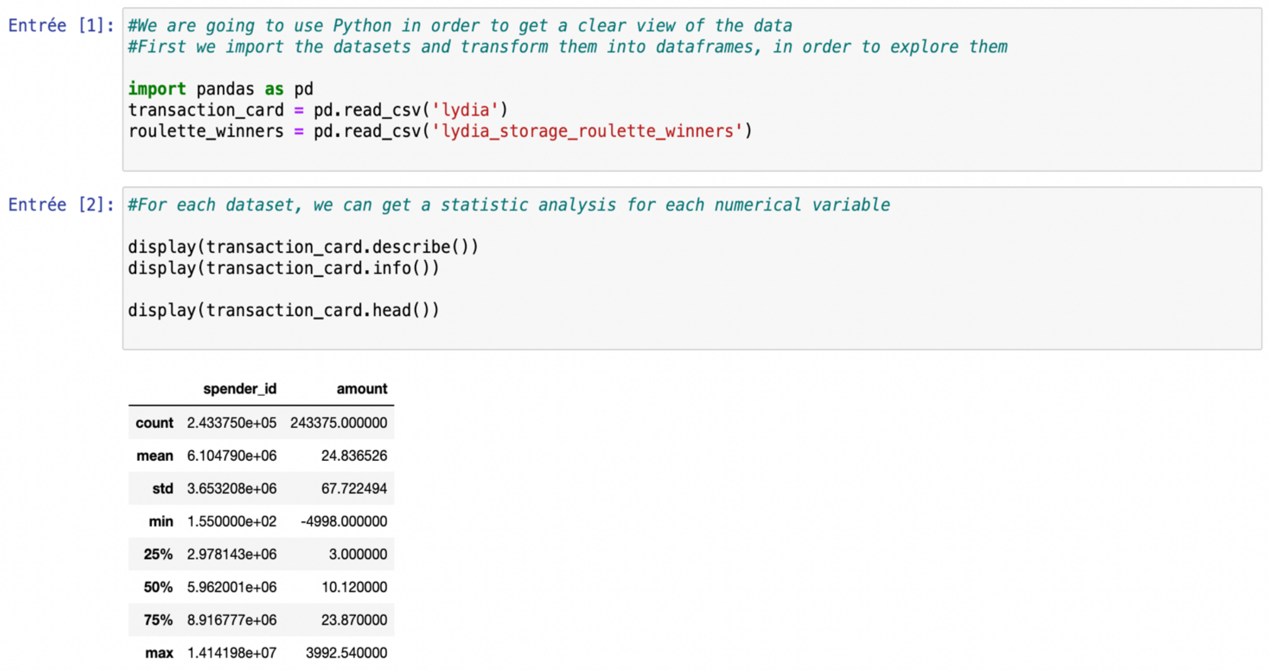
Lydia Data Analyst Case

Par Thomas ZILLIOX

Première étape : étude des datasets à l'aide du langage de programmation Python



Entrée [1]:

```
#We are going to use Python in order to get a clear view of the data
#First we import the datasets and transform them into dataframes, in order to explore them

import pandas as pd
transaction_card = pd.read_csv('lydia')
roulette_winners = pd.read_csv('lydia_storage_roulette_winners')
```

