

Comparing rentals of two different months

José Luis DORADO LADERA
Adrián Carcelen

20 de juny de 2022

Goal

- For every customer, we aim to compare the number of rentals between two different months.

Goal

- ▶ For every customer, we aim to compare the number of rentals between two different months.
- ▶ We already have a function that gives us the number of rentals in a given month for each customer.

Goal

- ▶ For every customer, we aim to compare the number of rentals between two different months.
- ▶ We already have a function that gives us the number of rentals in a given month for each customer.
- ▶ We will call that functions two times, one for each month we want to compare and use the information to make the comparison.

Goal

- ▶ For every customer, we aim to compare the number of rentals between two different months.
- ▶ We already have a function that gives us the number of rentals in a given month for each customer.
- ▶ We will call that functions two times, one for each month we want to compare and use the information to make the comparison.

Code

The screenshot shows a Jupyter Notebook with the following code and output:

```
def compare_rentals(df_month1, df_month2):
    merged_df = df_month1.merge(df_month2, on='customer_id', suffixes=('_month1', '_month2'), how='outer').fillna(0)
    return pd.DataFrame({
        'customer_id': merged_df['customer_id'],
        'difference': (merged_df['number of rentals month2'] - merged_df['number of rentals month1']).apply(int)
    })
```

Below the function definition, the notebook shows the execution of the function with two dataframes:

```
df_month1 = rental_count_month(rentals_month(cnx, 5, 2005))
df_month2 = rental_count_month(rentals_month(cnx, 6, 2005))
compare_rentals(df_month1, df_month2)
```

The output is a dataframe with 598 rows and 2 columns:

customer_id	difference
0	1
1	2
2	3
3	5
4	6
...	...
593	583
594	585
595	591
596	592
597	598

The bottom of the screenshot shows the terminal output:

```
akhrain@akhrain-Laptop:~$
```