SQL Request from Yandex Practicum course project.

Implementation in Python with usage of visualization.

ARPPU visualization

In [1]:

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

In [ ]:

```
%%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
```

In [3]:

```
#displaying results for ARPU SQL request
df = result.DataFrame()
display(df.head(11))
```

	arpu
0	41.1890941132294458

In [4]:

```
%%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
```

```
 * postgresql://postgres:***@localhost/1
6 rows affected.
Returning data to local variable result2
```

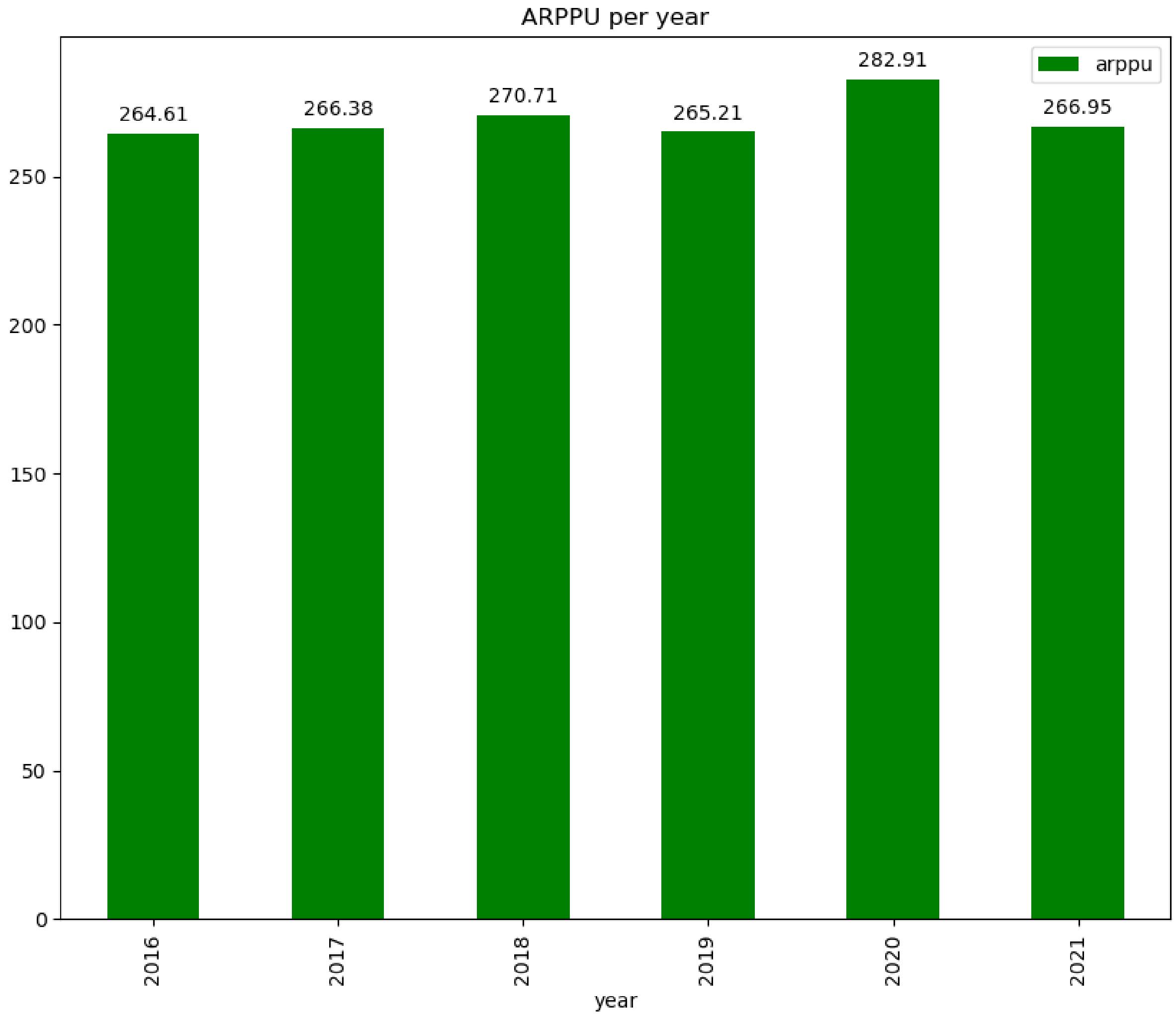
In [5]:

```
#displaying results for ARPPU SQL request
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

In [6]:

```
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()
```


ARPPU per year
arppu
year