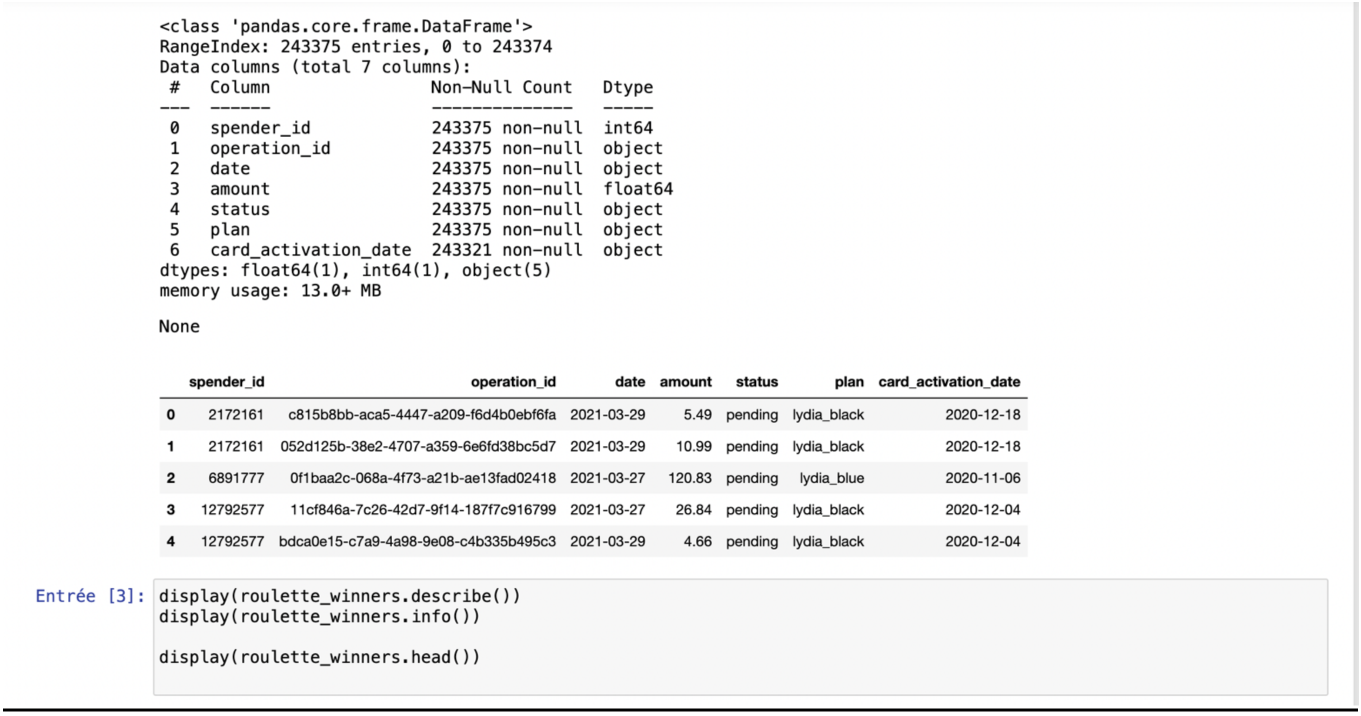
Entrée [2]:

```
#For each dataset, we can get a statistic analysis for each numerical variable

display(transaction_card.describe())
display(transaction_card.info())

display(transaction_card.head())
```

	spender_id	amount
count	2.433750e+05	243375.000000
mean	6.104790e+06	24.836526
std	3.653208e+06	67.722494
min	1.550000e+02	-4998.000000
25%	2.978143e+06	3.000000
50%	5.962001e+06	10.120000
75%	8.916777e+06	23.870000
max	1.414198e+07	3992.540000



