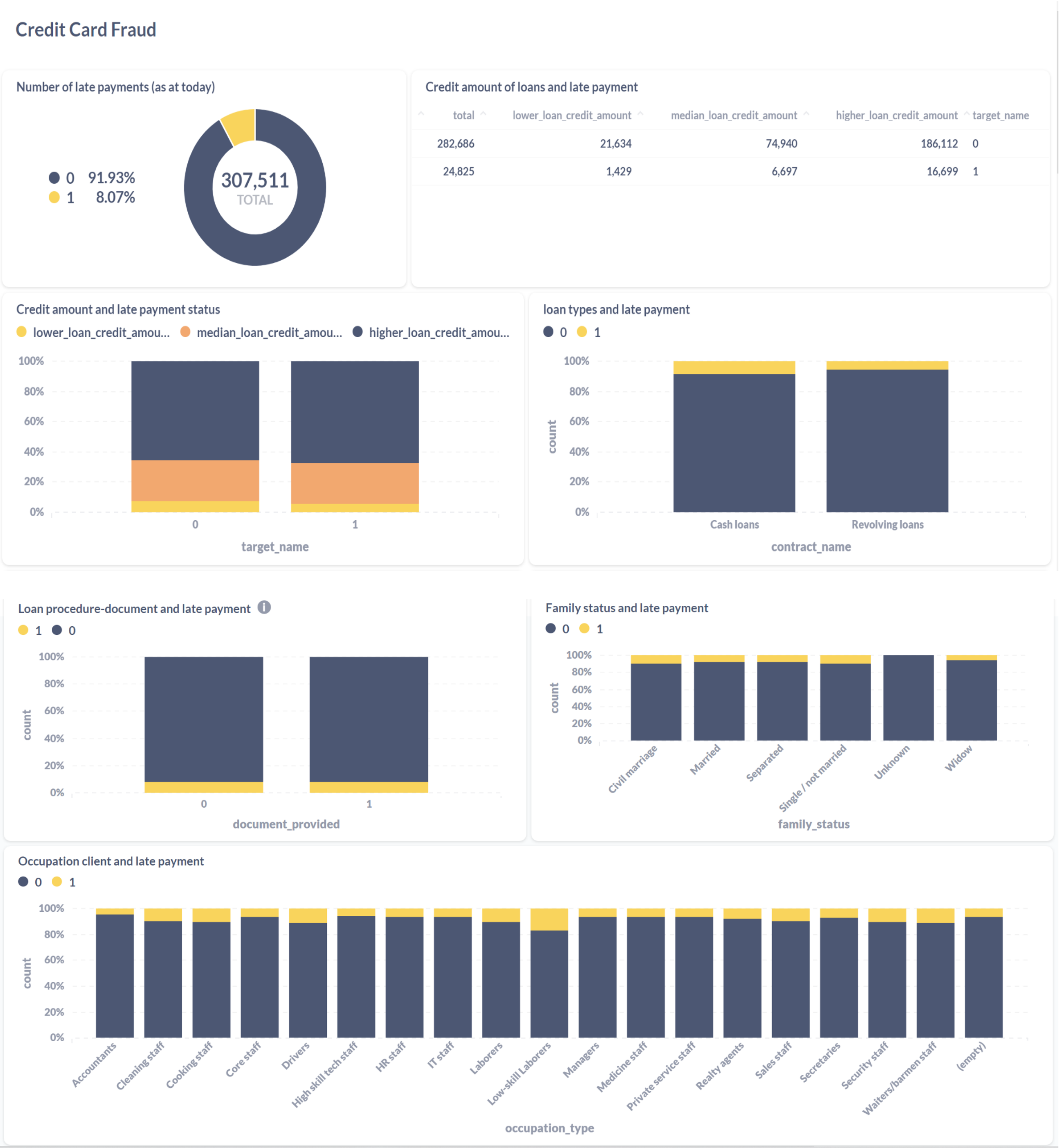
For the performance of SQL query and dashboard, we can get the result of each querying sentence lower than 3 minutes and each table can relate with each other easily, which is quite effective. But we need to optimize the performance in the dashboard because the data size in some tables is extremely large. Due to we combined columns in some tables, which made the total number of rows much more than the original dataset, and it spent a long time to build these tables.

Demo：
The entire dashboard shows the number of loan transactions and the number of late payments that occur. In addition, in order for the bank to better estimate the overdue risk, different factors and late payment situations are also shown, including the loan amount, loan type, loan procedures, borrower's personal information(family status, job, property, contact information,and income type).
In details, as can be clearly seen from the dashboard: (target name -1 ：late payment）

- By the count day,there were 307,511 transactions in total, among which late payments accounted for 8.09%.

- Borrowers with relatively large loans were more likely to be late on their payments.
- The probability of overdue payment for cash loan (8.3%) is higher than that for revolving loans (5.5%).
- There is no relationship between provision of relevant loan documents and late payment.
- Borrowers in civil marriage (9.9%) and Single/Not married (9.8%) were more likely to be late on payments than those in other status, and widow was the least likely to be late on loans (5.8%).
- Laborers(27.8%), especially low-skilled workers(17.2%), drivers(11.3%), waiters/bartenders(11.3%), security personnel(10.7%), cooks(10.4%), cleaners(9.6%), salespeople(9.6%), loan applicants in these categories are more likely to have a late payment.
- Borrowers who do not own cars or realty are more likely to fall behind on payments.
- Borrowers who provided contact information(8.2%) were more likely to be late on their payments than those who did not(7.9%) .
- Borrowers whose income type is maternity leave and unemployed are more likely to make late payments, while businessmen and students almost do not.

