

Лабораторна робота №3. Варіант 15

Виконав: ІП-13 Ал Хадам Мурат Резгович

Структури даних Pandas

Мета: Ознайомитись з основними структурами даних бібліотеки Pandas: Series DataFrame, операціями над ними. Навчитись використовувати групування.

Завдання:

- Вивести інформацію про набір даних, основні статистичні характеристики, типи ознак. Які ознаки є категоріальними, а які – кількісними?
- За допомогою зрізів зробити копію частини набору даних і зберегти до нового об'єкту DataFrame. Призначити йому власні індекси рядків та стовпців. Додати новий рядок.
- Використовуючи початковий DataFrame: а) Знайти кількість товарів у кожному блоці; б) Вивести товари з середнім пріоритетом, які планується відправити літаком та які були замовлені чоловіками. Знайти загальну вартість цих товарів. в) Додати новий стовпець, який покаже, чи знижка на товар більше, ніж по 10 доларів на кожну попередню покупку (наприклад, знижка=20, попередніх покупок – 3, не перевищує). г) Додати новий стовпець, який містить середній рейтинг покупок для товарів у даному блоці.

1

```
In [30]:
import pandas as pd

df = pd.read_csv('CommerceShipping.csv')
df

Out[30]:
```

	ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Prior_discount
0	1	D	Flight	4	2	177	3	0
1	2	F	Flight	4	5	216	2	0
2	3	A	Flight	2	2	183	4	0
3	4	B	Flight	3	3	176	4	0
4	5	C	Flight	2	2	184	3	0
...
1452	1453	D	Ship	4	5	143	3	0
1453	1454	F	Ship	6	3	223	6	0
1454	1455	A	Ship	3	5	230	2	0
1455	1456	B	Ship	4	5	241	4	0
1456	1457	C	Ship	3	5	156	3	0

1457 rows x 12 columns

```
Основні статистичні характеристики

In [31]:
df.describe()

Out[31]:
```

	ID	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Discount_offered	Weight_in_gms
count	1457.000000	1457.000000	1457.000000	1457.000000	1457.000000	1457.000000	1457.000000
mean	729.000000	3.732327	2.986960	194.671929	3.277282	32.793411	2345.960879
std	420.743984	1.018917	1.404162	45.940277	1.424370	18.896276	945.996401
min	1.000000	2.000000	1.000000	96.000000	2.000000	1.000000	1004.000000
25%	365.000000	3.000000	2.000000	157.000000	2.000000	16.000000	1549.000000
50%	729.000000	4.000000	3.000000	190.000000	3.000000	33.000000	2200.000000
75%	1093.000000	4.000000	4.000000	234.000000	4.000000	49.000000	3067.000000
max	1457.000000	7.000000	5.000000	294.000000	10.000000	65.000000	7846.000000

```
Типи ознак

In [32]:
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1457 entries, 0 to 1456
Data columns (total 12 columns):
#   Column                Non-Null Count  Dtype  
---  --
0    ID                     1457 non-null  int64  
1    Warehouse_block       1457 non-null  object  
2    Mode_of_Shipment      1457 non-null  object  
3    Customer_care_calls   1457 non-null  int64  
4    Customer_rating       1457 non-null  int64  
5    Cost_of_the_Product   1457 non-null  int64  
6    Prior_purchases       1457 non-null  int64  
7    Product_importance    1457 non-null  object  
8    Gender                1457 non-null  object  
9    Discount_offered      1457 non-null  int64  
10   Weight_in_gms         1457 non-null  int64  
11   Reached_on_Time_Y.N   1457 non-null  int64  
dtypes: int64(8), object(4)
memory usage: 136.7+ KB

Які ознаки є категоріальними, а які – кількісними?

In [33]:
for col in df.columns:
    if df[col].dtype == 'object':
        print(f'{col} is categorical')
    else:
        print(f'{col} is not categorical')

ID is not categorical
Warehouse_block is categorical
Mode_of_Shipment is categorical
Customer_care_calls is not categorical
Customer_rating is not categorical
Cost_of_the_Product is not categorical
Prior_purchases is not categorical
Product_importance is categorical
Gender is categorical
Discount_offered is not categorical
Weight_in_gms is not categorical
Reached_on_Time_Y.N is not categorical
```

