Question 1:

a. By referring to Question1-Material.xlsx, provide us with a normalized data model design to minimize data redundancy.
Based on your normalized model, answer the below questions by providing us with the respective SQL queries:
b. Which are the 10 largest completed deposit transactions? Extract the amount, customer email, customer brand name and calendar date.
c. What is the total number and amount of failed deposit transactions per brand? Extract total number, amount, customer brand name, paymentstatus.
d. How much daily turnover and accounting revenue did each brand generate per product in the first 6 days of the year? Extract turnover, accountingrevenue, brand.
e. What is the average gamewin per product? Extract the average gamewin and product.
f. Which customers had a lifetime total turnover of 500 EUR or more and what was this total turnover amount? Extract turnover, customeremail

Solution

First let's have a look at data, we've given two tables in excel file.

Deposit Transactions

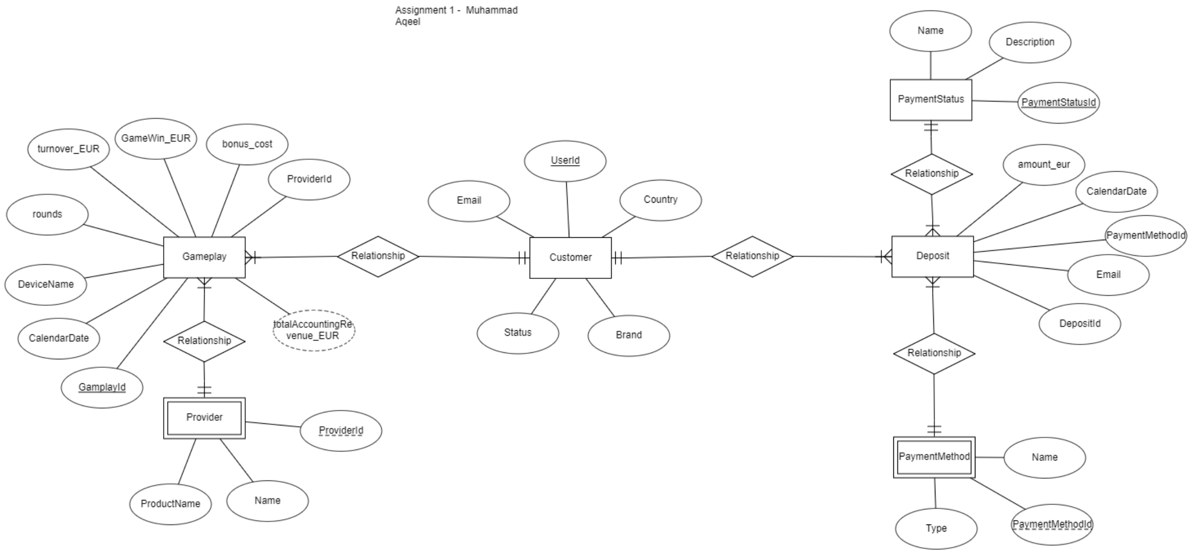
CalendarDate
CustomerEmail
CustomerBrandName
CustomerCountry
CustomerAccountStatus
PaymentMethodName
PaymentMethodType
PaymentStatusName
PaymentStatusDescription
amount_eur

Gamewin Transactions

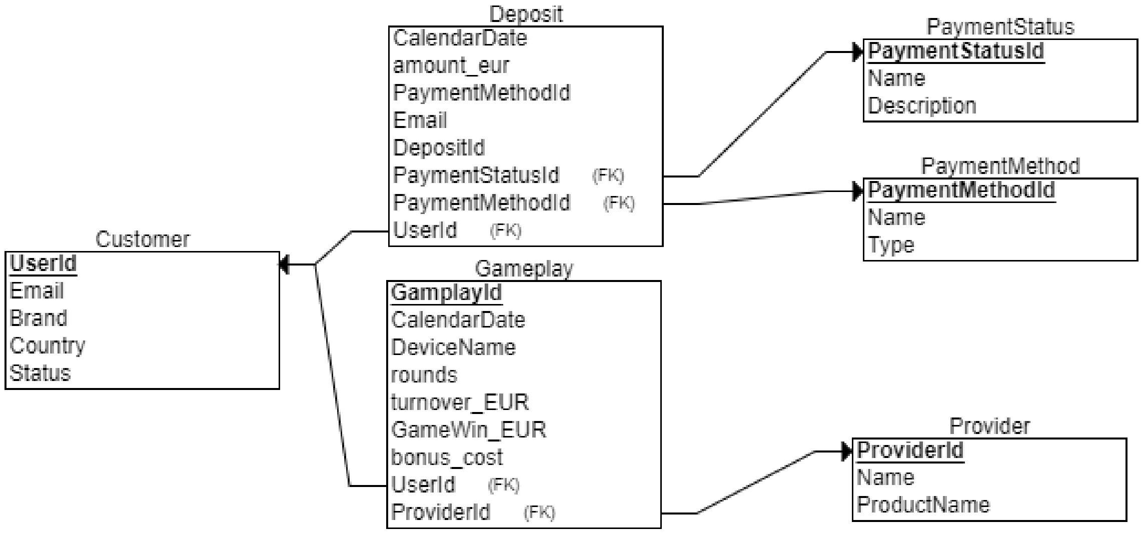
CalendarDate
CustomerEmail
CustomerBrand
CustomerCountry
CustomerStatus
ProviderName
ProviderProductName
DeviceName
rounds

turnover_EUR
gameWin_EUR
bonus cost
totalAccountingRevenue_EUR

We can normalize above two tables, following is ERD after normalization.



