```
# IMPORTANT: RUN THIS CELL IN ORDER TO IMPORT YOUR KAGGLE DATA SOURCES,
# THEN FEEL FREE TO DELETE THIS CELL.
# NOTE: THIS NOTEBOOK ENVIRONMENT DIFFERS FROM KAGGLE'S PYTHON
# ENVIRONMENT SO THERE MAY BE MISSING LIBRARIES USED BY YOUR
# NOTEBOOK.
import kagglehub
mishra5001_credit_card_path = kagglehub.dataset_download('mishra5001/credit-card')
manoreji_credit_card_fraud_detection_path = kagglehub.dataset_download('manoreji/credit-card-fraud-detection')

print('Data source import complete.')
```

```
# This Python 3 environment comes with many helpful analytics libraries installed
# It is defined by the kaggle/python Docker image: https://github.com/kaggle/docker-python
# For example, here's several helpful packages to load

import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

# Input data files are available in the read-only "../input/" directory
# For example, running this (by clicking run or pressing Shift+Enter) will list all files under the input directory

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# You can write up to 20GB to the current directory (/kaggle/working/) that gets preserved as output when you create a version using "Save &
# You can also write temporary files to /kaggle/temp/, but they won't be saved outside of the current session
```

```
/kaggle/input/credit-card/application_data.csv
/kaggle/input/credit-card/previous_application.csv
/kaggle/input/credit-card/columns_description.csv
/kaggle/input/credit-card-fraud-detection/credit-card
```

```
pwd
```

```
'/kaggle/working'
```

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
from sklearn.preprocessing import StandardScaler
```

```
df = pd.read_csv('/kaggle/input/credit-card/application_data.csv')
```

```
df.head()
```

```
/usr/local/lib/python3.11/dist-packages/pandas/io/formats/format.py:1458: RuntimeWarning: invalid value encountered in greater
  has_large_values = (abs_vals > 1e6).any()
/usr/local/lib/python3.11/dist-packages/pandas/io/formats/format.py:1459: RuntimeWarning: invalid value encountered in less
  has_small_values = ((abs_vals < 10 ** (-self.digits)) & (abs_vals > 0)).any()
/usr/local/lib/python3.11/dist-packages/pandas/io/formats/format.py:1459: RuntimeWarning: invalid value encountered in greater
  has_small_values = ((abs_vals < 10 ** (-self.digits)) & (abs_vals > 0)).any()
/usr/local/lib/python3.11/dist-packages/pandas/io/formats/format.py:1458: RuntimeWarning: invalid value encountered in greater
  has_large_values = (abs_vals > 1e6).any()
/usr/local/lib/python3.11/dist-packages/pandas/io/formats/format.py:1459: RuntimeWarning: invalid value encountered in less
  has_small_values = ((abs_vals < 10 ** (-self.digits)) & (abs_vals > 0)).any()
/usr/local/lib/python3.11/dist-packages/pandas/io/formats/format.py:1459: RuntimeWarning: invalid value encountered in greater
  has_small_values = ((abs_vals < 10 ** (-self.digits)) & (abs_vals > 0)).any()
```

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_AI
0	100002	1	Cash loans	M	N	Y	0	202500.0	406597.5	2
1	100003	0	Cash loans	F	N	N	0	270000.0	1293502.5	3
2	100004	0	Revolving loans	M	Y	Y	0	67500.0	135000.0	
3	100006	0	Cash loans	F	N	Y	0	135000.0	312682.5	2
4	100007	0	Cash loans	M	N	Y	0	121500.0	513000.0	2

5 rows × 122 columns

```
df.shape
```

```
(307511, 122)
```

```
df.dtypes
```

```
SK_ID_CURR                int64
TARGET                    int64
NAME_CONTRACT_TYPE       object
CODE_GENDER              object
FLAG_OWN_CAR             object
                          ...
AMT_REQ_CREDIT_BUREAU_DAY    float64
AMT_REQ_CREDIT_BUREAU_WEEK   float64
AMT_REQ_CREDIT_BUREAU_MON    float64
AMT_REQ_CREDIT_BUREAU_QRT    float64
AMT_REQ_CREDIT_BUREAU_YEAR   float64
Length: 122, dtype: object
```

```
pd.DataFrame(df.isnull().sum())
```

	0
SK_ID_CURR	0
TARGET	0
NAME_CONTRACT_TYPE	0
CODE_GENDER	0
FLAG_OWN_CAR	0
...	...
AMT_REQ_CREDIT_BUREAU_DAY	41519
AMT_REQ_CREDIT_BUREAU_WEEK	41519
AMT_REQ_CREDIT_BUREAU_MON	41519
AMT_REQ_CREDIT_BUREAU_QRT	41519
AMT_REQ_CREDIT_BUREAU_YEAR	41519

122 rows × 1 columns