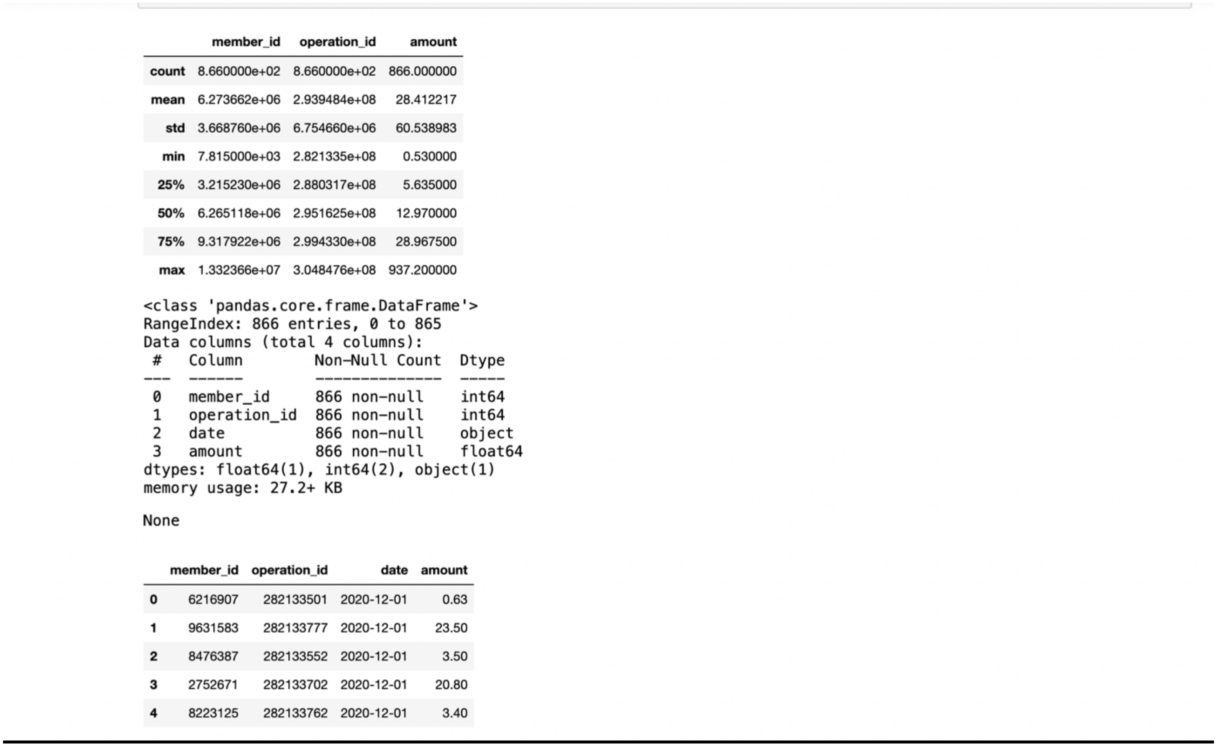
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 243375 entries, 0 to 243374
Data columns (total 7 columns):
 #   Column                Non-Null Count   Dtype  
---  ------                --------------   -----  
 0   spender_id            243375 non-null  int64  
 1   operation_id          243375 non-null  object 
 2   date                  243375 non-null  object 
 3   amount                243375 non-null  float64
 4   status                243375 non-null  object 
 5   plan                  243375 non-null  object 
 6   card_activation_date  243321 non-null  object 
dtypes: float64(1), int64(1), object(5)
memory usage: 13.0+ MB

None
```

	spender_id	operation_id	date	amount	status	plan	card_activation_date
0	2172161	c815b8bb-aca5-4447-a209-f6d4b0ebf6fa	2021-03-29	5.49	pending	lydia_black	2020-12-18
1	2172161	052d125b-38e2-4707-a359-6e6fd38bc5d7	2021-03-29	10.99	pending	lydia_black	2020-12-18
2	6891777	0f1baa2c-068a-4f73-a21b-ae13fad02418	2021-03-27	120.83	pending	lydia_blue	2020-11-06
3	12792577	11cf846a-7c26-42d7-9f14-187f7c916799	2021-03-27	26.84	pending	lydia_black	2020-12-04
4	12792577	bdca0e15-c7a9-4a98-9e08-c4b335b495c3	2021-03-29	4.66	pending	lydia_black	2020-12-04

```
Entrée [3]: display(roulette_winners.describe())
            display(roulette_winners.info())