Relational Schema of the diagram.


Deposit
CalendarDate
amount_eur
PaymentMethodId
Email
DepositId
PaymentStatusId (FK)
PaymentMethodId (FK)
UserId (FK)

PaymentStatus
PaymentStatusId
Name
Description

PaymentMethod
PaymentMethodId
Name
Type

Customer
UserId
Email
Brand
Country
Status

Gameplay
GamplayId
CalendarDate
DeviceName
rounds
turnover_EUR
GameWin_EUR
bonus_cost
UserId (FK)
ProviderId (FK)

Provider
ProviderId
Name
ProductName

Now as we've dumped our data in Postgres Database, Let's write and execute queries.

Note : as there was some issue in email data in one table, all of the email were ending with .om so we changed it to .com to clean data and then dumped.

In [22]:

```
# b. Which are the 10 largest completed deposit transactions?
# Extract the amount, customer email, customer brand name and calendar date.
query = '''select amount_eur ,
email ,
brand  ,
calendar_date
from deposit
left join customer on deposit.customer_id=customer.customer_id
order by amount_eur desc limit 10;'''

execute_query(query)
```

```
   amount_eur                          email     brand  calendar_date
0   99.954219  betsson7@testbetssongroup.com  BrandABC       20210105
1   99.393521  betsson6@testbetssongroup.com  BrandGHI       20210101
2   98.550624  betsson3@testbetssongroup.com  BrandABC       20210102
3   96.542504  betsson3@testbetssongroup.com  BrandABC       20210105
4   95.221557  betsson3@testbetssongroup.com  BrandABC       20210102
5   93.265848   betsson@testbetssongroup.com  BrandABC       20210104
6   92.778977   betsson@testbetssongroup.com  BrandABC       20210105
7   89.286755   betsson@testbetssongroup.com  BrandABC       20210102
8   88.798928  betsson4@testbetssongroup.com  BrandDEF       20210102
9   85.767210  betsson5@testbetssongroup.com  BrandGHI       20210102
```

In [36]:

```
#c. What is the total number and amount of failed deposit transactions per brand?
# Extract  total number, amount, customer brand name, paymentstatus.


query = '''select count(deposit_id),
sum(d.amount_eur) ,
c.brand ,
'Failed'  from deposit d
inner join payment_status ps
on d.payment_status_id = ps.payment_status_id and  ps.name = 'Failed'
left join customer c on d.customer_id = c.customer_id
group by c.brand;'''

execute_query(query)
```

```
   count         sum     brand ?column?
0      3  179.531986  BrandABC   Failed
1      3  137.549155  BrandDEF   Failed
2      1   85.767210  BrandGHI   Failed
```

In [31]:

```
# d. How much daily turnover and accounting revenue did each brand generate per p
# Extract turnover, accountingrevenue, brand.

query = '''select sum(g.turnover_eur) as sum_turnover , sum(gamewin_eur)  + sum(g
from gameplay g
inner join customer c on g.customer_id = c.customer_id and g.calendar_date <= 202
group by c.brand ;'''

execute_query(query)
```

```
   sum_turnover  sum_accountingrevenue     brand
0   2452.209478            -572.741053  BrandABC
1   1526.681294            -449.071244  BrandDEF
2   1060.926938            -335.830123  BrandGHI
```

In [32]:

```
# e. What is the average gamewin per product? Extract the average gamewin and pro
query = '''select AVG(g.gamewin_eur),  p.product_name from gameplay g
left join provider p on g.provider_id = p.provider_id
group by p.product_name;  '''

execute_query(query)
```

```
         avg     product_name
0 -42.303916             None
1  -6.182484  Games of Chance
2  -1.308644            Poker
3  -5.562201       Sportsbook
```

In [33]:

```
# f. Which customers had a lifetime total turnover of 500 EUR or more and what wo
# Extract turnover, customeremail, customer

query = '''select
sum(g.turnover_eur),
c.email
from gameplay g
left join customer c on g.customer_id = c.customer_id
group by c.email having sum(g.turnover_eur) >=500 ;'''

execute_query(query)
```

```
          sum                        email
0  567.746808  betsson4@testbetssongroup.com
1  563.241676  betsson7@testbetssongroup.com
2  711.688935  betsson3@testbetssongroup.com
3  906.865703   betsson@testbetssongroup.com
4  545.195125  betsson6@testbetssongroup.com
5  958.934486  betsson2@testbetssongroup.com
6  515.731814  betsson5@testbetssongroup.com
```