

```
In [3]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
```

```
In [4]: df = pd.read_csv('Banking.csv')
df.head(5)
```

Out[4]:

	Client ID	Name	Age	Location ID	Joined Bank	Banking Contact	Nationality	Occupation	Stru
0	IND81288	Raymond Mills	24	34324	06-05-2019	Anthony Torres	American	Safety Technician IV	
1	IND65833	Julia Spencer	23	42205	10-12-2001	Jonathan Hawkins	African	Software Consultant	
2	IND47499	Stephen Murray	27	7314	25-01-2010	Anthony Berry	European	Help Desk Operator	
3	IND72498	Virginia Garza	40	34594	28-03-2019	Steve Diaz	American	Geologist II	
4	IND60181	Melissa Sanders	46	41269	20-07-2012	Shawn Long	American	Assistant Professor	

5 rows × 25 columns



```
In [5]: df.shape
```

Out[5]: (3000, 25)

```
In [6]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000 entries, 0 to 2999
Data columns (total 25 columns):
 #   Column           Non-Null Count  Dtype  
 --- 
 0   Client ID        3000 non-null   object  
 1   Name              3000 non-null   object  
 2   Age               3000 non-null   int64   
 3   Location ID      3000 non-null   int64   
 4   Joined Bank       3000 non-null   object  
 5   Banking Contact  3000 non-null   object  
 6   Nationality       3000 non-null   object  
 7   Occupation        3000 non-null   object  
 8   Fee Structure    3000 non-null   object  
 9   Loyalty Classification 3000 non-null   object  
 10  Estimated Income 3000 non-null   float64 
 11  Superannuation Savings 3000 non-null   float64 
 12  Amount of Credit Cards 3000 non-null   int64  
 13  Credit Card Balance 3000 non-null   float64 
 14  Bank Loans        3000 non-null   float64 
 15  Bank Deposits    3000 non-null   float64 
 16  Checking Accounts 3000 non-null   float64 
 17  Saving Accounts   3000 non-null   float64 
 18  Foreign Currency Account 3000 non-null   float64 
 19  Business Lending  3000 non-null   float64 
 20  Properties Owned 3000 non-null   int64  
 21  Risk Weighting    3000 non-null   int64  
 22  BRId              3000 non-null   int64  
 23  GenderId          3000 non-null   int64  
 24  IAId              3000 non-null   int64  
dtypes: float64(9), int64(8), object(8)
memory usage: 586.1+ KB
```

In [7]: `# Generate descriptive statistics for the dataframe
df.describe()`

Out[7]:

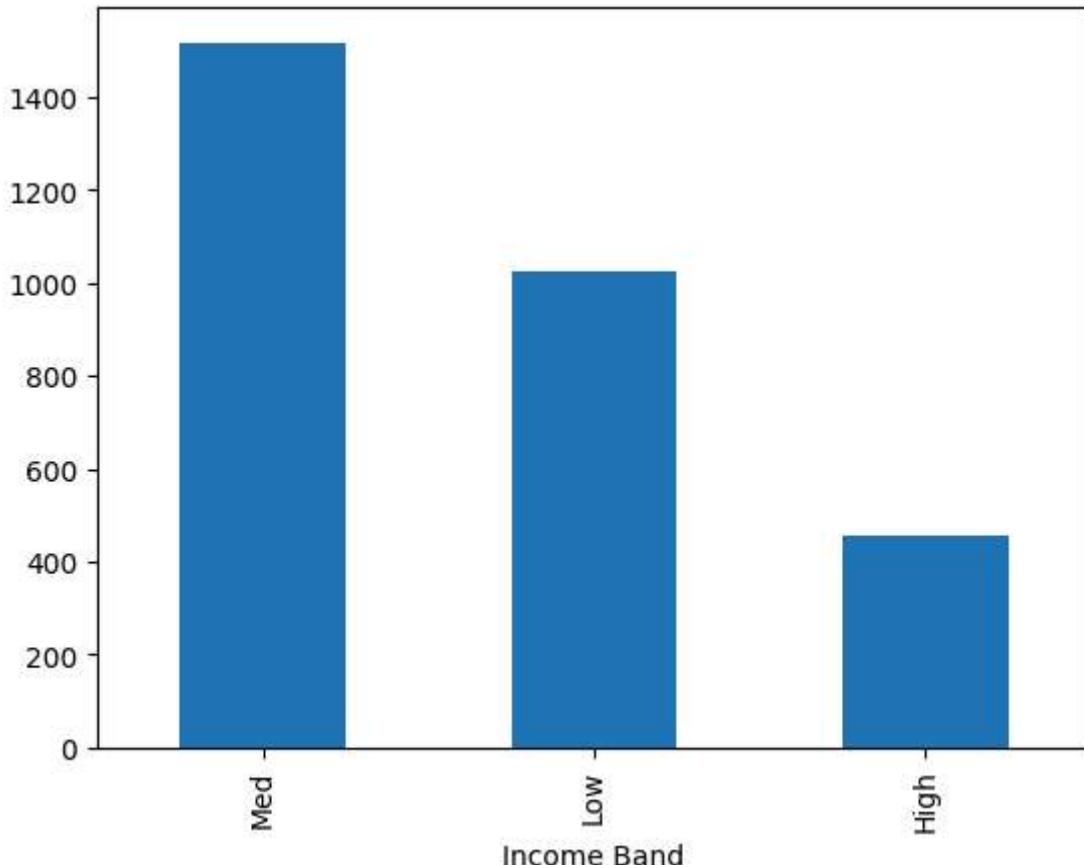
	Age	Location ID	Estimated Income	Superannuation Savings	Amount of Credit Cards	Credit Ba
count	3000.000000	3000.000000	3000.000000	3000.000000	3000.000000	3000.0
mean	51.039667	21563.323000	171305.034263	25531.599673	1.463667	3176.2
std	19.854760	12462.273017	111935.808209	16259.950770	0.676387	2497.0
min	17.000000	12.000000	15919.480000	1482.030000	1.000000	1.1
25%	34.000000	10803.500000	82906.595000	12513.775000	1.000000	1236.6
50%	51.000000	21129.500000	142313.480000	22357.355000	1.000000	2560.8
75%	69.000000	32054.500000	242290.305000	35464.740000	2.000000	4522.6
max	85.000000	43369.000000	522330.260000	75963.900000	3.000000	13991.9

In [8]: `bins = [0, 100000, 300000, float('inf')]
labels = ['Low', 'Med', 'High']`

```
df['Income Band'] = pd.cut(df['Estimated Income'], bins=bins, labels=labels, rig
```

```
In [9]: df['Income Band'].value_counts().plot(kind='bar')
```

```
Out[9]: <Axes: xlabel='Income Band'>
```



```
In [10]: # Examine the distribution of unique categories in categorical columns
categorical_cols = df[['BRIId", "GenderId", "IAId", "Amount of Credit Cards", "Na
for col in categorical_cols:
    print(f"Value Counts for '{col}':")
    display(df[col].value_counts())
```

```
Value Counts for 'BRIId':
```

```
BRIId
3    1352
1     660
2     495
4     493
```

```
Name: count, dtype: int64
```

```
Value Counts for 'GenderId':
```

```
GenderId
2    1512
1    1488
```

```
Name: count, dtype: int64
```

```
Value Counts for 'IAId':
```

IAId

1	177
2	177
3	177
4	177
8	177
9	176
13	176
12	176
10	176
11	176
14	176
15	176
6	89
5	89
7	89
16	88
17	88
18	88
19	88
20	88
21	88
22	88

Name: count, dtype: int64

Value Counts for 'Amount of Credit Cards':

Amount of Credit Cards

1	1922
2	765
3	313

Name: count, dtype: int64

Value Counts for 'Nationality':

Nationality

European	1309
Asian	754
American	507
Australian	254
African	176

Name: count, dtype: int64

Value Counts for 'Occupation':

Occupation

Associate Professor	28
Structural Analysis Engineer	28
Recruiter	25
Account Coordinator	24
Human Resources Manager	24
	..
Office Assistant IV	8
Automation Specialist I	7
Computer Systems Analyst I	6
Developer III	5
Senior Sales Associate	4

Name: count, Length: 195, dtype: int64

Value Counts for 'Fee Structure':