```
df['TARGET'].value_counts()
```

```
TARGET
0    282686
1     24825
Name: count, dtype: int64
```

```python
df['TARGET'].value_counts(normalize=True)*100
```

```
TARGET
0    91.927118
1     8.072882
Name: proportion, dtype: float64
```

```python
df['TARGET'].mean()*100
```

```
8.072881945686495
```

```python
fraud_count = df['TARGET'].value_counts()[1]
non_fraud_count = df['TARGET'].value_counts()[0]
print(f"Fraud Count: {fraud_count}, Non-Fraud Count: {non_fraud_count}")
```

```
Fraud Count: 24825, Non-Fraud Count: 282686
```

```python
df.TARGET
```

```
0         1
1         0
2         0
3         0
4         0
         ..
307506    0
307507    0
307508    0
307509    1
307510    0
Name: TARGET, Length: 307511, dtype: int64
```

```python
df['TARGET'].value_counts()[1] / len(df) * 100
```

```
8.072881945686495
```

```python
df.corr
```

```
/usr/local/lib/python3.11/dist-packages/pandas/io/formats/format.py:1458: RuntimeWarning: invalid value encountered in greater
  has_large_values = (abs_vals > 1e6).any()
/usr/local/lib/python3.11/dist-packages/pandas/io/formats/format.py:1459: RuntimeWarning: invalid value encountered in less
  has_small_values = ((abs_vals < 10 ** (-self.digits)) & (abs_vals > 0)).any()
/usr/local/lib/python3.11/dist-packages/pandas/io/formats/format.py:1459: RuntimeWarning: invalid value encountered in greater
  has_small_values = ((abs_vals < 10 ** (-self.digits)) & (abs_vals > 0)).any()
<bound method DataFrame.corr of         SK_ID_CURR  TARGET NAME_CONTRACT_TYPE CODE_GENDER FLAG_OWN_CAR  \
0           100002       1         Cash loans           M            N
1           100003       0         Cash loans           F            N
2           100004       0    Revolving loans           M            Y
3           100006       0         Cash loans           F            N
4           100007       0         Cash loans           M            N
...            ...     ...                ...         ...          ...
307506      456251       0         Cash loans           M            N
307507      456252       0         Cash loans           F            N
307508      456253       0         Cash loans           F            N
307509      456254       1         Cash loans           F            N
307510      456255       0         Cash loans           F            N

        FLAG_OWN_REALTY  CNT_CHILDREN  AMT_INCOME_TOTAL  AMT_CREDIT  \
0                     Y             0          202500.0    406597.5
1                     N             0          270000.0   1293502.5
2                     Y             0           67500.0    135000.0
3                     Y             0          135000.0    312682.5
4                     Y             0          121500.0    513000.0
...                 ...           ...               ...         ...
307506                N             0          157500.0    254700.0
307507                Y             0           72000.0    269550.0
307508                Y             0          153000.0    677664.0
307509                Y             0          171000.0    370107.0
307510                N             0          157500.0    675000.0

        AMT_ANNUITY  ...  FLAG_DOCUMENT_18 FLAG_DOCUMENT_19 FLAG_DOCUMENT_20  \
```

```
0         24700.5  ...              0           0           0
1         35698.5  ...              0           0           0
2          6750.0  ...              0           0           0
3         29686.5  ...              0           0           0
4         21865.5  ...              0           0           0
...           ...  ...            ...         ...         ...
307506    27558.0  ...              0           0           0
307507    12001.5  ...              0           0           0
307508    29979.0  ...              0           0           0
307509    20205.0  ...              0           0           0
307510    49117.5  ...              0           0           0

        FLAG_DOCUMENT_21 AMT_REQ_CREDIT_BUREAU_HOUR AMT_REQ_CREDIT_BUREAU_DAY  \
0                      0                        0.0                        0.0
1                      0                        0.0                        0.0
2                      0                        0.0                        0.0
3                      0                        NaN                        NaN
4                      0                        0.0                        0.0
...                  ...                        ...                        ...
307506                 0                        NaN                        NaN
307507                 0                        NaN                        NaN
307508                 0                        1.0                        0.0
307509                 0                        0.0                        0.0
307510                 0                        0.0                        0.0
```

```python
df_numeric = df.select_dtypes(include=[float, int])
corr_matrix = df_numeric.corr()
plt.figure(figsize=(12,8))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', fmt='.2f')
plt.title('Correlation Matrix Between Features')
plt.show()
```

