Fraud Detection using SQL

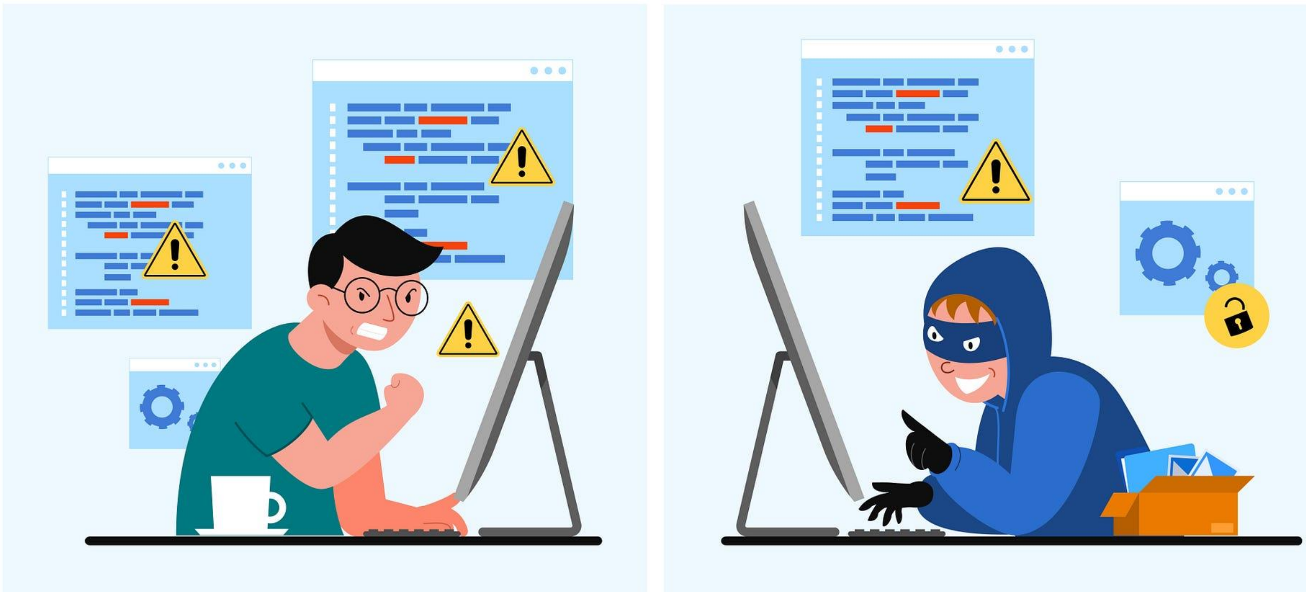
BY: Tarun S Gowda

In the modern digital era, where technology has become an integral part of daily life, the volume of online transactions and data exchanges has grown exponentially. With this rise in digital interactions comes an increased risk of fraudulent activities, which have evolved in complexity and scale. Fraud not only results in substantial financial losses for businesses and consumers but also undermines trust in digital platforms, banking systems, and e-commerce services. As a result, fraud detection has become a critical area of focus for organizations aiming to safeguard their assets and maintain customer confidence. This project centers around the concept of fraud detection using Structured Query Language (SQL), a domain-specific language designed for managing and analyzing structured data in relational databases. SQL is widely adopted across industries and remains a foundational tool in data analytics, especially when working with large volumes of transactional data.

The primary goal of this project is to explore how SQL can be effectively used to detect potentially fraudulent activities through rule-based analysis and pattern recognition. SQL enables users to write queries that identify anomalies such as unusually large transactions, multiple transactions from the same user in a short time frame, geographic inconsistencies, or repeated failed login attempts. These indicators, when captured and analyzed correctly, can serve as early warning signs of fraud. Unlike advanced machine learning models, SQL does not require extensive computational resources or training data, making it an accessible and practical option for organizations that need fast, explainable, and easily deployable fraud detection systems. Moreover, SQL integrates seamlessly with most existing relational database systems, allowing fraud detection mechanisms to be built directly into current infrastructures without the need for additional platforms or tools.

This project not only implements a variety of fraud detection queries using SQL but also emphasizes the importance of designing efficient, scalable, and adaptable queries that can evolve alongside emerging fraud patterns. While SQL may not independently detect sophisticated, previously unseen fraud schemes, it is extremely effective as a first line of defense—filtering and flagging suspicious activities for further investigation. In addition, SQL serves as a powerful data preprocessing and feature engineering tool, playing a critical role when used in combination with machine learning or real-time monitoring systems. By the end of this project, the implementation will demonstrate how SQL-based fraud detection can significantly enhance data oversight, improve operational security, and contribute to a larger, more robust anti-fraud strategy. The project underscores that while technology continues to evolve, fundamental tools like SQL remain invaluable in the ongoing battle against digital fraud.

Introduction



Fraud is everywhere these days—whether you are a small Fastfood shop or a large international business. While there are emerging technologies that employ machine learning and artificial intelligence to detect fraud, many instances of fraud detection still require strong data analytics to find abnormal charges.

Application of new SQL skills to analyze historical credit card transactions and consumption patterns in order to identify possible fraudulent transactions.

Accomplish three main tasks:

1. Data Modeling: Define a database model to store the credit card transactions data and create a new PostgreSQL database using your model.
2. Data Engineering: Create a database schema on PostgreSQL and populate your database from the CSV files provided.
3. Data Analysis: Analyze the data to identify possible fraudulent transactions
   .

Literature survey

Existing System

Increasingly, machine learning is being used in fraud prevention and detection due to its ability to analyze large quantities of data, identify patterns, and adapt to new information. Some common applications of machine learning in fraud prevention include:

Anomaly detection
Machine-learning algorithms can identify unusual patterns or deviations from normal behavior in transactional data. By "training" on historical data, the algorithms learn to recognize legitimate transactions and flag suspicious activities that may indicate fraud.

Risk scoring
Machine-learning models can assign risk scores to transactions or user accounts based on various factors, such as transaction amount, location, frequency, and past behavior. Higher risk scores indicate a higher likelihood of fraud, enabling organizations to prioritize their resources and focus on specific transactions or accounts that warrant further investigation.

Network analysis
Fraudulent actors often collaborate and form networks to carry out their activities. Machine-learning techniques, like graph analysis, can help uncover these networks by analyzing relationships between entities (such as users, accounts, or devices) and identifying unusual connections or clusters.

Text analysis
Machine-learning algorithms can analyze unstructured text data, such as emails, social media posts, or customer reviews, to identify patterns or keywords that may indicate fraud or scams.

Identity verification
Machine-learning models can analyze and verify user-provided information, such as images of identification documents or facial recognition data, to ensure that an individual is who they claim to be and prevent identity theft.

Adaptive learning
One of the key strengths of machine learning is its ability to learn and adapt to new information. As fraudulent actors change their tactics, machine-learning models can be retrained on new data, allowing them to stay up to date and better equipped to detect emerging fraud patterns

Using machine learning in fraud prevention can be a powerful way for organizations to enhance their detection capabilities, reduce the risk of false positives, and improve overall security and customer experience.

Proposed System

I, Tarun in this independent SQL project idea, I have worked with multiple CSV files to create a database schema and database, then query your new database to identify potentially fraudulent transactions. Fraud is everywhere these days, and while there are emerging technologies that employ machine learning and artificial intelligence to detect fraud, many instances still require strong data analytics to find abnormal charges. You'll apply SQL skills to analyze historical credit card transactions and consumption patterns in order to identify possible fraudulent transactions.

System Requirement Specification

Hardware Requirements

Fraud Detection SQL

The hardware requirements for working with **SQL** and **Python** depend heavily on the **specific use case**—whether it's light development, data analysis, machine learning, or working with large-scale databases. Here's a breakdown by general usage level:

Python Hardware Requirements

Basic Scripting / Development

- **CPU**: Dual-core (Intel i3 or AMD equivalent)
- **RAM**: 4–8 GB
- **Storage**: SSD preferred, 10–20 GB free
- **GPU**: Not required

Data Analysis / Web Development

- **CPU**: Quad-core (Intel i5 or better)
- **RAM**: 8–16 GB
- **Storage**: SSD, 50+ GB free
- **GPU**: Optional (helpful for visualizations or specific tasks)

Machine Learning / AI

- **CPU**: 6-core or better (Intel i7/i9, AMD Ryzen 7/9)
- **RAM**: 16–64+ GB
- **Storage**: SSD, 100+ GB (datasets take space)
- **GPU**: NVIDIA GPU with CUDA support (e.g., RTX 3060 or better)

SQL Hardware Requirements

Local Development / Testing (e.g., SQLite, MySQL, PostgreSQL

- **CPU**: Dual-core or quad-core
- **RAM**: 4–8 GB
- **Storage**: SSD, 20+ GB free
- **GPU**: Not required

Production Database Server (medium to large scale)

- **CPU**: Multi-core (8+ cores recommended)
- **RAM**: 32–128+ GB (RAM is crucial for caching and performance)
- **Storage**: Fast SSDs or NVMe (RAID for redundancy), sized based on DB size
- **GPU**: Not required unless using GPU-accelerated databases (rare)

Tarun S Gowda

Fraud Detection SQL

Combo Setup (Python + SQL + Dev Workstation)

If you're doing data analysis, data science, or full-stack development:

- **CPU**: Quad-core or better
- **RAM**: 16 GB (32 GB for comfort with large datasets)
- **Storage**: 512 GB SSD
- **GPU**: Optional, but helpful for ML or data visualization

Software Requirements

- SQL
- PostgreSQL
- Entity Relationship Diagrams (ERDs)
- Python
- Pandas
- Plotly Express
- SQLAlchemy

Prerequisites

To successfully complete this project, you should be comfortable with the following:

- SQL fundamentals like querying, grouping data, joins, subqueries, and filtering data
- Creating database schemas and tables
- Importing data from CSV files into a database
- Basic Python programming for data analysis

- Using Python libraries like Pandas for data manipulation and analysis

Step-by-Step Instructions

1. Define a database model to store the credit card transactions data and create a new PostgreSQL database using your model. Create an entity relationship diagram (ERD) by inspecting the provided CSV files.
2. Create a database schema on PostgreSQL and populate your database from the CSV files provided. Import the data from the corresponding CSV files after creating the schema.
3. Analyze the data to identify possible fraudulent transactions:
    a. Find the top 100 highest transactions during early morning hours (7-9 AM)
    b. Count transactions less than $2.00 per cardholder to check for hacked cards
    c. Identify the top 5 merchants prone to being hacked with small transactions
    d. Create views for reusable queries
    e. Create a report for fraudulent transactions of top customers using Python and data visualization libraries
    f. Analyze outlier/anomalous transactions using techniques like standard deviation and interquartile range

Expected Outcomes

Upon completing this SQL project, you'll have gained valuable skills and experience, including:

- Designing a database schema to store financial transactions data
- Querying a database using complex SQL statements to uncover anomalous transactions
- Applying data analysis techniques to identify outliers and potentially fraudulent activity
- Communicating findings through visualizations and a report

CSV Files

id, name

1, Robert Johnson

2, Shane Shaffer

3, Elizabeth Sawyer

4, Danielle Green

5, Sara Cooper

6, Beth Hernandez

7, Sean Taylor

8, Michael Floyd

9, Laurie Gibbs

10, Matthew Gutierrez

11, Brandon Pineda

12, Megan Price

13, John Martin

14, Gary Jacobs

15, Kyle Tucker

16, Crystal Clark

17, Michael Carroll

18, Malik Carlson

19, Peter Mckay

20, Kevin Spencer

21, Dana Washington

22, Austin Johnson

23, Mark Lewis

24, Stephanie Dalton

25, Nancy Contreras

Credit_card.csv

card, id_card_holder

3.51711E+15, 1

4.76105E+18, 1

4.86676E+18, 2

6.75911E+11, 2

3.00783E+13,	3
4.26369E+15,	4
5.84227E+11,	4
4.27647E+12,	5
4.26849E+15,	5
3.58135E+15,	6
4.15984E+18,	6
3.51695E+15,	7
4.53999E+15,	7
4.83448E+15,	8
3.00633E+13,	8
3.0182E+13,	9
4.96292E+18,	10
4.16531E+18,	10
2.13194E+14,	10
1.80099E+14,	11
4.64401E+18,	11
4.02791E+15,	11
5.0188E+11,	12
5.29719E+15,	12
3.76028E+14,	12
4.71177E+15,	13
5.13584E+15,	13
3.56195E+15,	13

5.17595E+15,	14
4.72378E+18,	15
6.50024E+15,	15
5.03843E+11,	16
5.5706E+15,	16
5.50071E+15,	16
6.01199E+15,	17
4.498E+12,	18
3.4412E+14,	18
4.7432E+18,	19
5.36178E+15,	19
3.56107E+15,	19
3.53565E+15,	20
4.50641E+15,	20
4.58696E+18,	20
4.2791E+18,	21
5.01809E+11,	22
4.74104E+12,	23
4.18816E+15,	23
4.15072E+15,	23
4.6819E+12,	24
3.0143E+13,	24
3.5822E+15,	24
4.31965E+12,	25

3.72415E+14, 25

Merchant.csv

id	name	name	id_merchant_category
2	1	Murphy, Heath and Fields	1
3	2	Riggs-Adams	1
4	3	Sanders, Parks and Mcfarland	2
5	4	Mccarty-Thomas	3
6	5	Miller-Blevins	4
7	6	Wilson and Sons	1
8	7	Gomez-Kelly	4
9	8	Russell-Thomas	1
10	9	Curry, Scott and Richardson	3
11	10	Herrera Group	1
12	11	Stanton Group	4
13	12	Bell, Gonzalez and Lowe	4
14	13	Giles and Sons	4
15	14	Osborne-Page	2
16	15	Long, Harrell and Johnson	5
17	16	Bryant, Thomas and Collins	4
18	17	Bauer-Cole	3
19	18	Romero-Jordan	5
20	19	Santos-Fitzgerald	4
21	20	Kim-Lopez	2
22	21	Robertson-Smith	4
23	22	Dalton, Cameron and Jones	3
24	23	Wilson, Roberts and Davenport	5
25	24	Rodgers, Johnston and Macias	5
26	25	Vaughn, Wilson and Hall	1
27	26	Smith-Stephens	2
28	27	Horn Ltd	2
29	28	Hess-Fischer	5
30	29	Browning-Cantu	4
31	30	Atkinson Ltd	3

32	31	Fisher, Salazar and Thomas	5
33	32	Norton, Burton and Smith	5
34	33	Vasquez-Parker	3
35	34	Combs-Jones	5
36	35	Jarvis-Turner	4
37	36	Hamilton-Mcfarland	1
38	37	Nguyen, Bautista and Williams	4
39	38	Brown LLC	3
40	39	Young-Navarro	5
41	40	Cox, Montgomery and Morgan	3
42	41	Ford, Williams and Dunn	4
43	42	Kennedy-Chen	3
44	43	Wallace and Sons	2
45	44	Little-Floyd	4
46	45	Velazquez Ltd	4
47	46	Miller, Chavez and Cobb	5
48	47	Martin Inc	1
49	48	Baker Inc	5
50	49	Davis, Lowe and Baxter	5
51	50	Johnson-Watts	1
52	51	Fisher-Bolton	1
53	52	Jensen-Stanley	5
54	53	Wallace PLC	4
55	54	Berry-Lopez	1
56	55	Johnson, Rivas and Anderson	1
57	56	Smith PLC	3
58	57	Thornton-Williams	4
59	58	Young, Hull and Williams	4
60	59	Williams Group	3
61	60	Smith-Richards	4
62	61	Richardson, Smith and Jordan	5
63	62	Cooper, Carpenter and Jackson	5
64	63	Reed Group	5
65	64	Cline, Myers and Strong	1
66	65	Allen, Ramos and Carroll	4
67	66	Robles Inc	3
68	67	Maxwell, Tapia and Villanueva	2
69	68	Ramirez-Carr	2

70	69	Dominguez PLC	5
71	70	White-Hall	3
72	71	Greene LLC	1
73	72	Lopez-Kelly	1
74	73	Colon Ltd	3
75	74	Skinner-Williams	4
76	75	Martinez Group	1
77	76	Lowe PLC	1
78	77	Brown, Ballard and Glass	1
79	78	Ruiz-Anderson	4
80	79	Lee LLC	1
81	80	Kelly, Dyer and Schmitt	5
82	81	Fowler and Sons	5
83	82	Day-Murray	5
84	83	Solis Group	5
85	84	Marshall-Rojas	2
86	85	Patton-Rivera	3
87	86	Walker, Campbell and Sullivan	5
88	87	Griffin-Woodard	3
89	88	Armstrong PLC	5
90	89	Kelley-Roberts	5
91	90	Brown-Cunningham	4
92	91	Turner Ltd	4
93	92	Garcia-White	4
94	93	Rodriguez-Parker	5
95	94	Yoder-Zavala	5
96	95	Baxter-Smith	1
97	96	Johnson-Fuller	4
98	97	Ruiz-Chavez	3
99	98	Rivera PLC	4
100	99	Bond, Lewis and Rangel	1
101	100	Townsend-Anderson	1
102	101	Whitehead-Sexton	4
103	102	Walters-Ward	1
104	103	Jones, Clark and Hoover	2
105	104	Mcdaniel, Hines and Mcfarland	2
106	105	Garcia and Sons	4
107	106	Carter-Blackwell	4
108	107	Rowe-Abbott	4
109	108	Best Inc	1
110	109	Collins LLC	2

111	110	Rodriguez, Dunlap and Nunez	2
112	111	Padilla-Clements	2
113	112	Greer Inc	3
114	113	Edwards-Aguirre	2
115	114	Greene-Wood	3
116	115	Williams Inc	4
117	116	Ferguson Ltd	2
118	117	Mitchell Group	2
119	118	Maldonado Group	2
120	119	Henderson and Sons	1
121	120	Vega, Jones and Castro	5
122	121	Fleming, Smith and Collins	3
123	122	Perry and Sons	3
124	123	Boone, Davis and Townsend	4
125	124	Mccarty PLC	1
126	125	Russell and Sons	4
127	126	Bartlett and Sons	4
128	127	Williams, Wright and Wagner	2
129	128	Pitts, Salinas and Garcia	2
130	129	Sweeney-Paul	2
131	130	Brown, Estrada and Powers	2
132	131	Harrison, Newton and Hansen	1
133	132	Pugh-Williams	3
134	133	Scott, Hess and Finley	3
135	134	Jenkins, Peterson and Beck	1
136	135	Jacobs, Torres and Walker	3
137	136	Martinez-Robinson	3
138	137	Garcia PLC	5
139	138	Mccullough-Murphy	5
140	139	Kidd-Lopez	5
141	140	Wheeler-Moreno	5
142	141	Wood-Ramirez	3
143	142	Thomas-Garcia	5
144	143	Guzman, Garcia and Church	3
145	144	Walker, Deleon and Wolf	1
146	145	Hood-Phillips	3
147	146	Pitts, Smith and Gonzalez	4
148	147	Marshall-Lopez	5

149	148	Huerta, Keith and Walters	5
150	149	Clark and Sons	5
151	150	Johnson and Sons	2

merchant_category.csv

1	id	name
2	1	restaurant
3	2	coffee shop
4	3	bar
5	4	pub
6	5	food truck

transaction.csv

id	date	amount	card	id_merchant	
37	1995	2018-01-05 01:10:27	5.09	501809222273	46
38	1346	2018-01-05 03:11:18	12.2	4268491956169254	69
39	2176	2018-01-05 05:47:28	4.64	5175947111814778	103
40	1552	2018-01-05 06:26:45	10.74	372414832802279	86
41	487	2018-01-05 06:27:06	4.6	4159836738768855913	48
42	2077	2018-01-05 07:19:27	1.36	344119623920892	30
43	314	2018-01-05 10:42:55	5.31	3561072557118696	8
44	77	2018-01-05 16:58:08	4.57	5135837688671496	91
45	1570	2018-01-06 01:43:44	10.04	5500708021555307	34
46	2439	2018-01-06 02:16:41	1.33	4866761290278198714	127
47	2258	2018-01-06 04:43:33	5.31	5297187379298983	42
48	1867	2018-01-06 05:13:20	10.82	4866761290278198714	70
49	846	2018-01-06 06:46:24	2.7	213193946980303	146

50	2890	2018-01-06 07:00:32	18.35	3561954487988605	76
51	3291	2018-01-06 08:42:50	18.72	3535651398328201	76
52	2557	2018-01-06 08:49:34	16.28	4268491956169254	101
53	325	2018-01-06 23:33:29	7.6	376027549341849	141
54	2663	2018-01-07 00:30:25	17.84	4159836738768855913	112
55	3457	2018-01-07 01:10:54	175.0	344119623920892	12
56	447	2018-01-07 07:33:17	7.07	4834483169177062	14
57	1249	2018-01-07 10:31:33	10.9	5297187379298983	135
58	2108	2018-01-07 14:57:23	2.93	4319653513507	137
59	3028	2018-01-07 15:10:27	17.29	4866761290278198714	126
60	149	2018-01-07 16:50:22	5.83	4159836738768855913	81
61	2304	2018-01-07 20:35:00	2.5	6011987562414062	104
62	1384	2018-01-08 00:26:21	10.04	3581345943543942	64
63	1371	2018-01-08 01:12:52	10.4	5500708021555307	121
64	1291	2018-01-08 02:34:32	1029.0	3581345943543942	145
65	1065	2018-01-08 04:15:30	11.49	4741042733274	130
66	1340	2018-01-08 04:29:20	10.64	501809222273	116
67	1544	2018-01-08 06:27:34	10.48	4027907156459098	68
68	522	2018-01-08 10:48:26	11.06	4743204091443101526	113
69	812	2018-01-08 11:15:36	333.0	344119623920892	95
70	1640	2018-01-08 11:21:20	11.98	5500708021555307	133
71	2550	2018-01-08 11:35:55	15.84	4150721559116778	15
72	2095	2018-01-08 14:06:25	3.52	180098539019105	95

73	3130	2018-01-08 16:38:22	21.27	5297187379298983	106
74	3085	2018-01-08 19:31:23	17.43	584226564303	19
75	1816	2018-01-08 20:10:59	11.55	344119623920892	102
76	3480	2018-01-08 21:44:34	1.68	4681896441519	78
77	1188	2018-01-08 21:44:58	10.64	5175947111814778	84
78	379	2018-01-09 11:14:56	7.07	501809222273	136
79	687	2018-01-09 21:06:16	2.7	4506405265172173	67
80	2558	2018-01-09 21:43:47	16.37	4681896441519	64
81	754	2018-01-10 00:25:40	1.39	372414832802279	50
82	608	2018-01-10 03:17:20	0.87	503842928916	129
83	2818	2018-01-10 03:24:29	16.64	3561954487988605	65
84	865	2018-01-10 03:55:54	2.3	4150721559116778	19
85	998	2018-01-10 10:07:20	10.91	675911140852	78
86	1436	2018-01-10 13:41:23	11.5	3517111172421930	49
87	1938	2018-01-10 14:50:30	5.64	5361779664174555	44
88	602	2018-01-10 15:56:41	1.22	503842928916	27
89	3289	2018-01-10 19:53:09	18.41	4159836738768855913	23
90	1361	2018-01-10 23:03:42	10.7	3535651398328201	70
91	496	2018-01-11 00:39:12	3.23	503842928916	68
92	156	2018-01-11 09:27:30	5.55	6011987562414062	107
93	315	2018-01-11 09:38:33	7.94	5135837688671496	146
94	3498	2018-01-11 09:58:11	1.62	5500708021555307	149
95	3331	2018-01-11 10:51:08	1.24	30063281385429	7

96	2128	2018-01-11 12:46:01	3.97	4834483169177062	44
97	2188	2018-01-11 13:20:31	229.0	5570600642865857	115
98	1716	2018-01-11 16:18:55	10.27	5361779664174555	46
99	2146	2018-01-11 19:36:21	1.72	4761049645711555811	99
100	2779	2018-01-11 22:55:51	19.5	4268491956169254	8
101	1048	2018-01-12 02:18:05	11.52	3582198969197591	2
102	2170	2018-01-12 02:48:29	1.84	30142966699187	2
103	2025	2018-01-12 04:33:17	5.32	4188164051171486	10
104	451	2018-01-12 06:34:53	6.19	30181963913340	40
105	444	2018-01-12 13:30:20	5.92	4506405265172173	38
106	416	2018-01-12 13:59:09	4.45	501809222273	85
107	291	2018-01-12 16:29:03	4.69	4268491956169254	144
108	3373	2018-01-12 18:04:04	1.88	4681896441519	44
109	270	2018-01-12 22:14:25	6.4	4276466390111	112
110	1937	2018-01-12 23:38:53	1.1	4506405265172173	64
111	2835	2018-01-13 02:02:40	17.12	376027549341849	2
112	3175	2018-01-13 03:06:36	16.27	4268491956169254	24
113	450	2018-01-13 03:20:50	4.89	213193946980303	58
114	2222	2018-01-13 04:49:38	3.63	4506405265172173	53
115	2755	2018-01-13 06:07:45	19.24	4962915017023706562	32
116	582	2018-01-13 14:36:43	1.35	4150721559116778	102
117	1546	2018-01-13 15:22:46	10.15	4711773125020499	108
118	718	2018-01-13 18:34:36	2.22	4263694062533017	9

119	3445	2018-01-13 22:18:00	2.11	30142966699187	3
120	2386	2018-01-13 22:26:57	3.79	5500708021555307	106
121	2843	2018-01-14 03:20:21	14.69	584226564303	133
122	3023	2018-01-14 05:02:22	17.84	372414832802279	52
123	2879	2018-01-14 06:19:11	21.11	3535651398328201	74
124	830	2018-01-14 07:06:57	3.15	584226564303	96
125	2521	2018-01-14 07:33:54	15.21	4741042733274	21
126	1482	2018-01-14 07:50:44	10.23	376027549341849	38
127	1916	2018-01-14 10:04:25	12.33	4165305432349489280	55
128	1560	2018-01-14 13:30:29	10.94	3517111172421930	19
129	3159	2018-01-14 15:03:26	16.67	213193946980303	30
130	604	2018-01-14 15:31:50	1.42	376027549341849	97
131	970	2018-01-14 17:29:28	10.31	4506405265172173	40
132	3359	2018-01-15 03:37:27	1.47	4834483169177062	139
133	2787	2018-01-15 06:23:30	15.87	4539990688484983	70
134	2622	2018-01-15 10:27:56	15.51	4761049645711555811	8
135	1382	2018-01-15 11:52:41	11.67	4159836738768855913	139
136	3105	2018-01-15 11:55:38	17.16	3561954487988605	101
137	2581	2018-01-15 19:37:59	18.61	3561954487988605	74
138	591	2018-01-15 20:53:52	2.95	4681896441519	98
139	2714	2018-01-16 00:57:00	15.33	4506405265172173	149
140	3333	2018-01-16 02:26:16	1.65	372414832802279	31
141	3068	2018-01-16 03:22:49	13.29	5297187379298983	12

142	3065	2018-01-16 04:12:01	16.89	4711773125020499	109
143	2655	2018-01-16 06:29:35	17.64	675911140852	136
144	2866	2018-01-16 08:02:04	17.36	30078299053512	116
145	680	2018-01-16 13:48:12	1.84	4268491956169254	144
146	1755	2018-01-16 13:51:42	13.52	4159836738768855913	54
147	753	2018-01-16 18:45:54	1.48	4962915017023706562	86
148	665	2018-01-16 19:19:48	2.55	344119623920892	99
149	3278	2018-01-17 00:03:25	18.05	4279104135293225293	133
150	651	2018-01-17 00:18:59	3.6	4159836738768855913	44
151	246	2018-01-17 00:22:04	5.41	4743204091443101526	105
152	2971	2018-01-17 02:02:41	15.18	213193946980303	75
153	2875	2018-01-17 04:56:56	16.46	376027549341849	42
154	758	2018-01-17 06:36:33	1.02	180098539019105	147
155	1551	2018-01-17 06:59:43	10.0	4586962917519654607	149
156	1889	2018-01-17 07:31:03	12.67	4498002758300	89
157	1593	2018-01-17 07:56:09	11.91	4681896441519	2
158	30	2018-01-17 09:09:33	3.82	4711773125020499	111
159	733	2018-01-17 09:25:09	2.85	3561954487988605	18
160	878	2018-01-17 11:09:35	12.27	376027549341849	146
161	3340	2018-01-17 15:02:38	1.72	4498002758300	115
162	105	2018-01-17 16:57:27	1.25	4268491956169254	65
163	840	2018-01-17 19:16:40	2.11	4268491956169254	132
164	2314	2018-01-17 20:06:45	2.69	3561072557118696	29