Fee Structure

High	1476
Mid	962
Low	562

Name: count, dtype: int64

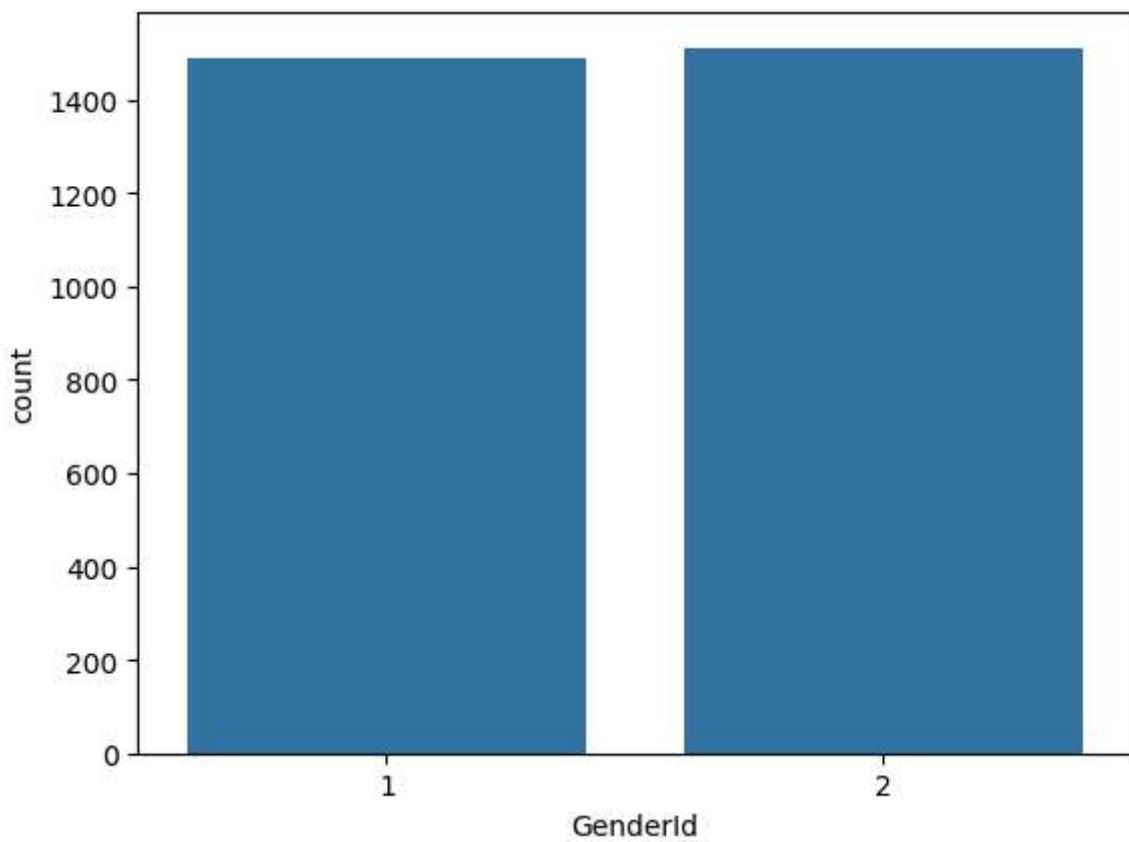
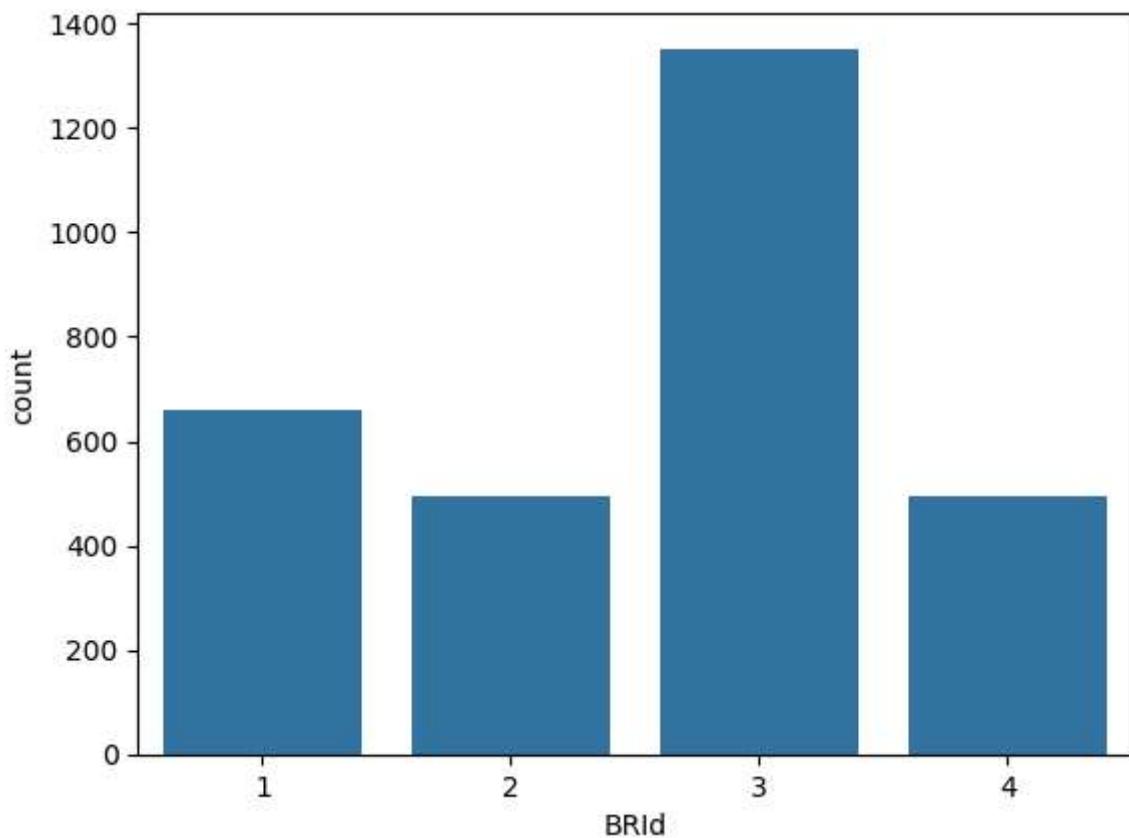
Value Counts for 'Loyalty Classification':

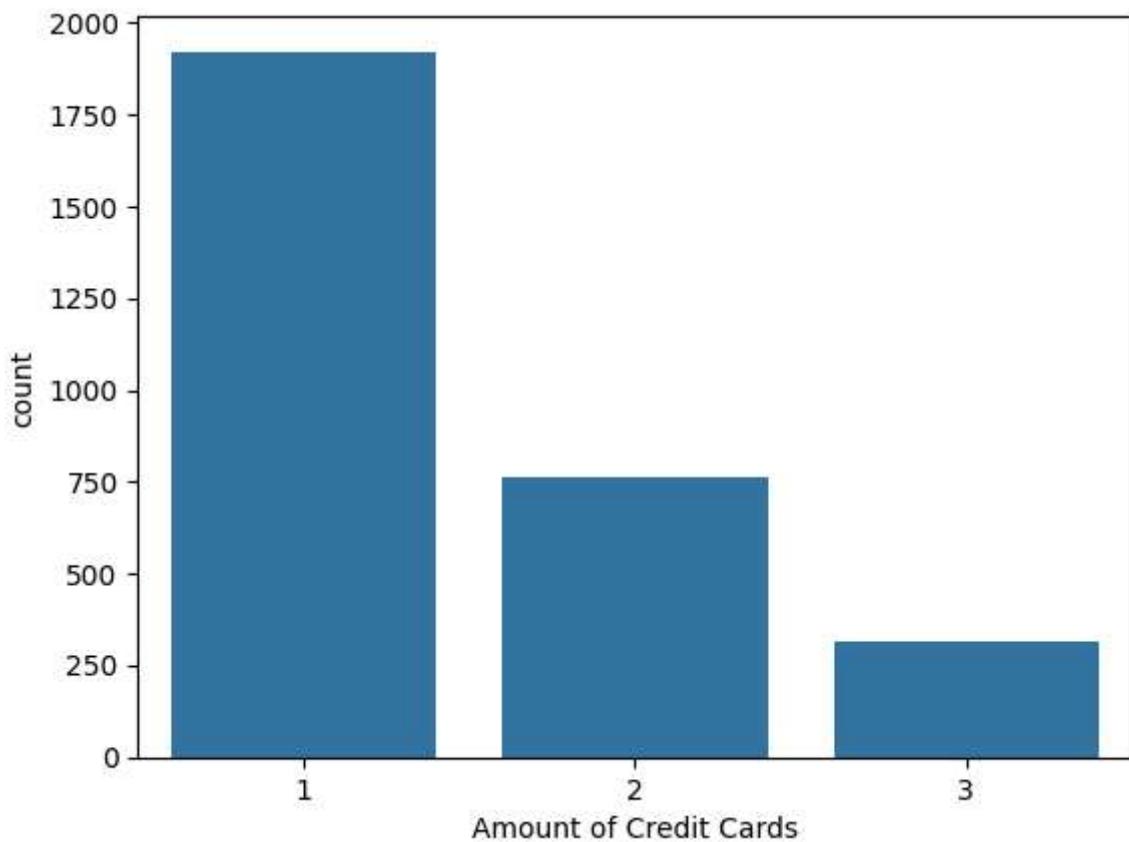
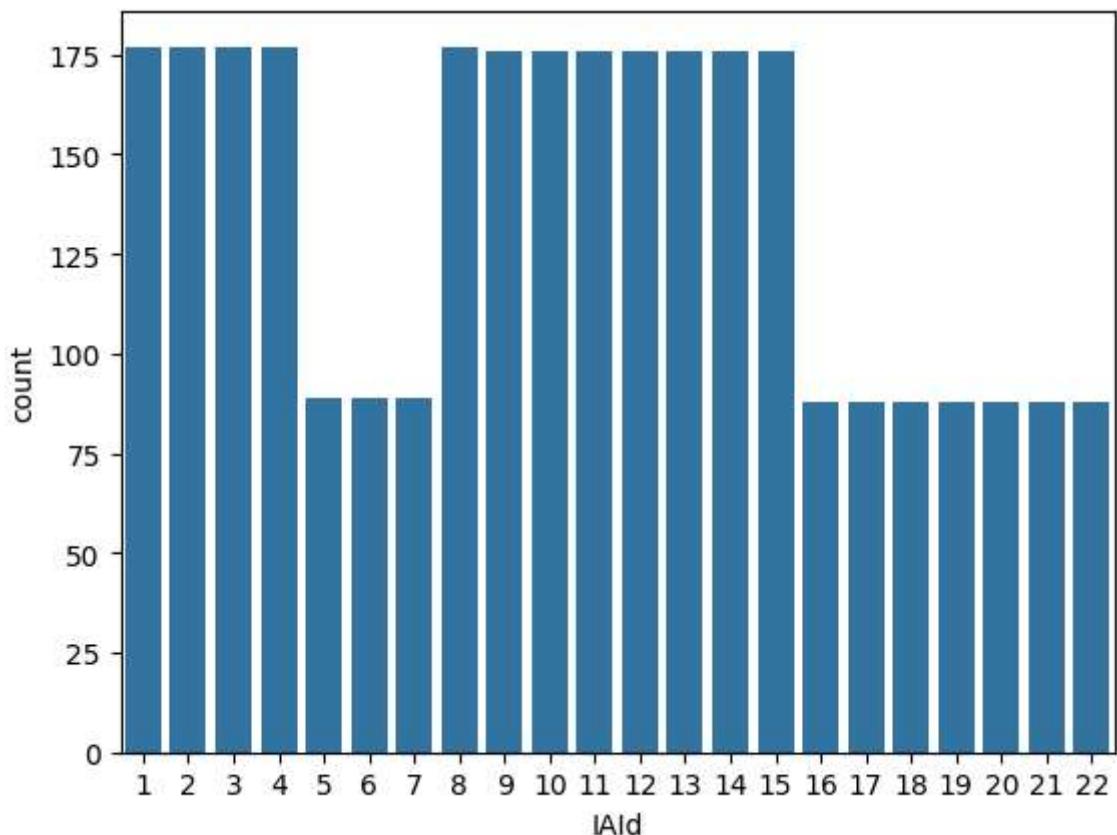
```
Loyalty Classification
Jade        1331
Silver      767
Gold         585
Platinum    317
Name: count, dtype: int64
Value Counts for 'Properties Owned':
Properties Owned
2      777
1      776
3      742
0      705
Name: count, dtype: int64
Value Counts for 'Risk Weighting':
Risk Weighting
2      1222
1      836
3      460
4      322
5      160
Name: count, dtype: int64
Value Counts for 'Income Band':
Income Band
Med       1517
Low       1027
High      456
Name: count, dtype: int64
```