This dashboard can clearly provide visual data for bank risk management personnel to understand the characteristics of customers who pay late from various aspects, so as to carry out risk assessment in the future. In addition, the dashboard can be updated dynamically as long as the latest data is uploaded to the database, which is very convenient and flexible for all departments of the bank to check.



Credit Card Fraud

Number of late payments (as at today)

Credit amount of loans and late payment

total	lower_loan_credit_amount	median_loan_credit_amount	higher_loan_credit_amount	target_name
282,686	21,634	74,940	186,112	0
24,825	1,429	6,697	16,699	1

Credit amount and late payment status

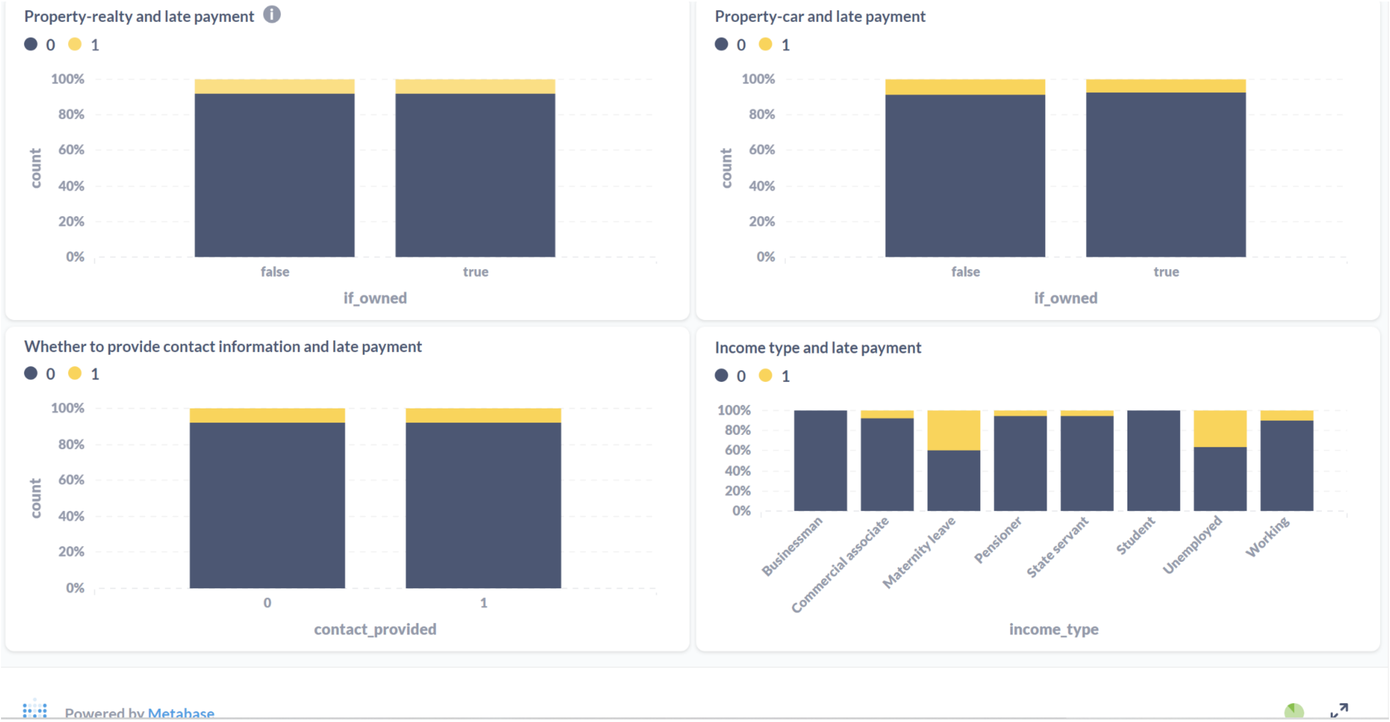
● lower_loan_credit_amou...
● median_loan_credit_amou...
● higher_loan_credit_amou...
100%
80%
60%
40%
20%
0%
0
1
target_name

loan types and late payment

Loan procedure-document and late payment

Family status and late payment

Occupation client and late payment



http://127.0.0.1:3000/dashboard/1-credit-card-fraud

Conclusion

Our project aims to reduce credit fraud risk by establishing a comprehensive customer database management system, which allows the financial institution to identify and understand small business lending more effectively. For the purpose of characterizing loan applicators precisely, we collected large size real financial loan data to build our relational database and developed this relational database to the third normalization form, with 23 different factors (tables). For the purpose of understanding small business lending better, it is important to reveal the driving factors behind each loan default, and we achieved this by implementing a detailed customer interaction plan. In this plan, we designed 12 analytical procedures that derive valuable insights from both the product side and the customer side to the C-level officer. Specifically, these procedures are SQL queries that generate quick quantitative business values from our relational database. Successfully carrying out our customer interaction plan will help the institution not only reduce the loan default risk of borrowers but also expand its user population.