165	3466	2018-01-18 03:12:14	3.44	344119623920892	5
166	2633	2018-01-18 08:18:50	17.54	4165305432349489280	104
167	2838	2018-01-18 08:58:13	17.53	180098539019105	137
168	3406	2018-01-18 11:17:37	1.24	30181963913340	116
169	2662	2018-01-18 12:41:06	15.86	4319653513507	43
170	2119	2018-01-18 14:13:09	2.61	376027549341849	26
171	3095	2018-01-18 14:39:01	19.0	501879657465	15
172	1303	2018-01-18 18:49:21	10.19	4159836738768855913	116
173	2882	2018-01-19 01:11:13	16.38	5500708021555307	82
174	784	2018-01-19 01:34:25	2.59	4723783028106084756	47
175	1410	2018-01-19 03:01:33	10.39	4498002758300	132
176	2956	2018-01-19 05:03:56	17.53	4962915017023706562	82
177	3381	2018-01-19 07:08:43	2.51	4681896441519	8
178	2253	2018-01-19 10:15:55	3.93	4761049645711555811	13
179	2015	2018-01-19 11:41:42	3.27	213193946980303	124
180	1183	2018-01-19 12:28:12	10.4	180098539019105	60
181	917	2018-01-19 13:15:49	10.16	3535651398328201	81
182	679	2018-01-19 14:23:21	2.47	4150721559116778	131
183	458	2018-01-19 17:44:59	5.96	6500236164848279	138
184	1245	2018-01-19 20:12:31	11.58	4866761290278198714	132
185	2238	2018-01-20 01:13:21	3.7	5361779664174555	107
186	2411	2018-01-20 02:23:13	2.17	4263694062533017	109
187	1954	2018-01-20 03:18:02	3.38	30063281385429	93

188	2791	2018-01-20 03:19:33	17.64	30078299053512	37
189	302	2018-01-20 04:32:42	4.51	501809222273	83
190	1266	2018-01-20 07:47:30	10.58	30142966699187	34
191	1456	2018-01-20 08:35:34	11.96	501879657465	121
192	256	2018-01-20 09:56:37	9.81	4159836738768855913	69
193	2445	2018-01-20 10:50:36	2.17	4711773125020499	113
194	729	2018-01-20 13:44:37	1.83	501879657465	119
195	1615	2018-01-20 16:31:01	10.51	584226564303	10
196	1320	2018-01-20 17:30:16	13.84	4506405265172173	109
197	193	2018-01-20 19:46:39	8.03	4263694062533017	8
198	1037	2018-01-20 20:11:29	10.12	376027549341849	37
199	277	2018-01-20 21:29:46	4.87	180098539019105	113
200	2537	2018-01-20 22:58:27	16.44	6500236164848279	18
201	1417	2018-01-20 23:29:57	10.58	4723783028106084756	110
202	870	2018-01-21 00:03:41	2.32	4711773125020499	130
203	3006	2018-01-21 01:30:19	20.29	5135837688671496	136
204	132	2018-01-21 02:01:49	8.61	5175947111814778	37
205	2844	2018-01-21 05:38:08	16.29	3516952396080247	106
206	494	2018-01-21 06:47:24	5.34	4498002758300	85
207	1892	2018-01-21 13:18:10	10.38	4506405265172173	15
208	227	2018-01-21 15:37:39	5.51	213193946980303	21
209	322	2018-01-21 16:05:43	2.88	5175947111814778	75
210	1492	2018-01-21 17:27:37	10.28	501809222273	148

211	1338	2018-01-21 20:28:17	10.24	4723783028106084756	63
212	565	2018-01-21 23:04:02	2.22	372414832802279	149
213	1704	2018-01-22 02:08:47	10.38	4681896441519	98
214	1442	2018-01-22 08:07:03	1131.0	5570600642865857	144
215	2044	2018-01-22 11:52:29	1.67	180098539019105	30
216	1057	2018-01-22 14:56:43	10.75	5570600642865857	117
217	1935	2018-01-22 16:26:22	2.35	4723783028106084756	96
218	1364	2018-01-22 22:48:50	11.63	6011987562414062	150
219	1509	2018-01-23 01:51:00	11.3	4506405265172173	134
220	3499	2018-01-23 06:23:05	2.67	3516952396080247	51
221	2667	2018-01-23 06:29:37	1678.0	501879657465	92
222	1379	2018-01-23 08:07:03	10.47	4866761290278198714	7
223	538	2018-01-23 08:18:10	1.48	5500708021555307	35
224	2070	2018-01-23 10:30:44	4.34	4027907156459098	110
225	110	2018-01-23 12:06:35	6.38	5135837688671496	94
226	2319	2018-01-23 16:13:22	5.58	584226564303	33
227	392	2018-01-23 18:14:18	7.15	30063281385429	88
228	318	2018-01-24 01:31:16	3.54	3535651398328201	81
229	2212	2018-01-24 03:43:35	2.78	3581345943543942	68
230	263	2018-01-24 07:14:27	5.56	5500708021555307	36
231	1463	2018-01-24 07:55:45	10.96	6500236164848279	107
232	168	2018-01-24 08:28:08	5.0	4268491956169254	141
233	3196	2018-01-24 09:31:37	15.22	376027549341849	83

	234	942	2018-01-24 10:54:25	10.69	30063281385429	2
	235	3093	2018-01-24 12:41:48	17.21	4268491956169254	133
	236	1231	2018-01-24 13:06:03	13.39	30181963913340	91
	237	2913	2018-01-24 13:17:19	1691.0	4761049645711555811	14
	238	3144	2018-01-24 14:45:28	18.15	4681896441519	27
	239	2973	2018-01-24 16:42:45	18.59	3561954487988605	90
	240	2265	2018-01-24 23:15:56	5.78	4539990688484983	11
	241	148	2018-01-25 00:52:59	8.7	5175947111814778	132
	242	273	2018-01-25 07:41:11	5.74	4711773125020499	20
	243	1502	2018-01-25 08:58:28	11.35	376027549341849	141
	244	178	2018-01-25 11:54:44	6.4	5500708021555307	72
	245	1841	2018-01-25 14:38:43	10.17	4165305432349489280	24
	246	1252	2018-01-25 16:52:06	10.1	30181963913340	80
	247	2880	2018-01-26 05:10:21	16.54	584226564303	13
	248	1758	2018-01-26 06:11:05	10.12	3561072557118696	124
	249	938	2018-01-26 09:30:15	10.76	4498002758300	30
	250	969	2018-01-26 11:32:35	11.39	675911140852	67
	251	1730	2018-01-26 13:10:00	11.95	213193946980303	11
2478		2018-01-26 14:06:47	17.58	5361779664174555	106	
	253	299	2018-01-26 20:32:12	7.0	4027907156459098	78
	254	2454	2018-01-26 21:32:03	17.08	4743204091443101526	81

255	103	2018-01-27 00:11:12	5.87	344119623920892	64
256	3261	2018-01-27 00:56:55	16.14	4723783028106084756	94
257	2636	2018-01-27 04:07:35	18.58	4539990688484983	49
258	3452	2018-01-27 08:24:11	3.08	3582198969197591	43
259	3284	2018-01-27 08:36:58	15.15	6500236164848279	71
260	639	2018-01-27 14:00:31	1.88	6500236164848279	35
261	2382	2018-01-27 18:02:40	1.83	4268491956169254	85
262	1495	2018-01-27 19:10:04	11.93	213193946980303	45
263	142	2018-01-27 23:35:44	6.12	4276466390111	135
264	1374	2018-01-28 05:11:34	13.03	4268491956169254	109
265	2978	2018-01-28 14:38:33	19.93	3517111172421930	71
266	911	2018-01-28 14:45:12	10.71	5135837688671496	77
267	2686	2018-01-28 17:01:39	17.62	5570600642865857	82
268	3280	2018-01-28 18:51:37	18.64	6500236164848279	147
269	744	2018-01-28 19:30:38	0.72	6011987562414062	36
270	231	2018-01-28 20:44:10	6.52	4279104135293225293	44
271	2660	2018-01-28 21:41:42	18.8	4027907156459098	105
272	3168	2018-01-29 02:00:47	18.65	4498002758300	90
273	2852	2018-01-29 02:49:08	18.19	4723783028106084756	40
274	549	2018-01-29 02:51:23	3.68	3561072557118696	140
275	2115	2018-01-29 05:15:42	3.54	4962915017023706562	45
276	1914	2018-01-29 06:32:49	10.24	3517111172421930	49
277	536	2018-01-29 07:01:26	2.65	4150721559116778	5

278	681	2018-01-29 07:09:51	2.46	4723783028106084756	138
279	1798	2018-01-29 11:06:43	14.19	4165305432349489280	59
280	356	2018-01-29 11:52:48	5.14	4761049645711555811	131
281	2014	2018-01-29 12:54:56	3.93	4743204091443101526	142
282	2029	2018-01-29 13:38:21	2.06	3561072557118696	22
283	3365	2018-01-29 14:09:08	2.37	5135837688671496	92
284	2473	2018-01-29 17:25:24	17.99	3561072557118696	34
285	1020	2018-01-29 19:47:14	10.25	5175947111814778	91
286	2076	2018-01-29 22:49:11	5.33	4506405265172173	45
287	2664	2018-01-29 23:55:06	16.73	584226564303	10
288	1486	2018-01-30 02:32:22	10.15	3561072557118696	87
289	2177	2018-01-30 04:24:37	2.72	30181963913340	133
290	2782	2018-01-30 04:52:53	17.05	5570600642865857	147
291	2646	2018-01-30 06:49:12	15.97	4279104135293225293	82
292	1027	2018-01-30 09:15:16	10.53	3561072557118696	79
293	2047	2018-01-30 14:21:12	2.86	3535651398328201	32
294	2132	2018-01-30 14:21:48	4.15	4188164051171486	85
295	3205	2018-01-30 16:34:45	16.91	3517111172421930	112
296	26	2018-01-30 16:48:54	4.04	376027549341849	74
297	1243	2018-01-30 17:53:56	10.85	4279104135293225293	64
298	1415	2018-01-30 18:31:00	1177.0	4319653513507	64
299	2186	2018-01-30 22:14:39	3.65	4159836738768855913	102
300	3126	2018-01-31 00:51:40	16.43	4188164051171486	48

301	3310	2018-01-31 01:24:31	15.99	4159836738768855913	105
302	1080	2018-01-31 02:09:59	10.09	4586962917519654607	118
303	647	2018-01-31 05:46:43	2.75	4319653513507	81
304	1729	2018-01-31 06:01:51	10.28	4506405265172173	26
305	1545	2018-01-31 09:42:00	10.49	3517111172421930	77
306	2219	2018-01-31 10:34:42	4.17	584226564303	78
307	2954	2018-01-31 11:06:22	18.78	4711773125020499	48
308	777	2018-01-31 13:17:35	0.58	4741042733274	63
309	1405	2018-01-31 18:25:03	11.25	4279104135293225293	127
310	3286	2018-01-31 23:44:43	16.02	4506405265172173	44
311	1657	2018-02-01 08:49:24	10.26	4723783028106084756	115
312	45	2018-02-01 09:29:56	6.3	4539990688484983	116
313	1024	2018-02-01 11:05:57	10.49	4741042733274	138
314	899	2018-02-01 11:15:49	10.65	4279104135293225293	65
315	1279	2018-02-01 18:37:46	10.27	4150721559116778	145
316	989	2018-02-01 20:33:09	11.62	3582198969197591	4
317	1516	2018-02-01 20:44:12	10.24	3517111172421930	51
318	2150	2018-02-01 21:52:43	1.54	4743204091443101526	43
319	2793	2018-02-02 03:11:47	15.95	4834483169177062	27
320	2776	2018-02-02 10:56:03	15.52	4644008655884311378	95
321	1360	2018-02-02 11:31:33	10.75	372414832802279	135
322	2638	2018-02-02 16:57:12	15.27	4681896441519	53
323	3491	2018-02-02 17:43:38	2.17	5500708021555307	64

324	1752	2018-02-02 17:56:30	10.15	501809222273	127
325	2456	2018-02-02 18:46:46	18.64	4741042733274	148
326	13	2018-02-02 21:41:12	4.29	6500236164848279	148
327	248	2018-02-03 03:04:53	5.24	3582198969197591	83
328	1715	2018-02-03 06:57:16	12.39	5570600642865857	69
329	2245	2018-02-03 07:45:02	4.53	5570600642865857	80
330	1701	2018-02-03 12:53:24	10.23	4761049645711555811	95
331	488	2018-02-03 14:33:11	7.05	30181963913340	3
332	638	2018-02-03 15:45:32	1.17	3516952396080247	19
333	2719	2018-02-03 16:27:12	15.98	4723783028106084756	114
334	3181	2018-02-03 16:34:30	18.53	3561072557118696	86
335	3395	2018-02-03 18:05:39	1.41	4866761290278198714	65
336	1435	2018-02-03 18:48:33	13.01	30181963913340	75
337	3216	2018-02-03 19:32:30	17.94	3516952396080247	104
338	2389	2018-02-04 02:22:17	4.79	30063281385429	27
339	11	2018-02-04 04:00:18	6.9	6500236164848279	125
340	2011	2018-02-04 09:22:22	2.41	213193946980303	94
341	1996	2018-02-04 16:13:14	2.92	5361779664174555	4
342	1981	2018-02-04 18:44:31	2.68	4741042733274	104
343	810	2018-02-04 19:02:09	0.51	3561954487988605	75
344	3281	2018-02-04 20:48:29	18.11	4539990688484983	98
345	2256	2018-02-04 22:25:14	5.94	4268491956169254	43
346	1505	2018-02-05 03:47:29	10.27	4268491956169254	17

347	2688	2018-02-05 08:37:24	17.85	180098539019105	1
348	3063	2018-02-05 09:39:22	15.73	503842928916	75
349	2086	2018-02-05 15:54:02	1.52	30063281385429	145
350	1740	2018-02-05 17:32:22	10.58	30181963913340	39
351	3237	2018-02-05 17:36:46	17.7	3535651398328201	50
352	607	2018-02-05 19:19:11	2.09	4539990688484983	87
353	1270	2018-02-05 19:35:31	10.08	501879657465	52
354	1281	2018-02-05 20:08:01	10.92	4027907156459098	34
355	3297	2018-02-05 20:43:26	17.61	180098539019105	35
356	881	2018-02-05 21:59:07	10.81	372414832802279	65
357	1819	2018-02-05 23:22:08	10.96	4159836738768855913	67
358	1079	2018-02-06 04:34:49	10.86	4586962917519654607	132
359	3494	2018-02-06 10:55:49	2.49	4761049645711555811	100
360	372	2018-02-06 10:58:29	4.83	376027549341849	72
361	611	2018-02-06 18:29:49	2.53	3561072557118696	102
362	1998	2018-02-06 20:53:47	4.58	180098539019105	9
363	1603	2018-02-06 22:44:18	10.32	30181963913340	45
364	2418	2018-02-07 00:20:11	5.97	372414832802279	104
365	2072	2018-02-07 01:36:19	5.64	4761049645711555811	63
366	230	2018-02-07 02:48:38	7.68	4263694062533017	63
367	1115	2018-02-07 03:45:39	10.21	4741042733274	42
368	1473	2018-02-07 15:53:22	10.33	3561072557118696	49
369	859	2018-02-07 23:35:00	2.87	501809222273	70

370	205	2018-02-08 04:15:41	7.02	30063281385429	96
371	2878	2018-02-08 05:12:18	18.32	4866761290278198714	57
372	2085	2018-02-08 08:36:01	2.48	501809222273	85
373	748	2018-02-08 10:27:37	3.48	4962915017023706562	139
374	3473	2018-02-08 11:06:06	2.45	5361779664174555	127
375	3060	2018-02-08 12:15:41	15.39	675911140852	135
376	364	2018-02-08 16:35:44	7.15	4268491956169254	97
377	3102	2018-02-08 19:56:26	15.71	4741042733274	88
378	121	2018-02-08 21:10:00	5.52	584226564303	124
379	1927	2018-02-09 00:54:12	1.72	4586962917519654607	3
380	3436	2018-02-09 07:38:19	1.16	503842928916	12
381	2409	2018-02-09 11:38:37	445.0	3516952396080247	112
382	657	2018-02-09 15:30:03	1.21	3582198969197591	101
383	1265	2018-02-09 19:25:44	11.11	501879657465	114
384	1142	2018-02-10 02:01:54	11.43	5570600642865857	68
385	3133	2018-02-10 03:34:18	15.25	4150721559116778	122
386	2665	2018-02-10 05:36:19	15.92	30142966699187	126
387	2629	2018-02-10 10:29:14	20.01	3581345943543942	148
388	1386	2018-02-10 10:55:31	10.14	4150721559116778	35
389	3313	2018-02-10 11:17:10	12.97	4743204091443101526	135
390	1556	2018-02-10 11:56:25	11.33	4506405265172173	10
391	375	2018-02-10 12:04:36	7.39	4268491956169254	105
392	3418	2018-02-10 13:26:43	1.66	4276466390111	72

393	3013	2018-02-10 16:22:57	17.3	5570600642865857	150
394	2801	2018-02-10 16:43:22	15.67	3516952396080247	25
395	1487	2018-02-10 18:53:22	10.74	213193946980303	105
396	206	2018-02-11 00:03:19	7.51	501809222273	25
397	1419	2018-02-11 02:48:31	12.98	3561954487988605	115
398	1446	2018-02-11 06:30:43	10.93	4962915017023706562	65
399	738	2018-02-11 08:25:36	1.26	3582198969197591	38
400	2894	2018-02-11 14:38:30	15.76	4723783028106084756	111
401	556	2018-02-11 16:04:02	2.74	4159836738768855913	63
402	988	2018-02-11 17:50:29	11.46	4027907156459098	131
403	1449	2018-02-11 21:30:00	10.77	584226564303	142
404	2290	2018-02-11 22:11:53	4.56	4761049645711555811	127
405	2554	2018-02-11 22:51:15	16.2	4539990688484983	118
406	127	2018-02-12 03:44:20	3.69	4319653513507	116
407	2126	2018-02-12 10:31:41	5.47	4188164051171486	98
408	3029	2018-02-12 15:16:31	19.8	5570600642865857	39
409	430	2018-02-12 15:26:58	7.73	4761049645711555811	150
3410	2018-03-20 08:58:40	2.06	4276466390111	130	
797	236	2018-03-20 10:19:25	852.0	501879657465	35
798	916	2018-03-20 12:58:34	12.16	4723783028106084756	109

799	1821	2018-03-20 13:05:54	1011.0	30142966699187	141
800	3003	2018-03-20 15:18:33	18.41	5361779664174555	139
801	1127	2018-03-20 15:47:33	11.54	4506405265172173	53
802	3407	2018-03-20 17:15:15	1.64	4866761290278198714	60
803	2056	2018-03-21 01:10:18	3.2	4711773125020499	100
804	1207	2018-03-21 04:04:04	10.54	3516952396080247	107
805	552	2018-03-21 08:07:16	1.77	5570600642865857	51
806	2966	2018-03-21 10:17:56	18.18	501809222273	79
807	2707	2018-03-21 14:26:38	16.19	180098539019105	12
808	1905	2018-03-21 16:22:37	11.7	503842928916	12
809	79	2018-03-21 17:16:06	7.24	4681896441519	137
810	2502	2018-03-21 18:06:25	16.71	4741042733274	90
811	2985	2018-03-21 20:15:49	16.3	4644008655884311378	124
812	2916	2018-03-21 21:49:48	15.95	4268491956169254	145
813	107	2018-03-22 00:52:06	5.39	4644008655884311378	138
814	1971	2018-03-22 08:20:15	4.98	4165305432349489280	49
815	288	2018-03-22 10:13:54	6.82	4866761290278198714	112
816	1344	2018-03-22 10:34:09	10.45	3561954487988605	33
817	2853	2018-03-22 11:56:50	18.3	3517111172421930	35
818	1407	2018-03-22 13:13:49	11.2	4743204091443101526	76
819	601	2018-03-22 13:35:24	2.3	4681896441519	55
820	3188	2018-03-22 14:16:15	13.88	30063281385429	19
821	945	2018-03-22 23:42:48	11.36	6011987562414062	59

822	694	2018-03-23 07:13:23	1.46	376027549341849	121
823	541	2018-03-23 07:49:52	2.79	376027549341849	41
824	2907	2018-03-23 09:31:49	14.26	503842928916	43
825	1161	2018-03-23 12:13:43	11.46	4681896441519	99
826	1886	2018-03-23 15:04:43	10.45	213193946980303	120
827	2849	2018-03-23 17:23:57	18.88	5135837688671496	136
828	2292	2018-03-23 18:20:40	3.41	30078299053512	8
829	3379	2018-03-24 00:51:57	3.08	4743204091443101526	87
830	2965	2018-03-24 01:13:41	15.83	5135837688671496	133
831	950	2018-03-24 02:22:57	11.11	4761049645711555811	137
832	1474	2018-03-24 04:51:58	10.04	4681896441519	149
833	1587	2018-03-24 09:19:46	12.34	5500708021555307	42
834	1804	2018-03-24 11:29:42	11.94	3561072557118696	80
835	1709	2018-03-24 12:10:08	10.14	4741042733274	114
836	3104	2018-03-24 13:54:41	18.64	4723783028106084756	14
837	1375	2018-03-24 14:58:56	10.91	6500236164848279	104
838	2626	2018-03-24 17:21:08	18.1	5570600642865857	64
839	3485	2018-03-24 18:16:00	1.81	4027907156459098	119
840	1297	2018-03-24 22:30:49	10.15	6011987562414062	83
841	1784	2018-03-24 22:45:14	10.78	180098539019105	44
842	2548	2018-03-25 00:54:15	15.48	5361779664174555	107
843	27	2018-03-25 05:17:49	6.7	4586962917519654607	95
844	46	2018-03-25 08:27:43	5.17	30078299053512	19

845	1116	2018-03-25 12:33:24	11.05	376027549341849	89
846	2037	2018-03-25 13:07:44	2.62	4188164051171486	65
847	2648	2018-03-25 13:31:56	19.6	344119623920892	93
848	1922	2018-03-25 15:00:40	12.27	4711773125020499	108
849	1118	2018-03-25 15:27:30	11.38	5361779664174555	139
850	1317	2018-03-25 17:32:11	11.37	4506405265172173	36
851	1993	2018-03-25 19:05:08	4.57	3561954487988605	137
852	566	2018-03-26 02:44:07	3.11	4506405265172173	60
853	2385	2018-03-26 05:08:48	4.43	4263694062533017	97
854	1620	2018-03-26 07:41:59	1009.0	30181963913340	111
855	2895	2018-03-26 08:08:27	19.34	5500708021555307	96
856	204	2018-03-26 08:14:09	6.2	501809222273	45
857	529	2018-03-26 12:28:38	2.16	376027549341849	53
858	1447	2018-03-26 18:39:32	10.27	4834483169177062	16
859	2139	2018-03-26 22:58:10	2.85	4962915017023706562	75
860	951	2018-03-27 02:40:54	13.77	4263694062533017	48
861	766	2018-03-27 07:04:02	0.78	4268491956169254	48
862	1215	2018-03-27 12:20:26	11.01	30078299053512	95
863	2865	2018-03-27 17:52:28	17.38	30142966699187	43
864	2453	2018-03-27 18:10:00	15.79	5297187379298983	30
865	583	2018-03-27 19:33:15	2.55	5500708021555307	23
866	1936	2018-03-27 21:24:13	4.39	4165305432349489280	92
867	1989	2018-03-28 04:11:22	3.05	5297187379298983	70

868	2953	2018-03-28 05:30:09	19.55	30142966699187	23
869	2570	2018-03-28 07:36:02	16.36	4165305432349489280	67
870	3405	2018-03-28 11:15:53	2.45	4159836738768855913	24
871	1711	2018-03-28 13:00:17	11.63	3561072557118696	35
872	2792	2018-03-28 14:44:02	18.68	30142966699187	149
873	3151	2018-03-28 14:52:41	16.68	4539990688484983	119
874	2496	2018-03-28 19:25:33	14.22	4150721559116778	46
875	1810	2018-03-28 23:13:53	10.92	3517111172421930	82
876	1387	2018-03-28 23:56:44	10.46	4962915017023706562	26
877	3347	2018-03-29 00:33:24	1.41	584226564303	2
878	478	2018-03-29 06:38:59	5.39	5361779664174555	3
879	154	2018-03-29 11:42:05	6.57	3581345943543942	150
880	2642	2018-03-29 13:21:16	16.07	4711773125020499	74
881	576	2018-03-29 14:43:30	1.46	4276466390111	138
882	1734	2018-03-29 17:18:07	10.59	4711773125020499	83
883	3364	2018-03-29 18:31:14	3.21	376027549341849	149
884	752	2018-03-29 18:33:39	2.53	344119623920892	21
885	2495	2018-03-29 20:01:40	18.62	4866761290278198714	11
886	1974	2018-03-30 05:38:41	4.91	4159836738768855913	18
887	400	2018-03-30 06:01:36	7.55	4027907156459098	59
888	1862	2018-03-30 08:12:28	12.96	4268491956169254	103
889	1537	2018-03-30 08:28:18	10.53	4188164051171486	13
890	457	2018-03-30 12:57:30	6.27	4159836738768855913	33

891	3106	2018-03-30 15:39:04	19.87	501809222273	33
892	817	2018-03-30 16:04:04	1.52	4743204091443101526	104
893	1153	2018-03-30 17:01:55	10.58	4761049645711555811	3
894	2220	2018-03-30 17:36:58	1.26	3517111172421930	61
895	2743	2018-03-30 18:46:40	17.2	503842928916	138
896	507	2018-03-30 21:04:57	3.08	4498002758300	80
897	274	2018-03-30 22:54:28	6.43	4268491956169254	126
898	1145	2018-03-30 23:00:32	10.79	4723783028106084756	135
899	1072	2018-03-31 00:50:30	11.53	3517111172421930	139
900	1271	2018-03-31 03:26:27	10.46	4711773125020499	53
901	3042	2018-03-31 04:35:13	16.53	5570600642865857	90
902	2941	2018-03-31 06:08:07	14.21	4276466390111	87
903	2435	2018-03-31 08:48:53	3.65	5361779664174555	129
904	2897	2018-03-31 09:18:18	13.98	5175947111814778	75
905	2653	2018-03-31 12:54:47	18.59	4743204091443101526	109
906	909	2018-03-31 16:18:26	11.41	4743204091443101526	30
907	3335	2018-03-31 17:56:51	2.35	30181963913340	111
908	3235	2018-03-31 20:12:10	21.04	372414832802279	103
909	390	2018-03-31 21:50:44	4.22	4962915017023706562	24
910	179	2018-03-31 22:41:23	6.82	30142966699187	92
911	1934	2018-03-31 23:13:53	1.56	344119623920892	142
912	739	2018-04-01 01:48:11	1.71	30078299053512	19
913	774	2018-04-01 07:17:21	100.0	4319653513507	111