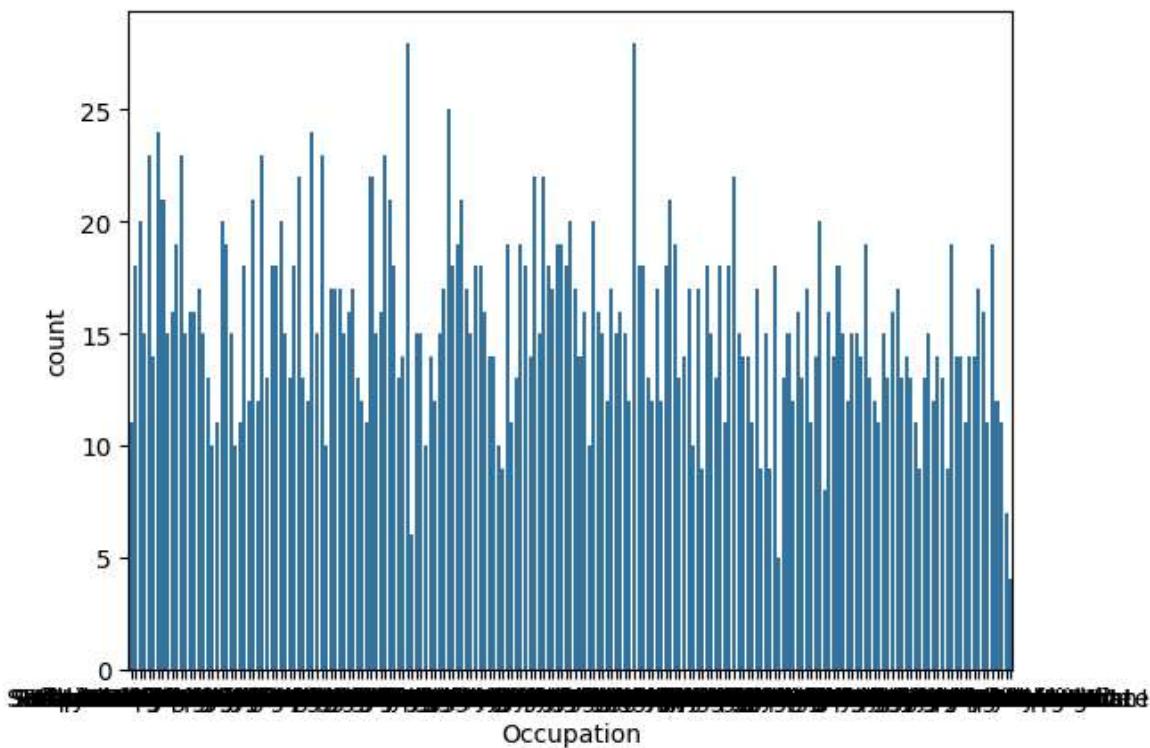
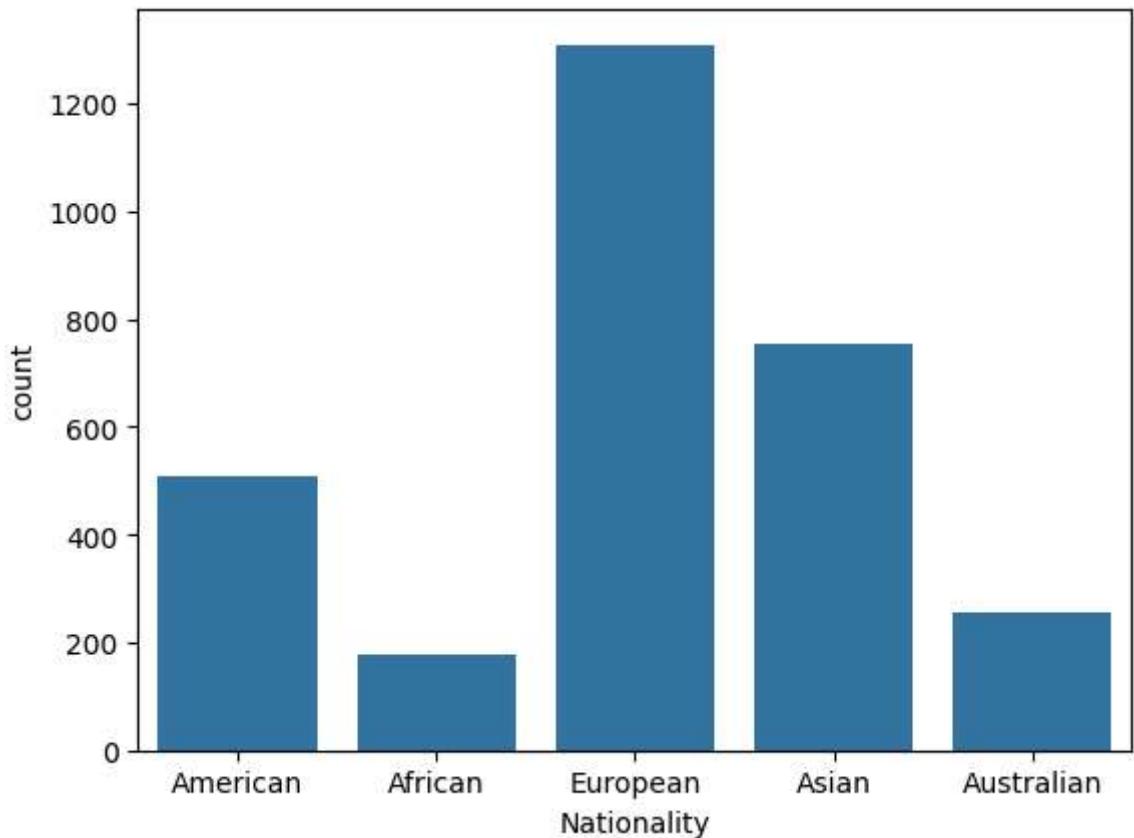
In []:

