3_ARPU_ARPPU

December 12, 2022

SQL request project Yandex Practicum (personal visuzalization implementation in Python)

ARPPU visualization

[ ]:
```
import pandas as pd
%load_ext sql
%sql postgresql://postgres:sqltest123@localhost/1
```

[ ]:
```
%%sql result <<
SELECT SUM(tso.total_amt) / COUNT(DISTINCT tsu.user_id) AS ARPU
FROM tools_shop.users AS tsu
LEFT JOIN tools_shop.orders AS tso ON tsu.user_id = tso.user_id
```

[3]:
```
df = result.DataFrame()
display(df.head(11))
```

```
                 arpu
0  41.1890941132294458
```

[ ]:
```
%%sql result2 <<
SELECT EXTRACT(YEAR FROM o.created_at) AS year,
ROUND(sum(o.total_amt)/COUNT (DISTINCT o.user_id), 2)::float AS ARPPU
FROM tools_shop.orders o
GROUP BY 1
```

[8]:
```
df2 = result2.DataFrame()
display(df2.head(11))
```

```
   year   arppu
0  2016  264.61
1  2017  266.38
2  2018  270.71
3  2019  265.21
4  2020  282.91
5  2021  266.95
```

[23]:
```
import pandas as pd
import matplotlib.pyplot as plt
plot = df2.sort_values(by='year').plot(
```

```
    x='year',
    y='arppu',
    kind='bar',
    figsize=(10, 8),
    color = 'green',
    title='ARPPU per year')
for p in plot.patches:
    plot.annotate(format(p.get_height(), '.2f'),
                  (p.get_x() + p.get_width() / 2., p.get_height()),
                  ha = 'center', va = 'center',
                  xytext = (0, 9),
                  textcoords = 'offset points')
plt.show()
```