914	2332	2018-04-01 11:54:51	1.08	675911140852	140
915	1177	2018-04-01 13:53:19	14.36	213193946980303	40
916	2669	2018-04-01 15:55:53	18.67	5570600642865857	82
917	1402	2018-04-01 17:19:09	10.81	4644008655884311378	123
918	2419	2018-04-01 19:06:02	1.29	3582198969197591	54
919	2564	2018-04-01 19:59:57	17.75	4711773125020499	83
920	1930	2018-04-01 20:08:33	4.18	4681896441519	136
921	2124	2018-04-01 21:08:23	2.62	4319653513507	82
922	294	2018-04-02 01:50:15	7.08	372414832802279	138
923	902	2018-04-02 14:52:44	11.24	4539990688484983	42
924	2616	2018-04-02 18:34:50	17.15	372414832802279	80
925	1806	2018-04-02 22:19:53	11.15	4644008655884311378	26
926	386	2018-04-03 01:38:27	5.08	4150721559116778	8
927	1359	2018-04-03 03:23:37	1077.0	344119623920892	100
928	1314	2018-04-04 00:47:59	10.28	3561954487988605	38
929	955	2018-04-04 01:34:45	10.29	5361779664174555	145
930	518	2018-04-04 06:41:06	6.07	30078299053512	122
931	1107	2018-04-04 10:11:32	10.56	4150721559116778	102
932	2466	2018-04-04 12:57:15	16.61	4723783028106084756	53
933	2471	2018-04-05 01:29:17	16.22	4743204091443101526	124
934	592	2018-04-05 02:07:01	2.04	3582198969197591	4
935	2529	2018-04-05 04:38:57	13.83	6011987562414062	57
936	3195	2018-04-05 06:19:44	17.54	376027549341849	134

937	460	2018-04-05 12:46:25	5.53	4268491956169254	128
938	2624	2018-04-05 19:52:03	14.39	30181963913340	93
939	1604	2018-04-05 22:50:44	11.37	180098539019105	92
940	3228	2018-04-06 00:53:35	15.8	4268491956169254	26
941	2225	2018-04-06 02:30:41	1.44	4263694062533017	72
942	2689	2018-04-06 02:42:53	13.57	5135837688671496	60
943	629	2018-04-06 13:03:31	1.43	3561954487988605	118
944	2042	2018-04-06 13:45:51	1.59	584226564303	141
945	3495	2018-04-06 14:08:53	2.94	4743204091443101526	80
946	861	2018-04-06 16:57:28	1.02	180098539019105	50
947	2809	2018-04-06 17:25:52	15.32	4586962917519654607	135
948	2366	2018-04-06 17:48:38	1.78	4188164051171486	56
949	2137	2018-04-06 19:58:27	4.44	376027549341849	34
950	2771	2018-04-07 04:37:00	15.11	5361779664174555	142
951	3053	2018-04-07 07:18:20	14.8	5297187379298983	23
952	3412	2018-04-07 08:18:17	2.58	503842928916	120
953	338	2018-04-07 09:06:38	3.22	4681896441519	124
954	1389	2018-04-07 09:36:01	10.38	4027907156459098	99
955	1964	2018-04-07 14:02:37	5.55	30063281385429	59
956	1642	2018-04-07 15:15:19	10.8	5135837688671496	24
957	2964	2018-04-07 15:21:18	17.64	4279104135293225293	65
958	396	2018-04-07 16:40:23	6.5	675911140852	122
959	1750	2018-04-07 20:28:00	10.92	376027549341849	20

960	2243	2018-04-08 04:28:30	3.79	5500708021555307	71
961	398	2018-04-08 05:03:56	7.18	4279104135293225293	50
962	1341	2018-04-08 06:03:50	1063.0	4319653513507	16
963	1251	2018-04-08 07:06:20	11.73	3561954487988605	83
964	327	2018-04-08 09:08:35	4.34	4279104135293225293	71
965	1662	2018-04-08 09:46:08	10.53	5175947111814778	65
966	1508	2018-04-08 11:31:27	11.26	675911140852	108
967	2129	2018-04-08 14:37:51	4.58	5297187379298983	105
968	1944	2018-04-08 16:34:54	1.49	3535651398328201	109
969	1113	2018-04-08 18:03:55	10.15	372414832802279	84
970	1130	2018-04-08 18:14:22	10.06	4319653513507	96
971	220	2018-04-08 18:49:37	8.07	5361779664174555	74
972	3460	2018-04-08 20:15:13	1.21	3561072557118696	37
973	2873	2018-04-08 21:13:01	16.24	4711773125020499	124
974	1312	2018-04-08 21:40:55	10.14	30063281385429	26
975	3184	2018-04-09 02:20:29	17.81	4165305432349489280	78
976	1427	2018-04-09 04:09:54	10.01	3535651398328201	54
977	3190	2018-04-09 04:21:58	16.43	4723783028106084756	107
978	2675	2018-04-09 06:41:33	21.02	5175947111814778	50
979	3020	2018-04-09 07:08:11	17.73	4150721559116778	61
980	2320	2018-04-09 08:30:25	1.99	4539990688484983	124
981	3423	2018-04-09 10:24:32	283.0	4761049645711555811	50
982	600	2018-04-09 10:33:41	1.91	30063281385429	76

983	2278	2018-04-09 10:45:20	3.65	5361779664174555	113
984	329	2018-04-09 18:28:25	269.0	4319653513507	36
985	2904	2018-04-09 18:40:01	17.76	4279104135293225293	116
986	489	2018-04-09 18:59:37	8.31	4743204091443101526	14
987	2201	2018-04-09 19:04:52	1.77	5361779664174555	9
988	3329	2018-04-09 20:12:09	3.41	180098539019105	98
989	940	2018-04-09 21:08:59	10.47	30063281385429	103
990	1092	2018-04-09 21:26:54	11.24	4644008655884311378	77
991	287	2018-04-09 23:29:21	5.54	584226564303	11
992	1515	2018-04-10 02:51:58	10.71	3517111172421930	91
993	408	2018-04-10 06:08:01	543.0	3516952396080247	63
994	892	2018-04-10 09:14:37	10.32	5297187379298983	14
995	3185	2018-04-10 12:54:34	15.81	584226564303	115
996	1493	2018-04-10 13:16:51	10.3	4761049645711555811	134
997	503	2018-04-10 17:14:29	7.33	4268491956169254	86
998	894	2018-04-10 17:25:44	10.13	4027907156459098	67
999	2317	2018-04-10 17:32:28	3.86	30063281385429	45
1000	2052	2018-04-10 17:43:42	1.5	6500236164848279	40
1001	1163	2018-04-10 23:03:20	10.24	4319653513507	101
1002	1400	2018-04-11 00:35:22	10.48	4962915017023706562	21
1003	764	2018-04-11 02:50:37	325.0	30181963913340	45
1004	1186	2018-04-11 04:05:46	12.35	5135837688671496	119
1005	3247	2018-04-11 04:29:45	20.13	4268491956169254	126

1006	2889	2018-04-11 05:08:59	19.28	4159836738768855913	145
1007	2816	2018-04-11 08:26:52	16.07	5135837688671496	34
1008	313	2018-04-11 16:45:26	4.28	30063281385429	10
1009	3330	2018-04-11 18:29:01	2.12	4150721559116778	88
1010	1880	2018-04-12 01:35:50	10.34	3535651398328201	14

SQL/schema.sql

```
-- Exported from QuickDBD: https://www.quickdatabasediagrams.com/
 -- Link to schema: https://app.quickdatabasediagrams.com/#/d/s8Wnqm
-- NOTE! If you have used non-SQL datatypes in your design, you will have to change these here.

--Card Holder

CREATE TABLE "card_holder" (
    "id" SERIAL NOT NULL,
    "name" VARCHAR(50) NOT NULL,
     CONSTRAINT "pk_card_holder" PRIMARY KEY (
```

```
        "id"
    )
);
-- Credit Card
CREATE TABLE "credit_card" (
    "card" VARCHAR(20) NOT NULL,
    "id_card_holder" INT NOT NULL,
     CONSTRAINT "pk_credit_card" PRIMARY KEY (
        "card"
     )
);

-- Merchant

CREATE TABLE "merchant" (
        "id" SERIAL NOT NULL,
        "name" VARCHAR(255) NOT NULL,
      "id_merchant_category" INT NOT NULL,
       CONSTRAINT "pk_merchant" PRIMARY KEY (
          "id"
    )
);

-- Merchant Category
```

Fraud Detection SQL

```
CREATE TABLE "merchant_category"
    ( "id" SERIAL NOT NULL,
      "name" VARCHAR(50) NOT NULL,
      CONSTRAINT "pk_merchant_category" PRIMARY KEY (
          "id"
       )
);
-- Transaction


CREATE TABLE "transaction' (
"id" INT NOT NULL,
    "date" TIMESTAMP NOT NULL,
    "amount" FLOAT NOT NULL,
 "card" VARCHAR(20) NOT NULL,
 "id_merchant" INT NOT NULL,
 CONSTRAINT "pk_transaction" PRIMARY KEY (
        "id"
     )
 );
ALTER TABLE "credit_card" ADD CONSTRAINT "fk_credit_card_id_card_holder" FOREIGN KEY("id_card_holder")
 REFERENCES "card_holder" ("id");
ALTER TABLE "credit_card" ADD CONSTRAINT "check_credit_card_length" CHECK (char_length("card") <= 20);
```

Tarun S Gowda

--ALTER TABLE "credit_card" DROP CONSTRAINT check_credit_card_length

ALTER TABLE "merchant" ADD CONSTRAINT "fk_merchant_id_merchant_category" FOREIGN KEY("id_merchant_category")

REFERENCES "merchant_category" ("id");

ALTER TABLE "transaction" ADD CONSTRAINT "fk_transaction_card" FOREIGN KEY("card") REFERENCES "credit_card" ("card");

ALTER TABLE "transaction" ADD CONSTRAINT "fk_transaction_id_merchant" FOREIGN KEY("id_merchant")

REFERENCES "merchant" ("id");

Seed.sql

INSERT INTO card_holder VALUES

(1, 'Robert Johnson'),

(2, 'Shane Shaffer'),

(3, 'Elizabeth Sawyer'),

(4, 'Danielle Green'),

(5, 'Sara Cooper'),

(6, 'Beth Hernandez'),

(7, 'Sean Taylor'),

(8, 'Michael Floyd'),

(9, 'Laurie Gibbs'),

(10, 'Matthew Gutierrez'),

(11, 'Brandon Pineda'),

(12, 'Megan Price'),

(13, 'John Martin'),

(14, 'Gary Jacobs'),

```
    (15, 'Kyle Tucker'),
    (16, 'Crystal Clark'),
    (17, 'Michael Carroll'),
    (18, 'Malik Carlson'),
    (19, 'Peter Mckay'),
    (20, 'Kevin Spencer'),
    (21, 'Dana Washington'),
    (22, 'Austin Johnson'),
    (23, 'Mark Lewis'),
    (24, 'Stephanie Dalton'),
    (25, 'Nancy Contreras');

INSERT INTO credit_card VALUES
    (3517111172421930, 1),
    (4761049645711555811, 1),
    (4866761290278198714, 2),
    (675911140852, 2),
    (30078299053512, 3),
    (4263694062533017, 4),
    (584226564303, 4),
    (4276466390111, 5),
    (4268491956169254, 5),
    (3581345943543942, 6),
    (4159836738768855913, 6),
```

(3516952396080247, 7),

(4539990688484983, 7),

(4834483169177062, 8),

(30063281385429, 8),

(30181963913340, 9),

(4962915017023706562, 10),

(4165305432349489280, 10),

(213193946980303, 10),

(180098539019105, 11),

(4644008655884311378, 11),

(4027907156459098, 11),

(501879657465, 12),

(5297187379298983, 12),

(376027549341849, 12),

(4711773125020499, 13),

(5135837688671496, 13),

(3561954487988605, 13),

(5175947111814778, 14),

(4723783028106084756, 15),

(6500236164848279, 15),

(503842928916, 16),

(5570600642865857, 16),

(5500708021555307, 16),

(6011987562414062, 17),

```
    (4498002758300, 18),
    (344119623920892, 18),
    (4743204091443101526, 19),
    (5361779664174555, 19),
    (3561072557118696, 19),
    (3535651398328201, 20),
    (4506405265172173, 20),
    (4586962917519654607, 20),
    (4279104135293225293, 21),
    (501809222273, 22),
    (4741042733274, 23),
    (4188164051171486, 23),
    (4150721559116778, 23),
    (4681896441519, 24),
    (30142966699187, 24),
    (3582198969197591, 24),
    (4319653513507, 25),
    (372414832802279, 25);

INSERT INTO merchant_category VALUES
    (1, 'restaurant'),
    (2, 'coffee shop'),
    (3, 'bar'),
    (4, 'pub'),
```

```
    (5, 'food truck');


INSERT INTO merchant VALUES
    (1, 'Murphy, Heath and Fields', 1),
    (2, 'Riggs-Adams', 1),
    (3, 'Sanders, Parks and Mcfarland', 2),
    (4, 'Mccarty-Thomas', 3),
    (5, 'Miller-Blevins', 4),
    (6, 'Wilson and Sons', 1),
    (7, 'Gomez-Kelly', 4),
    (8, 'Russell-Thomas', 1),
    (9, 'Curry, Scott and Richardson', 3),
    (10, 'Herrera Group', 1),
    (11, 'Stanton Group', 4),
    (12, 'Bell, Gonzalez and Lowe', 4),
    (13, 'Giles and Sons', 4),
    (14, 'Osborne-Page', 2),
    (15, 'Long, Harrell and Johnson', 5),
    (16, 'Bryant, Thomas and Collins', 4),
    (17, 'Bauer-Cole', 3),
    (18, 'Romero-Jordan', 5),
    (19, 'Santos-Fitzgerald', 4),
    (20, 'Kim-Lopez', 2),
    (21, 'Robertson-Smith', 4),
```

```
(22, 'Dalton, Cameron and Jones', 3),
(23, 'Wilson, Roberts and Davenport', 5),
(24, 'Rodgers, Johnston and Macias', 5),
(25, 'Vaughn, Wilson and Hall', 1),
(26, 'Smith-Stephens', 2),
(27, 'Horn Ltd', 2),
(28, 'Hess-Fischer', 5),
(29, 'Browning-Cantu', 4),
(30, 'Atkinson Ltd', 3),
(31, 'Fisher, Salazar and Thomas', 5),
(32, 'Norton, Burton and Smith', 5),
(33, 'Vasquez-Parker', 3),
(34, 'Combs-Jones', 5),
(35, 'Jarvis-Turner', 4),
(36, 'Hamilton-Mcfarland', 1),
(37, 'Nguyen, Bautista and Williams', 4),
(38, 'Brown LLC', 3),
(39, 'Young-Navarro', 5),
(40, 'Cox, Montgomery and Morgan', 3),
(41, 'Ford, Williams and Dunn', 4),
(42, 'Kennedy-Chen', 3),
(43, 'Wallace and Sons', 2),
(44, 'Little-Floyd', 4),
(45, 'Velazquez Ltd', 4),
```

(46, 'Miller, Chavez and Cobb', 5),

(47, 'Martin Inc', 1),

(48, 'Baker Inc', 5),

(49, 'Davis, Lowe and Baxter', 5),

(50, 'Johnson-Watts', 1),

(51, 'Fisher-Bolton', 1),

(52, 'Jensen-Stanley', 5),

(53, 'Wallace PLC', 4),

(54, 'Berry-Lopez', 1),

(55, 'Johnson, Rivas and Anderson', 1),

(56, 'Smith PLC', 3),

(57, 'Thornton-Williams', 4),

(58, 'Young, Hull and Williams', 4),

(59, 'Williams Group', 3),

(60, 'Smith-Richards', 4),

(61, 'Richardson, Smith and Jordan', 5),

(62, 'Cooper, Carpenter and Jackson', 5),

(63, 'Reed Group', 5),

(64, 'Cline, Myers and Strong', 1),

(65, 'Allen, Ramos and Carroll', 4),

(66, 'Robles Inc', 3),

(67, 'Maxwell, Tapia and Villanueva', 2),

(68, 'Ramirez-Carr', 2),

(69, 'Dominguez PLC', 5),

```
(70, 'White-Hall', 3),
(71, 'Greene LLC', 1),
(72, 'Lopez-Kelly', 1),
(73, 'Colon Ltd', 3),
(74, 'Skinner-Williams', 4),
(75, 'Martinez Group', 1),
(76, 'Lowe PLC', 1),
(77, 'Brown, Ballard and Glass', 1),
(78, 'Ruiz-Anderson', 4),
(79, 'Lee LLC', 1),
(80, 'Kelly, Dyer and Schmitt', 5),
(81, 'Fowler and Sons', 5),
(82, 'Day-Murray', 5),
(83, 'Solis Group', 5),
(84, 'Marshall-Rojas', 2),
(85, 'Patton-Rivera', 3),
(86, 'Walker, Campbell and Sullivan', 5),
(87, 'Griffin-Woodard', 3),
(88, 'Armstrong PLC', 5),
(89, 'Kelley-Roberts', 5),
(90, 'Brown-Cunningham', 4),
(91, 'Turner Ltd', 4),
(92, 'Garcia-White', 4),
(93, 'Rodriguez-Parker', 5),
```

```
(94, 'Yoder-Zavala', 5),
(95, 'Baxter-Smith', 1),
(96, 'Johnson-Fuller', 4),
(97, 'Ruiz-Chavez', 3),
(98, 'Rivera PLC', 4),
(99, 'Bond, Lewis and Rangel', 1),
(100, 'Townsend-Anderson', 1),
(101, 'Whitehead-Sexton', 4),
(102, 'Walters-Ward', 1),
(103, 'Jones, Clark and Hoover', 2),
(104, 'Mcdaniel, Hines and Mcfarland', 2),
(105, 'Garcia and Sons', 4),
(106, 'Carter-Blackwell', 4),
(107, 'Rowe-Abbott', 4),
(108, 'Best Inc', 1),
(109, 'Collins LLC', 2),
(110, 'Rodriguez, Dunlap and Nunez', 2),
(111, 'Padilla-Clements', 2),
(112, 'Greer Inc', 3),
(113, 'Edwards-Aguirre', 2),
(114, 'Greene-Wood', 3),
(115, 'Williams Inc', 4),
(116, 'Ferguson Ltd', 2),
(117, 'Mitchell Group', 2),
```

```
(118, 'Maldonado Group', 2),
(119, 'Henderson and Sons', 1),
(120, 'Vega, Jones and Castro', 5),
(121, 'Fleming, Smith and Collins', 3),
(122, 'Perry and Sons', 3),
(123, 'Boone, Davis and Townsend', 4),
(124, 'Mccarty PLC', 1),
(125, 'Russell and Sons', 4),
(126, 'Bartlett and Sons', 4),
(127, 'Williams, Wright and Wagner', 2),
(128, 'Pitts, Salinas and Garcia', 2),
(129, 'Sweeney-Paul', 2),
(130, 'Brown, Estrada and Powers', 2),
(131, 'Harrison, Newton and Hansen', 1),
(132, 'Pugh-Williams', 3),
(133, 'Scott, Hess and Finley', 3),
(134, 'Jenkins, Peterson and Beck', 1),
(135, 'Jacobs, Torres and Walker', 3),
(136, 'Martinez-Robinson', 3),
(137, 'Garcia PLC', 5),
(138, 'Mccullough-Murphy', 5),
(139, 'Kidd-Lopez', 5),
(140, 'Wheeler-Moreno', 5),
(141, 'Wood-Ramirez', 3),
```

```
    (142, 'Thomas-Garcia', 5),
    (143, 'Guzman, Garcia and Church', 3),
    (144, 'Walker, Deleon and Wolf', 1),
    (145, 'Hood-Phillips', 3),
    (146, 'Pitts, Smith and Gonzalez', 4),
    (147, 'Marshall-Lopez', 5),
    (148, 'Huerta, Keith and Walters', 5),
    (149, 'Clark and Sons', 5),
    (150, 'Johnson and Sons', 2);

INSERT INTO transaction VALUES
    (222, '2018-01-01 21:35:10', 6.22, 3561954487988605, 69),
    (2045, '2018-01-01 21:43:12', 3.83, 5135837688671496, 85),
    (395, '2018-01-01 22:41:21', 9.61, 213193946980303, 82),
    (3309, '2018-01-01 23:13:30', 19.03, 4263694062533017, 5),
    (567, '2018-01-01 23:15:10', 2.95, 4498002758300, 64),
    (1683, '2018-01-02 01:13:21', 11.24, 4263694062533017, 127),
    (2083, '2018-01-02 02:06:21', 1.46, 4319653513507, 93),
    (3488, '2018-01-02 04:36:45', 3.36, 4506405265172173, 136),
    (2635, '2018-01-02 05:45:43', 16.69, 5297187379298983, 120),
    (432, '2018-01-02 10:13:09', 8.55, 5175947111814778, 70),
    (2918, '2018-01-02 11:28:35', 15.85, 180098539019105, 16),
    (2020, '2018-01-02 13:17:15', 2.64, 501879657465, 84),
    (1084, '2018-01-02 15:10:33', 10.52, 6500236164848279, 39),
```

```
(3490, '2018-01-02 16:14:55', 3.12, 3517111172421930, 21),
(1179, '2018-01-02 19:40:46', 10.44, 4741042733274, 55),
(99, '2018-01-02 23:27:46', 1031.0, 501879657465, 95),
(191, '2018-01-03 05:13:19', 2.76, 3535651398328201, 149),
(2387, '2018-01-03 08:44:46', 3.89, 4027907156459098, 74),
(2512, '2018-01-03 10:18:20', 14.57, 4268491956169254, 106),
(793, '2018-01-03 10:34:16', 2.63, 30181963913340, 35),
(533, '2018-01-03 15:23:58', 1.39, 4962915017023706562, 100),
(1077, '2018-01-03 18:16:55', 10.27, 501809222273, 84),
(2120, '2018-01-03 21:04:28', 1.91, 3561072557118696, 108),
(1302, '2018-01-04 00:29:04', 12.45, 3561954487988605, 39),
(2022, '2018-01-04 01:28:18', 2.81, 4644008655884311378, 89),
(2806, '2018-01-04 01:35:21', 20.33, 3516952396080247, 112),
(2922, '2018-01-04 03:00:19', 17.59, 501809222273, 100),
(2650, '2018-01-04 03:05:18', 1685.0000000000002, 3516952396080247, 80),
(1510, '2018-01-04 06:34:37', 10.22, 4188164051171486, 80),
(139, '2018-01-04 11:15:55', 5.72, 3535651398328201, 99),
(1331, '2018-01-04 13:41:56', 12.7, 5135837688671496, 122),
(530, '2018-01-04 14:38:12', 2.84, 3535651398328201, 73),
(2859, '2018-01-04 17:19:14', 15.48, 5175947111814778, 72),
(2694, '2018-01-04 19:24:31', 14.31, 4586962917519654607, 134),
(2127, '2018-01-05 00:49:22', 2.32, 4188164051171486, 136),
(1995, '2018-01-05 01:10:27', 5.09, 501809222273, 46),
(1346, '2018-01-05 03:11:18', 12.2, 4268491956169254, 69),
```

(2176, '2018-01-05 05:47:28', 4.64, 5175947111814778, 103),

(1552, '2018-01-05 06:26:45', 10.74, 372414832802279, 86),

(487, '2018-01-05 06:27:06', 4.6, 4159836738768855913, 48),

(2077, '2018-01-05 07:19:27', 1.36, 344119623920892, 30),

(314, '2018-01-05 10:42:55', 5.31, 3561072557118696, 8),

(77, '2018-01-05 16:58:08', 4.57, 5135837688671496, 91),

(1570, '2018-01-06 01:43:44', 10.04, 5500708021555307, 34),

(2439, '2018-01-06 02:16:41', 1.33, 4866761290278198714, 127),

(2258, '2018-01-06 04:43:33', 5.31, 5297187379298983, 42),

(1867, '2018-01-06 05:13:20', 10.82, 4866761290278198714, 70),

(846, '2018-01-06 06:46:24', 2.7, 213193946980303, 146),

(2890, '2018-01-06 07:00:32', 18.35, 3561954487988605, 76),

(3291, '2018-01-06 08:42:50', 18.72, 3535651398328201, 76),

(2557, '2018-01-06 08:49:34', 16.28, 4268491956169254, 101),

(325, '2018-01-06 23:33:29', 7.6, 376027549341849, 141),

(2663, '2018-01-07 00:30:25', 17.84, 4159836738768855913, 112),

(3457, '2018-01-07 01:10:54', 175.0, 344119623920892, 12),

(447, '2018-01-07 07:33:17', 7.07, 4834483169177062, 14),

(1249, '2018-01-07 10:31:33', 10.9, 5297187379298983, 135),

(2108, '2018-01-07 14:57:23', 2.93, 4319653513507, 137),

(3028, '2018-01-07 15:10:27', 17.29, 4866761290278198714, 126),

(149, '2018-01-07 16:50:22', 5.83, 4159836738768855913, 81),

(2304, '2018-01-07 20:35:00', 2.5, 6011987562414062, 104),

(1384, '2018-01-08 00:26:21', 10.04, 3581345943543942, 64),

```
(1371, '2018-01-08 01:12:52', 10.4, 5500708021555307, 121),
(1291, '2018-01-08 02:34:32', 1029.0, 3581345943543942, 145),
(1065, '2018-01-08 04:15:30', 11.49, 4741042733274, 130),
(1340, '2018-01-08 04:29:20', 10.64, 501809222273, 116),
(1544, '2018-01-08 06:27:34', 10.48, 4027907156459098, 68),
(522, '2018-01-08 10:48:26', 11.06, 4743204091443101526, 113),
(812, '2018-01-08 11:15:36', 333.0, 344119623920892, 95),
(1640, '2018-01-08 11:21:20', 11.98, 5500708021555307, 133),
(2550, '2018-01-08 11:35:55', 15.84, 4150721559116778, 15),
(2095, '2018-01-08 14:06:25', 3.52, 180098539019105, 95),
(3130, '2018-01-08 16:38:22', 21.27, 5297187379298983, 106),
(3085, '2018-01-08 19:31:23', 17.43, 584226564303, 19),
(1816, '2018-01-08 20:10:59', 11.55, 344119623920892, 102),
(3480, '2018-01-08 21:44:34', 1.68, 4681896441519, 78),
(1188, '2018-01-08 21:44:58', 10.64, 5175947111814778, 84),
(379, '2018-01-09 11:14:56', 7.07, 501809222273, 136),
(687, '2018-01-09 21:06:16', 2.7, 4506405265172173, 67),
(2558, '2018-01-09 21:43:47', 16.37, 4681896441519, 64),
(754, '2018-01-10 00:25:40', 1.39, 372414832802279, 50),
(608, '2018-01-10 03:17:20', 0.87, 503842928916, 129),
(2818, '2018-01-10 03:24:29', 16.64, 3561954487988605, 65),
(865, '2018-01-10 03:55:54', 2.3, 4150721559116778, 19),
(998, '2018-01-10 10:07:20', 10.91, 675911140852, 78),
(1436, '2018-01-10 13:41:23', 11.5, 3517111172421930, 49),
```