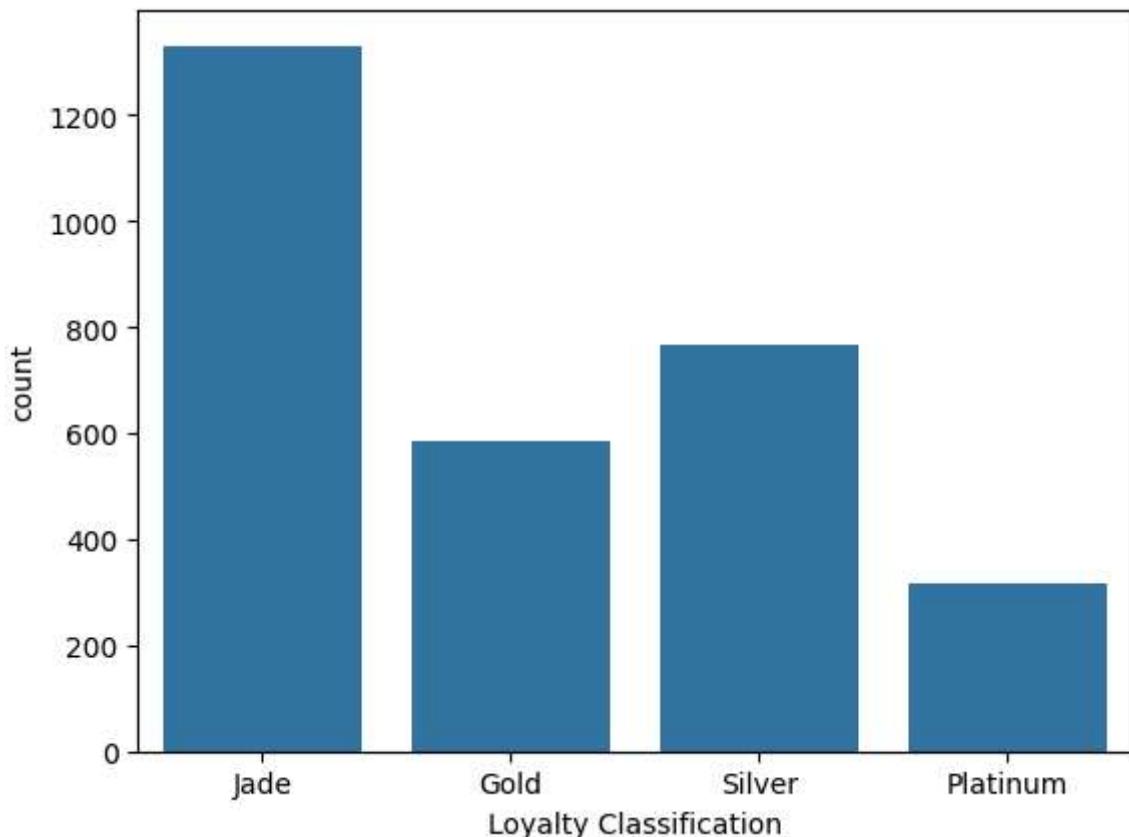
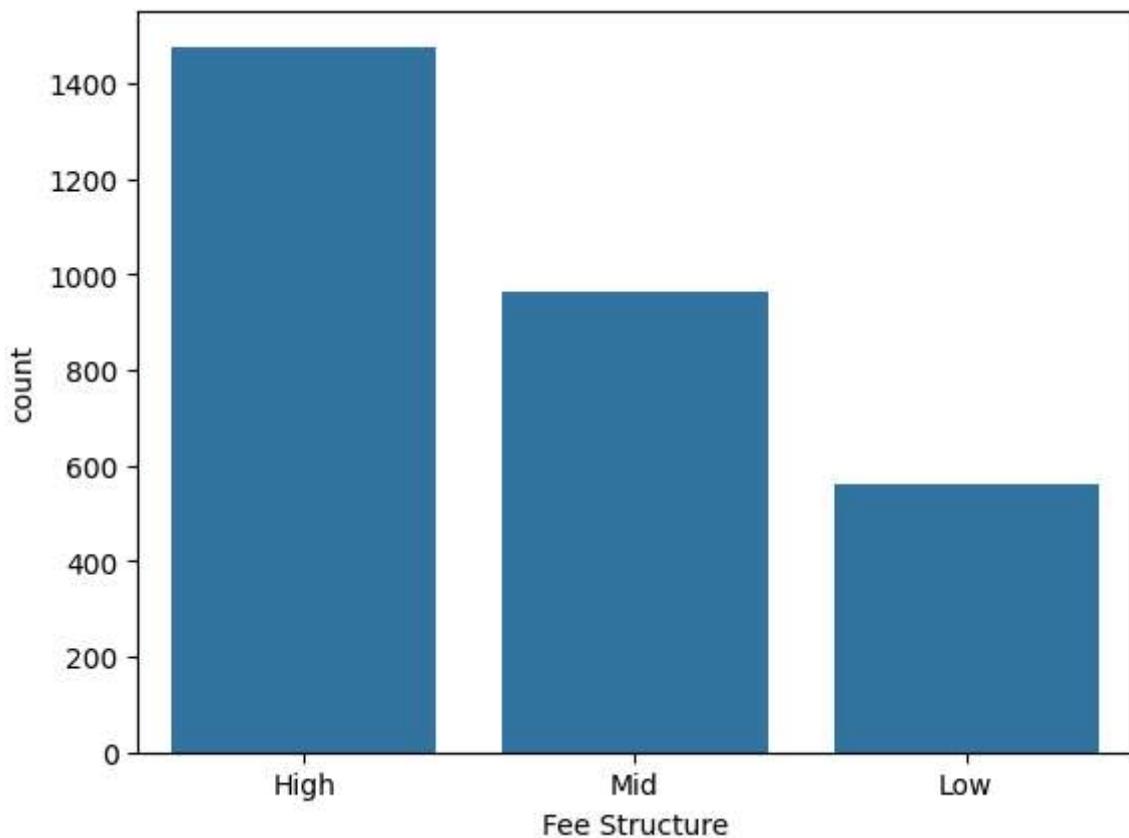
Universal Analysis

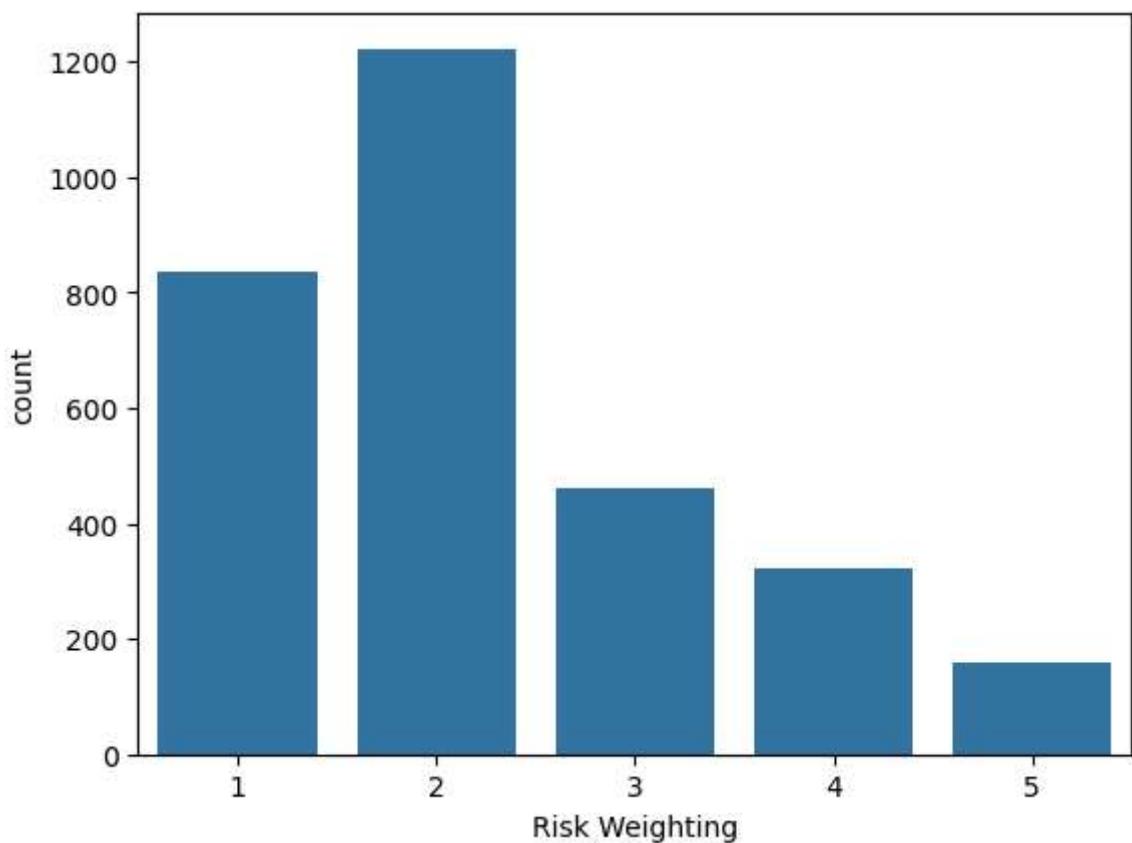
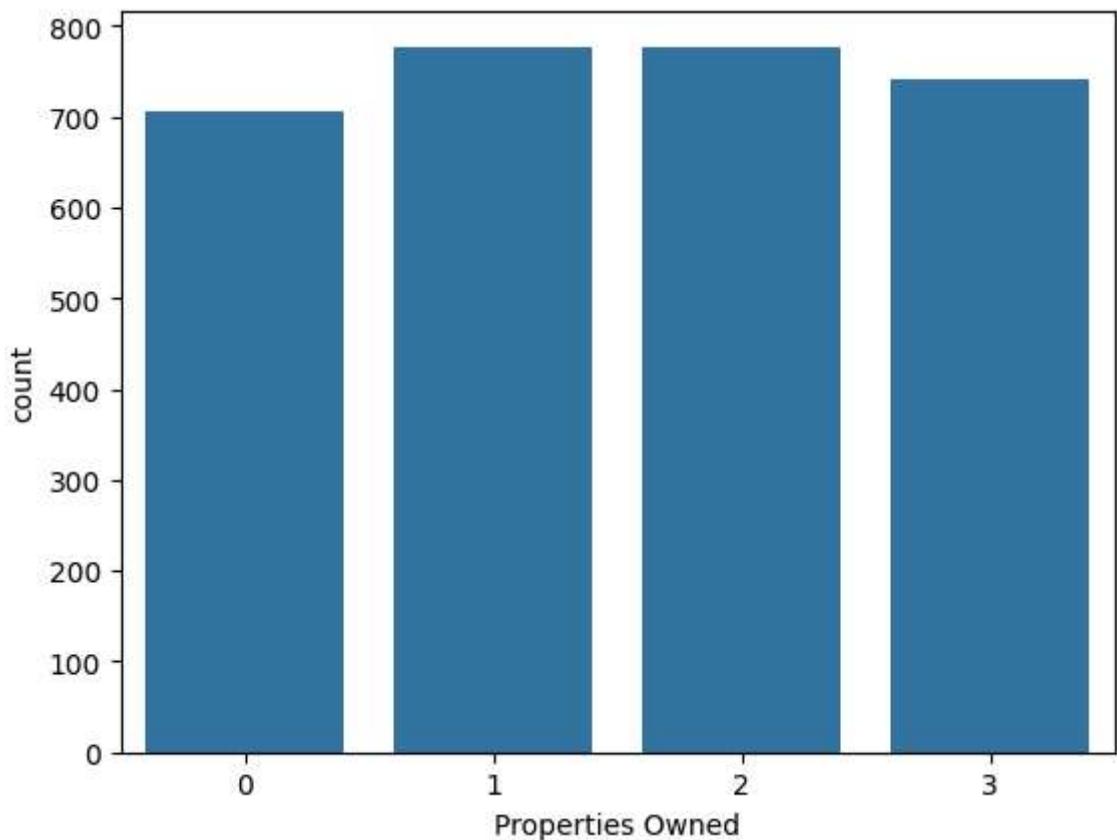
```
In [11]: for i, predictor in enumerate(df[["BRId", "GenderId", "IAId", "Amount of Credit"])
    plt.figure(i)
    sns.countplot(data=df, x=predictor)
```