            display(roulette_winners.head())
```

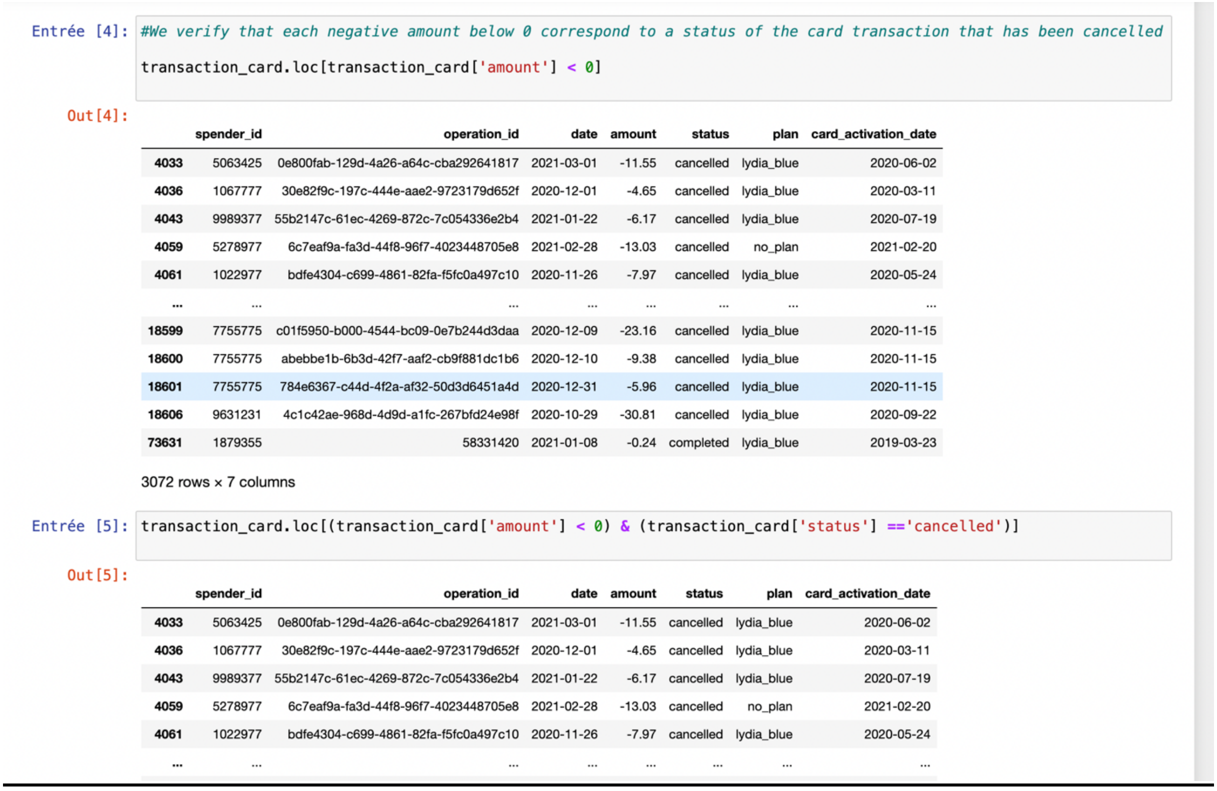


	member_id	operation_id	amount
count	8.660000e+02	8.660000e+02	866.000000
mean	6.273662e+06	2.939484e+08	28.412217
std	3.668760e+06	6.754660e+06	60.538983
min	7.815000e+03	2.821335e+08	0.530000
25%	3.215230e+06	2.880317e+08	5.635000
50%	6.265118e+06	2.951625e+08	12.970000
75%	9.317922e+06	2.994330e+08	28.967500
max	1.332366e+07	3.048476e+08	937.200000

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 866 entries, 0 to 865
Data columns (total 4 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   member_id     866 non-null    int64  
 1   operation_id  866 non-null    int64  
 2   date          866 non-null    object 
 3   amount        866 non-null    float64
dtypes: float64(1), int64(2), object(1)
memory usage: 27.2+ KB

None
```

	member_id	operation_id	date	amount
0	6216907	282133501	2020-12-01	0.63
1	9631583	282133777	2020-12-01	23.50
2	8476387	282133552	2020-12-01	3.50
3	2752671	282133702	2020-12-01	20.80
4	8223125	282133762	2020-12-01	3.40



Entrée [4]:

```
#We verify that each negative amount below 0 correspond to a status of the card transaction that has been cancelled

transaction_card.loc[transaction_card['amount'] < 0]
```

Out[4]:

	spender_id	operation_id	date	amount	status	plan	card_activation_date
4033	5063425	0e800fab-129d-4a26-a64c-cba292641817	2021-03-01	-11.55	cancelled	lydia_blue	2020-06-02
4036	1067777	30e82f9c-197c-444e-aae2-9723179d652f	2020-12-01	-4.65	cancelled	lydia_blue	2020-03-11
4043	9989377	55b2147c-61ec-4269-872c-7c054336e2b4	2021-01-22	-6.17	cancelled	lydia_blue	2020-07-19
4059	5278977	6c7eaf9a-fa3d-44f8-96f7-4023448705e8	2021-02-28	-13.03	cancelled	no_plan	2021-02-20
4061	1022977	bdfe4304-c699-4861-82fa-f5fc0a497c10	2020-11-26	-7.97	cancelled	lydia_blue	2020-05-24
...	...	...	...	...	...	...	...
18599	7755775	c01f5950-b000-4544-bc09-0e7b244d3daa	2020-12-09	-23.16	cancelled	lydia_blue	2020-11-15
18600	7755775	abebbe1b-6b3d-42f7-aaf2-cb9f881dc1b6	2020-12-10	-9.38	cancelled	lydia_blue	2020-11-15
18601	7755775	784e6367-c44d-4f2a-af32-50d3d6451a4d	2020-12-31	-5.96	cancelled	lydia_blue	2020-11-15
18606	9631231	4c1c42ae-968d-4d9d-a1fc-267bfd24e98f	2020-10-29	-30.81	cancelled	lydia_blue	2020-09-22
73631	1879355	58331420	2021-01-08	-0.24	completed	lydia_blue	2019-03-23