(1938, '2018-01-10 14:50:30', 5.64, 5361779664174555, 44),

(602, '2018-01-10 15:56:41', 1.22, 503842928916, 27),

(3289, '2018-01-10 19:53:09', 18.41, 4159836738768855913, 23),

(1361, '2018-01-10 23:03:42', 10.7, 3535651398328201, 70),

(496, '2018-01-11 00:39:12', 3.23, 503842928916, 68),

(156, '2018-01-11 09:27:30', 5.55, 6011987562414062, 107),

(315, '2018-01-11 09:38:33', 7.94, 5135837688671496, 146),

(3498, '2018-01-11 09:58:11', 1.62, 5500708021555307, 149),

(3331, '2018-01-11 10:51:08', 1.24, 30063281385429, 7),

(2128, '2018-01-11 12:46:01', 3.97, 4834483169177062, 44),

(2188, '2018-01-11 13:20:31', 229.0, 5570600642865857, 115),

(1716, '2018-01-11 16:18:55', 10.27, 5361779664174555, 46),

(2146, '2018-01-11 19:36:21', 1.72, 4761049645711555811, 99),

(2779, '2018-01-11 22:55:51', 19.5, 4268491956169254, 8),

(1048, '2018-01-12 02:18:05', 11.52, 3582198969197591, 2),

(2170, '2018-01-12 02:48:29', 1.84, 30142966699187, 2),

(2025, '2018-01-12 04:33:17', 5.32, 4188164051171486, 10),

(451, '2018-01-12 06:34:53', 6.19, 30181963913340, 40),

(444, '2018-01-12 13:30:20', 5.92, 4506405265172173, 38),

(416, '2018-01-12 13:59:09', 4.45, 501809222273, 85),

(291, '2018-01-12 16:29:03', 4.69, 4268491956169254, 144),

(3373, '2018-01-12 18:04:04', 1.88, 4681896441519, 44),

(270, '2018-01-12 22:14:25', 6.4, 4276466390111, 112),

(1937, '2018-01-12 23:38:53', 1.1, 4506405265172173, 64),

```
(2835, '2018-01-13 02:02:40', 17.12, 376027549341849, 2),
(3175, '2018-01-13 03:06:36', 16.27, 4268491956169254, 24),
(450, '2018-01-13 03:20:50', 4.89, 213193946980303, 58),
(2222, '2018-01-13 04:49:38', 3.63, 4506405265172173, 53),
(2755, '2018-01-13 06:07:45', 19.24, 4962915017023706562, 32),
(582, '2018-01-13 14:36:43', 1.35, 4150721559116778, 102),
(1546, '2018-01-13 15:22:46', 10.15, 4711773125020499, 108),
(718, '2018-01-13 18:34:36', 2.22, 4263694062533017, 9),
(3445, '2018-01-13 22:18:00', 2.11, 30142966699187, 3),
(2386, '2018-01-13 22:26:57', 3.79, 5500708021555307, 106),
(2843, '2018-01-14 03:20:21', 14.69, 584226564303, 133),
(3023, '2018-01-14 05:02:22', 17.84, 372414832802279, 52),
(2879, '2018-01-14 06:19:11', 21.11, 3535651398328201, 74),
(830, '2018-01-14 07:06:57', 3.15, 584226564303, 96),
(2521, '2018-01-14 07:33:54', 15.21, 4741042733274, 21),
(1482, '2018-01-14 07:50:44', 10.23, 376027549341849, 38),
(1916, '2018-01-14 10:04:25', 12.33, 4165305432349489280, 55),
(1560, '2018-01-14 13:30:29', 10.94, 3517111172421930, 19),
(3159, '2018-01-14 15:03:26', 16.67, 213193946980303, 30),
(604, '2018-01-14 15:31:50', 1.42, 376027549341849, 97),
(970, '2018-01-14 17:29:28', 10.31, 4506405265172173, 40),
(3359, '2018-01-15 03:37:27', 1.47, 4834483169177062, 139),
(2787, '2018-01-15 06:23:30', 15.87, 4539990688484983, 70),
(2622, '2018-01-15 10:27:56', 15.51, 4761049645711555811, 8),
```

```
(1382, '2018-01-15 11:52:41', 11.67, 4159836738768855913, 139),
(3105, '2018-01-15 11:55:38', 17.16, 3561954487988605, 101),
(2581, '2018-01-15 19:37:59', 18.61, 3561954487988605, 74),
(591, '2018-01-15 20:53:52', 2.95, 4681896441519, 98),
(2714, '2018-01-16 00:57:00', 15.33, 4506405265172173, 149),
(3333, '2018-01-16 02:26:16', 1.65, 372414832802279, 31),
(3068, '2018-01-16 03:22:49', 13.29, 5297187379298983, 12),
(3065, '2018-01-16 04:12:01', 16.89, 4711773125020499, 109),
(2655, '2018-01-16 06:29:35', 17.64, 675911140852, 136),
(2866, '2018-01-16 08:02:04', 17.36, 30078299053512, 116),
(680, '2018-01-16 13:48:12', 1.84, 4268491956169254, 144),
(1755, '2018-01-16 13:51:42', 13.52, 4159836738768855913, 54),
(753, '2018-01-16 18:45:54', 1.48, 4962915017023706562, 86),
(665, '2018-01-16 19:19:48', 2.55, 344119623920892, 99),
(3278, '2018-01-17 00:03:25', 18.05, 4279104135293225293, 133),
(651, '2018-01-17 00:18:59', 3.6, 4159836738768855913, 44),
(246, '2018-01-17 00:22:04', 5.41, 4743204091443101526, 105),
(2971, '2018-01-17 02:02:41', 15.18, 213193946980303, 75),
(2875, '2018-01-17 04:56:56', 16.46, 376027549341849, 42),
(758, '2018-01-17 06:36:33', 1.02, 180098539019105, 147),
(1551, '2018-01-17 06:59:43', 10.0, 4586962917519654607, 149),
(1889, '2018-01-17 07:31:03', 12.67, 4498002758300, 89),
(1593, '2018-01-17 07:56:09', 11.91, 4681896441519, 2),
(30, '2018-01-17 09:09:33', 3.82, 4711773125020499, 111),
```

(733, '2018-01-17 09:25:09', 2.85, 3561954487988605, 18),

(878, '2018-01-17 11:09:35', 12.27, 376027549341849, 146),

(3340, '2018-01-17 15:02:38', 1.72, 4498002758300, 115),

(105, '2018-01-17 16:57:27', 1.25, 4268491956169254, 65),

(840, '2018-01-17 19:16:40', 2.11, 4268491956169254, 132),

(2314, '2018-01-17 20:06:45', 2.69, 3561072557118696, 29),

(3466, '2018-01-18 03:12:14', 3.44, 344119623920892, 5),

(2633, '2018-01-18 08:18:50', 17.54, 4165305432349489280, 104),

(2838, '2018-01-18 08:58:13', 17.53, 180098539019105, 137),

(3406, '2018-01-18 11:17:37', 1.24, 30181963913340, 116),

(2662, '2018-01-18 12:41:06', 15.86, 4319653513507, 43),

(2119, '2018-01-18 14:13:09', 2.61, 376027549341849, 26),

(3095, '2018-01-18 14:39:01', 19.0, 501879657465, 15),

(1303, '2018-01-18 18:49:21', 10.19, 4159836738768855913, 116),

(2882, '2018-01-19 01:11:13', 16.38, 5500708021555307, 82),

(784, '2018-01-19 01:34:25', 2.59, 4723783028106084756, 47),

(1410, '2018-01-19 03:01:33', 10.39, 4498002758300, 132),

(2956, '2018-01-19 05:03:56', 17.53, 4962915017023706562, 82),

(3381, '2018-01-19 07:08:43', 2.51, 4681896441519, 8),

(2253, '2018-01-19 10:15:55', 3.93, 4761049645711555811, 13),

(2015, '2018-01-19 11:41:42', 3.27, 213193946980303, 124),

(1183, '2018-01-19 12:28:12', 10.4, 180098539019105, 60),

(917, '2018-01-19 13:15:49', 10.16, 3535651398328201, 81),

(679, '2018-01-19 14:23:21', 2.47, 4150721559116778, 131),

(458, '2018-01-19 17:44:59', 5.96, 6500236164848279, 138),

(1245, '2018-01-19 20:12:31', 11.58, 4866761290278198714, 132),

(2238, '2018-01-20 01:13:21', 3.7, 5361779664174555, 107),

(2411, '2018-01-20 02:23:13', 2.17, 4263694062533017, 109),

(1954, '2018-01-20 03:18:02', 3.38, 30063281385429, 93),

(2791, '2018-01-20 03:19:33', 17.64, 30078299053512, 37),

(302, '2018-01-20 04:32:42', 4.51, 501809222273, 83),

(1266, '2018-01-20 07:47:30', 10.58, 30142966699187, 34),

(1456, '2018-01-20 08:35:34', 11.96, 501879657465, 121),

(256, '2018-01-20 09:56:37', 9.81, 4159836738768855913, 69),

(2445, '2018-01-20 10:50:36', 2.17, 4711773125020499, 113),

(729, '2018-01-20 13:44:37', 1.83, 501879657465, 119),

(1615, '2018-01-20 16:31:01', 10.51, 584226564303, 10),

(1320, '2018-01-20 17:30:16', 13.84, 4506405265172173, 109),

(193, '2018-01-20 19:46:39', 8.03, 4263694062533017, 8),

(1037, '2018-01-20 20:11:29', 10.12, 376027549341849, 37),

(277, '2018-01-20 21:29:46', 4.87, 180098539019105, 113),

(2537, '2018-01-20 22:58:27', 16.44, 6500236164848279, 18),

(1417, '2018-01-20 23:29:57', 10.58, 4723783028106084756, 110),

(870, '2018-01-21 00:03:41', 2.32, 4711773125020499, 130),

(3006, '2018-01-21 01:30:19', 20.29, 5135837688671496, 136),

(132, '2018-01-21 02:01:49', 8.61, 5175947111814778, 37),

(2844, '2018-01-21 05:38:08', 16.29, 3516952396080247, 106),

(494, '2018-01-21 06:47:24', 5.34, 4498002758300, 85),

(1892, '2018-01-21 13:18:10', 10.38, 4506405265172173, 15),

(227, '2018-01-21 15:37:39', 5.51, 213193946980303, 21),

(322, '2018-01-21 16:05:43', 2.88, 5175947111814778, 75),

(1492, '2018-01-21 17:27:37', 10.28, 501809222273, 148),

(1338, '2018-01-21 20:28:17', 10.24, 4723783028106084756, 63),

(565, '2018-01-21 23:04:02', 2.22, 372414832802279, 149),

(1704, '2018-01-22 02:08:47', 10.38, 4681896441519, 98),

(1442, '2018-01-22 08:07:03', 1131.0, 5570600642865857, 144),

(2044, '2018-01-22 11:52:29', 1.67, 180098539019105, 30),

(1057, '2018-01-22 14:56:43', 10.75, 5570600642865857, 117),

(1935, '2018-01-22 16:26:22', 2.35, 4723783028106084756, 96),

(1364, '2018-01-22 22:48:50', 11.63, 6011987562414062, 150),

(1509, '2018-01-23 01:51:00', 11.3, 4506405265172173, 134),

(3499, '2018-01-23 06:23:05', 2.67, 3516952396080247, 51),

(2667, '2018-01-23 06:29:37', 1678.0, 501879657465, 92),

(1379, '2018-01-23 08:07:03', 10.47, 4866761290278198714, 7),

(538, '2018-01-23 08:18:10', 1.48, 5500708021555307, 35),

(2070, '2018-01-23 10:30:44', 4.34, 4027907156459098, 110),

(110, '2018-01-23 12:06:35', 6.38, 5135837688671496, 94),

(2319, '2018-01-23 16:13:22', 5.58, 584226564303, 33),

(392, '2018-01-23 18:14:18', 7.15, 30063281385429, 88),

(318, '2018-01-24 01:31:16', 3.54, 3535651398328201, 81),

(2212, '2018-01-24 03:43:35', 2.78, 3581345943543942, 68),

(263, '2018-01-24 07:14:27', 5.56, 5500708021555307, 36),

```
(1463, '2018-01-24 07:55:45', 10.96, 6500236164848279, 107),
(168, '2018-01-24 08:28:08', 5.0, 4268491956169254, 141),
(3196, '2018-01-24 09:31:37', 15.22, 376027549341849, 83),
(942, '2018-01-24 10:54:25', 10.69, 30063281385429, 2),
(3093, '2018-01-24 12:41:48', 17.21, 4268491956169254, 133),
(1231, '2018-01-24 13:06:03', 13.39, 30181963913340, 91),
(2913, '2018-01-24 13:17:19', 1691.0, 4761049645711555811, 14),
(3144, '2018-01-24 14:45:28', 18.15, 4681896441519, 27),
(2973, '2018-01-24 16:42:45', 18.59, 3561954487988605, 90),
(2265, '2018-01-24 23:15:56', 5.78, 4539990688484983, 11),
(148, '2018-01-25 00:52:59', 8.7, 5175947111814778, 132),
(273, '2018-01-25 07:41:11', 5.74, 4711773125020499, 20),
(1502, '2018-01-25 08:58:28', 11.35, 376027549341849, 141),
(178, '2018-01-25 11:54:44', 6.4, 5500708021555307, 72),
(1841, '2018-01-25 14:38:43', 10.17, 4165305432349489280, 24),
(1252, '2018-01-25 16:52:06', 10.1, 30181963913340, 80),
(2880, '2018-01-26 05:10:21', 16.54, 584226564303, 13),
(1758, '2018-01-26 06:11:05', 10.12, 3561072557118696, 124),
(938, '2018-01-26 09:30:15', 10.76, 4498002758300, 30),
(969, '2018-01-26 11:32:35', 11.39, 675911140852, 67),
(1730, '2018-01-26 13:10:00', 11.95, 213193946980303, 11),
(2478, '2018-01-26 14:06:47', 17.58, 5361779664174555, 106),
(299, '2018-01-26 20:32:12', 7.0, 4027907156459098, 78),
(2454, '2018-01-26 21:32:03', 17.08, 4743204091443101526, 81),
```

(103, '2018-01-27 00:11:12', 5.87, 344119623920892, 64),

(3261, '2018-01-27 00:56:55', 16.14, 4723783028106084756, 94),

(2636, '2018-01-27 04:07:35', 18.58, 4539990688484983, 49),

(3452, '2018-01-27 08:24:11', 3.08, 3582198969197591, 43),

(3284, '2018-01-27 08:36:58', 15.15, 6500236164848279, 71),

(639, '2018-01-27 14:00:31', 1.88, 6500236164848279, 35),

(2382, '2018-01-27 18:02:40', 1.83, 4268491956169254, 85),

(1495, '2018-01-27 19:10:04', 11.93, 213193946980303, 45),

(142, '2018-01-27 23:35:44', 6.12, 4276466390111, 135),

(1374, '2018-01-28 05:11:34', 13.03, 4268491956169254, 109),

(2978, '2018-01-28 14:38:33', 19.93, 3517111172421930, 71),

(911, '2018-01-28 14:45:12', 10.71, 5135837688671496, 77),

(2686, '2018-01-28 17:01:39', 17.62, 5570600642865857, 82),

(3280, '2018-01-28 18:51:37', 18.64, 6500236164848279, 147),

(744, '2018-01-28 19:30:38', 0.72, 6011987562414062, 36),

(231, '2018-01-28 20:44:10', 6.52, 4279104135293225293, 44),

(2660, '2018-01-28 21:41:42', 18.8, 4027907156459098, 105),

(3168, '2018-01-29 02:00:47', 18.65, 4498002758300, 90),

(2852, '2018-01-29 02:49:08', 18.19, 4723783028106084756, 40),

(549, '2018-01-29 02:51:23', 3.68, 3561072557118696, 140),

(2115, '2018-01-29 05:15:42', 3.54, 4962915017023706562, 45),

(1914, '2018-01-29 06:32:49', 10.24, 3517111172421930, 49),

(536, '2018-01-29 07:01:26', 2.65, 4150721559116778, 5),

(681, '2018-01-29 07:09:51', 2.46, 4723783028106084756, 138),

```
    (1798, '2018-01-29 11:06:43', 14.19, 4165305432349489280, 59),
    (356, '2018-01-29 11:52:48', 5.14, 4761049645711555811, 131),
    (2014, '2018-01-29 12:54:56', 3.93, 4743204091443101526, 142),
    (2029, '2018-01-29 13:38:21', 2.06, 3561072557118696, 22),
    (3365, '2018-01-29 14:09:08', 2.37, 5135837688671496, 92),
(2473, '2018-01-29 17:25:24', 17.99, 3561072557118696, 34),
    (1020, '2018-01-29 19:47:14', 10.25, 5175947111814778, 91),
    (2076, '2018-01-29 22:49:11', 5.33, 4506405265172173, 45),
    (2664, '2018-01-29 23:55:06', 16.73, 584226564303, 10),
    (1486, '2018-01-30 02:32:22', 10.15, 3561072557118696, 87),
    (2177, '2018-01-30 04:24:37', 2.72, 30181963913340, 133),
    (2782, '2018-01-30 04:52:53', 17.05, 5570600642865857, 147),
    (2646, '2018-01-30 06:49:12', 15.97, 4279104135293225293, 82),
    (1027, '2018-01-30 09:15:16', 10.53, 3561072557118696, 79),
    (2047, '2018-01-30 14:21:12', 2.86, 3535651398328201, 32),
    (2132, '2018-01-30 14:21:48', 4.15, 4188164051171486, 85),
    (3205, '2018-01-30 16:34:45', 16.91, 3517111172421930, 112),
    (26, '2018-01-30 16:48:54', 4.04, 376027549341849, 74),
    (1243, '2018-01-30 17:53:56', 10.85, 4279104135293225293, 64),
    (1415, '2018-01-30 18:31:00', 1177.0, 4319653513507, 64),
    (2186, '2018-01-30 22:14:39', 3.65, 4159836738768855913, 102),
    (3126, '2018-01-31 00:51:40', 16.43, 4188164051171486, 48),
    (3310, '2018-01-31 01:24:31', 15.99, 4159836738768855913, 105),
    (1080, '2018-01-31 02:09:59', 10.09, 4586962917519654607, 118),
```

```
(647, '2018-01-31 05:46:43', 2.75, 4319653513507, 81),
(1729, '2018-01-31 06:01:51', 10.28, 4506405265172173, 26),
(1545, '2018-01-31 09:42:00', 10.49, 3517111172421930, 77),
(2219, '2018-01-31 10:34:42', 4.17, 584226564303, 78),
(2954, '2018-01-31 11:06:22', 18.78, 4711773125020499, 48),
(777, '2018-01-31 13:17:35', 0.58, 4741042733274, 63),
(1405, '2018-01-31 18:25:03', 11.25, 4279104135293225293, 127),
(3286, '2018-01-31 23:44:43', 16.02, 4506405265172173, 44),
(1657, '2018-02-01 08:49:24', 10.26, 4723783028106084756, 115),
(45, '2018-02-01 09:29:56', 6.3, 4539990688484983, 116),
(1024, '2018-02-01 11:05:57', 10.49, 4741042733274, 138),
(899, '2018-02-01 11:15:49', 10.65, 4279104135293225293, 65),
(1279, '2018-02-01 18:37:46', 10.27, 4150721559116778, 145),
(989, '2018-02-01 20:33:09', 11.62, 3582198969197591, 4),
(1516, '2018-02-01 20:44:12', 10.24, 3517111172421930, 51),
(2150, '2018-02-01 21:52:43', 1.54, 4743204091443101526, 43),
(2793, '2018-02-02 03:11:47', 15.95, 4834483169177062, 27),
(2776, '2018-02-02 10:56:03', 15.52, 4644008655884311378, 95),
(1360, '2018-02-02 11:31:33', 10.75, 372414832802279, 135),
(2638, '2018-02-02 16:57:12', 15.27, 4681896441519, 53),
(3491, '2018-02-02 17:43:38', 2.17, 5500708021555307, 64),
(1752, '2018-02-02 17:56:30', 10.15, 501809222273, 127),
(2456, '2018-02-02 18:46:46', 18.64, 4741042733274, 148),
(13, '2018-02-02 21:41:12', 4.29, 6500236164848279, 148),
```

```
(248, '2018-02-03 03:04:53', 5.24, 3582198969197591, 83),
(1715, '2018-02-03 06:57:16', 12.39, 5570600642865857, 69),
(2245, '2018-02-03 07:45:02', 4.53, 5570600642865857, 80),
(1701, '2018-02-03 12:53:24', 10.23, 4761049645711555811, 95),
(488, '2018-02-03 14:33:11', 7.05, 30181963913340, 3),
(638, '2018-02-03 15:45:32', 1.17, 3516952396080247, 19),
(2719, '2018-02-03 16:27:12', 15.98, 4723783028106084756, 114),
(3181, '2018-02-03 16:34:30', 18.53, 3561072557118696, 86),
(3395, '2018-02-03 18:05:39', 1.41, 4866761290278198714, 65),
(1435, '2018-02-03 18:48:33', 13.01, 30181963913340, 75),
(3216, '2018-02-03 19:32:30', 17.94, 3516952396080247, 104),
(2389, '2018-02-04 02:22:17', 4.79, 30063281385429, 27),
(11, '2018-02-04 04:00:18', 6.9, 6500236164848279, 125),
(2011, '2018-02-04 09:22:22', 2.41, 213193946980303, 94),
(1996, '2018-02-04 16:13:14', 2.92, 5361779664174555, 4),
(1981, '2018-02-04 18:44:31', 2.68, 4741042733274, 104),
(810, '2018-02-04 19:02:09', 0.51, 3561954487988605, 75),
(3281, '2018-02-04 20:48:29', 18.11, 4539990688484983, 98),
(2256, '2018-02-04 22:25:14', 5.94, 4268491956169254, 43),
(1505, '2018-02-05 03:47:29', 10.27, 4268491956169254, 17),
(2688, '2018-02-05 08:37:24', 17.85, 180098539019105, 1),
(3063, '2018-02-05 09:39:22', 15.73, 503842928916, 75),
(2086, '2018-02-05 15:54:02', 1.52, 30063281385429, 145),
(1740, '2018-02-05 17:32:22', 10.58, 30181963913340, 39),
```

(3237, '2018-02-05 17:36:46', 17.7, 3535651398328201, 50),

(607, '2018-02-05 19:19:11', 2.09, 4539990688484983, 87),

(1270, '2018-02-05 19:35:31', 10.08, 501879657465, 52),

(1281, '2018-02-05 20:08:01', 10.92, 4027907156459098, 34),

(3297, '2018-02-05 20:43:26', 17.61, 180098539019105, 35),

(881, '2018-02-05 21:59:07', 10.81, 372414832802279, 65),

(1819, '2018-02-05 23:22:08', 10.96, 4159836738768855913, 67),

(1079, '2018-02-06 04:34:49', 10.86, 4586962917519654607, 132),

(3494, '2018-02-06 10:55:49', 2.49, 4761049645711555811, 100),

(372, '2018-02-06 10:58:29', 4.83, 376027549341849, 72),

(611, '2018-02-06 18:29:49', 2.53, 3561072557118696, 102),

(1998, '2018-02-06 20:53:47', 4.58, 180098539019105, 9),

(1603, '2018-02-06 22:44:18', 10.32, 30181963913340, 45),

(2418, '2018-02-07 00:20:11', 5.97, 372414832802279, 104),

(2072, '2018-02-07 01:36:19', 5.64, 4761049645711555811, 63),

(230, '2018-02-07 02:48:38', 7.68, 4263694062533017, 63),

(1115, '2018-02-07 03:45:39', 10.21, 4741042733274, 42),

(1473, '2018-02-07 15:53:22', 10.33, 3561072557118696, 49),

(859, '2018-02-07 23:35:00', 2.87, 501809222273, 70),

(205, '2018-02-08 04:15:41', 7.02, 30063281385429, 96),

(2878, '2018-02-08 05:12:18', 18.32, 4866761290278198714, 57),

(2085, '2018-02-08 08:36:01', 2.48, 501809222273, 85),

(748, '2018-02-08 10:27:37', 3.48, 4962915017023706562, 139),

(3473, '2018-02-08 11:06:06', 2.45, 5361779664174555, 127),

```
(3060, '2018-02-08 12:15:41', 15.39, 675911140852, 135),
(364, '2018-02-08 16:35:44', 7.15, 4268491956169254, 97),
(3102, '2018-02-08 19:56:26', 15.71, 4741042733274, 88),
(121, '2018-02-08 21:10:00', 5.52, 584226564303, 124),
(1927, '2018-02-09 00:54:12', 1.72, 4586962917519654607, 3),
(3436, '2018-02-09 07:38:19', 1.16, 503842928916, 12),
(2409, '2018-02-09 11:38:37', 445.0, 3516952396080247, 112),
(657, '2018-02-09 15:30:03', 1.21, 3582198969197591, 101),
(1265, '2018-02-09 19:25:44', 11.11, 501879657465, 114),
(1142, '2018-02-10 02:01:54', 11.43, 5570600642865857, 68),
(3133, '2018-02-10 03:34:18', 15.25, 4150721559116778, 122),
(2665, '2018-02-10 05:36:19', 15.92, 30142966699187, 126),
(2629, '2018-02-10 10:29:14', 20.01, 3581345943543942, 148),
(1386, '2018-02-10 10:55:31', 10.14, 4150721559116778, 35),
(3313, '2018-02-10 11:17:10', 12.97, 4743204091443101526, 135),
(1556, '2018-02-10 11:56:25', 11.33, 4506405265172173, 10),
(375, '2018-02-10 12:04:36', 7.39, 4268491956169254, 105),
(3418, '2018-02-10 13:26:43', 1.66, 4276466390111, 72),
(3013, '2018-02-10 16:22:57', 17.3, 5570600642865857, 150),
(2801, '2018-02-10 16:43:22', 15.67, 3516952396080247, 25),
(1487, '2018-02-10 18:53:22', 10.74, 213193946980303, 105),
(206, '2018-02-11 00:03:19', 7.51, 501809222273, 25),
(1419, '2018-02-11 02:48:31', 12.98, 3561954487988605, 115),
(1446, '2018-02-11 06:30:43', 10.93, 4962915017023706562, 65),
```

(738, '2018-02-11 08:25:36', 1.26, 3582198969197591, 38),

(2894, '2018-02-11 14:38:30', 15.76, 4723783028106084756, 111),

(556, '2018-02-11 16:04:02', 2.74, 4159836738768855913, 63),

(988, '2018-02-11 17:50:29', 11.46, 4027907156459098, 131),

(1449, '2018-02-11 21:30:00', 10.77, 584226564303, 142),

(2290, '2018-02-11 22:11:53', 4.56, 4761049645711555811, 127),

(2554, '2018-02-11 22:51:15', 16.2, 4539990688484983, 118),

(127, '2018-02-12 03:44:20', 3.69, 4319653513507, 116),

(2126, '2018-02-12 10:31:41', 5.47, 4188164051171486, 98),

(3029, '2018-02-12 15:16:31', 19.8, 5570600642865857, 39),

(430, '2018-02-12 15:26:58', 7.73, 4761049645711555811, 150),

(1146, '2018-02-12 15:41:56', 10.08, 213193946980303, 49),

(3285, '2018-02-12 16:21:37', 16.93, 4711773125020499, 84),

(1833, '2018-02-12 21:29:58', 10.39, 4188164051171486, 66),

(3339, '2018-02-13 01:08:02', 1.5, 5175947111814778, 137),

(3465, '2018-02-13 01:31:17', 2.19, 4539990688484983, 1),

(1347, '2018-02-13 01:43:23', 10.36, 376027549341849, 51),

(3076, '2018-02-13 03:52:37', 17.21, 5297187379298983, 35),

(2926, '2018-02-13 09:24:07', 19.2, 30142966699187, 40),

(1848, '2018-02-13 10:34:23', 10.44, 3516952396080247, 130),

(1708, '2018-02-13 10:35:24', 10.78, 501879657465, 101),

(3155, '2018-02-13 13:36:50', 19.37, 4159836738768855913, 149),

(664, '2018-02-13 20:23:29', 1.92, 4723783028106084756, 133),

(1047, '2018-02-14 01:50:55', 10.24, 4741042733274, 81),

```
(2657, '2018-02-14 04:16:16', 17.77, 30078299053512, 4),
(2967, '2018-02-14 05:56:33', 13.85, 6500236164848279, 44),
(2563, '2018-02-14 06:14:36', 16.13, 4268491956169254, 39),
(2035, '2018-02-14 06:48:22', 4.75, 4159836738768855913, 32),
(490, '2018-02-14 08:06:06', 6.17, 4150721559116778, 97),
(3326, '2018-02-14 09:36:28', 2.73, 4498002758300, 1),
(3282, '2018-02-14 10:44:48', 17.99, 4711773125020499, 121),
(1836, '2018-02-14 11:14:37', 11.38, 3517111172421930, 18),
(2060, '2018-02-14 11:45:02', 3.54, 4159836738768855913, 144),
(195, '2018-02-14 14:16:24', 7.64, 4723783028106084756, 135),
(3038, '2018-02-14 14:49:22', 17.36, 4498002758300, 65),
(3016, '2018-02-14 16:46:34', 15.87, 5297187379298983, 83),
(863, '2018-02-14 17:22:35', 2.91, 4188164051171486, 20),
(371, '2018-02-14 18:59:49', 8.45, 5570600642865857, 4),
(340, '2018-02-14 21:50:14', 5.62, 3561072557118696, 51),
(1042, '2018-02-15 04:55:25', 10.29, 6011987562414062, 129),
(1786, '2018-02-15 10:22:06', 10.0, 30142966699187, 147),
(1714, '2018-02-15 10:24:56', 10.3, 5500708021555307, 92),
(1155, '2018-02-15 14:12:59', 10.92, 4723783028106084756, 11),
(891, '2018-02-15 14:49:29', 10.01, 501809222273, 6),
(3220, '2018-02-15 16:28:19', 17.86, 3581345943543942, 91),
(2891, '2018-02-15 19:33:13', 15.95, 3561072557118696, 118),
(572, '2018-02-15 21:35:19', 3.46, 3535651398328201, 139),
(670, '2018-02-16 02:59:59', 3.69, 213193946980303, 121),
```