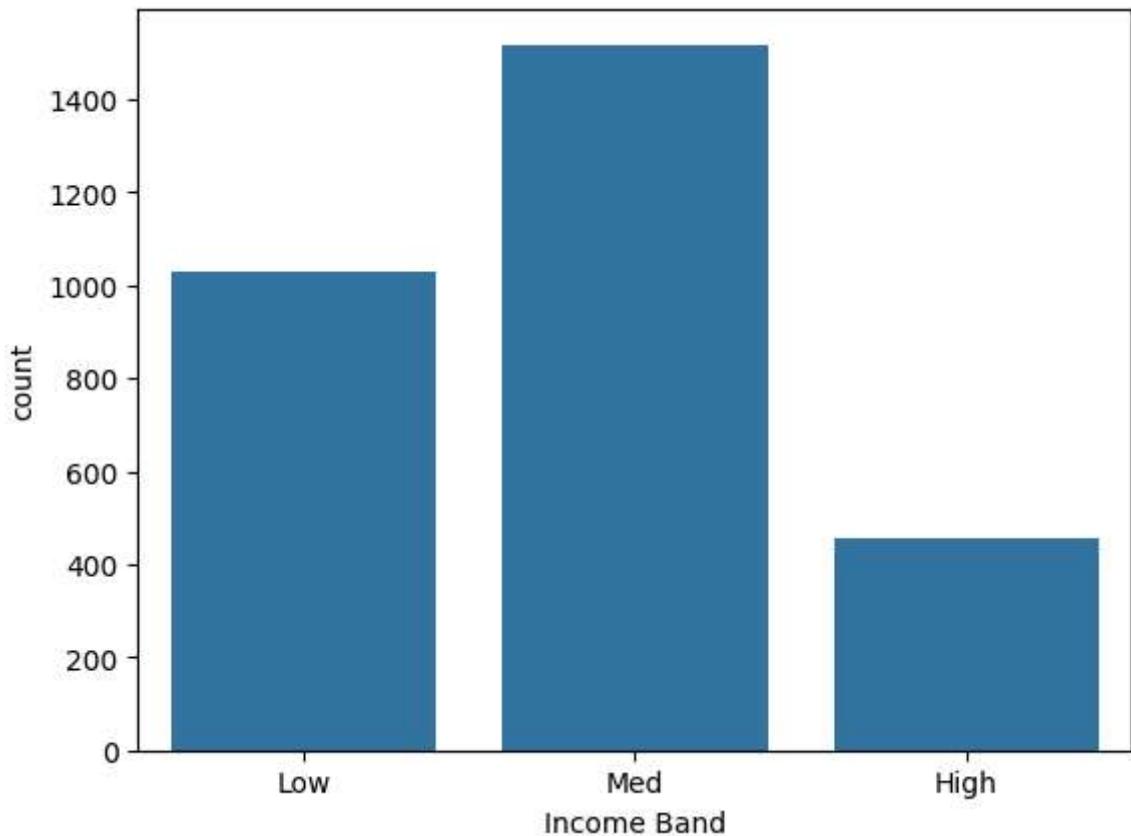






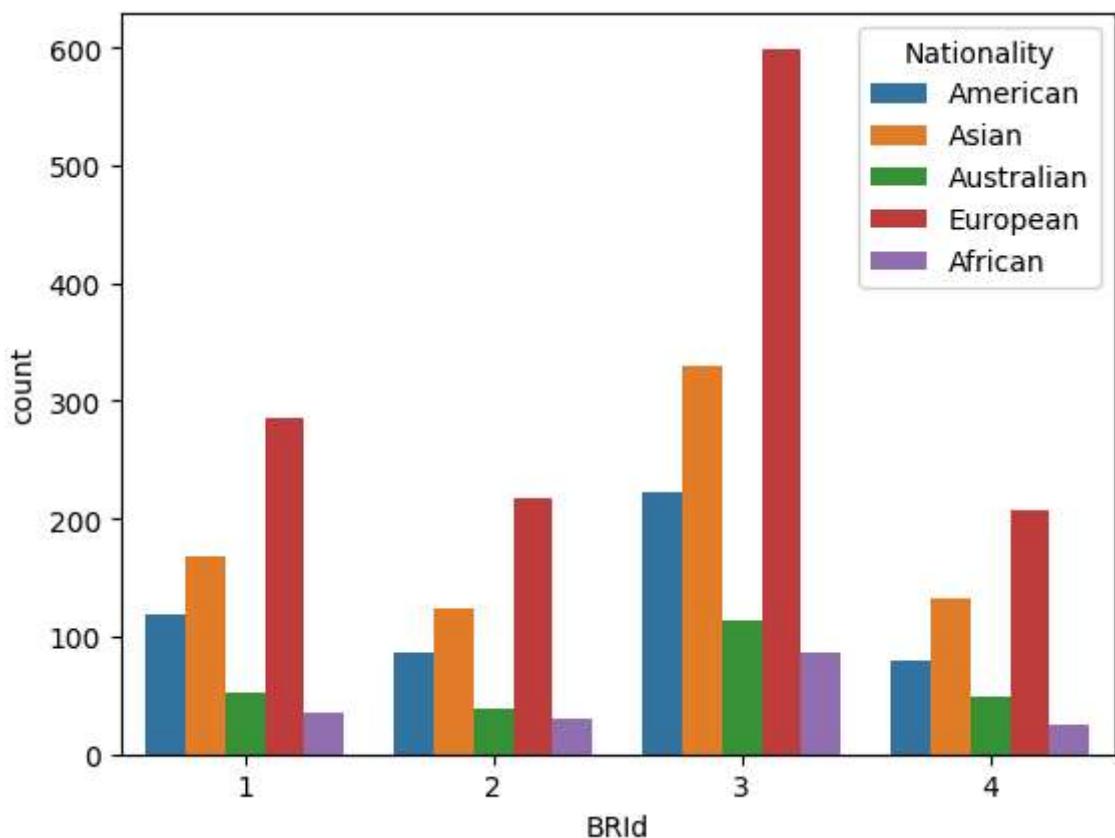


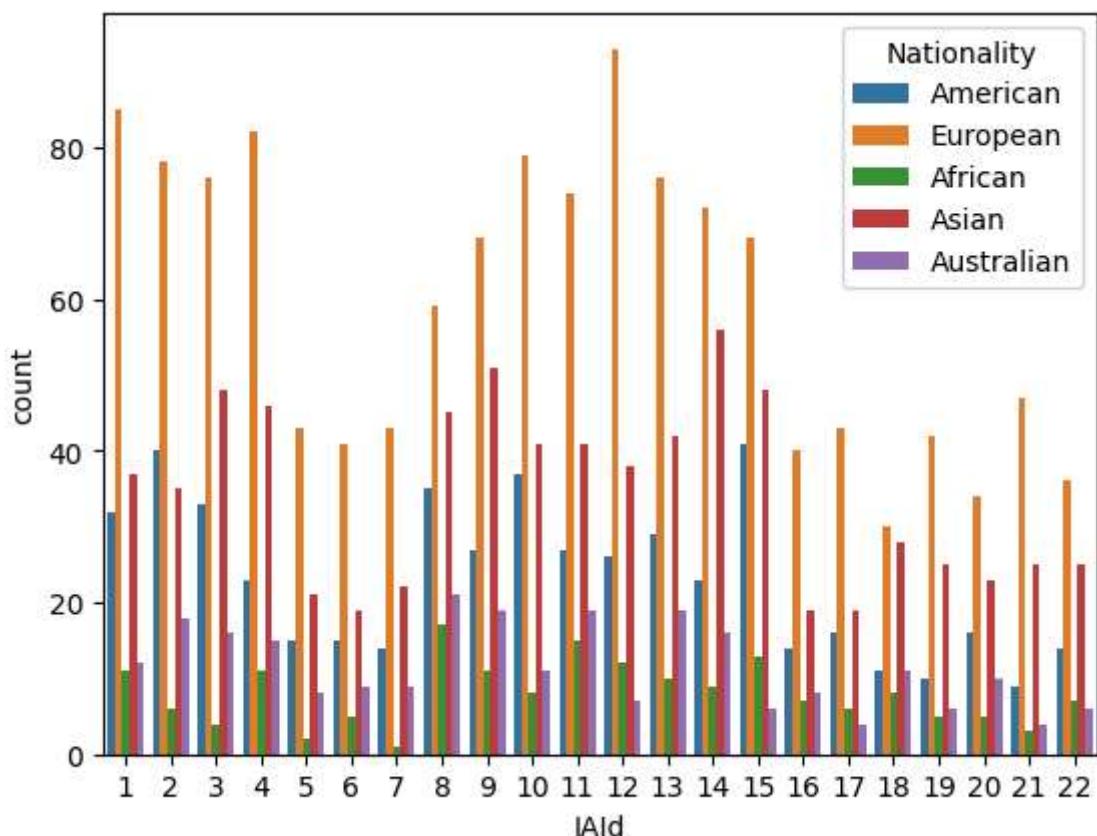
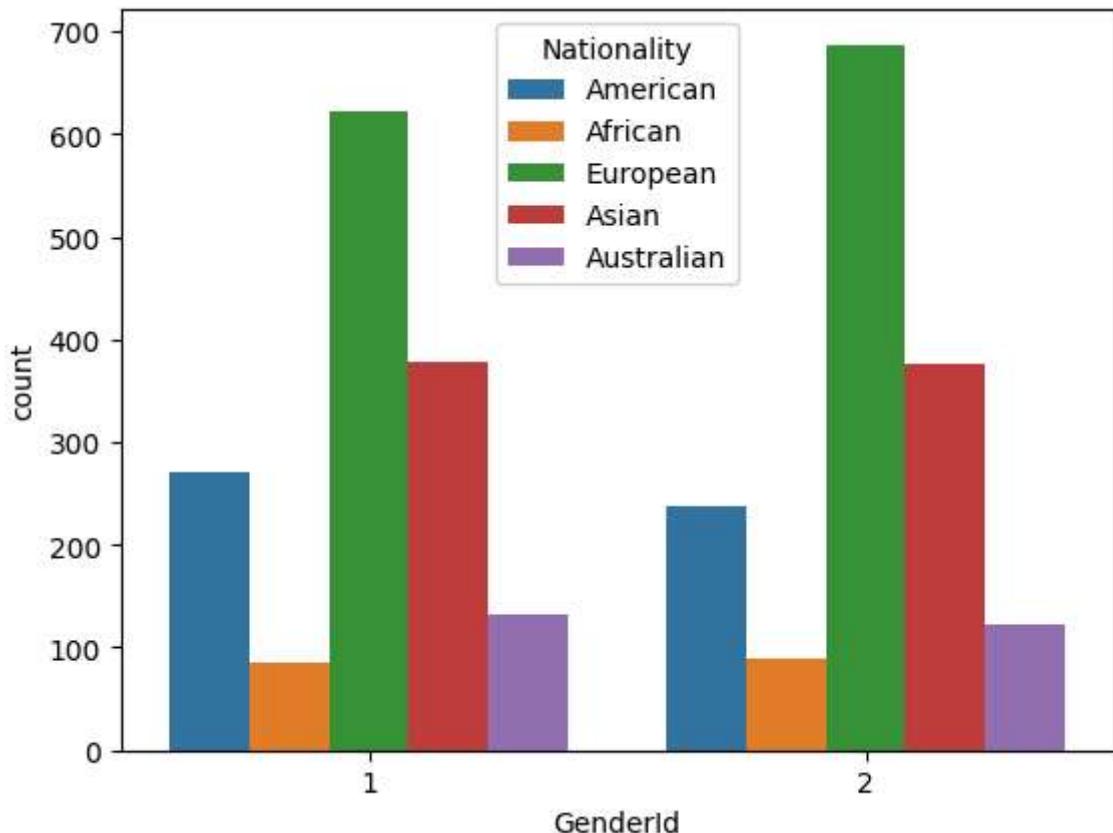