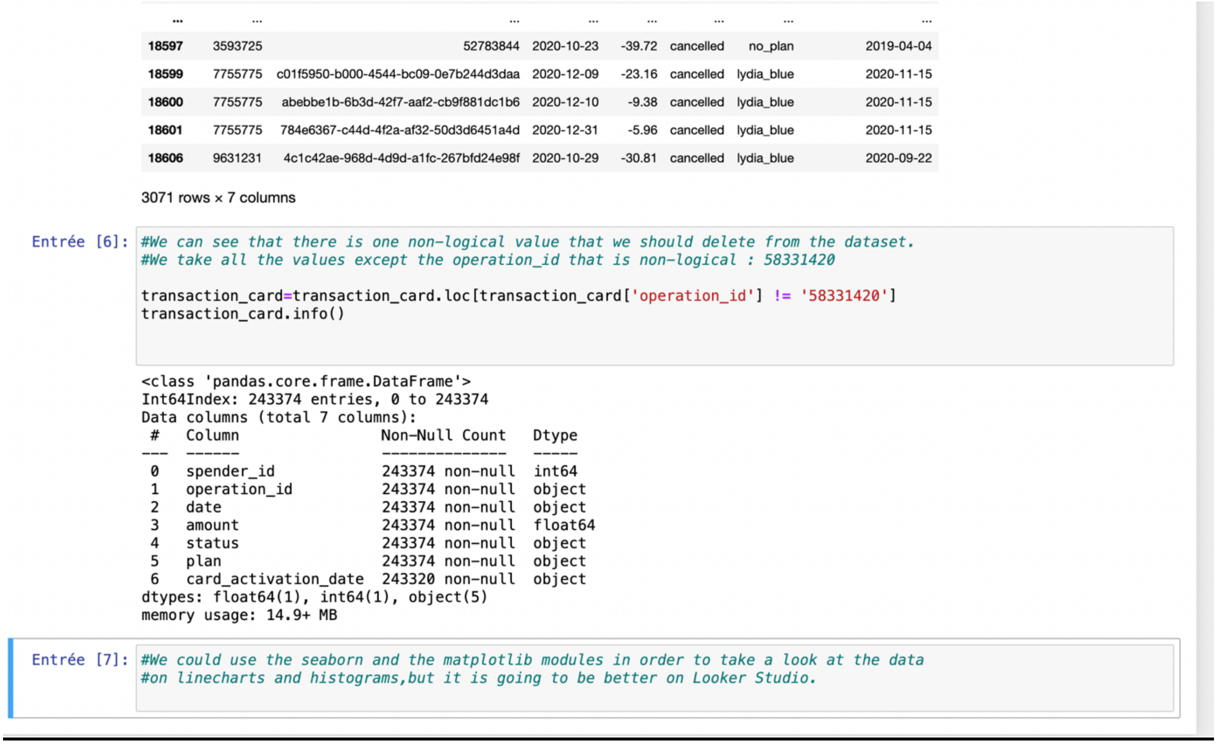
3072 rows × 7 columns

Entrée [5]:

```
transaction_card.loc[(transaction_card['amount'] < 0) & (transaction_card['status'] =='cancelled')]
```