(1322, '2018-02-16 05:33:51', 10.19, 4741042733274, 98),

(187, '2018-02-16 06:17:01', 5.83, 30142966699187, 40),

(1693, '2018-02-16 08:32:50', 10.0, 584226564303, 42),

(2830, '2018-02-16 10:00:20', 15.95, 30181963913340, 98),

(727, '2018-02-16 12:15:58', 3.11, 5361779664174555, 13),

(1469, '2018-02-16 12:31:27', 10.5, 4644008655884311378, 37),

(1029, '2018-02-16 13:45:35', 10.48, 5570600642865857, 100),

(3138, '2018-02-16 17:15:38', 16.69, 4681896441519, 67),

(2543, '2018-02-16 18:57:43', 17.63, 584226564303, 63),

(2995, '2018-02-16 20:08:20', 16.08, 501879657465, 15),

(2401, '2018-02-16 20:44:08', 5.87, 5297187379298983, 3),

(842, '2018-02-17 00:18:57', 2.71, 376027549341849, 122),

(2699, '2018-02-17 01:27:19', 1430.0, 5570600642865857, 71),

(1648, '2018-02-17 01:56:56', 10.76, 5500708021555307, 136),

(596, '2018-02-17 02:02:40', 2.57, 4962915017023706562, 47),

(2797, '2018-02-17 05:49:13', 17.93, 376027549341849, 30),

(750, '2018-02-17 05:55:46', 1.27, 3581345943543942, 48),

(3057, '2018-02-17 20:31:30', 18.03, 503842928916, 47),

(2670, '2018-02-17 21:37:21', 17.23, 3582198969197591, 25),

(2344, '2018-02-18 01:13:37', 2.51, 4027907156459098, 20),

(2640, '2018-02-18 07:25:42', 16.38, 3582198969197591, 94),

(34, '2018-02-18 08:19:55', 8.02, 5175947111814778, 67),

(2590, '2018-02-18 08:54:19', 18.3, 344119623920892, 69),

(2862, '2018-02-18 09:33:06', 18.06, 5570600642865857, 88),

```
(3468, '2018-02-18 10:09:01', 1.63, 4962915017023706562, 36),
(1764, '2018-02-18 11:36:44', 10.99, 30181963913340, 79),
(2683, '2018-02-18 13:04:21', 14.91, 501809222273, 103),
(1109, '2018-02-18 17:26:10', 11.73, 4962915017023706562, 2),
(501, '2018-02-18 17:37:23', 3.48, 3516952396080247, 41),
(871, '2018-02-18 17:48:32', 1.7, 180098539019105, 63),
(1928, '2018-02-18 18:28:54', 4.26, 4165305432349489280, 94),
(2630, '2018-02-18 20:43:22', 16.7, 4319653513507, 137),
(1525, '2018-02-18 22:19:41', 10.31, 4681896441519, 75),
(2116, '2018-02-19 00:17:41', 3.64, 376027549341849, 89),
(1748, '2018-02-19 03:12:43', 11.0, 584226564303, 148),
(1246, '2018-02-19 08:50:36', 12.17, 4761049645711555811, 149),
(310, '2018-02-19 10:53:19', 5.36, 30063281385429, 124),
(1679, '2018-02-19 11:31:50', 10.38, 4539990688484983, 70),
(2785, '2018-02-19 12:45:15', 15.59, 3517111172421930, 38),
(1480, '2018-02-19 16:00:43', 1072.0, 3516952396080247, 49),
(2250, '2018-02-19 16:15:50', 5.42, 4644008655884311378, 134),
(1860, '2018-02-19 17:56:39', 12.1, 4498002758300, 49),
(3098, '2018-02-19 22:48:25', 1839.0, 344119623920892, 95),
(1403, '2018-02-20 02:58:23', 10.9, 503842928916, 90),
(2933, '2018-02-20 06:24:25', 18.96, 4165305432349489280, 65),
(347, '2018-02-20 10:58:30', 4.6, 213193946980303, 66),
(1978, '2018-02-20 13:01:42', 4.35, 4539990688484983, 22),
(3166, '2018-02-20 17:03:54', 20.88, 584226564303, 106),
```

```
(2174, '2018-02-20 20:28:14', 3.82, 584226564303, 87),
(1082, '2018-02-20 22:07:53', 11.17, 3561954487988605, 70),
(675, '2018-02-21 00:25:34', 2.81, 4506405265172173, 105),
(3496, '2018-02-21 00:28:54', 2.08, 4962915017023706562, 85),
(1114, '2018-02-21 03:19:03', 10.71, 30142966699187, 118),
(3398, '2018-02-21 05:56:41', 2.88, 30142966699187, 9),
(2861, '2018-02-21 08:21:10', 18.98, 4268491956169254, 23),
(864, '2018-02-21 11:46:38', 3.38, 4165305432349489280, 77),
(2469, '2018-02-21 12:56:11', 15.62, 4962915017023706562, 142),
(2934, '2018-02-21 13:21:13', 15.78, 4761049645711555811, 5),
(22, '2018-02-21 17:59:47', 3.8, 4268491956169254, 120),
(3432, '2018-02-21 20:23:38', 2.31, 344119623920892, 90),
(2295, '2018-02-21 20:40:26', 5.2, 4962915017023706562, 30),
(1213, '2018-02-21 22:24:02', 10.07, 584226564303, 68),
(1137, '2018-02-21 23:56:07', 10.67, 3561954487988605, 77),
(1324, '2018-02-22 01:15:43', 12.35, 3517111172421930, 122),
(2214, '2018-02-22 05:56:02', 3.54, 4279104135293225293, 82),
(2952, '2018-02-22 06:40:21', 19.58, 3561072557118696, 121),
(1818, '2018-02-22 07:59:29', 11.64, 3581345943543942, 41),
(520, '2018-02-22 17:56:10', 7.34, 180098539019105, 146),
(623, '2018-02-22 17:59:19', 0.93, 4681896441519, 93),
(1483, '2018-02-22 22:15:06', 11.11, 4276466390111, 132),
(3447, '2018-02-23 00:41:12', 3.12, 5570600642865857, 10),
(3396, '2018-02-23 10:13:27', 1.26, 4319653513507, 26),
```

```
(3431, '2018-02-23 10:46:28', 2.48, 3561954487988605, 112),
(714, '2018-02-23 12:26:19', 2.63, 4319653513507, 44),
(312, '2018-02-23 13:04:55', 6.96, 4866761290278198714, 23),
(1929, '2018-02-23 13:05:54', 4.08, 4644008655884311378, 78),
(850, '2018-02-23 14:39:43', 0.73, 4539990688484983, 68),
(1141, '2018-02-23 18:49:22', 11.01, 372414832802279, 22),
(2596, '2018-02-23 18:53:29', 17.17, 4150721559116778, 51),
(1939, '2018-02-23 20:00:01', 3.26, 5175947111814778, 50),
(3204, '2018-02-24 00:03:28', 16.59, 4834483169177062, 64),
(624, '2018-02-24 02:18:56', 1.13, 501879657465, 145),
(1298, '2018-02-24 04:59:23', 11.36, 4279104135293225293, 45),
(666, '2018-02-24 07:59:19', 2.9, 5570600642865857, 139),
(2227, '2018-02-24 11:53:11', 1.13, 503842928916, 134),
(1533, '2018-02-24 13:57:21', 10.18, 501809222273, 124),
(605, '2018-02-24 14:48:28', 3.4, 4743204091443101526, 102),
(974, '2018-02-24 19:19:15', 10.16, 4279104135293225293, 2),
(630, '2018-02-24 19:20:58', 3.92, 4150721559116778, 82),
(1309, '2018-02-25 00:52:31', 12.07, 5297187379298983, 85),
(1945, '2018-02-25 01:59:45', 3.83, 5297187379298983, 25),
(3370, '2018-02-25 02:33:57', 2.7, 4268491956169254, 37),
(1356, '2018-02-25 07:37:03', 11.68, 5135837688671496, 107),
(3007, '2018-02-25 07:55:21', 18.14, 3535651398328201, 43),
(3100, '2018-02-25 08:23:37', 15.1, 4834483169177062, 138),
(3371, '2018-02-25 08:52:38', 2.55, 3561072557118696, 20),
```

```
(3154, '2018-02-25 09:36:59', 18.32, 4539990688484983, 75),
(2547, '2018-02-25 10:00:33', 18.03, 4165305432349489280, 61),
(1760, '2018-02-25 11:23:34', 10.49, 3517111172421930, 127),
(2112, '2018-02-25 13:40:42', 2.92, 30063281385429, 43),
(2559, '2018-02-25 16:09:42', 14.91, 4188164051171486, 58),
(2631, '2018-02-25 16:54:01', 18.53, 4165305432349489280, 91),
(3400, '2018-02-25 17:30:07', 2.85, 4743204091443101526, 113),
(708, '2018-02-26 01:52:16', 1.01, 675911140852, 81),
(1272, '2018-02-26 03:00:54', 10.35, 5297187379298983, 87),
(3349, '2018-02-26 03:29:02', 2.14, 4276466390111, 119),
(2402, '2018-02-26 03:56:40', 2.38, 3517111172421930, 81),
(492, '2018-02-26 07:25:47', 7.23, 376027549341849, 66),
(2842, '2018-02-26 07:31:20', 13.46, 4586962917519654607, 21),
(3207, '2018-02-26 09:57:26', 17.87, 4723783028106084756, 4),
(1796, '2018-02-26 10:39:29', 10.21, 3561954487988605, 52),
(52, '2018-02-27 00:02:20', 7.11, 376027549341849, 88),
(3008, '2018-02-27 01:20:09', 19.67, 4268491956169254, 95),
(17, '2018-02-27 05:09:28', 6.35, 4644008655884311378, 2),
(2003, '2018-02-27 06:33:42', 5.3, 503842928916, 102),
(2836, '2018-02-27 08:27:00', 18.52, 675911140852, 6),
(5, '2018-02-27 09:20:29', 6.03, 4150721559116778, 18),
(144, '2018-02-27 12:53:43', 11.36, 501809222273, 136),
(1842, '2018-02-27 15:27:32', 1145.0, 3581345943543942, 33),
(924, '2018-02-28 00:50:21', 10.21, 3517111172421930, 56),
```

```
(751, '2018-02-28 02:06:08', 0.91, 372414832802279, 79),
(625, '2018-02-28 04:36:28', 1.41, 3535651398328201, 101),
(138, '2018-02-28 09:00:15', 6.16, 4539990688484983, 2),
(2183, '2018-02-28 09:13:38', 2.59, 4962915017023706562, 139),
(3394, '2018-02-28 13:56:12', 1.18, 372414832802279, 28),
(1216, '2018-03-01 03:52:18', 11.34, 5500708021555307, 126),
(2424, '2018-03-01 10:05:17', 4.47, 213193946980303, 148),
(455, '2018-03-01 15:32:22', 10.46, 5570600642865857, 120),
(223, '2018-03-01 18:42:20', 6.13, 501879657465, 22),
(2644, '2018-03-01 20:03:47', 15.53, 3582198969197591, 39),
(1005, '2018-03-01 21:29:05', 1119.0, 30078299053512, 19),
(2572, '2018-03-01 23:00:57', 12.65, 5361779664174555, 135),
(433, '2018-03-01 23:17:42', 9.26, 5500708021555307, 121),
(2799, '2018-03-01 23:38:12', 20.22, 4027907156459098, 72),
(2321, '2018-03-02 02:48:43', 3.12, 6011987562414062, 127),
(101, '2018-03-02 03:35:56', 4.4, 30078299053512, 111),
(2277, '2018-03-02 06:42:40', 1.49, 501879657465, 126),
(1791, '2018-03-02 09:24:52', 10.56, 4834483169177062, 124),
(2325, '2018-03-02 11:05:27', 1.92, 213193946980303, 44),
(3336, '2018-03-02 11:42:48', 1.37, 5135837688671496, 63),
(366, '2018-03-02 12:56:48', 6.25, 213193946980303, 67),
(3001, '2018-03-02 12:57:18', 16.54, 5297187379298983, 31),
(2232, '2018-03-02 13:14:30', 1.1, 30063281385429, 76),
(1315, '2018-03-02 17:30:33', 10.19, 5297187379298983, 68),
```

(547, '2018-03-02 17:44:08', 2.05, 584226564303, 19),

(992, '2018-03-02 18:37:24', 10.02, 5500708021555307, 88),

(3118, '2018-03-02 18:53:41', 15.95, 30078299053512, 72),

(3420, '2018-03-02 19:34:07', 1.77, 4711773125020499, 23),

(1397, '2018-03-02 19:40:34', 10.06, 6011987562414062, 19),

(1150, '2018-03-02 23:23:52', 12.42, 4319653513507, 65),

(251, '2018-03-03 02:05:51', 3.49, 3517111172421930, 88),

(3357, '2018-03-03 02:17:54', 2.85, 30142966699187, 138),

(2524, '2018-03-03 05:07:39', 18.03, 180098539019105, 111),

(3238, '2018-03-03 08:42:02', 19.5, 4681896441519, 23),

(2394, '2018-03-03 11:05:28', 5.34, 4268491956169254, 1),

(667, '2018-03-03 11:08:50', 3.69, 5361779664174555, 126),

(3319, '2018-03-03 13:04:51', 18.44, 4276466390111, 90),

(1835, '2018-03-03 14:02:41', 10.28, 30078299053512, 37),

(1663, '2018-03-03 14:14:15', 11.32, 30142966699187, 84),

(3090, '2018-03-03 18:45:59', 17.94, 4498002758300, 130),

(1342, '2018-03-04 00:52:49', 10.85, 4834483169177062, 48),

(3136, '2018-03-04 03:56:11', 15.4, 4741042733274, 67),

(2413, '2018-03-04 04:38:49', 2.02, 4268491956169254, 12),

(622, '2018-03-04 06:20:22', 3.07, 5500708021555307, 81),

(332, '2018-03-04 07:04:06', 2.23, 5135837688671496, 48),

(420, '2018-03-04 10:13:36', 4.53, 213193946980303, 116),

(2155, '2018-03-04 12:58:48', 1.94, 584226564303, 117),

(70, '2018-03-04 13:32:00', 5.7, 30181963913340, 150),

(2789, '2018-03-04 15:50:53', 1534.0, 30181963913340, 118),

(827, '2018-03-04 17:41:10', 2.98, 5500708021555307, 141),

(787, '2018-03-04 19:14:58', 3.42, 5175947111814778, 129),

(1494, '2018-03-04 20:41:27', 12.29, 4159836738768855913, 99),

(610, '2018-03-04 20:48:10', 1.13, 376027549341849, 109),

(1211, '2018-03-04 22:35:20', 11.32, 4644008655884311378, 100),

(2050, '2018-03-04 23:28:50', 3.64, 584226564303, 79),

(3025, '2018-03-05 00:15:21', 15.73, 584226564303, 31),

(304, '2018-03-05 05:48:41', 5.98, 4743204091443101526, 75),

(2464, '2018-03-05 06:18:35', 15.48, 4681896441519, 85),

(2246, '2018-03-05 06:58:53', 2.05, 4539990688484983, 116),

(2486, '2018-03-05 07:34:15', 16.58, 4319653513507, 135),

(2451, '2018-03-05 08:26:08', 1617.0000000000002, 5570600642865857, 4),

(2928, '2018-03-05 10:19:07', 19.69, 4276466390111, 139),

(508, '2018-03-05 11:06:17', 5.01, 30063281385429, 51),

(580, '2018-03-05 11:39:50', 3.19, 4263694062533017, 111),

(3403, '2018-03-05 12:32:16', 1.68, 4268491956169254, 43),

(2327, '2018-03-05 14:58:09', 4.54, 4150721559116778, 39),

(2856, '2018-03-05 15:43:47', 17.06, 4866761290278198714, 142),

(2172, '2018-03-06 02:12:47', 3.83, 4539990688484983, 120),

(1448, '2018-03-06 03:48:18', 10.94, 4188164051171486, 66),

(219, '2018-03-06 04:01:25', 4.1, 4866761290278198714, 127),

(2005, '2018-03-06 05:30:11', 5.57, 3561072557118696, 37),

(2840, '2018-03-06 07:18:09', 1334.0, 4319653513507, 87),

```
(2602, '2018-03-06 11:34:55', 17.3, 6500236164848279, 66),
(438, '2018-03-06 12:39:14', 5.62, 3535651398328201, 46),
(2750, '2018-03-06 13:24:44', 18.8, 4188164051171486, 132),
(1903, '2018-03-06 16:09:53', 10.82, 344119623920892, 65),
(1030, '2018-03-06 20:15:28', 12.6, 4279104135293225293, 59),
(1611, '2018-03-07 00:05:21', 12.92, 344119623920892, 126),
(2911, '2018-03-07 02:46:57', 15.55, 503842928916, 77),
(1773, '2018-03-07 02:52:30', 10.92, 503842928916, 54),
(185, '2018-03-07 03:03:21', 6.46, 5500708021555307, 118),
(2368, '2018-03-07 04:00:16', 1.41, 180098539019105, 50),
(1918, '2018-03-07 05:49:44', 12.91, 4711773125020499, 140),
(353, '2018-03-07 09:22:27', 3.97, 4159836738768855913, 146),
(746, '2018-03-07 16:45:37', 2.88, 4319653513507, 84),
(283, '2018-03-07 17:14:46', 4.77, 4268491956169254, 101),
(2027, '2018-03-07 18:43:44', 1.26, 3561954487988605, 115),
(1276, '2018-03-07 19:07:30', 10.18, 4498002758300, 117),
(1133, '2018-03-07 19:14:00', 10.37, 4498002758300, 45),
(1406, '2018-03-07 22:35:31', 11.16, 501879657465, 117),
(2013, '2018-03-08 00:32:59', 4.89, 6500236164848279, 114),
(1023, '2018-03-08 04:08:16', 10.21, 501879657465, 32),
(3428, '2018-03-08 07:46:00', 1.37, 5175947111814778, 48),
(1771, '2018-03-08 09:02:09', 10.84, 4711773125020499, 61),
(1999, '2018-03-08 09:47:40', 2.99, 5297187379298983, 110),
(73, '2018-03-08 15:38:27', 8.19, 4027907156459098, 118),
```

```
(1582, '2018-03-08 16:31:22', 11.22, 4165305432349489280, 134),
(641, '2018-03-08 17:36:48', 3.23, 503842928916, 150),
(2239, '2018-03-08 18:56:27', 1.32, 4586962917519654607, 44),
(2573, '2018-03-08 20:11:49', 20.71, 30063281385429, 19),
(2620, '2018-03-08 20:28:05', 18.21, 4586962917519654607, 101),
(3097, '2018-03-09 00:13:59', 18.11, 6011987562414062, 70),
(2030, '2018-03-09 00:31:16', 3.63, 3517111172421930, 5),
(2226, '2018-03-09 02:18:29', 3.67, 4681896441519, 85),
(2117, '2018-03-09 03:31:09', 5.02, 4506405265172173, 66),
(2205, '2018-03-09 03:59:06', 2.04, 372414832802279, 108),
(38, '2018-03-09 04:51:38', 389.0, 3581345943543942, 76),
(1866, '2018-03-09 05:28:06', 11.77, 4027907156459098, 126),
(3176, '2018-03-09 05:47:19', 12.15, 344119623920892, 99),
(2449, '2018-03-09 11:21:07', 5.45, 30142966699187, 117),
(3197, '2018-03-09 11:41:54', 15.72, 501809222273, 68),
(463, '2018-03-09 12:18:16', 8.63, 4741042733274, 28),
(995, '2018-03-09 13:47:25', 10.18, 4506405265172173, 103),
(1032, '2018-03-09 13:55:45', 10.72, 30078299053512, 26),
(2609, '2018-03-09 14:53:23', 16.07, 3517111172421930, 117),
(1429, '2018-03-09 14:55:31', 10.54, 4834483169177062, 60),
(1149, '2018-03-09 16:30:56', 10.92, 4962915017023706562, 65),
(2822, '2018-03-09 23:28:44', 16.38, 213193946980303, 38),
(1366, '2018-03-09 23:41:26', 11.7, 3561954487988605, 2),
(290, '2018-03-10 00:20:50', 8.15, 5570600642865857, 44),
```

```
(269, '2018-03-10 00:47:29', 8.45, 4586962917519654607, 5),
(472, '2018-03-10 02:22:01', 5.09, 4159836738768855913, 135),
(1699, '2018-03-10 03:53:17', 10.86, 30181963913340, 97),
(586, '2018-03-10 05:41:14', 1.68, 5175947111814778, 69),
(373, '2018-03-10 06:05:39', 4.69, 4159836738768855913, 92),
(929, '2018-03-10 08:52:09', 13.53, 675911140852, 109),
(2270, '2018-03-10 13:05:20', 4.65, 30063281385429, 130),
(201, '2018-03-10 13:26:19', 6.79, 4498002758300, 115),
(2323, '2018-03-10 19:52:29', 4.13, 675911140852, 64),
(575, '2018-03-10 23:00:35', 0.76, 3517111172421930, 2),
(2863, '2018-03-10 23:42:01', 18.88, 4276466390111, 14),
(3081, '2018-03-11 00:57:07', 18.36, 5570600642865857, 48),
(960, '2018-03-11 03:59:13', 10.78, 213193946980303, 98),
(1646, '2018-03-11 04:02:13', 10.13, 5361779664174555, 145),
(1036, '2018-03-11 04:40:09', 12.67, 4962915017023706562, 9),
(1203, '2018-03-11 09:33:38', 10.08, 4866761290278198714, 34),
(1467, '2018-03-11 11:44:49', 10.2, 3517111172421930, 49),
(925, '2018-03-11 15:47:35', 10.79, 180098539019105, 142),
(418, '2018-03-11 17:34:44', 7.19, 5570600642865857, 109),
(3267, '2018-03-11 19:37:02', 13.57, 4319653513507, 70),
(1263, '2018-03-11 21:52:20', 10.4, 4276466390111, 39),
(3438, '2018-03-11 22:09:12', 3.35, 30063281385429, 78),
(3268, '2018-03-11 22:39:53', 18.05, 3535651398328201, 38),
(2610, '2018-03-12 00:44:01', 1530.0, 501879657465, 20),
```

(1566, '2018-03-12 01:00:24', 10.1, 372414832802279, 89),

(2682, '2018-03-12 04:06:40', 15.13, 5361779664174555, 143),

(2081, '2018-03-12 05:07:34', 3.79, 675911140852, 77),

(2749, '2018-03-12 05:29:57', 22.49, 6011987562414062, 147),

(1242, '2018-03-12 06:01:02', 10.69, 4276466390111, 73),

(3273, '2018-03-12 06:23:31', 17.37, 6500236164848279, 50),

(1194, '2018-03-12 08:57:11', 11.55, 5175947111814778, 88),

(2021, '2018-03-12 09:08:18', 1.65, 372414832802279, 13),

(2990, '2018-03-12 13:06:35', 20.53, 4711773125020499, 42),

(3314, '2018-03-12 14:09:43', 20.52, 5135837688671496, 135),

(2009, '2018-03-12 17:16:34', 3.08, 372414832802279, 76),

(1785, '2018-03-12 19:29:00', 10.14, 213193946980303, 143),

(807, '2018-03-12 20:04:21', 2.41, 180098539019105, 60),

(3368, '2018-03-12 21:31:02', 1.62, 501809222273, 70),

(1470, '2018-03-12 22:14:04', 10.84, 4150721559116778, 21),

(2287, '2018-03-13 01:03:47', 2.69, 376027549341849, 98),

(731, '2018-03-13 03:50:55', 2.97, 3517111172421930, 144),

(966, '2018-03-13 10:35:49', 10.0, 5500708021555307, 63),

(2358, '2018-03-13 11:42:46', 1.88, 4723783028106084756, 28),

(2615, '2018-03-13 18:40:48', 15.06, 213193946980303, 119),

(1064, '2018-03-13 20:05:06', 10.46, 344119623920892, 5),

(3413, '2018-03-13 22:45:15', 2.76, 4586962917519654607, 83),

(1576, '2018-03-14 13:39:02', 11.33, 4761049645711555811, 143),

(2254, '2018-03-14 16:12:53', 5.16, 4723783028106084756, 128),

(3383, '2018-03-14 16:32:44', 3.1, 4741042733274, 75),

(999, '2018-03-14 18:42:06', 10.39, 5175947111814778, 107),

(1864, '2018-03-14 20:45:55', 11.79, 30181963913340, 13),

(2790, '2018-03-14 21:33:29', 18.35, 501809222273, 102),

(1573, '2018-03-14 22:20:54', 10.02, 6011987562414062, 18),

(33, '2018-03-14 23:24:08', 7.22, 4268491956169254, 142),

(1503, '2018-03-15 01:47:05', 10.18, 584226564303, 128),

(164, '2018-03-15 03:52:08', 3.64, 4681896441519, 106),

(2018, '2018-03-15 06:35:54', 4.09, 4268491956169254, 126),

(2883, '2018-03-15 06:41:30', 15.24, 675911140852, 33),

(1258, '2018-03-15 07:30:36', 11.12, 213193946980303, 87),

(2532, '2018-03-15 10:11:12', 18.01, 4962915017023706562, 134),

(926, '2018-03-15 13:06:10', 10.9, 6011987562414062, 141),

(2410, '2018-03-15 18:24:19', 3.2, 4159836738768855913, 126),

(2293, '2018-03-16 01:07:59', 2.29, 4506405265172173, 104),

(2388, '2018-03-16 02:04:54', 4.2, 372414832802279, 54),

(271, '2018-03-16 04:29:47', 3.02, 4962915017023706562, 10),

(653, '2018-03-16 04:32:53', 2.7, 4188164051171486, 113),

(235, '2018-03-16 05:07:18', 6.45, 4681896441519, 74),

(2090, '2018-03-16 06:18:36', 2.57, 4279104135293225293, 147),

(2297, '2018-03-16 10:19:31', 1.97, 376027549341849, 87),

(1769, '2018-03-16 13:00:13', 10.09, 30078299053512, 68),

(62, '2018-03-16 16:57:27', 5.59, 4276466390111, 24),

(361, '2018-03-16 21:30:32', 9.72, 5570600642865857, 98),

(2236, '2018-03-16 21:38:11', 4.09, 5175947111814778, 40),

(141, '2018-03-17 00:58:15', 2.73, 4188164051171486, 34),

(826, '2018-03-17 01:16:34', 3.14, 4681896441519, 120),

(1255, '2018-03-17 08:07:52', 10.36, 30142966699187, 44),

(1958, '2018-03-17 10:10:48', 3.45, 4741042733274, 93),

(2136, '2018-03-17 14:17:13', 5.33, 3535651398328201, 28),

(2679, '2018-03-17 16:30:11', 17.59, 5175947111814778, 145),

(2062, '2018-03-17 16:30:43', 5.22, 213193946980303, 8),

(2053, '2018-03-17 18:22:07', 2.56, 4319653513507, 77),

(1433, '2018-03-17 21:08:36', 13.27, 4761049645711555811, 76),

(1013, '2018-03-17 21:41:05', 12.21, 4711773125020499, 134),

(1390, '2018-03-18 01:20:51', 12.13, 6500236164848279, 4),

(1911, '2018-03-18 07:23:02', 10.47, 503842928916, 51),

(1972, '2018-03-18 11:51:52', 4.07, 501879657465, 24),

(2684, '2018-03-18 12:29:39', 18.28, 372414832802279, 89),

Implementation (Coding)