```
/usr/local/lib/python3.11/dist-packages/matplotlib/colors.py:721: RuntimeWarning: invalid value encountered in less
    xa[xa < 0] = -1
```


Correlation Matrix Between Features
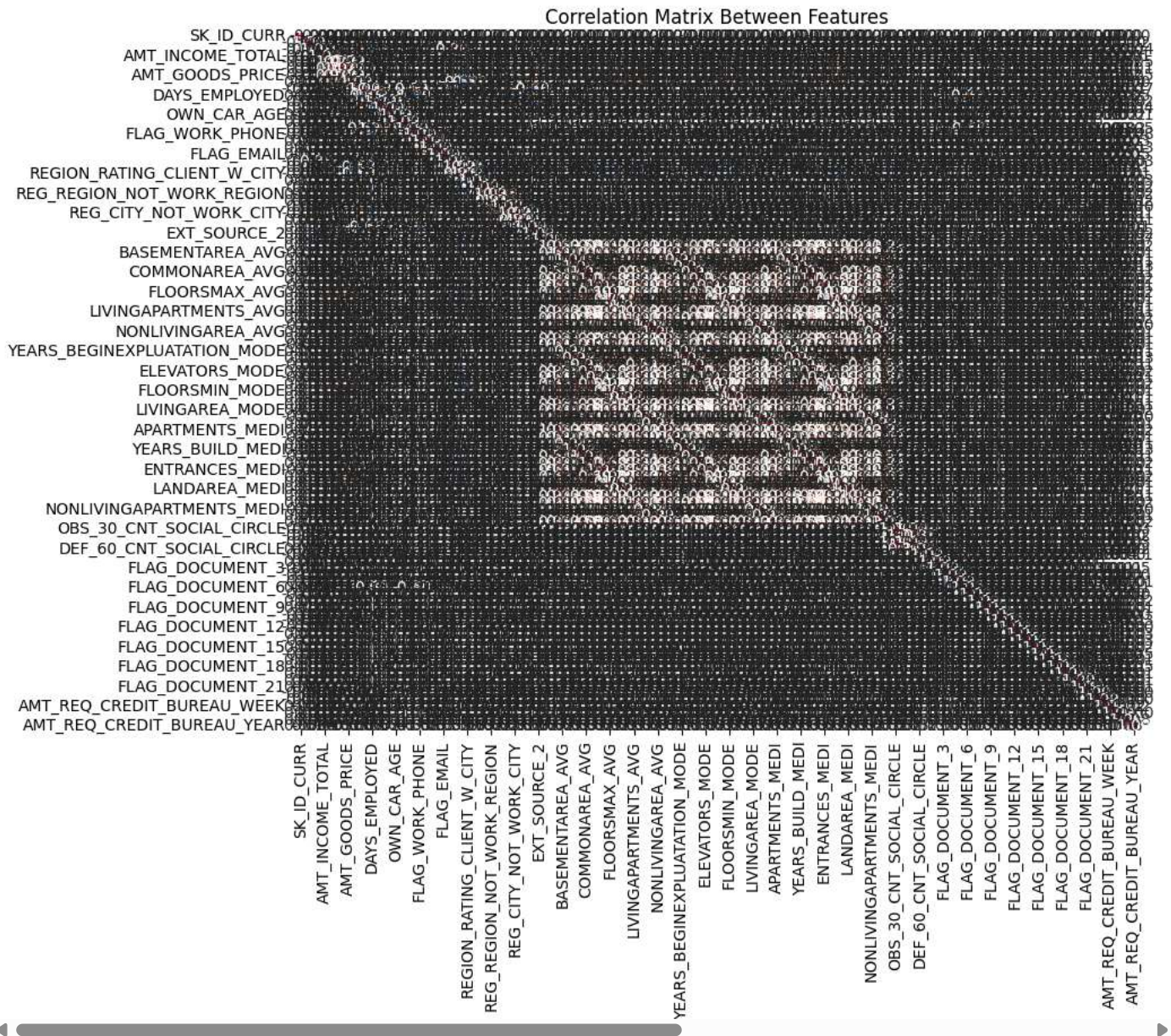
```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Assuming you have already loaded the dataset into the variable df
df_numeric = df.select_dtypes(include=[float, int])  # Select only numeric columns
corr_matrix = df_numeric.corr()  # Calculate correlation matrix

# Bar chart of correlations
plt.figure(figsize=(10, 6))
correlations = corr_matrix.unstack().sort_values(ascending=False).drop_duplicates()  # Use corr_matrix
correlations = correlations[correlations != 1]  # Remove self-correlations
correlations.head(10).plot(kind='bar')
plt.title('Top 10 Feature Correlations (Bar Chart)')
plt.ylabel('Correlation coefficient')
plt.show()

# Pie chart of correlations
top_correlations = correlations.head(5)  # Select top 5 correlations for pie chart
plt.figure(figsize=(8, 8))
plt.pie(top_correlations, labels=top_correlations.index, autopct='%1.1f%%', colors=sns.color_palette('pastel'))
plt.title('Top 5 Feature Correlations (Pie Chart)')
plt.show()
```

Top 10 Feature Correlations (Bar Chart)