Out[5]:

	spender_id	operation_id	date	amount	status	plan	card_activation_date
4033	5063425	0e800fab-129d-4a26-a64c-cba292641817	2021-03-01	-11.55	cancelled	lydia_blue	2020-06-02
4036	1067777	30e82f9c-197c-444e-aae2-9723179d652f	2020-12-01	-4.65	cancelled	lydia_blue	2020-03-11
4043	9989377	55b2147c-61ec-4269-872c-7c054336e2b4	2021-01-22	-6.17	cancelled	lydia_blue	2020-07-19
4059	5278977	6c7eaf9a-fa3d-44f8-96f7-4023448705e8	2021-02-28	-13.03	cancelled	no_plan	2021-02-20
4061	1022977	bdfe4304-c699-4861-82fa-f5fc0a497c10	2020-11-26	-7.97	cancelled	lydia_blue	2020-05-24
...	...	...	...	...	...	...	...
18597	3593725	52783844	2020-10-23	-39.72	cancelled	no_plan	2019-04-04
18599	7755775	c01f5950-b000-4544-bc09-0e7b244d3daa	2020-12-09	-23.16	cancelled	lydia_blue	2020-11-15
18600	7755775	abebbe1b-6b3d-42f7-aaf2-cb9f881dc1b6	2020-12-10	-9.38	cancelled	lydia_blue	2020-11-15
18601	7755775	784e6367-c44d-4f2a-af32-50d3d6451a4d	2020-12-31	-5.96	cancelled	lydia_blue	2020-11-15
18606	9631231	4c1c42ae-968d-4d9d-a1fc-267bfd24e98f	2020-10-29	-30.81	cancelled	lydia_blue	2020-09-22



3071 rows × 7 columns

Entrée [6]:

```
#We can see that there is one non-logical value that we should delete from the dataset.
#We take all the values except the operation_id that is non-logical : 58331420

transaction_card=transaction_card.loc[transaction_card['operation_id'] != '58331420']
transaction_card.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 243374 entries, 0 to 243374
Data columns (total 7 columns):
 #   Column                Non-Null Count   Dtype  
---  ------                --------------   -----  
 0   spender_id            243374 non-null  int64  
 1   operation_id          243374 non-null  object 
 2   date                  243374 non-null  object 
 3   amount                243374 non-null  float64
 4   status                243374 non-null  object 
 5   plan                  243374 non-null  object 
 6   card_activation_date  243320 non-null  object 
dtypes: float64(1), int64(1), object(5)
memory usage: 14.9+ MB
```

Entrée [7]:

```
#We could use the seaborn and the matplotlib modules in order to take a look at the data
#on linecharts and histograms,but it is going to be better on Looker Studio.
```

Deuxième étape : nettoyage des données à l'aide du langage SQL sur BigQuery

1) Recherche de la clé primaire

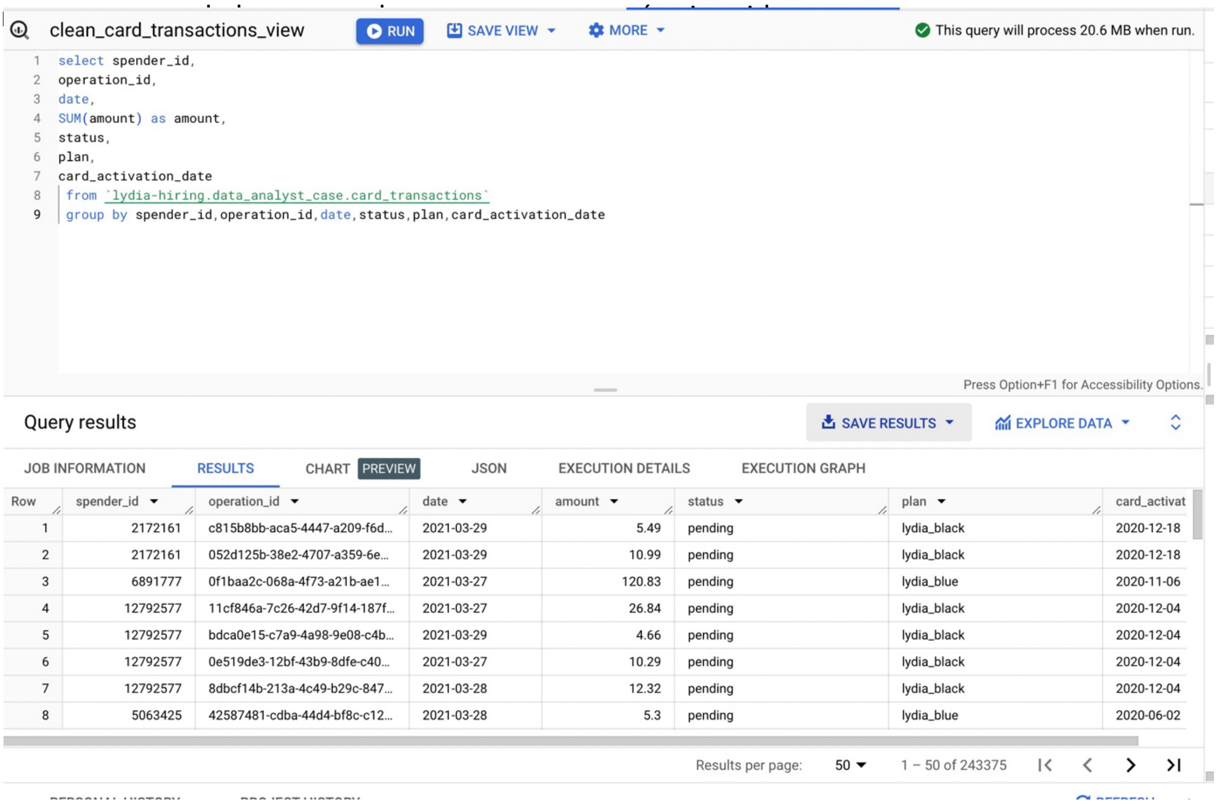
```
SELECT operation_id, COUNT(operation_id) FROM `lydia-
hiring.data_analyst_case.card_transactions`
group by operation_id
HAVING COUNT(operation_id) > 1
LIMIT 1000
```