Identifying Outliers using Standard Deviation

```
# initial imports
import pandas as pd
import numpy as np
import random
from sqlalchemy import create_engine
from numpy import mean
from numpy import std
from numpy import percentile
import plotly.express as px
```

```
# create a connection to the database
engine = create_engine("postgresql://postgres:Istay@10314@localhost:5432/fraud_detection")

# Loading data from the database

def execute_query(query):

    transaction_df = pd.read_sql(sql=query, con=engine, index_col='date', parse_dates='date')

    return transaction_df


# Loading data of daily transactions from jan to jun 2018 for card holder 25
query = f'SELECT a.id, a.name, b.card, c.date, c.amount, e.name as "category" \
        FROM public.card_holder a, public.credit_card b, public.transaction c, public.merchant d,
public.merchant_category e \
        WHERE a.id = b.id_card_holder AND b.card=c.card AND c.id_merchant=d.id AND
d.id_merchant_category=e.id'

transaction_df = execute_query(query)
transaction_df.head()
```

2018-01-01 21:35:10	13	John Martin	3561954487988605	6.22	food truck
2018-01-01 21:43:12	13	John Martin	5135837688671496	3.83	bar
2018-01-01 22:41:21	10	Matthew Gutierrez	213193946980303	9.61	food truck
2018-01-01 23:13:30	4	Danielle Green	4263694062533017	19.03	pub
2018-01-01 23:15:10	18	Malik Carlson	4498002758300	2.95	restaurant

```
# code a function to identify outliers based on standard deviation
# calculate summary statistics
data_mean, data_std = mean(transaction_df['amount']), std(transaction_df['amount'])

# identify outliers
```

```
cut_off = data_std * 3

lower, upper = data_mean - cut_off, data_mean + cut_off

# identify outliers
outliers = [x for x in transaction_df['amount'] if x < lower or x > upper]

print('Identified outliers: %d' % len(outliers))

# remove outliers
outliers_removed = [x for x in transaction_df['amount'] if x >= lower and x <= upper]

print('Non-outlier observations: %d' % len(outliers_removed))

transaction_df['outlier'] = (transaction_df['amount'] > upper) | (transaction_df['amount'] < lower)

outlier = transaction_df[transaction_df['outlier']==True]
outlier
```

```
Identified outliers: 77
Non-outlier observations: 3423
```

2018-01-02 23:27:46	12	Megan Price	501879657465	1031.0	restaurant	True
2018-01-04 03:05:18	7	Sean Taylor	3516952396080247	1685.0	food truck	True
2018-01-08 02:34:32	6	Beth Hernandez	3581345943543942	1029.0	bar	True
2018-01-22 08:07:03	16	Crystal Clark	5570600642865857	1131.0	restaurant	True
2018-01-23 06:29:37	12	Megan Price	501879657465	1678.0	pub	True
2018-01-24 13:17:19	1	Robert Johnson	4761049645711555811	1691.0	coffee shop	True
2018-01-30 18:31:00	25	Nancy Contreras	4319653513507	1177.0	restaurant	True
2018-02-17 01:27:19	16	Crystal Clark	5570600642865857	1430.0	restaurant	True

2018-02-19 16:00:43	7	Sean Taylor	3516952396080247	1072.0	food truck	True
2018-02-19 22:48:25	18	Malik Carlson	344119623920892	1839.0	restaurant	True
2018-02-27 15:27:32	6	Beth Hernandez	3581345943543942	1145.0	bar	True
2018-03-01 21:29:05	3	Elizabeth Sawyer	30078299053512	1119.0	pub	True
2018-03-04 15:50:53	9	Laurie Gibbs	30181963913340	1534.0	coffee shop	True
2018-03-05 08:26:08	16	Crystal Clark	5570600642865857	1617.0	bar	True
2018-03-06 07:18:09	25	Nancy Contreras	4319653513507	1334.0	bar	True
2018-03-12 00:44:01	12	Megan Price	501879657465	1530.0	coffee shop	True
2018-03-20 10:19:25	12	Megan Price	501879657465	852.0	pub	True
2018-03-20 13:05:54	24	Stephanie Dalton	30142966699187	1011.0	bar	True
2018-03-26 07:41:59	9	Laurie Gibbs	30181963913340	1009.0	coffee shop	True
2018-04-03 03:23:37	18	Malik Carlson	344119623920892	1077.0	restaurant	True
2018-04-08 06:03:50	25	Nancy Contreras	4319653513507	1063.0	pub	True
2018-04-18 23:23:29	7	Sean Taylor	3516952396080247	1086.0	coffee shop	True
2018-04-21 19:41:51	6	Beth Hernandez	3581345943543942	2108.0	coffee shop	True
2018-05-08 13:21:01	24	Stephanie Dalton	30142966699187	1901.0	restaurant	True
2018-05-13 06:31:20	25	Nancy Contreras	4319653513507	1046.0	food truck	True
2018-05-29 02:55:08	16	Crystal Clark	5570600642865857	1203.0	food truck	True
2018-06-03 20:02:28	18	Malik Carlson	344119623920892	1814.0	pub	True
2018-06-04 03:46:15	25	Nancy Contreras	4319653513507	1162.0	pub	True
2018-06-06 21:50:17	25	Nancy Contreras	4319653513507	749.0	restaurant	True
2018-06-10 04:54:27	9	Laurie Gibbs	30181963913340	1795.0	pub	True
2018-09-04 01:35:39	1	Robert Johnson	4761049645711555811	1790.0	coffee shop	True

2018-09-06 08:28:55	1	Robert Johnson	4761049645711555811	1017.0	bar	True
2018-09-06 21:55:02	1	Robert Johnson	4761049645711555811	1056.0	restaurant	True
2018-09-10 22:49:41	18	Malik Carlson	344119623920892	1176.0	restaurant	True
2018-09-11 15:16:47	6	Beth Hernandez	3581345943543942	1856.0	food truck	True
2018-09-23 19:20:23	12	Megan Price	501879657465	1075.0	pub	True
2018-09-25 23:23:21	9	Laurie Gibbs	30181963913340	1095.0	food truck	True
2018-09-26 08:48:40	1	Robert Johnson	4761049645711555811	1060.0	restaurant	True
2018-10-07 14:40:34	3	Elizabeth Sawyer	30078299053512	757.0	bar	True
2018-10-07 18:29:20	9	Laurie Gibbs	30181963913340	1179.0	pub	True
2018-10-19 01:07:37	3	Elizabeth Sawyer	30078299053512	1053.0	restaurant	True
2018-11-13 17:07:25	16	Crystal Clark	5570600642865857	1911.0	restaurant	True
2018-11-17 05:30:43	18	Malik Carlson	344119623920892	1769.0	food truck	True
2018-11-20 05:24:28	3	Elizabeth Sawyer	30078299053512	1054.0	bar	True
2018-11-25 20:44:07	12	Megan Price	501879657465	1123.0	bar	True
2018-11-27 15:36:05	12	Megan Price	501879657465	1802.0	bar	True
2018-11-27 17:20:29	6	Beth Hernandez	3581345943543942	1279.0	restaurant	True
2018-11-27 17:27:34	1	Robert Johnson	4761049645711555811	1660.0	pub	True
2018-12-03 02:38:52	16	Crystal Clark	5570600642865857	1014.0	restaurant	True
2018-12-07 07:22:03	1	Robert Johnson	4761049645711555811	1894.0	bar	True
2018-12-13 12:09:58	18	Malik Carlson	344119623920892	1154.0	restaurant	True
2018-12-13 15:51:59	7	Sean Taylor	3516952396080247	2249.0	food truck	True
2018-12-14 08:51:41	12	Megan Price	501879657465	748.0	pub	True
2018-12-18 13:33:37	25	Nancy Contreras	4319653513507	1074.0	coffee shop	True
2018-12-18 17:20:33	7	Sean Taylor	3516952396080247	1296.0	bar	True
2018-12-19 16:10:03	9	Laurie Gibbs	30181963913340	1724.0	pub	True

2018-12-21 09:56:32	24	Stephanie Dalton	30142966699187	1301.0	pub	True
2018-12-24 15:55:06	16	Crystal Clark	5570600642865857	1634.0	pub	True
2018-12-25 19:10:42	24	Stephanie Dalton	30142966699187	1035.0	pub	True
2018-12-30 23:23:09	1	Robert Johnson	4761049645711555811	1033.0	pub	True

77 rows × 6 columns

```
# find anomalous transactions for 3 random card holders
import datetime
start_time = datetime.time(7,0,0)
end_time = datetime.time(9,0,0)

anomalous_transactions = outlier.between_time(start_time, end_time).sort_values('amount',
ascending=False)

px.scatter(anomalous_transactions, x='name', y='amount', color='category')
```

Identifying Outliers Using Interquartile Range

```
# code a function to identify outliers based on interquartile range
# calculate interquartile range
q25, q75 = percentile(transaction_df['amount'], 25), percentile(transaction_df['amount'], 75)
iqr = q75 - q25
print('Percentiles: 25th=%.3f, 75th=%.3f, IQR=%.3f' % (q25, q75, iqr))
# calculate the outlier cutoff
cut_off = iqr * 1.5
lower, upper = q25 - cut_off, q75 + cut_off
# identify outliers
outliers_2 = [x for x in transaction_df['amount'] if x < lower or x > upper]
print('Identified outliers: %d' % len(outliers_2))
# remove outliers
outliers_removed_2 = [x for x in transaction_df['amount'] if x >= lower and x <= upper]
print('Non-outlier observations: %d' % len(outliers_removed_2))

transaction_df['outlier'] = (transaction_df['amount'] > upper) | (transaction_df['amount'] < lower)

outlier_2 = transaction_df[transaction_df['outlier']==True]
outlier_2
```

```
Percentiles: 25th=3.735, 75th=14.648, IQR=10.913
Identified outliers: 110
Non-outlier observations: 3390
```

2018-01-02 23:27:46	12	Megan Price	501879657465	1031.0	restaurant	True
2018-01-04 03:05:18	7	Sean Taylor	3516952396080247	1685.0	food truck	True
2018-01-07 01:10:54	18	Malik Carlson	344119623920892	175.0	pub	True
2018-01-08 02:34:32	6	Beth Hernandez	3581345943543942	1029.0	bar	True
2018-01-08 11:15:36	18	Malik Carlson	344119623920892	333.0	restaurant	True
2018-01-11 13:20:31	16	Crystal Clark	5570600642865857	229.0	pub	True
2018-01-22 08:07:03	16	Crystal Clark	5570600642865857	1131.0	restaurant	True
2018-01-23 06:29:37	12	Megan Price	501879657465	1678.0	pub	True
2018-01-24 13:17:19	1	Robert Johnson	4761049645711555811	1691.0	coffee shop	True
2018-01-30 18:31:00	25	Nancy Contreras	4319653513507	1177.0	restaurant	True
2018-02-09 11:38:37	7	Sean Taylor	3516952396080247	445.0	bar	True
2018-02-17 01:27:19	16	Crystal Clark	5570600642865857	1430.0	restaurant	True
2018-02-19 16:00:43	7	Sean Taylor	3516952396080247	1072.0	food truck	True
2018-02-19 22:48:25	18	Malik Carlson	344119623920892	1839.0	restaurant	True
2018-02-27 15:27:32	6	Beth Hernandez	3581345943543942	1145.0	bar	True
2018-03-01 21:29:05	3	Elizabeth Sawyer	30078299053512	1119.0	pub	True
2018-03-04 15:50:53	9	Laurie Gibbs	30181963913340	1534.0	coffee shop	True
2018-03-05 08:26:08	16	Crystal Clark	5570600642865857	1617.0	bar	True
2018-03-06 07:18:09	25	Nancy Contreras	4319653513507	1334.0	bar	True
2018-03-09 04:51:38	6	Beth Hernandez	3581345943543942	389.0	restaurant	True
2018-03-12 00:44:01	12	Megan Price	501879657465	1530.0	coffee shop	True
2018-03-20 10:19:25	12	Megan Price	501879657465	852.0	pub	True
2018-03-20 13:05:54	24	Stephanie Dalton	30142966699187	1011.0	bar	True
2018-03-26 07:41:59	9	Laurie Gibbs	30181963913340	1009.0	coffee shop	True
2018-04-01 07:17:21	25	Nancy Contreras	4319653513507	100.0	coffee shop	True
2018-04-03 03:23:37	18	Malik Carlson	344119623920892	1077.0	restaurant	True
2018-04-08 06:03:50	25	Nancy Contreras	4319653513507	1063.0	pub	True
2018-04-09 10:24:32	1	Robert Johnson	4761049645711555811	283.0	restaurant	True
2018-04-09 18:28:25	25	Nancy Contreras	4319653513507	269.0	restaurant	True
2018-04-10 06:08:01	7	Sean Taylor	3516952396080247	543.0	food truck	True
2018-10-11 23:29:33	3	Elizabeth Sawyer	30078299053512	206.0	restaurant	True
2018-10-16 13:27:33	1	Robert Johnson	4761049645711555811	484.0	food truck	True
2018-10-19 01:07:37	3	Elizabeth Sawyer	30078299053512	1053.0	restaurant	True
2018-10-19 12:32:37	16	Crystal Clark	5570600642865857	178.0	food truck	True
2018-10-23 22:47:13	16	Crystal Clark	5570600642865857	393.0	food truck	True
2018-10-28 02:12:58	25	Nancy Contreras	4319653513507	137.0	pub	True
2018-11-13 05:58:47	24	Stephanie Dalton	30142966699187	466.0	bar	True
2018-11-13 17:07:25	16	Crystal Clark	5570600642865857	1911.0	restaurant	True
2018-11-17 05:30:43	18	Malik Carlson	344119623920892	1769.0	food truck	True
2018-11-20 05:24:28	3	Elizabeth Sawyer	30078299053512	1054.0	bar	True

2018-11-23 09:08:05	12	Megan Price	501879657465	233.0	restaurant	True
2018-11-25 20:44:07	12	Megan Price	501879657465	1123.0	bar	True
2018-11-26 20:54:39	1	Robert Johnson	4761049645711555811	267.0	food truck	True
2018-11-27 15:36:05	12	Megan Price	501879657465	1802.0	bar	True
2018-11-27 17:20:29	6	Beth Hernandez	3581345943543942	1279.0	restaurant	True
2018-11-27 17:27:34	1	Robert Johnson	4761049645711555811	1660.0	pub	True
2018-12-03 02:38:52	16	Crystal Clark	5570600642865857	1014.0	restaurant	True
2018-12-05 19:24:27	9	Laurie Gibbs	30181963913340	57.0	bar	True
2018-12-07 07:22:03	1	Robert Johnson	4761049645711555811	1894.0	bar	True
2018-12-13 12:09:58	18	Malik Carlson	344119623920892	1154.0	restaurant	True
2018-12-13 15:51:59	7	Sean Taylor	3516952396080247	2249.0	food truck	True
2018-12-14 08:51:41	12	Megan Price	501879657465	748.0	pub	True
2018-12-18 13:33:37	25	Nancy Contreras	4319653513507	1074.0	coffee shop	True
2018-12-18 17:20:33	7	Sean Taylor	3516952396080247	1296.0	bar	True
2018-12-19 16:10:03	9	Laurie Gibbs	30181963913340	1724.0	pub	True
2018-12-21 09:56:32	24	Stephanie Dalton	30142966699187	1301.0	pub	True
2018-12-24 15:55:06	16	Crystal Clark	5570600642865857	1634.0	pub	True
2018-12-25 19:10:42	24	Stephanie Dalton	30142966699187	1035.0	pub	True
2018-12-28 16:20:31	3	Elizabeth Sawyer	30078299053512	313.0	pub	True
2018-12-30 23:23:09	1	Robert Johnson	4761049645711555811	1033.0	pub	True

110 rows × 6 columns

```
# find anomalous transactions for 3 random card holders
anomalous_transactions2 = outlier_2.between_time(start_time, end_time).sort_values('amount',
ascending=False)
anomalous_transactions2
```

2018-12-07 07:22:03	1	Robert Johnson	4761049645711555811	1894.0	bar	True

2018-03-05 08:26:08	16	Crystal Clark	5570600642865857	1617.0	bar	True
2018-03-06 07:18:09	25	Nancy Contreras	4319653513507	1334.0	bar	True
2018-01-22 08:07:03	16	Crystal Clark	5570600642865857	1131.0	restaurant	True
2018-09-26 08:48:40	1	Robert Johnson	4761049645711555811	1060.0	restaurant	True
2018-09-06 08:28:55	1	Robert Johnson	4761049645711555811	1017.0	bar	True
2018-03-26 07:41:59	9	Laurie Gibbs	30181963913340	1009.0	coffee shop	True
2018-12-14 08:51:41	12	Megan Price	501879657465	748.0	pub	True
2018-04-01 07:17:21	25	Nancy Contreras	4319653513507	100.0	coffee shop	True

```
px.scatter(anomalous_transactions2, x='name', y='amount', color='category')
```

For Outlier calculation using standard deviation, results in 77 records whereas using Interquartile range results in 110 records. There seems to be fraudulent transactions in Bar category wherein amount spent between 7-9 AM in the Bar

Visual Data Analysis of Fraudulent Transactions

```
# initial imports
import pandas as pd
import calendar
import plotly.express as px
import hvplot.pandas
from sqlalchemy import create_engine
import datetime
```

In [51]:

```
# create a connection to the database
engine = create_engine("postgresql://postgres:Istay@10314@localhost:5432/fraud_detection")
```

In [52]:

```
# Loading data from the database

def execute_query(query):

    transaction_df = pd.read_sql(sql=query, con=engine, index_col='date', parse_dates='date')

    return transaction_df

def fraud_transactions(df):
    start_time = datetime.time(7,0,0)
    end_time = datetime.time(9,0,0)
    return df.between_time(start_time, end_time).sort_values('amount', ascending=False)
```

In [58]:

```
# Loading data for card holder 2 and 18 from the database
query = f'SELECT a.id, a.name, b.card, c.date, c.amount, e.name as "category" \
        FROM public.card_holder a, public.credit_card b, public.transaction c, public.merchant d,
public.merchant_category e \
        WHERE a.id = b.id_card_holder AND b.card=c.card AND c.id_merchant=d.id AND
d.id_merchant_category=e.id AND b.card=c.card'

transaction_df = execute_query(query)

suspect_df = fraud_transactions(transaction_df).head(100)
suspect_df
```

2018-12-07 07:22:03	1	Robert Johnson	4761049645711555811	1894.00	bar
2018-03-05 08:26:08	16	Crystal Clark	5570600642865857	1617.00	bar
2018-03-06 07:18:09	25	Nancy Contreras	4319653513507	1334.00	bar
2018-01-22 08:07:03	16	Crystal Clark	5570600642865857	1131.00	restaurant
2018-09-26 08:48:40	1	Robert Johnson	4761049645711555811	1060.00	restaurant
2018-09-06 08:28:55	1	Robert Johnson	4761049645711555811	1017.00	bar
2018-03-26 07:41:59	9	Laurie Gibbs	30181963913340	1009.00	coffee shop
2018-12-14 08:51:41	12	Megan Price	501879657465	748.00	pub
2018-04-01 07:17:21	25	Nancy Contreras	4319653513507	100.00	coffee shop
2018-08-26 07:15:18	20	Kevin Spencer	4506405265172173	23.13	food truck
2018-08-28 07:17:14	10	Matthew Gutierrez	4165305432349489280	20.71	coffee shop
2018-10-07 08:16:54	20	Kevin Spencer	4586962917519654607	20.44	food truck

2018-10-18 07:16:04	16	Crystal Clark	5500708021555307	19.86	pub
2018-03-03 08:42:02	24	Stephanie Dalton	4681896441519	19.50	food truck
2018-06-04 08:30:28	4	Danielle Green	584226564303	19.49	food truck
2018-03-26 08:08:27	16	Crystal Clark	5500708021555307	19.34	pub
2018-06-01 08:56:59	11	Brandon Pineda	180098539019105	19.33	restaurant
2018-12-23 07:39:47	10	Matthew Gutierrez	4962915017023706562	19.27	pub
2018-09-23 07:30:56	15	Kyle Tucker	4723783028106084756	19.02	restaurant
2018-02-21 08:21:10	5	Sara Cooper	4268491956169254	18.98	food truck
2018-07-04 08:51:29	10	Matthew Gutierrez	213193946980303	18.93	coffee shop
2018-08-02 07:44:24	19	Peter Mckay	5361779664174555	18.93	food truck
2018-06-09 07:29:28	23	Mark Lewis	4188164051171486	18.82	pub
2018-06-02 07:45:48	23	Mark Lewis	4150721559116778	18.76	food truck
2018-01-06 08:42:50	20	Kevin Spencer	3535651398328201	18.72	restaurant
2018-06-24 07:28:48	14	Gary Jacobs	5175947111814778	18.60	pub
2018-06-23 07:12:06	8	Michael Floyd	4834483169177062	18.56	restaurant
2018-08-14 08:38:49	18	Malik Carlson	4498002758300	18.54	restaurant
2018-10-07 07:08:08	24	Stephanie Dalton	3582198969197591	18.53	food truck
2018-02-27 08:27:00	2	Shane Shaffer	675911140852	18.52	restaurant
2018-12-11 07:42:50	1	Robert Johnson	3517111172421930	14.92	restaurant
2018-04-07 07:18:20	12	Megan Price	5297187379298983	14.80	food truck
2018-08-02 07:13:49	7	Sean Taylor	3516952396080247	14.42	coffee shop
2018-12-15 08:34:15	25	Nancy Contreras	372414832802279	14.36	food truck
2018-04-24 07:41:49	15	Kyle Tucker	6500236164848279	14.30	food truck
2018-10-22 07:41:56	13	John Martin	4711773125020499	13.83	coffee shop
2018-09-03 07:27:55	12	Megan Price	376027549341849	13.59	pub
2018-03-10 08:52:09	2	Shane Shaffer	675911140852	13.53	coffee shop
2018-04-27 08:08:38	19	Peter Mckay	3561072557118696	13.52	food truck
2018-02-26 07:31:20	20	Kevin Spencer	4586962917519654607	13.46	pub
2018-10-27 08:04:44	23	Mark Lewis	4150721559116778	12.98	coffee shop
2018-05-01 07:40:00	15	Kyle Tucker	6500236164848279	12.98	bar
2018-03-30 08:12:28	5	Sara Cooper	4268491956169254	12.96	coffee shop
2018-01-17 07:31:03	18	Malik Carlson	4498002758300	12.67	food truck
2018-08-08 08:36:23	11	Brandon Pineda	180098539019105	12.52	coffee shop
2018-07-30 08:49:16	5	Sara Cooper	4276466390111	12.50	food truck
2018-11-30 08:52:20	7	Sean Taylor	4539990688484983	12.32	coffee shop
2018-12-29 08:11:55	18	Malik Carlson	4498002758300	12.25	restaurant

2018-02-19 08:50:36	1	Robert Johnson	4761049645711555811	12.17	food truck
2018-01-20 08:35:34	12	Megan Price	501879657465	11.96	bar
2018-07-31 07:44:59	24	Stephanie Dalton	4681896441519	11.93	bar
2018-01-17 07:56:09	24	Stephanie Dalton	4681896441519	11.91	restaurant
2018-06-08 07:25:30	25	Nancy Contreras	372414832802279	11.88	coffee shop
2018-12-31 08:22:17	23	Mark Lewis	4188164051171486	11.87	restaurant
2018-08-30 08:46:49	4	Danielle Green	584226564303	11.86	bar
2018-04-08 07:06:20	13	John Martin	3561954487988605	11.73	food truck
2018-09-15 08:33:49	16	Crystal Clark	5500708021555307	11.72	coffee shop
2018-12-18 07:45:28	21	Dana Washington	4279104135293225293	11.70	food truck
2018-02-25 07:37:03	13	John Martin	5135837688671496	11.68	pub
2018-05-04 08:21:59	11	Brandon Pineda	4027907156459098	11.65	pub

100 rows × 5 columns

```
px.scatter(suspect_df, x=suspect_df.index, y='amount', color='category', size='amount',
hover_data=['name', 'category'], labels = {'x' : 'Date', 'y' : 'Amount'}, title='Early Hour
Transactions')
```

Data Analysis Questions 1

Use hvPlot to create a line plot showing a time series from the transactions along all the year for **card holders 2 and 18**. In order to contrast the patterns of both card holders, create a line plot containing both lines. What difference do you observe between the consumption patterns? Does the difference could be a fraudulent transaction? Explain your rationale.

```
# Loading data for card holder 2 and 18 from the database
query = f'SELECT a.id, a.name, b.card, c.date, c.amount, e.name as "category" \
        FROM public.card_holder a, public.credit_card b, public.transaction c, public.merchant d,
public.merchant_category e \
        WHERE a.id = b.id_card_holder AND b.card=c.card AND c.id_merchant=d.id AND
d.id_merchant_category=e.id AND a.id IN (2, 18) AND b.card=c.card'

transaction_df = execute_query(query)
transaction_df.head()

data_by_id = transaction_df.groupby('id')
data_by_id
```

```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x0000021B074868C8>
```

In [61]:

```
# plot for cardholder 2

cardholder_2 = data_by_id.get_group(2)['amount'].hvplot(ylabel='Amount', xlabel='Date',
title="Transactions by Id Holder 2")
cardholder_2
```

```
WARNING:param.CurvePlot03139: title_format is deprecated. Please use title instead
WARNING:param.CurvePlot03139: title_format is deprecated. Please use title instead
```

:

```
# plot for cardholder 18

cardholder_18 = data_by_id.get_group(18)['amount'].hvplot(ylabel='Amount', xlabel='Date',
title="Transactions by Id Holder 18")
cardholder_18
```

```
WARNING:param.CurvePlot03302: title_format is deprecated. Please use title instead
WARNING:param.CurvePlot03302: title_format is deprecated. Please use title instead
```

```
# combined plot for card holders 2 and 18
cardholder_2 * cardholder_18
```

```
WARNING:param.OverlayPlot04094: title_format is deprecated. Please use title instead
WARNING:param.OverlayPlot04094: title_format is deprecated. Please use title instead
```

Conclusions for Question 1

The consumption pattern for both the id holder is very different. Id Holder 2 makes too many small transactions. Id Holder 18 has transactions ranging till $1839. Id Holder 2 is more suspectable to fraudulent transactions

Data Analysis Question 2

Use `Plotly Express` to create a series of six box plots, one for each month, in order to identify how many outliers could be per month for **card holder id 25**. By observing the consumption patters, do you see any anomalies? Write your own conclusions about your insights.

```
# Loading data of daily transactions from jan to jun 2018 for card holder 25
card_holder_id = 25
start_date = '2018-01-01'
end_date = '2018-07-01'
query = f'SELECT a.id, a.name, b.card, c.date, c.amount, e.name as "category" \
        FROM public.card_holder a, public.credit_card b, public.transaction c, public.merchant d,
public.merchant_category e \
```

```
        WHERE a.id = b.id_card_holder AND b.card=c.card AND c.id_merchant=d.id AND
d.id_merchant_category=e.id AND a.id={card_holder_id} \
        AND date BETWEEN \'{start_date} 00:00:00\'::timestamp AND \'{end_date} 00:00:00\'::timestamp'
transaction_df = execute_query(query)
transaction_df.head()
```

2018-01-02 02:06:21	25	Nancy Contreras	4319653513507	1.46	food truck
2018-01-05 06:26:45	25	Nancy Contreras	372414832802279	10.74	food truck
2018-01-07 14:57:23	25	Nancy Contreras	4319653513507	2.93	food truck
2018-01-10 00:25:40	25	Nancy Contreras	372414832802279	1.39	restaurant
2018-01-14 05:02:22	25	Nancy Contreras	372414832802279	17.84	food truck

```
# change the numeric month to month names
transaction_df['Month'] = transaction_df.index.month_name()
transaction_df.head()
```

2018-01-02 02:06:21	25	Nancy Contreras	4319653513507	1.46	food truck	January
2018-01-05 06:26:45	25	Nancy Contreras	372414832802279	10.74	food truck	January
2018-01-07 14:57:23	25	Nancy Contreras	4319653513507	2.93	food truck	January
2018-01-10 00:25:40	25	Nancy Contreras	372414832802279	1.39	restaurant	January
2018-01-14 05:02:22	25	Nancy Contreras	372414832802279	17.84	food truck	January

```
# creating the six box plots using plotly express
px.box(transaction_df, y='amount', hover_data=['category','card'],x='Month', color='category',
title='Transactions by Id Holder 25')
```

