In [ ]:
```
import pandas as pd
import numpy as np
# read csv file
df = pd.read_csv("bank-loan.csv", sep=",")
df.head()
```

Out[ ]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	...	FLAG_DOCUMEN
0	100002	1	Cash loans	M	N	Y	0	202500.0	406597.5	24700.5	...	
1	100003	0	Cash loans	F	N	N	0	270000.0	1293502.5	35698.5	...	
2	100004	0	Revolving loans	M	Y	Y	0	67500.0	135000.0	6750.0	...	
3	100006	0	Cash loans	F	N	Y	0	135000.0	312682.5	29686.5	...	
4	100007	0	Cash loans	M	N	Y	0	121500.0	513000.0	21865.5	...	

5 rows × 122 columns

In [ ]:
```
df.drop('SK_ID_CURR', inplace=True, axis=1)
df.head()
```

Out[ ]:

	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	...	FLAG_DO
0	1	Cash loans	M	N	Y	0	202500.0	406597.5	24700.5	351000.0	...	
1	0	Cash loans	F	N	N	0	270000.0	1293502.5	35698.5	1129500.0	...	
2	0	Revolving loans	M	Y	Y	0	67500.0	135000.0	6750.0	135000.0	...	
3	0	Cash loans	F	N	Y	0	135000.0	312682.5	29686.5	297000.0	...	
4	0	Cash loans	M	N	Y	0	121500.0	513000.0	21865.5	513000.0	...	

5 rows × 121 columns

In [ ]:
```
# remove columns which have more than 80% missing values
df = df.dropna(thresh=df.shape[0]*0.8,axis=1)
df.head()
```

Out[ ]:

	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	...	FLAG_DO
0	1	Cash loans	M	N	Y	0	202500.0	406597.5	24700.5	351000.0	...	
1	0	Cash loans	F	N	N	0	270000.0	1293502.5	35698.5	1129500.0	...	
2	0	Revolving loans	M	Y	Y	0	67500.0	135000.0	6750.0	135000.0	...	
3	0	Cash loans	F	N	Y	0	135000.0	312682.5	29686.5	297000.0	...	
4	0	Cash loans	M	N	Y	0	121500.0	513000.0	21865.5	513000.0	...	

5 rows × 71 columns

In [ ]:
```
#process colums with string values without sklearn, keep the original data frame
df_string = df.select_dtypes(include=['object'])
df_string = df_string.apply(lambda x: pd.factorize(x)[0])
df[df_string.columns] = df_string
df.head()
```

Out[ ]:

	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	...	FLAG_DO
0	1	0	0	0	0	0	202500.0	406597.5	24700.5	351000.0	...	
1	0	0	1	0	1	0	270000.0	1293502.5	35698.5	1129500.0	...	
2	0	1	0	1	0	0	67500.0	135000.0	6750.0	135000.0	...	
3	0	0	1	0	0	0	135000.0	312682.5	29686.5	297000.0	...	
4	0	0	0	0	0	0	121500.0	513000.0	21865.5	513000.0	...	

5 rows × 71 columns

In [ ]:
```
# normalize the data without sklearn
df = (df - df.min()) / (df.max() - df.min())
df.head()
```

Out[ ]:

	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	...	FLAG_DO
0	1.0	0.0	0.0	0.0	0.0	0.0	0.001512	0.090287	0.090032	0.077441	...	
1	0.0	0.0	0.5	0.0	1.0	0.0	0.002089	0.311736	0.132924	0.271605	...	
2	0.0	1.0	0.0	1.0	0.0	0.0	0.000358	0.022472	0.022025	0.023569	...	
3	0.0	0.0	0.5	0.0	0.0	0.0	0.000935	0.066837	0.109477	0.063973	...	
4	0.0	0.0	0.0	0.0	0.0	0.0	0.000819	0.116854	0.078975	0.117845	...	

5 rows × 71 columns

In [ ]:
```
df.fillna(df.mean(), inplace=True)
df.head()
```

Out[ ]:

	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	...	FLAG_DO
0	1.0	0.0	0.0	0.0	0.0	0.0	0.001512	0.090287	0.090032	0.077441	...	
1	0.0	0.0	0.5	0.0	1.0	0.0	0.002089	0.311736	0.132924	0.271605	...	
2	0.0	1.0	0.0	1.0	0.0	0.0	0.000358	0.022472	0.022025	0.023569	...	
3	0.0	0.0	0.5	0.0	0.0	0.0	0.000935	0.066837	0.109477	0.063973	...	
4	0.0	0.0	0.0	0.0	0.0	0.0	0.000819	0.116854	0.078975	0.117845	...	