=> Il y a 676 opération_id en double, cependant :

```
SELECT *
FROM `lydia-hiring.data_analyst_case.card_transactions`
WHERE operation_id IN (
    SELECT operation_id
    FROM `lydia-hiring.data_analyst_case.card_transactions`
    GROUP BY operation_id
    HAVING COUNT(operation_id) > 1
)
```

Présence de deux montants différents pour la même opération dans plusieurs dizaines de cas.

=> Après vérification auprès de Monsieur Arthur Du Peloux, il n'y a à priori pas d'erreurs donc

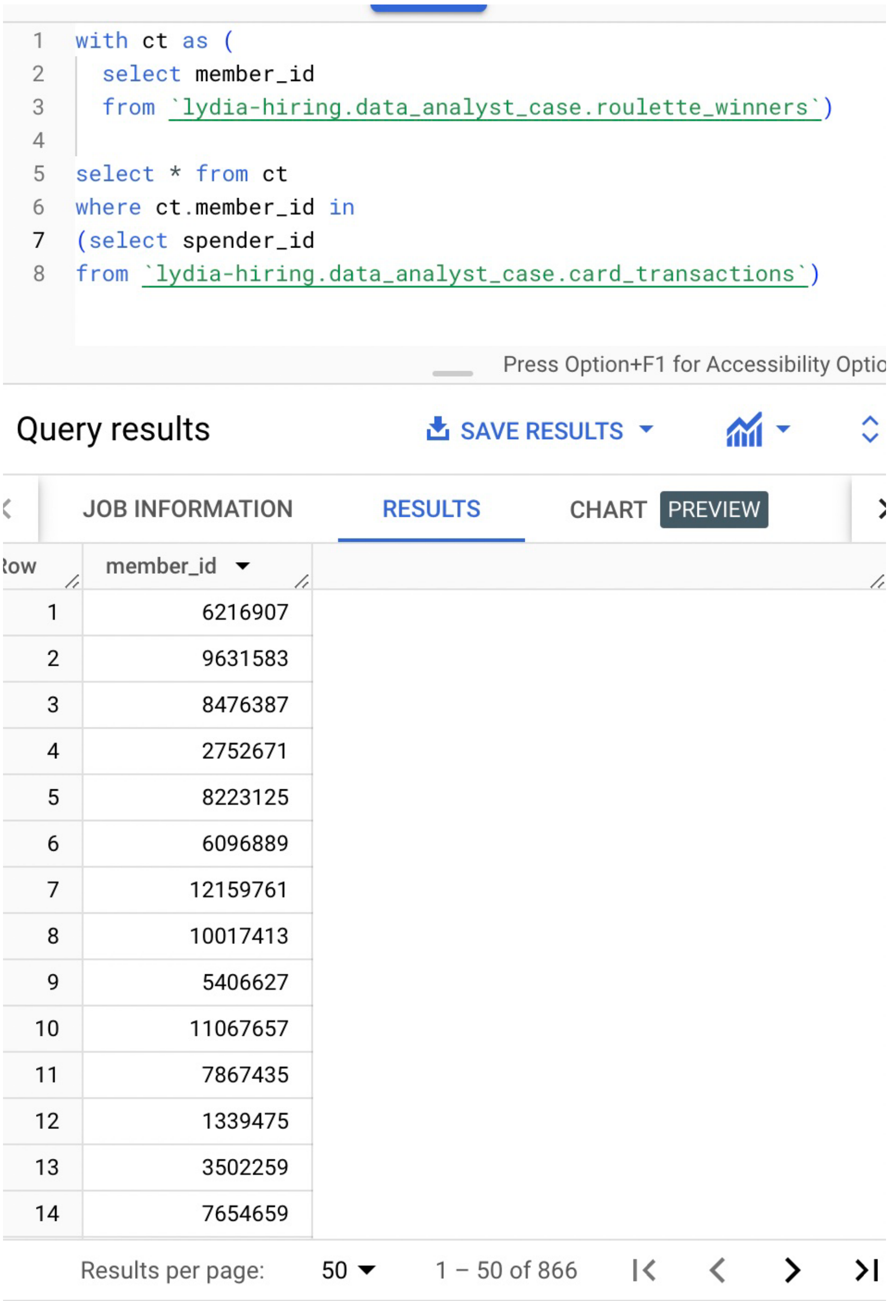

clean_card_transactions_view
RUN
SAVE VIEW
MORE
This query will process 20.6 MB when run.

```
select spender_id,
operation_id,
date,
SUM(amount) as amount,
status,
plan,
card_activation_date
 from `lydia-hiring.data_analyst_case.card_transactions`
 group by spender_id,operation_id,date,status,plan,card_activation_date
```

Row	spender_id	operation_id	date	amount	status	plan	card_activat
1	2172161	c815b8bb-aca5-4447-a209-f6d...	2021-03-29	5.49	pending	lydia_black	2020-12-18
2	2172161	052d125b-38e2-4707-a359-6e...	2021-03-29	10.99	pending	lydia_black	2020-12-18
3	6891777	0f1baa2c-068a-4f73-a21b-ae1...	2021-03-27	120.83	pending	lydia_blue	2020-11-06
4	12792577	11cf846a-7c26-42d7-9f14-187f...	2021-03-27	26.84	pending	lydia_black	2020-12-04