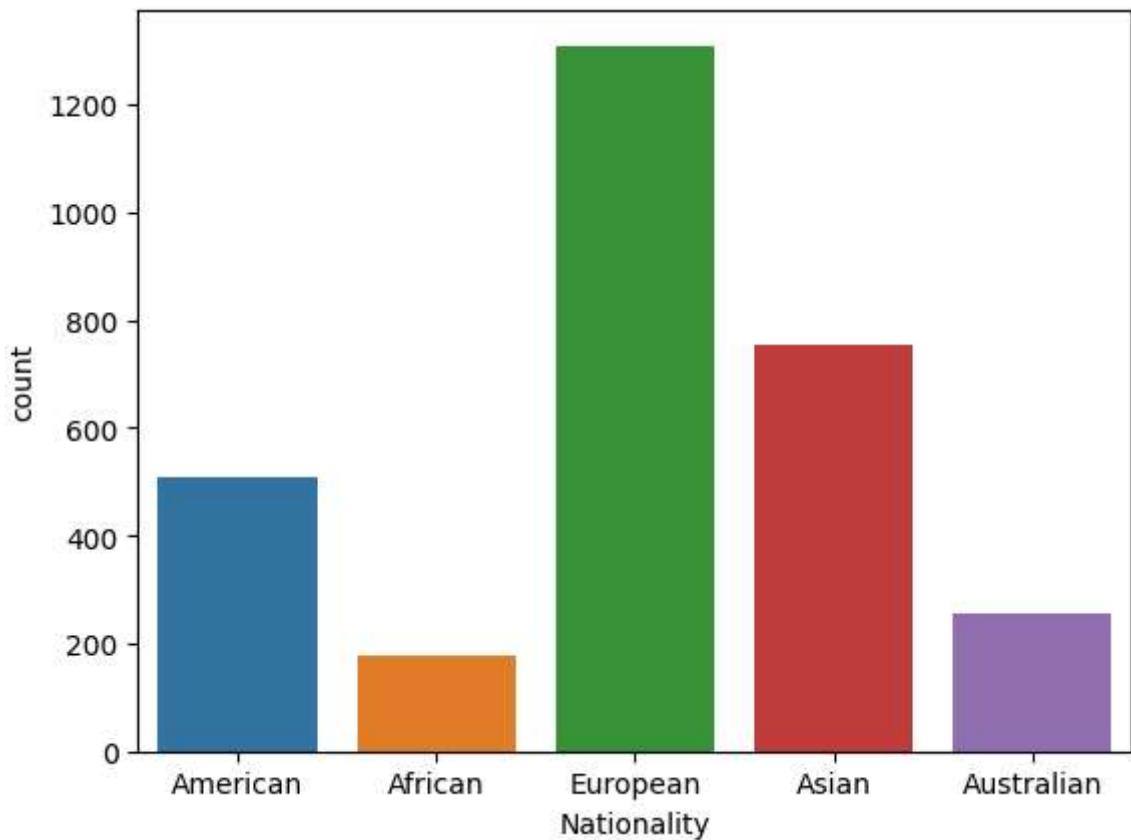
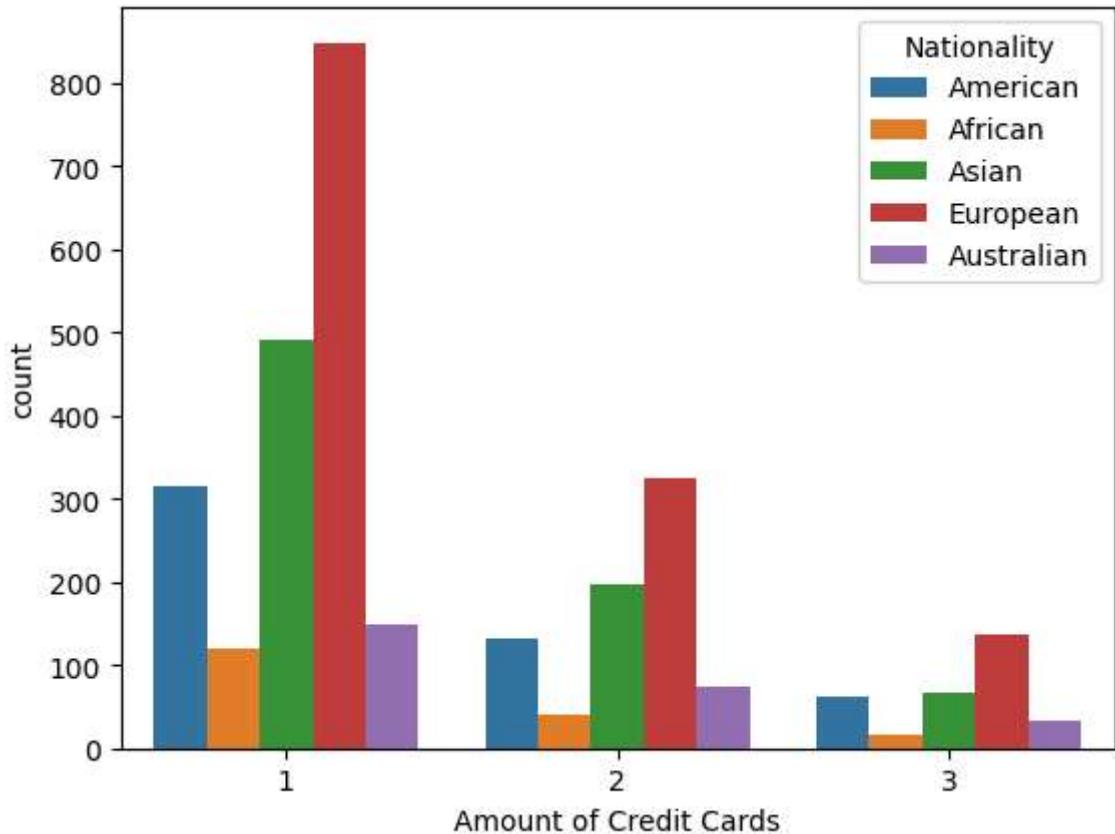