2

За допомогою зрізів зробити копію частини набору даних і зберегти до нового об'єкту DataFrame. Призначити йому власні індекси рядків та стовпців. Додати новий рядок.

```
In [34]:
df_copy = df.iloc[1:6, 1:5]
df_copy.reset_index(drop=True)
df_copy

Out[34]:
```

	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating
1	F	Flight	4	5
2	A	Flight	2	2
3	B	Flight	3	3
4	C	Flight	2	2
5	F	Flight	3	1

```
In [35]:
df_copy.index = ['11', '12', '13', '14', '15']
df_copy.columns = ['Warehouse', 'Mode', 'Calls', 'Rating']
df_copy

Out[35]:
```

	Warehouse	Mode	Calls	Rating
11	F	Flight	4	5
12	A	Flight	2	2
13	B	Flight	3	3
14	C	Flight	2	2
15	F	Flight	3	1

```
In [36]:
new_row = {
    'Warehouse' : 'AA',
    'Mode' : 'Test Flight',
    'Calls' : 100,
    'Rating' : 101
}
df_copy.loc['new_row'] = new_row
df_copy

Out[36]:
```

	Warehouse	Mode	Calls	Rating
11	F	Flight	4	5
12	A	Flight	2	2
13	B	Flight	3	3
14	C	Flight	2	2
15	F	Flight	3	1
new_row	AA	Test Flight	100	101

3

Використовуючи початковий DataFrame: а) Знайти кількість товарів у кожному блоці;

```
In [37]:
df

Out[37]:
```

	ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Prior_discount
0	1	D	Flight	4	2	177	3	0
1	2	F	Flight	4	5	216	2	0
2	3	A	Flight	2	2	183	4	0
3	4	B	Flight	3	3	176	4	0
4	5	C	Flight	2	2	184	3	0
...
1452	1453	D	Ship	4	5	143	3	0
1453	1454	F	Ship	6	3	223	6	0
1454	1455	A	Ship	3	5	230	2	0
1455	1456	B	Ship	4	5	241	4	0
1456	1457	C	Ship	3	5	156	3	0

1457 rows x 12 columns

```
In [38]:
total_purchases = df.groupby('Warehouse_block')['Prior_purchases'].sum()
total_purchases

Out[38]:
Warehouse_block
A      820
B      795
C      787
D      806
F     1567
Name: Prior_purchases, dtype: int64
```

б) Вивести товари з середнім пріоритетом, які планується відправити літаком та які були замовлені чоловіками. Знайти загальну вартість цих товарів.

```
In [39]:
purchases_by_flight = df[
    (df['Mode_of_Shipment'] == 'Flight') & (df['Gender'] == 'M') & (df['Product_importance'] == 'medium')]
purchases_by_flight

Out[39]:
```

	ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Prior_discount
3	4	B	Flight	3	3	176	4	0
10	11	C	Flight	3	4	189	2	0
137	138	F	Flight	3	5	162	2	0
151	152	F	Flight	5	3	98	5	0
269	270	F	Flight	4	5	208	3	0
274	275	C	Flight	3	5	229	6	0
280	281	C	Flight	3	1	205	3	0
281	282	F	Flight	5	4	147	3	0
285	286	B	Flight	2	4	181	4	0
407	408	F	Flight	4	5	193	3	0
408	409	D	Flight	3	2	192	3	0
409	410	F	Flight	2	5	166	3	0
410	411	A	Flight	3	2	264	3	0
411	412	B	Flight	3	4	200	2	0
412	413	C	Flight	4	3	142	4	0
421	422	F	Flight	4	2	251	2	0
423	424	B	Flight	4	4	229	2	0
426	427	D	Flight	3	4	167	3	0
548	549	A	Flight	2	4	182	3	0
551	552	F	Flight	3	5	233	3	0
557	558	F	Flight	2	3	198	2	0
558	559	D	Flight	4	2	164	3	0
559	560	F	Flight	5	2	271	3	0
562	563	C	Flight	3	2	137	2	0
682	683	C	Flight	4	4	143	2	0
693	694	B	Flight	3	3	207	7	0
698	699	A	Flight	4	3	264	3	0
701	702	F	Flight	4	2	197	7	0
826	827	C	Flight	5	2	232	2	0
828	829	D	Flight	4	2	230	3	0
833	834	F	Flight	3	4	145	6	0
835	836	F	Flight	3	5	218	4	0
962	963	A	Flight	3	2	180	4	0
973	974	F	Flight	5	4	229	3	0
975	976	B	Flight	5	5	249	3	0
1091	1092	F	Flight	3	5	154	3	0
1094	1095	A	Flight	3	1	185	3	0
1100	1101	A	Flight	5	2	260	2	0
1102	1103	C	Flight	4	1	227	3	0
1110	1111	D	Flight	5	1	196	3	0
1239	1240	B	Flight	4	5	157	3	0
1242	1243	D	Flight	4	1	250	2	0
1245	1246	B	Flight	4	3	258	3	0
1367	1368	F	Flight	5	3	165	2	0
1371	1372	B	Flight	5	4	268	3	0
1376	1377	A	Flight	5	1	274	3	0

```
In [40]:
print('Total cost is', purchases_by_flight['Cost_of_the_Product'].sum())

Total cost is 9272

в) Додати новий стовпець, який показує, чи знижка на товар більше, ніж по 10 доларів на кожну попередню покупку (наприклад, знижка=20, попередніх покупок – 3, не перевищує).

In [41]:
df['is_discount_more_10'] = df.apply(lambda row:
    True if row['Discount_offered'] / row['Prior_purchases'] > 10
    else False, axis=1)
df

Out[41]:
```

	ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Prior_discount	is_discount_more_10
0	1	D	Flight	4	2	177	3	0	False
1	2	F	Flight	4	5	216	2	0	False
2	3	A	Flight	2	2	183	4	0	False
3	4	B	Flight	3	3	176	4	0	False
4	5	C	Flight	2	2	184	3	0	False
...
1452	1453	D	Ship	4	5	143	3	0	False
1453	1454	F	Ship	6	3	223	6	0	False
1454	1455	A	Ship	3	5	230	2	0	False
1455	1456	B	Ship	4	5	241	4	0	False
1456	1457	C	Ship	3	5	156	3	0	False

1457 rows x 13 columns

г) Додати новий стовпець, який містить середній рейтинг покупок для товарів у даному блоці.

```
In [42]:
average_rate_by_block = df.groupby('Warehouse_block')['Customer_rating'].mean()
average_rate_by_block

Out[42]:
Warehouse_block
A      2.938272
B      2.971193
C      3.028807
D      3.061728
F      2.960825
Name: Customer_rating, dtype: float64
```

In [43]:

```
df['Average_rate_by_block'] = df['Warehouse_block'].map(average_rate_by_block)
```

df

Out[43]:

	ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Pn
0	1		D	Flight	4	2	177	3
1	2		F	Flight	4	5	216	2
2	3		A	Flight	2	2	183	4
3	4		B	Flight	3	3	176	4
4	5		C	Flight	2	2	184	3
...
1452	1453		D	Ship	4	5	143	3
1453	1454		F	Ship	6	3	223	6
1454	1455		A	Ship	3	5	230	2
1455	1456		B	Ship	4	5	241	4
1456	1457		C	Ship	3	5	156	3

1457 rows × 14 columns