5 rows × 71 columns

In [ ]:
```
df.describe()
```

Out[ ]:

	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	...
count	307511.000000	307511.000000	307511.000000	307511.000000	307511.000000	307511.000000	307511.000000	307511.000000	307511.000000	307511.000000	...
mean	0.080729	0.095213	0.329185	0.340108	0.306327	0.021950	0.001224	0.138334	0.099423	0.124179	...
std	0.272419	0.293509	0.237142	0.473746	0.460968	0.038006	0.002027	0.100497	0.056525	0.092101	...
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...
25%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000742	0.056180	0.058143	0.049383	...
50%	0.000000	0.000000	0.500000	0.000000	0.000000	0.000000	0.001039	0.116987	0.090821	0.102132	...
75%	0.000000	0.000000	0.500000	1.000000	1.000000	0.052632	0.001512	0.190674	0.128624	0.159371	...
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	...

8 rows × 71 columns

In [ ]:
```
df.to_csv("bank-loan-processed.csv", index=False)
new_df = pd.read_csv("bank-loan-processed.csv", sep=",")
new_df.head()
```

Out[ ]:

	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	...	FLAG_DO
0	1.0	0.0	0.0	0.0	0.0	0.0	0.001512	0.090287	0.090032	0.077441	...	
1	0.0	0.0	0.5	0.0	1.0	0.0	0.002089	0.311736	0.132924	0.271605	...	
2	0.0	1.0	0.0	1.0	0.0	0.0	0.000358	0.022472	0.022025	0.023569	...	
3	0.0	0.0	0.5	0.0	0.0	0.0	0.000935	0.066837	0.109477	0.063973	...	
4	0.0	0.0	0.0	0.0	0.0	0.0	0.000819	0.116854	0.078975	0.117845	...	

5 rows × 71 columns

In [ ]:
```
new_df.corr()
```

Out[ ]:

	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY
TARGET	1.000000	-0.030896	-0.054718	-0.021851	0.006148	0.019187	-0.003963	-0.030369	-0.012817
NAME_CONTRACT_TYPE	-0.030896	1.000000	0.009867	0.004022	-0.067177	0.029898	-0.003531	-0.221648	-0.241543
CODE_GENDER	-0.054718	0.009867	1.000000	-0.345815	-0.044396	-0.047367	-0.074724	-0.021614	-0.077002
FLAG_OWN_CAR	-0.021851	0.004022	-0.345815	1.000000	0.002817	0.102023	0.083383	0.116225	0.141586
FLAG_OWN_REALTY	0.006148	-0.067177	-0.044396	0.002817	1.000000	0.002366	-0.002934	0.039270	0.005225
...	...	...	...	...	...	...	...	...	...
AMT_REQ_CREDIT_BUREAU_DAY	0.002464	-0.004732	-0.001061	0.000535	0.008645	-0.000342	0.002868	0.003964	0.002018
AMT_REQ_CREDIT_BUREAU_WEEK	0.000718	-0.014144	0.001439	0.000227	-0.006972	-0.002277	0.002326	-0.001192	0.012815
AMT_REQ_CREDIT_BUREAU_MON	-0.011356	-0.013286	-0.008260	0.019149	0.004180	-0.010101	0.024063	0.050934	0.036148
AMT_REQ_CREDIT_BUREAU_QRT	-0.001842	-0.020307	0.006920	-0.009291	-0.014414	-0.007324	0.004734	0.014896	0.009348
AMT_REQ_CREDIT_BUREAU_YEAR	0.019160	-0.046539	0.016969	-0.033968	-0.062927	-0.038834	0.011388	-0.045318	-0.010452

71 rows × 71 columns

In [ ]:
```
# heatmap
import matplotlib.pyplot as plt
plt.matshow(new_df.corr(), cmap="winter")
plt.colorbar()
plt.xticks(np.arange(len(new_df.columns)), new_df.columns, rotation=90, fontsize=5)
plt.yticks(np.arange(len(new_df.columns)), new_df.columns, fontsize=5)
plt.rcParams["figure.figsize"] = [20,20]
plt.rcParams["figure.dpi"] = 300
plt.draw()
plt.savefig("heatmap.png", dpi=300, bbox_inches='tight')
plt.show()
```


In [ ]:
```
new_df.plot.box(rot=90, fontsize=4)
```

Out[ ]:
```
<Axes: >
```