5	12792577	bdca0e15-c7a9-4a98-9e08-c4b...	2021-03-29	4.66	pending	lydia_black	2020-12-04
6	12792577	0e519de3-12bf-43b9-8dfe-c40...	2021-03-27	10.29	pending	lydia_black	2020-12-04
7	12792577	8dbcf14b-213a-4c49-b29c-847...	2021-03-28	12.32	pending	lydia_black	2020-12-04
8	5063425	42587481-cdba-44d4-bf8c-c12...	2021-03-28	5.3	pending	lydia_blue	2020-06-02

=> Après comparaison des dates présentes dans les deux tables, on remarque que la table transaction_card commence deux mois avant celle de la roulette_winners table et termine deux mois après.
(01-10-2020/01-12-2020 et 09-01-2021 / 29-03-2021)

=> On vérifie bien que tous les member_id de la roulette_winners table soient dans la card_transaction_table en tant que spender_id.



```
with ct as (
  select member_id
  from `lydia-hiring.data_analyst_case.roulette_winners`)

select * from ct
where ct.member_id in
(select spender_id
from `lydia-hiring.data_analyst_case.card_transactions`)
```

Row	member_id
1	6216907
2	9631583
3	8476387
4	2752671
5	8223125
6	6096889
7	12159761
8	10017413
9	5406627
10	11067657
11	7867435
12	1339475
13	3502259
14	7654659

Results per page: 50 1 – 50 of 866

=> Enfin, on peut réaliser les dashboards sur Looker Studio. (Voir pdf)