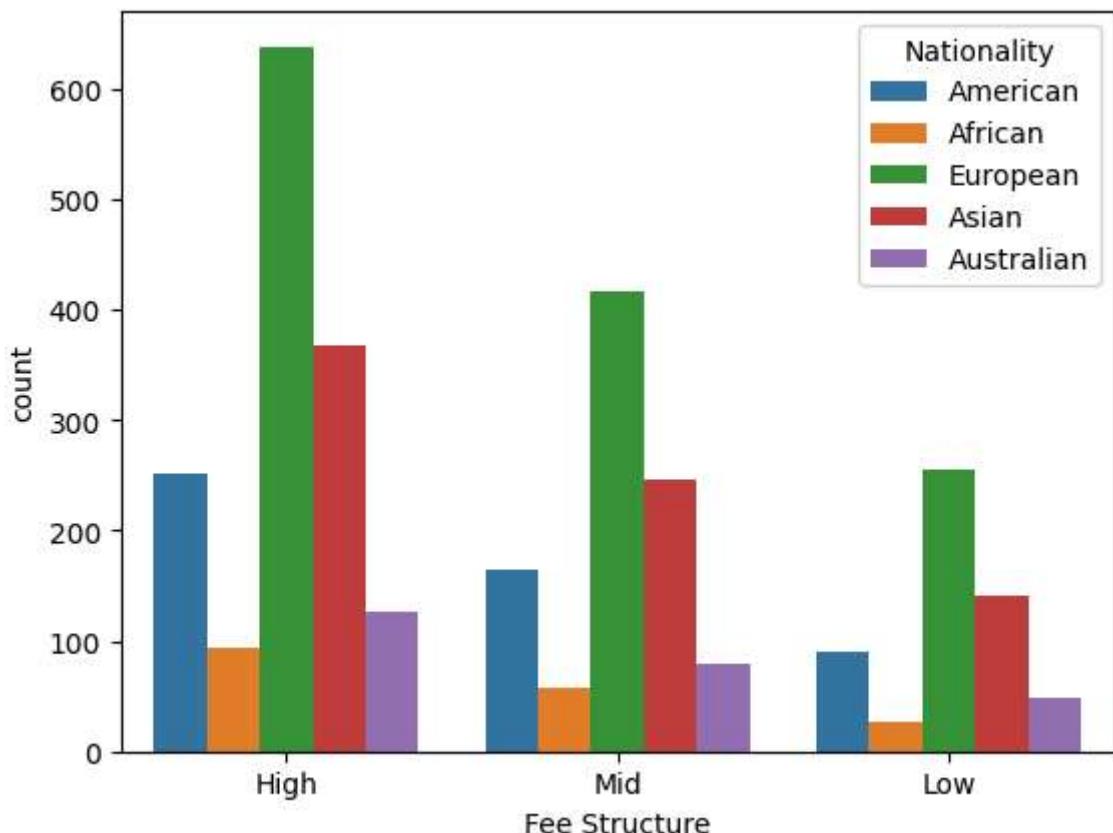
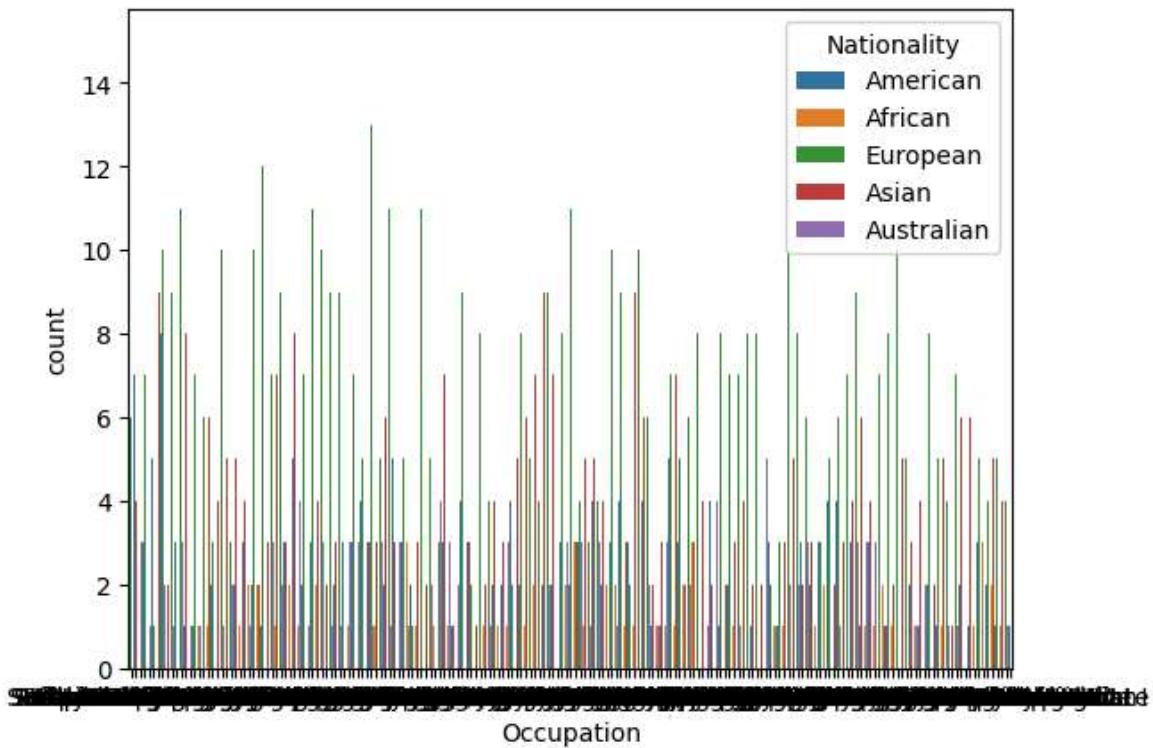
Bivariate Analysis

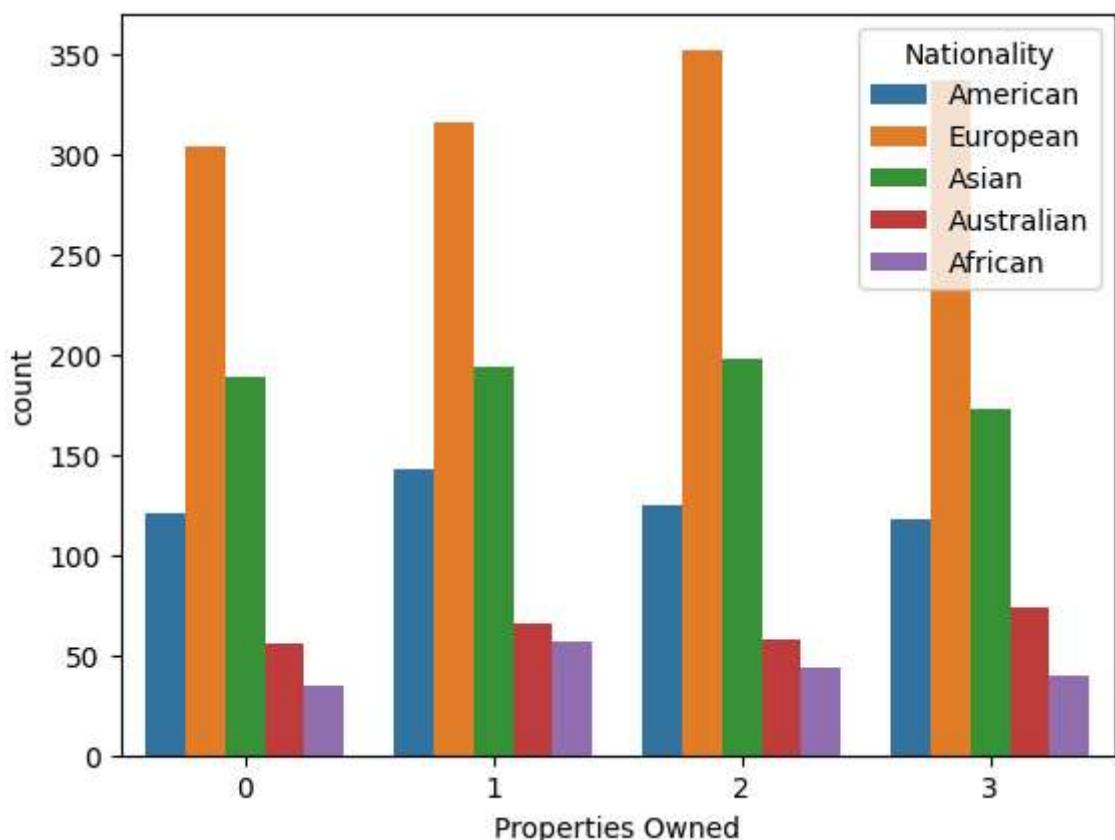
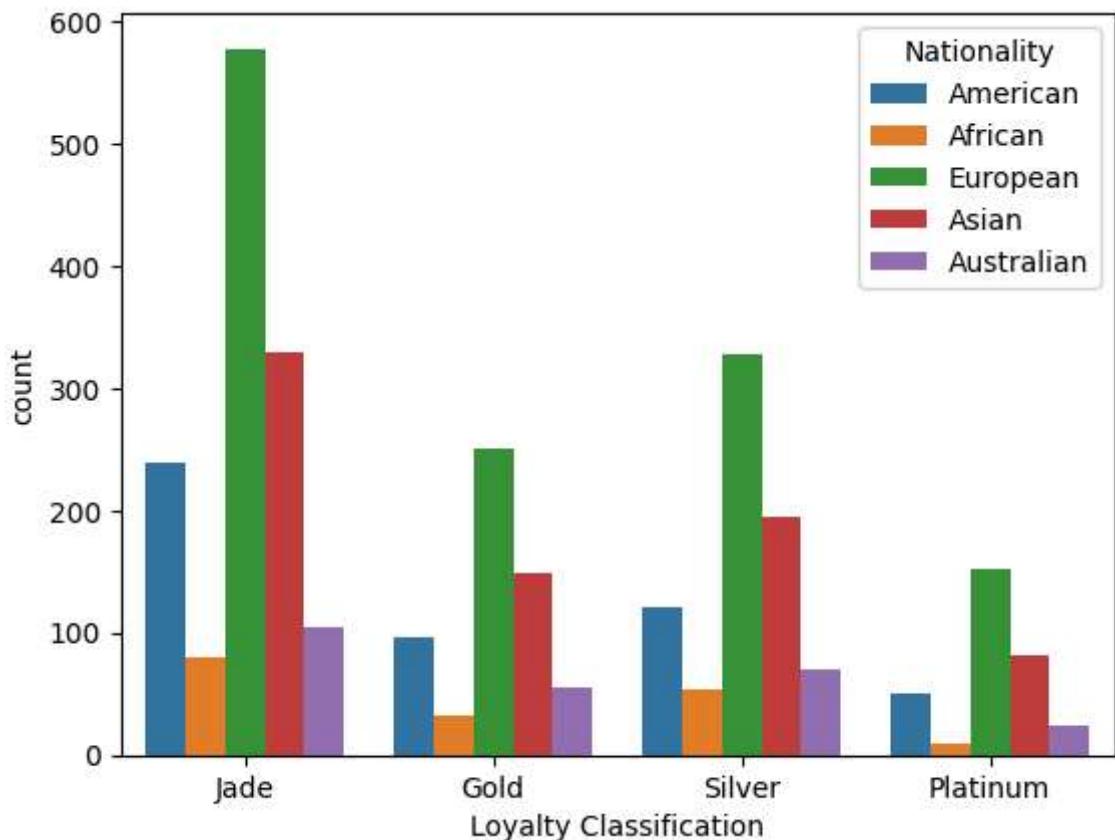
```
In [12]: for i, predictor in enumerate(df[["BRIId", "GenderId", "IAId", "Amount of Credit"]]):  
    plt.figure(i)  
    sns.countplot(data=df, x=predictor, hue='Nationality')
```