Conclusions for Question 2

There seems to be fraudulent transactions pertaining to Restaurant & Food Truck category where Food Truck is ranging from *1.46 to* 1046

Identifying Outliers using Standard Deviation

```
# initial imports
import pandas as pd
```

```
import numpy as np
import random
from sqlalchemy import create_engine
from numpy import mean
from numpy import std
from numpy import percentile
import plotly.express as px
```

In [92]:

```
# create a connection to the database
engine = create_engine("postgresql://postgres:Istay@10314@localhost:5432/fraud_detection")

# Loading data from the database

def execute_query(query):

    transaction_df = pd.read_sql(sql=query, con=engine, index_col='date', parse_dates='date')

    return transaction_df
```

In [93]:

```
# Loading data of daily transactions from jan to jun 2018 for card holder 25
query = f'SELECT a.id, a.name, b.card, c.date, c.amount, e.name as "category" \
        FROM public.card_holder a, public.credit_card b, public.transaction c, public.merchant d,
public.merchant_category e \
        WHERE a.id = b.id_card_holder AND b.card=c.card AND c.id_merchant=d.id AND
d.id_merchant_category=e.id'

transaction_df = execute_query(query)
transaction_df.head()
```

2018-01-01 21:35:10	13	John Martin	3561954487988605	6.22	food truck
2018-01-01 21:43:12	13	John Martin	5135837688671496	3.83	bar
2018-01-01 22:41:21	10	Matthew Gutierrez	213193946980303	9.61	food truck
2018-01-01 23:13:30	4	Danielle Green	4263694062533017	19.03	pub
2018-01-01 23:15:10	18	Malik Carlson	4498002758300	2.95	restaurant

```
# code a function to identify outliers based on standard deviation
# calculate summary statistics
data_mean, data_std = mean(transaction_df['amount']), std(transaction_df['amount'])

# identify outliers
cut_off = data_std * 3

lower, upper = data_mean - cut_off, data_mean + cut_off

# identify outliers
outliers = [x for x in transaction_df['amount'] if x < lower or x > upper]

print('Identified outliers: %d' % len(outliers))
```

```
# remove outliers
outliers_removed = [x for x in transaction_df['amount'] if x >= lower and x <= upper]

print('Non-outlier observations: %d' % len(outliers_removed))

transaction_df['outlier'] = (transaction_df['amount'] > upper) | (transaction_df['amount'] < lower)

outlier = transaction_df[transaction_df['outlier']==True]
outlier
```

```
Identified outliers: 77
Non-outlier observations: 3423
```

2018-01-02 23:27:46	12	Megan Price	501879657465	1031.0	restaurant	True
2018-01-04 03:05:18	7	Sean Taylor	3516952396080247	1685.0	food truck	True
2018-01-08 02:34:32	6	Beth Hernandez	3581345943543942	1029.0	bar	True
2018-01-22 08:07:03	16	Crystal Clark	5570600642865857	1131.0	restaurant	True
2018-01-23 06:29:37	12	Megan Price	501879657465	1678.0	pub	True
2018-01-24 13:17:19	1	Robert Johnson	4761049645711555811	1691.0	coffee shop	True
2018-01-30 18:31:00	25	Nancy Contreras	4319653513507	1177.0	restaurant	True
2018-02-17 01:27:19	16	Crystal Clark	5570600642865857	1430.0	restaurant	True
2018-02-19 16:00:43	7	Sean Taylor	3516952396080247	1072.0	food truck	True
2018-02-19 22:48:25	18	Malik Carlson	344119623920892	1839.0	restaurant	True

2018-02-27 15:27:32	6	Beth Hernandez	3581345943543942	1145.0	bar	True
2018-03-01 21:29:05	3	Elizabeth Sawyer	30078299053512	1119.0	pub	True
2018-03-04 15:50:53	9	Laurie Gibbs	30181963913340	1534.0	coffee shop	True
2018-03-05 08:26:08	16	Crystal Clark	5570600642865857	1617.0	bar	True
2018-03-06 07:18:09	25	Nancy Contreras	4319653513507	1334.0	bar	True
2018-03-12 00:44:01	12	Megan Price	501879657465	1530.0	coffee shop	True
2018-03-20 10:19:25	12	Megan Price	501879657465	852.0	pub	True
2018-03-20 13:05:54	24	Stephanie Dalton	30142966699187	1011.0	bar	True
2018-03-26 07:41:59	9	Laurie Gibbs	30181963913340	1009.0	coffee shop	True
2018-04-03 03:23:37	18	Malik Carlson	344119623920892	1077.0	restaurant	True
2018-04-08 06:03:50	25	Nancy Contreras	4319653513507	1063.0	pub	True
2018-04-18 23:23:29	7	Sean Taylor	3516952396080247	1086.0	coffee shop	True
2018-04-21 19:41:51	6	Beth Hernandez	3581345943543942	2108.0	coffee shop	True
2018-05-08 13:21:01	24	Stephanie Dalton	30142966699187	1901.0	restaurant	True
2018-05-13 06:31:20	25	Nancy Contreras	4319653513507	1046.0	food truck	True

2018-05-29 02:55:08	16	Crystal Clark	5570600642865857	1203.0	food truck	True
2018-06-03 20:02:28	18	Malik Carlson	344119623920892	1814.0	pub	True
2018-06-04 03:46:15	25	Nancy Contreras	4319653513507	1162.0	pub	True
2018-06-06 21:50:17	25	Nancy Contreras	4319653513507	749.0	restaurant	True
2018-06-10 04:54:27	9	Laurie Gibbs	30181963913340	1795.0	pub	True
2018-09-04 01:35:39	1	Robert Johnson	4761049645711555811	1790.0	coffee shop	True
2018-09-06 08:28:55	1	Robert Johnson	4761049645711555811	1017.0	bar	True
2018-09-06 21:55:02	1	Robert Johnson	4761049645711555811	1056.0	restaurant	True
2018-09-10 22:49:41	18	Malik Carlson	344119623920892	1176.0	restaurant	True
2018-09-11 15:16:47	6	Beth Hernandez	3581345943543942	1856.0	food truck	True
2018-09-23 19:20:23	12	Megan Price	501879657465	1075.0	pub	True
2018-09-25 23:23:21	9	Laurie Gibbs	30181963913340	1095.0	food truck	True
2018-09-26 08:48:40	1	Robert Johnson	4761049645711555811	1060.0	restaurant	True
2018-10-07 14:40:34	3	Elizabeth Sawyer	30078299053512	757.0	bar	True
2018-10-07 18:29:20	9	Laurie Gibbs	30181963913340	1179.0	pub	True

2018-10-19 01:07:37	3	Elizabeth Sawyer	30078299053512	1053.0	restaurant	True
2018-11-13 17:07:25	16	Crystal Clark	5570600642865857	1911.0	restaurant	True
2018-11-17 05:30:43	18	Malik Carlson	344119623920892	1769.0	food truck	True
2018-11-20 05:24:28	3	Elizabeth Sawyer	30078299053512	1054.0	bar	True
2018-11-25 20:44:07	12	Megan Price	501879657465	1123.0	bar	True
2018-11-27 15:36:05	12	Megan Price	501879657465	1802.0	bar	True
2018-11-27 17:20:29	6	Beth Hernandez	3581345943543942	1279.0	restaurant	True
2018-11-27 17:27:34	1	Robert Johnson	4761049645711555811	1660.0	pub	True
2018-12-03 02:38:52	16	Crystal Clark	5570600642865857	1014.0	restaurant	True
2018-12-07 07:22:03	1	Robert Johnson	4761049645711555811	1894.0	bar	True
2018-12-13 12:09:58	18	Malik Carlson	344119623920892	1154.0	restaurant	True
2018-12-13 15:51:59	7	Sean Taylor	3516952396080247	2249.0	food truck	True
2018-12-14 08:51:41	12	Megan Price	501879657465	748.0	pub	True
2018-12-18 13:33:37	25	Nancy Contreras	4319653513507	1074.0	coffee shop	True
2018-12-18 17:20:33	7	Sean Taylor	3516952396080247	1296.0	bar	True

2018-12-19 16:10:03	9	Laurie Gibbs	30181963913340	1724.0	pub	True
2018-12-21 09:56:32	24	Stephanie Dalton	30142966699187	1301.0	pub	True
2018-12-24 15:55:06	16	Crystal Clark	5570600642865857	1634.0	pub	True
2018-12-25 19:10:42	24	Stephanie Dalton	30142966699187	1035.0	pub	True
2018-12-30 23:23:09	1	Robert Johnson	4761049645711555811	1033.0	pub	True

77 rows × 6 columns

```
# find anomalous transactions for 3 random card holders
import datetime
start_time = datetime.time(7,0,0)
end_time = datetime.time(9,0,0)

anomalous_transactions = outlier.between_time(start_time, end_time).sort_values('amount', ascending=False)

px.scatter(anomalous_transactions, x='name', y='amount', color='category', title='Anomalous Transactions')
```

Identifying Outliers Using Interquartile Range

```
# code a function to identify outliers based on interquartile range
# calculate interquartile range
q25, q75 = percentile(transaction_df['amount'], 25), percentile(transaction_df['amount'], 75)
iqr = q75 - q25
print('Percentiles: 25th=%.3f, 75th=%.3f, IQR=%.3f' % (q25, q75, iqr))
# calculate the outlier cutoff
cut_off = iqr * 1.5
lower, upper = q25 - cut_off, q75 + cut_off
# identify outliers
outliers_2 = [x for x in transaction_df['amount'] if x < lower or x > upper]
print('Identified outliers: %d' % len(outliers_2))
# remove outliers
outliers_removed_2 = [x for x in transaction_df['amount'] if x >= lower and x <= upper]
print('Non-outlier observations: %d' % len(outliers_removed_2))

transaction_df['outlier'] = (transaction_df['amount'] > upper) | (transaction_df['amount'] < lower)

outlier_2 = transaction_df[transaction_df['outlier']==True]
```

```
outlier_2
```

```
Percentiles: 25th=3.735, 75th=14.648, IQR=10.913
Identified outliers: 110
Non-outlier observations: 3390
```

2018-01-02 23:27:46	12	Megan Price	501879657465	1031.0	restaurant	True
2018-01-04 03:05:18	7	Sean Taylor	3516952396080247	1685.0	food truck	True
2018-01-07 01:10:54	18	Malik Carlson	344119623920892	175.0	pub	True
2018-01-08 02:34:32	6	Beth Hernandez	3581345943543942	1029.0	bar	True
2018-01-08 11:15:36	18	Malik Carlson	344119623920892	333.0	restaurant	True
2018-01-11 13:20:31	16	Crystal Clark	5570600642865857	229.0	pub	True
2018-01-22 08:07:03	16	Crystal Clark	5570600642865857	1131.0	restaurant	True
2018-01-23 06:29:37	12	Megan Price	501879657465	1678.0	pub	True
2018-01-24 13:17:19	1	Robert Johnson	4761049645711555811	1691.0	coffee shop	True
2018-01-30 18:31:00	25	Nancy Contreras	4319653513507	1177.0	restaurant	True
2018-02-09 11:38:37	7	Sean Taylor	3516952396080247	445.0	bar	True
2018-02-17 01:27:19	16	Crystal Clark	5570600642865857	1430.0	restaurant	True
2018-02-19 16:00:43	7	Sean Taylor	3516952396080247	1072.0	food truck	True
2018-02-19 22:48:25	18	Malik Carlson	344119623920892	1839.0	restaurant	True
2018-02-27 15:27:32	6	Beth Hernandez	3581345943543942	1145.0	bar	True
2018-03-01 21:29:05	3	Elizabeth Sawyer	30078299053512	1119.0	pub	True
2018-03-04 15:50:53	9	Laurie Gibbs	30181963913340	1534.0	coffee shop	True
2018-03-05 08:26:08	16	Crystal Clark	5570600642865857	1617.0	bar	True
2018-03-06 07:18:09	25	Nancy Contreras	4319653513507	1334.0	bar	True
2018-03-09 04:51:38	6	Beth Hernandez	3581345943543942	389.0	restaurant	True
2018-03-12 00:44:01	12	Megan Price	501879657465	1530.0	coffee shop	True
2018-03-20 10:19:25	12	Megan Price	501879657465	852.0	pub	True
2018-03-20 13:05:54	24	Stephanie Dalton	30142966699187	1011.0	bar	True
2018-03-26 07:41:59	9	Laurie Gibbs	30181963913340	1009.0	coffee shop	True
2018-04-01 07:17:21	25	Nancy Contreras	4319653513507	100.0	coffee shop	True
2018-04-03 03:23:37	18	Malik Carlson	344119623920892	1077.0	restaurant	True
2018-04-08 06:03:50	25	Nancy Contreras	4319653513507	1063.0	pub	True
2018-04-09 10:24:32	1	Robert Johnson	4761049645711555811	283.0	restaurant	True
2018-04-09 18:28:25	25	Nancy Contreras	4319653513507	269.0	restaurant	True
2018-04-10 06:08:01	7	Sean Taylor	3516952396080247	543.0	food truck	True
2018-10-11 23:29:33	3	Elizabeth Sawyer	30078299053512	206.0	restaurant	True
2018-10-16 13:27:33	1	Robert Johnson	4761049645711555811	484.0	food truck	True
2018-10-19 01:07:37	3	Elizabeth Sawyer	30078299053512	1053.0	restaurant	True
2018-10-19 12:32:37	16	Crystal Clark	5570600642865857	178.0	food truck	True

2018-10-23 22:47:13	16	Crystal Clark	5570600642865857	393.0	food truck	True
2018-10-28 02:12:58	25	Nancy Contreras	4319653513507	137.0	pub	True
2018-11-13 05:58:47	24	Stephanie Dalton	30142966699187	466.0	bar	True
2018-11-13 17:07:25	16	Crystal Clark	5570600642865857	1911.0	restaurant	True
2018-11-17 05:30:43	18	Malik Carlson	344119623920892	1769.0	food truck	True
2018-11-20 05:24:28	3	Elizabeth Sawyer	30078299053512	1054.0	bar	True
2018-11-23 09:08:05	12	Megan Price	501879657465	233.0	restaurant	True
2018-11-25 20:44:07	12	Megan Price	501879657465	1123.0	bar	True
2018-11-26 20:54:39	1	Robert Johnson	4761049645711555811	267.0	food truck	True
2018-11-27 15:36:05	12	Megan Price	501879657465	1802.0	bar	True
2018-11-27 17:20:29	6	Beth Hernandez	3581345943543942	1279.0	restaurant	True
2018-11-27 17:27:34	1	Robert Johnson	4761049645711555811	1660.0	pub	True
2018-12-03 02:38:52	16	Crystal Clark	5570600642865857	1014.0	restaurant	True
2018-12-05 19:24:27	9	Laurie Gibbs	30181963913340	57.0	bar	True
2018-12-07 07:22:03	1	Robert Johnson	4761049645711555811	1894.0	bar	True
2018-12-13 12:09:58	18	Malik Carlson	344119623920892	1154.0	restaurant	True
2018-12-13 15:51:59	7	Sean Taylor	3516952396080247	2249.0	food truck	True
2018-12-14 08:51:41	12	Megan Price	501879657465	748.0	pub	True
2018-12-18 13:33:37	25	Nancy Contreras	4319653513507	1074.0	coffee shop	True
2018-12-18 17:20:33	7	Sean Taylor	3516952396080247	1296.0	bar	True
2018-12-19 16:10:03	9	Laurie Gibbs	30181963913340	1724.0	pub	True
2018-12-21 09:56:32	24	Stephanie Dalton	30142966699187	1301.0	pub	True
2018-12-24 15:55:06	16	Crystal Clark	5570600642865857	1634.0	pub	True
2018-12-25 19:10:42	24	Stephanie Dalton	30142966699187	1035.0	pub	True
2018-12-28 16:20:31	3	Elizabeth Sawyer	30078299053512	313.0	pub	True
2018-12-30 23:23:09	1	Robert Johnson	4761049645711555811	1033.0	pub	True

110 rows × 6 columns

```
# find anomalous transactions for 3 random card holders
anomalous_transactions2 = outlier_2.between_time(start_time, end_time).sort_values('amount',
ascending=False)
anomalous_transactions2
```

2018-12-07 07:22:03	1	Robert Johnson	4761049645711555811	1894.0	bar	True

2018-03-05 08:26:08	16	Crystal Clark	5570600642865857	1617.0	bar	True
2018-03-06 07:18:09	25	Nancy Contreras	4319653513507	1334.0	bar	True
2018-01-22 08:07:03	16	Crystal Clark	5570600642865857	1131.0	restaurant	True
2018-09-26 08:48:40	1	Robert Johnson	4761049645711555811	1060.0	restaurant	True
2018-09-06 08:28:55	1	Robert Johnson	4761049645711555811	1017.0	bar	True
2018-03-26 07:41:59	9	Laurie Gibbs	30181963913340	1009.0	coffee shop	True
2018-12-14 08:51:41	12	Megan Price	501879657465	748.0	pub	True
2018-04-01 07:17:21	25	Nancy Contreras	4319653513507	100.0	coffee shop	True

```
px.scatter(anomalous_transactions2, x='name', y='amount', color='category', title="Early Hour Transactions")
```

For Outlier calculation using standard deviation, results in 77 records whereas using Interquartile range results in 110 records. There seems to be fraudulent transactions in Bar category wherein amount spent between 7-9 AM in the Bar

Visual Data Analysis of Fraudulent Transactions

```python
# initial imports
import pandas as pd
import calendar
import plotly.express as px
import hvplot.pandas
from sqlalchemy import create_engine
import datetime
```

```python
# create a connection to the database
engine = create_engine("postgresql://postgres:Istay@10314@localhost:5432/fraud_detection")
```

```python
# Loading data from the database

def execute_query(query):

    transaction_df = pd.read_sql(sql=query, con=engine, index_col='date', parse_dates='date')

    return transaction_df

def fraud_transactions(df):
    start_time = datetime.time(7,0,0)
    end_time = datetime.time(9,0,0)
    return df.between_time(start_time, end_time).sort_values('amount', ascending=False)
```

```python
# Loading data for card holder 2 and 18 from the database
query = f'SELECT a.id, a.name, b.card, c.date, c.amount, e.name as "category" \
        FROM public.card_holder a, public.credit_card b, public.transaction c, public.merchant d,
public.merchant_category e \
        WHERE a.id = b.id_card_holder AND b.card=c.card AND c.id_merchant=d.id AND
d.id_merchant_category=e.id AND b.card=c.card'

transaction_df = execute_query(query)

suspect_df = fraud_transactions(transaction_df).head(100)
suspect_df
```

2018-12-07 07:22:03	1	Robert Johnson	4761049645711555811	1894.00	bar
2018-03-05 08:26:08	16	Crystal Clark	5570600642865857	1617.00	bar
2018-03-06 07:18:09	25	Nancy Contreras	4319653513507	1334.00	bar
2018-01-22 08:07:03	16	Crystal Clark	5570600642865857	1131.00	restaurant

2018-09-26 08:48:40	1	Robert Johnson	4761049645711555811	1060.00	restaurant
2018-09-06 08:28:55	1	Robert Johnson	4761049645711555811	1017.00	bar
2018-03-26 07:41:59	9	Laurie Gibbs	30181963913340	1009.00	coffee shop
2018-12-14 08:51:41	12	Megan Price	501879657465	748.00	pub
2018-04-01 07:17:21	25	Nancy Contreras	4319653513507	100.00	coffee shop
2018-08-26 07:15:18	20	Kevin Spencer	4506405265172173	23.13	food truck
2018-08-28 07:17:14	10	Matthew Gutierrez	4165305432349489280	20.71	coffee shop
2018-10-07 08:16:54	20	Kevin Spencer	4586962917519654607	20.44	food truck
2018-10-18 07:16:04	16	Crystal Clark	5500708021555307	19.86	pub
2018-03-03 08:42:02	24	Stephanie Dalton	4681896441519	19.50	food truck
2018-06-04 08:30:28	4	Danielle Green	584226564303	19.49	food truck
2018-03-26 08:08:27	16	Crystal Clark	5500708021555307	19.34	pub
2018-06-01 08:56:59	11	Brandon Pineda	180098539019105	19.33	restaurant
2018-12-23 07:39:47	10	Matthew Gutierrez	4962915017023706562	19.27	pub
2018-09-23 07:30:56	15	Kyle Tucker	4723783028106084756	19.02	restaurant
2018-02-21 08:21:10	5	Sara Cooper	4268491956169254	18.98	food truck
2018-07-04 08:51:29	10	Matthew Gutierrez	213193946980303	18.93	coffee shop
2018-08-02 07:44:24	19	Peter Mckay	5361779664174555	18.93	food truck
2018-06-09 07:29:28	23	Mark Lewis	4188164051171486	18.82	pub
2018-06-02 07:45:48	23	Mark Lewis	4150721559116778	18.76	food truck
2018-01-06 08:42:50	20	Kevin Spencer	3535651398328201	18.72	restaurant
2018-06-24 07:28:48	14	Gary Jacobs	5175947111814778	18.60	pub
2018-06-23 07:12:06	8	Michael Floyd	4834483169177062	18.56	restaurant
2018-08-14 08:38:49	18	Malik Carlson	4498002758300	18.54	restaurant
2018-10-07 07:08:08	24	Stephanie Dalton	3582198969197591	18.53	food truck
2018-02-27 08:27:00	2	Shane Shaffer	675911140852	18.52	restaurant
2018-12-11 07:42:50	1	Robert Johnson	3517111172421930	14.92	restaurant
2018-04-07 07:18:20	12	Megan Price	5297187379298983	14.80	food truck
2018-08-02 07:13:49	7	Sean Taylor	3516952396080247	14.42	coffee shop
2018-12-15 08:34:15	25	Nancy Contreras	372414832802279	14.36	food truck
2018-04-24 07:41:49	15	Kyle Tucker	6500236164848279	14.30	food truck
2018-10-22 07:41:56	13	John Martin	4711773125020499	13.83	coffee shop
2018-09-03 07:27:55	12	Megan Price	376027549341849	13.59	pub
2018-03-10 08:52:09	2	Shane Shaffer	675911140852	13.53	coffee shop
2018-04-27 08:08:38	19	Peter Mckay	3561072557118696	13.52	food truck
2018-02-26 07:31:20	20	Kevin Spencer	4586962917519654607	13.46	pub
2018-10-27 08:04:44	23	Mark Lewis	4150721559116778	12.98	coffee shop
2018-05-01 07:40:00	15	Kyle Tucker	6500236164848279	12.98	bar
2018-03-30 08:12:28	5	Sara Cooper	4268491956169254	12.96	coffee shop
2018-01-17 07:31:03	18	Malik Carlson	4498002758300	12.67	food truck
2018-08-08 08:36:23	11	Brandon Pineda	180098539019105	12.52	coffee shop

2018-07-30 08:49:16	5	Sara Cooper	4276466390111	12.50	food truck
2018-11-30 08:52:20	7	Sean Taylor	4539990688484983	12.32	coffee shop
2018-12-29 08:11:55	18	Malik Carlson	4498002758300	12.25	restaurant
2018-02-19 08:50:36	1	Robert Johnson	4761049645711555811	12.17	food truck
2018-01-20 08:35:34	12	Megan Price	501879657465	11.96	bar
2018-07-31 07:44:59	24	Stephanie Dalton	4681896441519	11.93	bar
2018-01-17 07:56:09	24	Stephanie Dalton	4681896441519	11.91	restaurant
2018-06-08 07:25:30	25	Nancy Contreras	372414832802279	11.88	coffee shop
2018-12-31 08:22:17	23	Mark Lewis	4188164051171486	11.87	restaurant
2018-08-30 08:46:49	4	Danielle Green	584226564303	11.86	bar
2018-04-08 07:06:20	13	John Martin	3561954487988605	11.73	food truck
2018-09-15 08:33:49	16	Crystal Clark	5500708021555307	11.72	coffee shop
2018-12-18 07:45:28	21	Dana Washington	4279104135293225293	11.70	food truck
2018-02-25 07:37:03	13	John Martin	5135837688671496	11.68	pub
2018-05-04 08:21:59	11	Brandon Pineda	4027907156459098	11.65	pub

100 rows × 5 columns

```
px.scatter(suspect_df, x=suspect_df.index, y='amount', color='category', size='amount',
hover_data=['name', 'category'], labels = {'x' : 'Date', 'y' : 'Amount'}, title='Early Hour
Transactions')
```

Data Analysis Questions 1

Use `hvPlot` to create a line plot showing a time series from the transactions along all the year for **card holders 2 and 18**. In order to contrast the patterns of both card holders, create a line plot containing both lines. What difference do you observe between the consumption patterns? Does the difference could be a fraudulent transaction? Explain your rationale.

```
# Loading data for card holder 2 and 18 from the database
query = f'SELECT a.id, a.name, b.card, c.date, c.amount, e.name as "category" \
        FROM public.card_holder a, public.credit_card b, public.transaction c, public.merchant d,
public.merchant_category e \
        WHERE a.id = b.id_card_holder AND b.card=c.card AND c.id_merchant=d.id AND
d.id_merchant_category=e.id AND a.id IN (2, 18) AND b.card=c.card'

transaction_df = execute_query(query)
transaction_df.head()

data_by_id = transaction_df.groupby('id')
data_by_id
```

```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x0000021B074868C8>
```

```
# plot for cardholder 2

cardholder_2 = data_by_id.get_group(2)['amount'].hvplot(ylabel='Amount', xlabel='Date',
title="Transactions by Id Holder 2")
cardholder_2
```

```
WARNING:param.CurvePlot03139: title_format is deprecated. Please use title instead
WARNING:param.CurvePlot03139: title_format is deprecated. Please use title instead
```

```
# plot for cardholder 18

cardholder_18 = data_by_id.get_group(18)['amount'].hvplot(ylabel='Amount', xlabel='Date',
title="Transactions by Id Holder 18")
cardholder_18
```

```
WARNING:param.CurvePlot03302: title_format is deprecated. Please use title instead
WARNING:param.CurvePlot03302: title_format is deprecated. Please use title instead
```

```
# combined plot for card holders 2 and 18
cardholder_2 * cardholder_18
```

```
WARNING:param.OverlayPlot04764: title_format is deprecated. Please use title instead
WARNING:param.OverlayPlot04764: title_format is deprecated. Please use title instead
```

Conclusions for Question 1

The consumption pattern for both the id holder is very different. Id Holder 2 makes too many small transactions. Id Holder 18 has transactions ranging till $1839. Id Holder 2 is more suspectable to fraudulent transactions

Data Analysis Question 2

Use `Plotly Express` to create a series of six box plots, one for each month, in order to identify how many outliers could be per month for **card holder id 25**. By observing the consumption patters, do you see any anomalies? Write your own conclusions about your insights.

```
# loading data of daily transactions from jan to jun 2018 for card holder 25
card_holder_id = 25
start_date = '2018-01-01'
end_date = '2018-07-01'
query = f'SELECT a.id, a.name, b.card, c.date, c.amount, e.name as "category" \
        FROM public.card_holder a, public.credit_card b, public.transaction c, public.merchant d,
public.merchant_category e \
        WHERE a.id = b.id_card_holder AND b.card=c.card AND c.id_merchant=d.id AND
d.id_merchant_category=e.id AND a.id={card_holder_id} \
        AND date BETWEEN \'{start_date} 00:00:00\'::timestamp AND \'{end_date} 00:00:00\'::timestamp'
transaction_df = execute_query(query)
transaction_df.head()
```

2018-01-02 02:06:21	25	Nancy Contreras	4319653513507	1.46	food truck
2018-01-05 06:26:45	25	Nancy Contreras	372414832802279	10.74	food truck
2018-01-07 14:57:23	25	Nancy Contreras	4319653513507	2.93	food truck
2018-01-10 00:25:40	25	Nancy Contreras	372414832802279	1.39	restaurant
2018-01-14 05:02:22	25	Nancy Contreras	372414832802279	17.84	food truck

```
# change the numeric month to month names
transaction_df['Month'] = transaction_df.index.month_name()
transaction_df.head()
```

