

```
In [1]: import pandas as pd
import re
```

```
In [3]: main = pd.read_csv('Datathon/main.csv')
main.head(10)
```

```
Out[3]:
```

	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	NAME_INCOME_TYPE	NAME_EDUCATION_
0	M	Y	Y	0.0	427500.0	Working	Higher educ
1	M	Y	Y	0.0	427500.0	Working	Higher educ
2	M	Y	Y	0.0	112500.0	Working	Secondary / secor s
3	F	N	Y	0.0	270000.0	Commercial associate	Secondary / secor s
4	F	NaN	Y	0.0	270000.0	Commercial associate	Secondary / secor s
5	F	N	Y	0.0	270000.0	Commercial associate	Secondary / secor s
6	F	N	Y	0.0	270000.0	Commercial associate	Secondary / secor s
7	F	N	Y	0.0	283500.0	Pensioner	Higher educ
8	F	N	Y	0.0	283500.0	Pensioner	Higher educ
9	F	N	Y	0.0	283500.0	Pensioner	Higher educ

```
In [4]: main.shape
```

```
Out[4]: (438757, 19)
```

```
In [5]: paid = pd.read_csv('Datathon/paid_record.csv')
paid.head()
```

```
Out[5]:
```

	ID	MONTHS_BALANCE	STATUS
0	5001711	0	X
1	5001711	-1	0

	ID	MONTHS_BALANCE	STATUS
2	5001711	-2	0
3	5001711	-3	0
4	5001712	0	C

```
In [6]: paid.shape
```

```
Out[6]: (1048575, 3)
```

```
In [39]: client1 = pd.Series([0,1,2,0,1,0,1,0,1,0,1,0])
         client1
```

```
Out[39]: 0    0
         1    1
         2    2
         3    0
         4    1
         5    0
         6    1
         7    0
         8    1
         9    0
        10    1
        11    0
         dtype: int64
```

```
In [40]: client1.mean()
```

```
Out[40]: 0.5833333333333334
```

```
In [41]: rolling_1 = client1.rolling(3, min_periods=1).mean()
         rolling_1
```

```
Out[41]: 0    0.000000
         1    0.500000
         2    1.000000
         3    1.000000
         4    1.000000
         5    0.333333
         6    0.666667
         7    0.333333
```

```
8      0.666667
9      0.333333
10     0.666667
11     0.333333
dtype: float64
```

```
In [42]: rolling_1.mean()
```

```
Out[42]: 0.5694444444444444
```

```
In [43]: client2 = pd.Series([0,1,2,3,4,0,0,0,0,0,0,0])
client2
```

```
Out[43]: 0      0
1      1
2      2
3      3
4      4
5      0
6      0
7      0
8      0
9      0
10     0
11     0
dtype: int64
```

```
In [44]: client2.mean()
```

```
Out[44]: 0.8333333333333334
```

```
In [51]: rolling_2 = client2.rolling(3, min_periods=1).mean()
rolling_2
```

```
Out[51]: 0      0.000000
1      0.500000
2      1.000000
3      2.000000
4      3.000000
5      2.333333
6      1.333333
7      0.000000
8      0.000000
9      0.000000
10     0.000000
```

11 0.000000
dtype: float64

```
In [52]: rolling_2.mean()
```

Out[52]: 0.8472222222222223

```
In [115... sample_paid = paid.sample(frac=0.2)
```

```
In [116... pivot = pd.pivot_table(sample_paid, index='ID', columns='MONTHS_BALANCE', values='STATUS',  
aggfunc=''.join).sort_values(by='MONTHS_BALANCE', axis=1)
```

```
In [117... pivot.head()
```

Out[117... MONTHS_BALANCE -60 -59 -58 -57 -56 -55 -54 -53 -52 -51 ... -9 -8 -7 -6 -5 -4 -3 -2 -1 0

ID

5001711	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	X
5001712	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	C	NaN	C	NaN	NaN	NaN	NaN	NaN
5001713	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	X	X	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
5001714	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	X	NaN	NaN	NaN	NaN	NaN	NaN
5001715	NaN	NaN	NaN	NaN	NaN	NaN	NaN	X	NaN	X	...	NaN	NaN	NaN	NaN	X	X	NaN	NaN	NaN	NaN

5 rows x 61 columns

```
In [118... pivot['all'] = pivot.apply(lambda x: ''.join(x.dropna().astype(str)), axis=1)
```

```
In [122... pivot['temp'] = pivot['all'].apply(lambda x: re.split('X|C|0', x))
```

```
In [133... pivot['total'] = pivot['temp'].apply(lambda x: str(x).split(','))
```