2018-01-02 02:06:21	25	Nancy Contreras	4319653513507	1.46	food truck	January
2018-01-05 06:26:45	25	Nancy Contreras	372414832802279	10.74	food truck	January
2018-01-07 14:57:23	25	Nancy Contreras	4319653513507	2.93	food truck	January
2018-01-10 00:25:40	25	Nancy Contreras	372414832802279	1.39	restaurant	January
2018-01-14 05:02:22	25	Nancy Contreras	372414832802279	17.84	food truck	January

```
# creating the six box plots using plotly express
px.box(transaction_df, y='amount', hover_data=['category','card'],x='Month', color='category',
title='Transactions by Id Holder 25')
```

Conclusions for Question 2

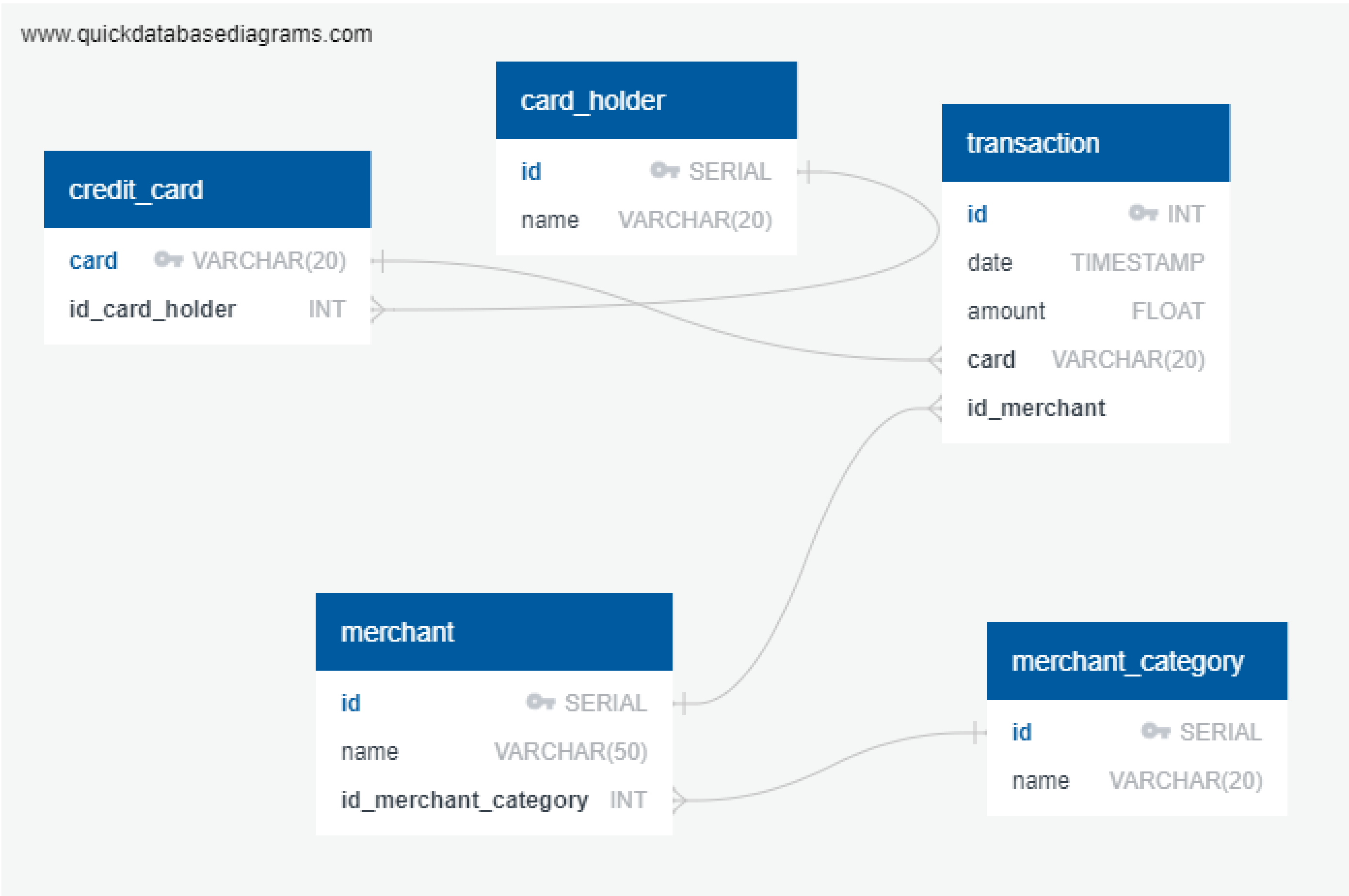
There seems to be fraudulent transactions pertaining to Restaurant & Food Truck category where Food Truck is ranging from *1.46 to* 1046

Data Modeling

Create an entity relationship diagram (ERD) by inspecting the provided CSV files.

Fraud Detection SQL

Note: For the credit_card table, the card column should be a VARCHAR(20) datatype rather than an INT. Tool used to develop ERD Quick Database Diagrams to create your model.



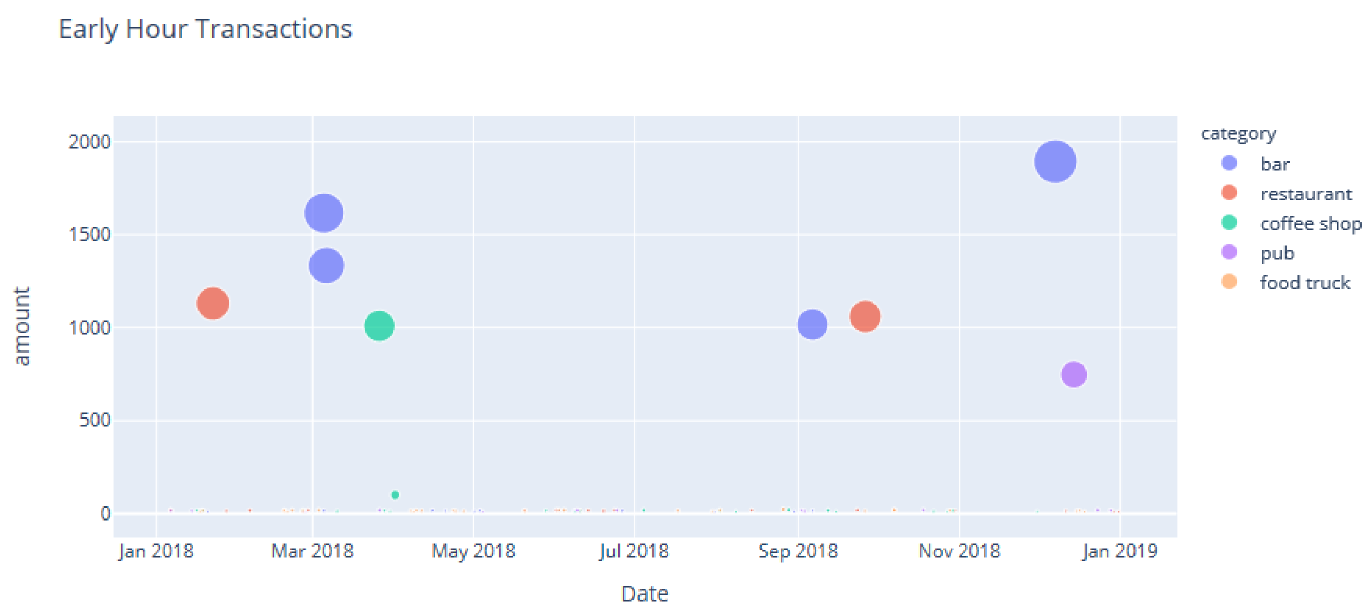
Data Engineering

Using your database model as a blueprint, create a database schema for each of your tables and relationships. Specify data types, primary keys, foreign keys, and any other constraints you defined.
After creating the database schema, import the data from the corresponding CSV files.

Data Analysis

Now that your data is prepared within the database, it's finally time to identify fraudulent transactions using SQL and Pandas DataFrames.
Top 100 highest transactions during early hours i.e. 7:00 to 9:00 AM

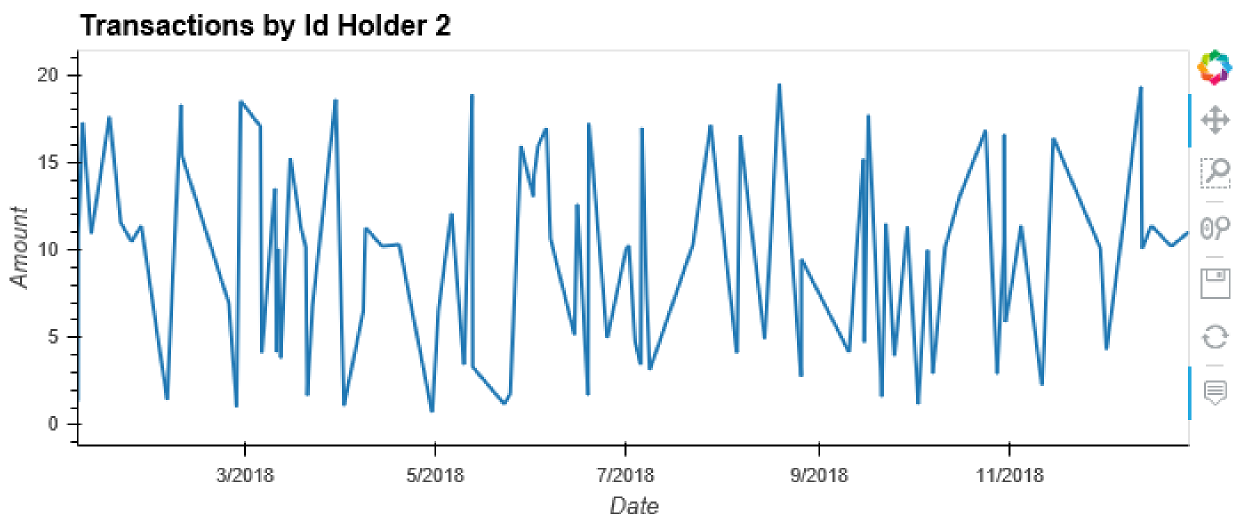
Tarun S Gowda

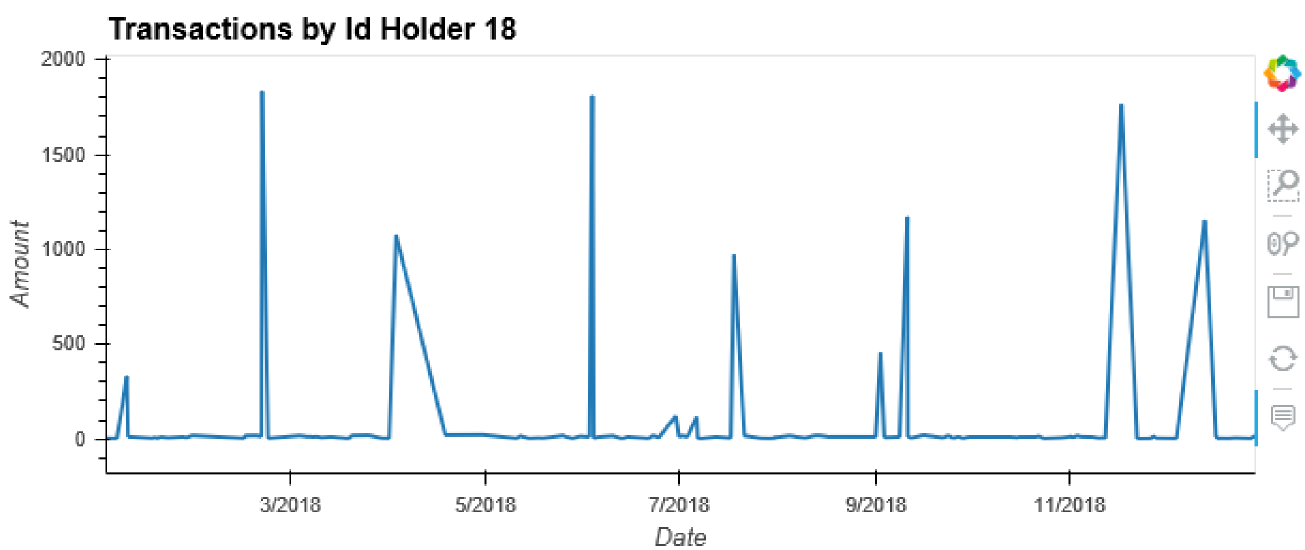


- Some fraudsters hack a credit card by making several small payments (generally less than $2.00), which are typically ignored by cardholders. Count the transactions that are less than $2.00 per cardholder. Is there any evidence to suggest that a credit card has been hacked? Explain your rationale.
- What are the top five merchants prone to being hacked using small transactions?
- Once you have a query that can be reused, create a view for each of the previous queries.

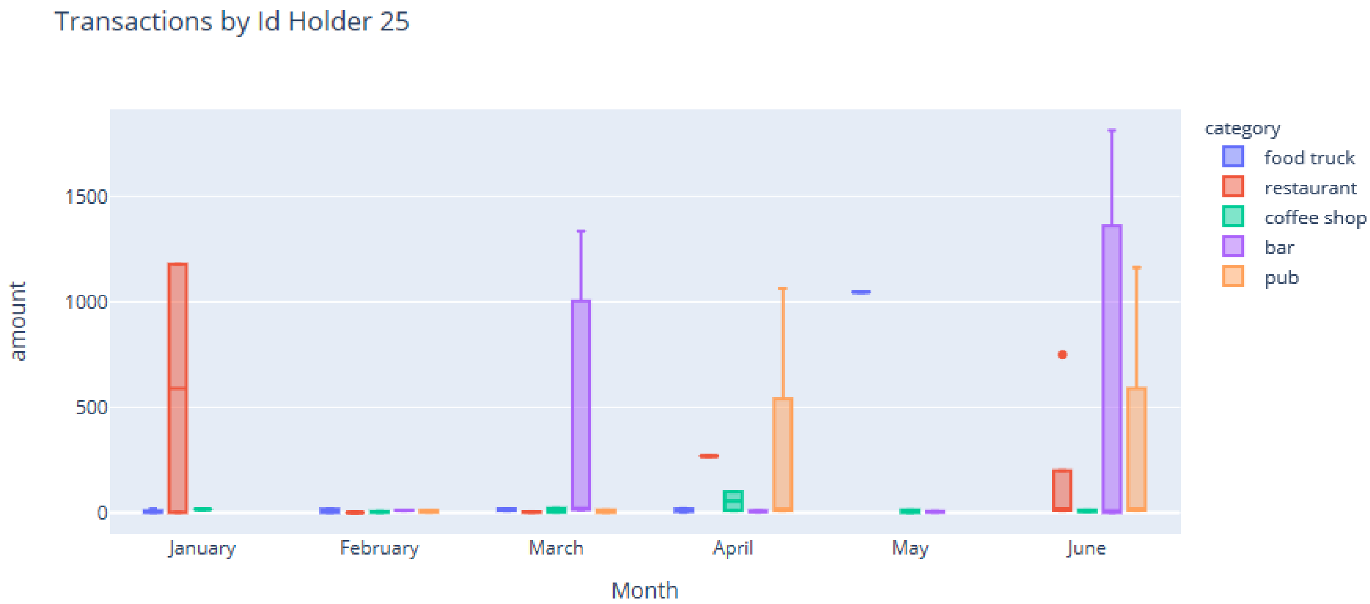
Created a report for fraudulent transactions of some top customers of the firm using Pandas, Plotly Express, hvPlot, and SQLAlchemy to create the visualizations.

- Fraudulent transactions in the history of two of the most important customers of the firm on the basis of their cardholders' IDs are 18 and 2.





- Observation : The consumption pattern for both the id holder is very different. Id Holder 2 makes too many small transactions. Id Holder 18 has transactions ranging till $1839. Id Holder 2 is more suspectable to fraudulent transactions
- The CEO of the firm's biggest customer suspects that someone has used her corporate credit card without authorization in the first quarter of 2018 to pay for several expensive restaurant bills. You are asked to find any anomalous transactions during that period.
    - Using Plotly Express, created a series of six box plots, one for each month, in order to identify how many outliers there are per month for cardholder ID 25.



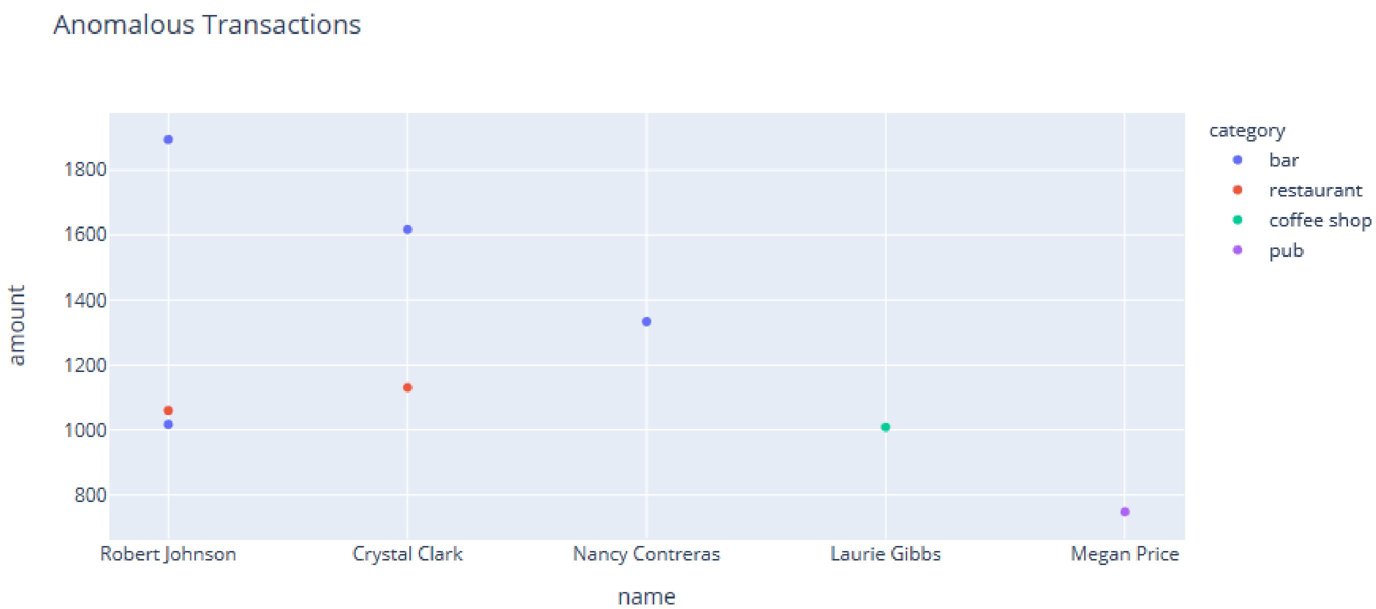
-
  - Observations : There seems to be fraudulent transactions pertaining to Restaurant & Food Truck category where Food Truck is ranging from $1.46 to $1046

Challenge

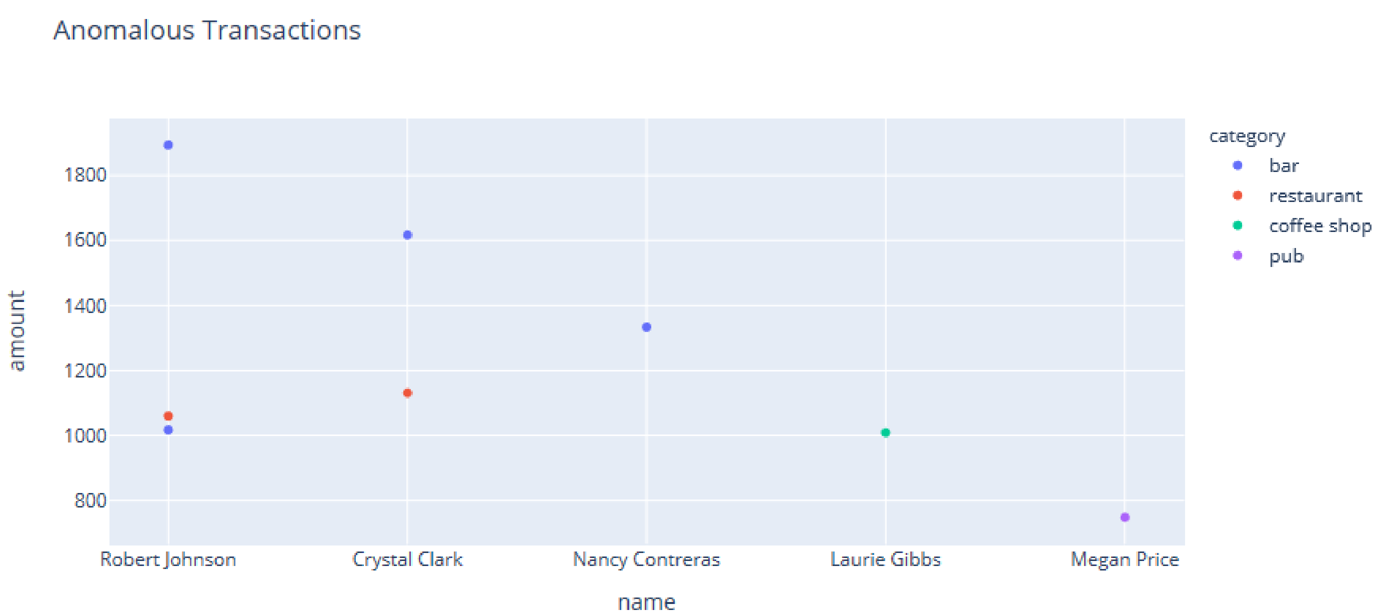
Another approach to identify fraudulent transactions is to look for outliers in the data. Standard deviation or quartiles are often used to detect outliers.

Identifying Outliers based on Standard Deviation

Fraud Detection SQL



Identifying Outliers based on Interquartile Range



Tarun S Gowda

Advantages of Fraud Detection Using SQL

1. **Direct Access to Data**
   a. SQL allows direct querying of transactional databases, which is useful for analyzing large volumes of financial or user behavior data in real-time.
2. **Simplicity and Readability**
   a. SQL is relatively easy to understand and write, especially for pattern-matching queries (e.g., finding duplicates, anomalies, or suspicious trends).
3. **Custom Rule Creation**
   a. You can define custom business rules using SQL, such as flagging transactions above a certain amount or multiple transactions from different locations in a short time.
4. **Integration with Existing Systems**
   a. Most organizations already use relational databases. SQL-based solutions integrate well with existing infrastructure without the need for additional tools.
5. **Low Cost**
   a. No need for additional software if you're using your existing RDBMS. It's a cost-effective way to start simple fraud detection.
6. **Automation Capabilities**
   a. SQL queries can be scheduled to run periodically using cron jobs or database jobs to detect fraud continuously.

Disadvantages of Fraud Detection Using SQL

1. **Limited to Known Patterns**
   a. SQL works best with rules-based detection. It struggles to detect unknown or evolving fraud patterns (unlike machine learning approaches).
2. **Lacks Advanced Analytics**
   a. No built-in support for statistical modeling, anomaly detection, or machine learning unless paired with external tools.
3. **Scalability Issues**
   a. Complex queries on large datasets can become slow and resource-intensive, especially without proper indexing or optimization.
4. **Hard to Maintain Complex Rules**
   a. As fraud detection rules grow in complexity, SQL queries become harder to read, maintain, and debug.
5. **No Real-Time Alerts by Default**
   a. SQL is not inherently event-driven. Without additional tools, it doesn't provide real-time fraud alerts or streaming data analysis.
6. **Security Risks**

a. Poorly written queries or over-permissioned accounts can expose sensitive data or create performance bottlenecks.

When to Use SQL for Fraud Detection

- Ideal for small to medium-sized datasets.
- Good for initial screening or prototyping rule-based fraud detection.
- Useful when fraud patterns are known and consistent.

Future Scope of Fraud Detection Using SQL

Continued Use as a Foundational Layer

- **Data Preprocessing & Feature Engineering:** SQL will remain essential for cleaning, transforming, and aggregating transactional data before feeding it into more advanced systems (like machine learning models).
- **Rule-Based Triggers:** Many industries will still rely on SQL to define hard-coded rules for detecting known fraud patterns, such as:
    - Sudden large withdrawals
    - Multiple failed login attempts
    - Rapid succession of transactions

2. Integration with Modern Technologies

- **SQL + AI/ML Integration:**
    - SQL will be used to extract features from databases, while Python or R can run machine learning models.
    - Tools like BigQuery ML, Snowflake, and Azure Synapse allow running ML models *using SQL itself*, bridging the gap.
- **Use in Real-Time Pipelines:**
    - With platforms like Apache Kafka and Flink, SQL is evolving into **streaming SQL** for real-time fraud detection (e.g., KSQL in Kafka).

3. Expansion with Cloud Data Warehouses

- Platforms like **Snowflake, BigQuery, and Amazon Redshift** are enabling advanced analytics directly through SQL.
- These cloud data warehouses provide scalability, allowing more complex fraud detection logic to be handled using SQL at larger scales.

4. Visualization and Reporting

- SQL will continue to be used to build dashboards and alerts in tools like **Tableau, Power BI**, and **Looker** for fraud monitoring.
- Analysts and fraud teams rely heavily on SQL to generate insights, KPIs, and reports from transactional data.

5. Low-Code / No-Code Enhancements

- Future tools may offer visual interfaces that generate SQL queries in the background, allowing non-technical users to set up fraud detection workflows using SQL-like logic.

Conclusion

Fraud detection using SQL continues to be a valuable and practical approach, especially in environments where transactional data is stored in relational databases and where known patterns of fraudulent behavior can be identified through clearly defined rules. SQL offers a straightforward, cost-effective method to implement fraud detection logic, enabling organizations to flag suspicious transactions, detect anomalies, and generate alerts based on business-specific criteria. Its accessibility and integration with existing database systems make it an ideal choice for initial screening processes and for building real-time dashboards and reports. While SQL excels at rule-based detection, it does face limitations when dealing with large-scale, dynamic, and evolving fraud patterns that require predictive capabilities or advanced anomaly detection. In such cases, SQL is best used in combination with more sophisticated tools such as machine learning algorithms or real-time data processing platforms. Nevertheless, the future of fraud detection still sees a prominent role for SQL, not as a standalone solution, but as a critical component of a multi-layered fraud detection system. Whether through data preprocessing, feature engineering, or integration into cloud data platforms, SQL continues to support the foundation of more complex analytical pipelines. As the threat of fraud becomes more advanced, SQL will remain a reliable tool for extracting meaningful patterns from structured data and enhancing the effectiveness of broader fraud detection strategies.

Biblogaphy

- Chandola, V., Banerjee, A., & Kumar, V. (2009). *Anomaly detection: A survey*. ACM Computing Surveys (CSUR), 41(3), 1-58. https://doi.org/10.1145/1541880.1541882

- Bolton, R. J., & Hand, D. J. (2002). *Statistical fraud detection: A review*. Statistical Science, 17(3), 235–255. https://doi.org/10.1214/ss/1042727940
- Microsoft. (2022). *Use SQL to detect fraud patterns in financial transactions*. Microsoft Learn. Retrieved from https://learn.microsoft.com/
- IBM. (2020). *SQL techniques for detecting data anomalies and fraud*. IBM Documentation. Retrieved from https://www.ibm.com/docs/en
- Redgate. (2021). *SQL Best Practices for Detecting Anomalies*. Redgate Hub. Retrieved from https://www.red-gate.com/hub/
- Mahendra, R., & Rani, S. (2021). *Anomaly Detection for Fraudulent Transactions Using SQL and Machine Learning*. International Journal of Computer Applications, 183(9), 21–27. https://doi.org/10.5120/ijca2021921361
- Google Cloud. (2023). *BigQuery ML for Fraud Detection*. Google Cloud Documentation. Retrieved from https://cloud.google.com/bigquery-ml/docs
- Rajput, A., & Thakur, R. (2020). *Comparative study on fraud detection techniques in e-commerce using SQL and machine learning*. Journal of Emerging Technologies and Innovative Research, 7(5), 112–118.
- Singh, A., & Kaur, P. (2019). *Role of SQL in detecting fraud patterns in banking datasets*. International Journal of Scientific Research in Computer Science and Engineering, 7(1), 25–30.