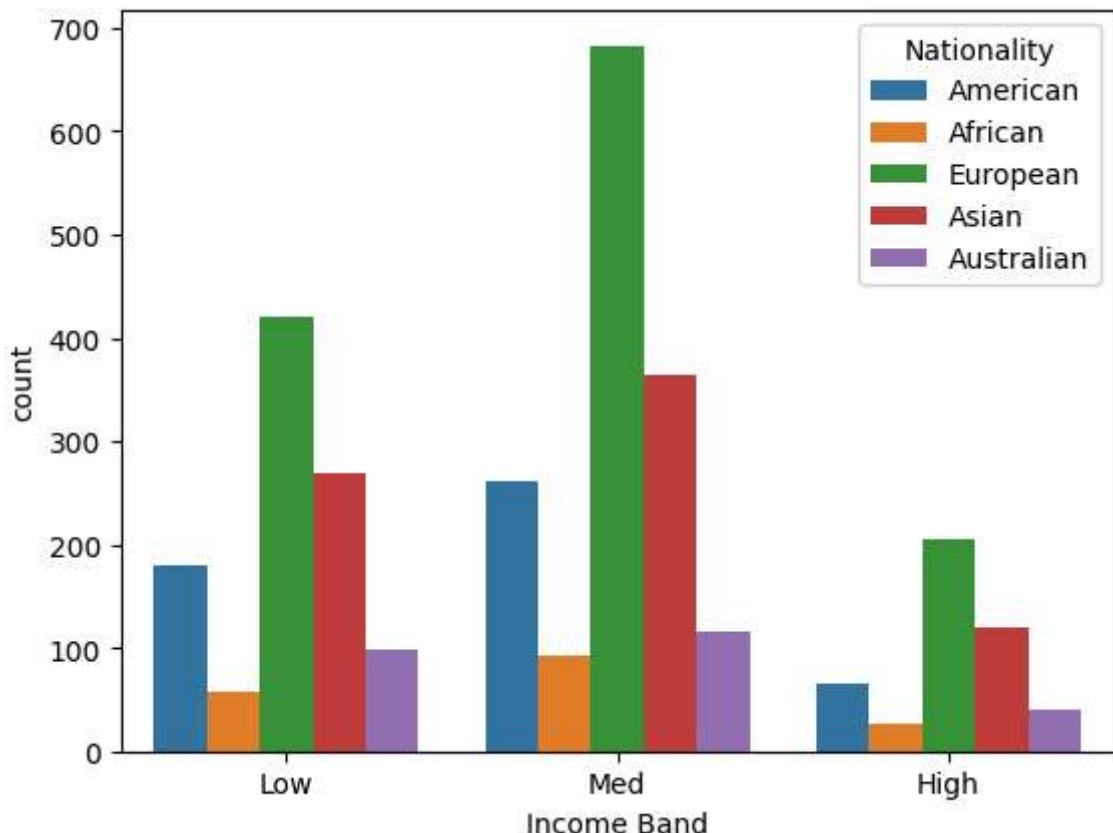
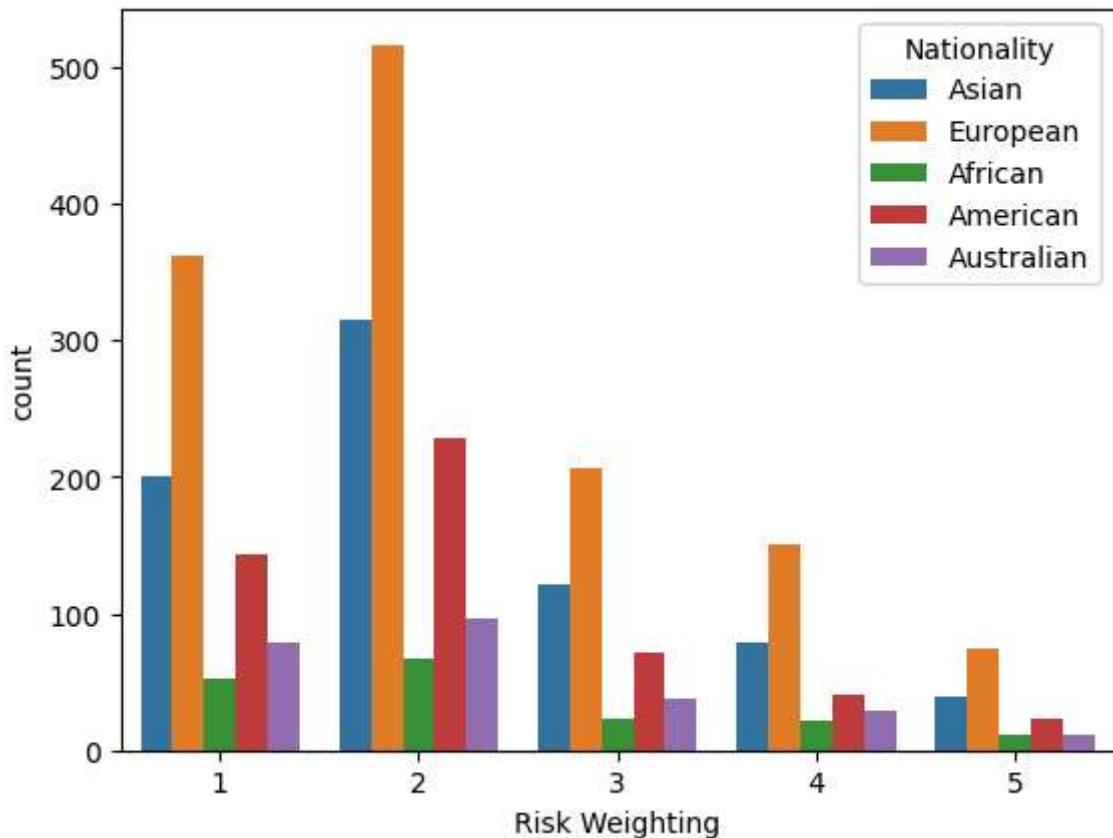








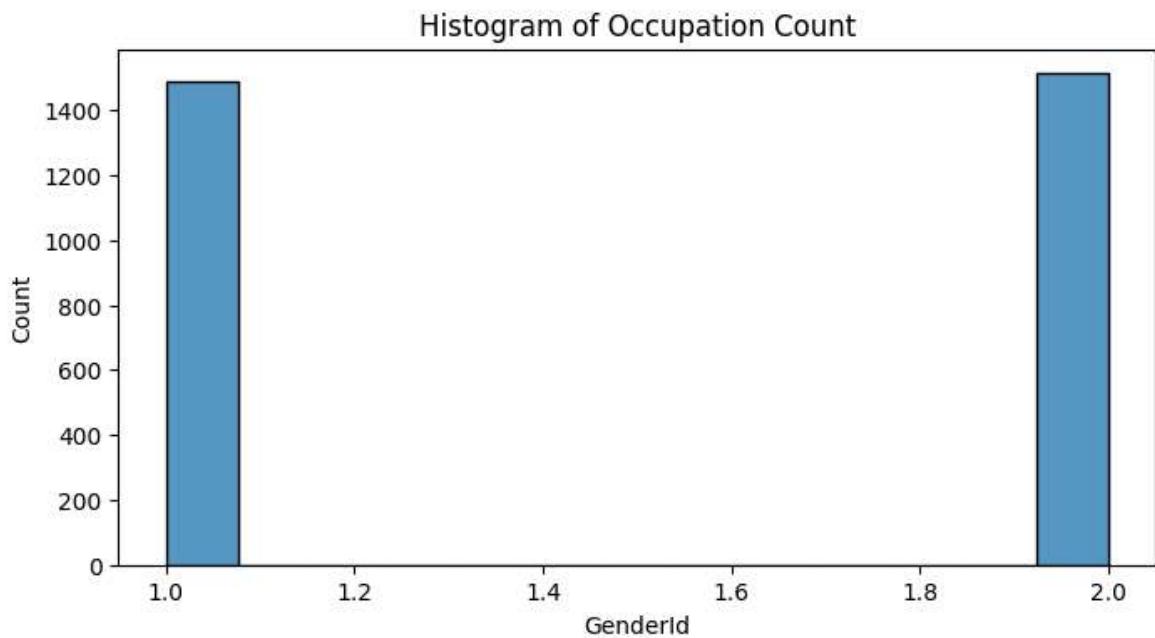
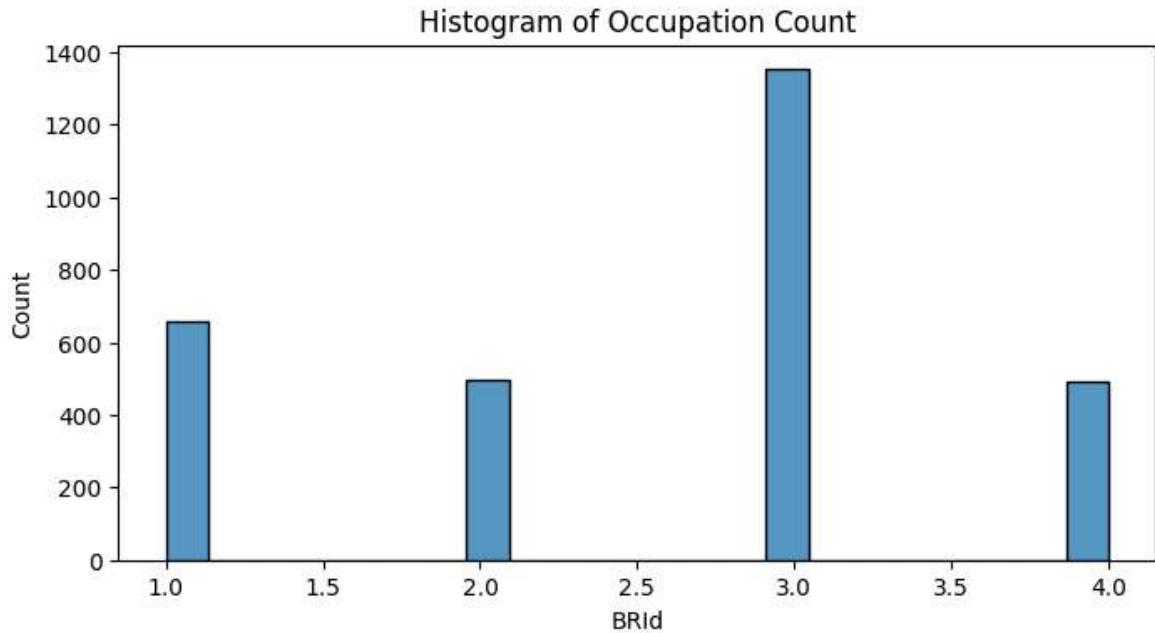




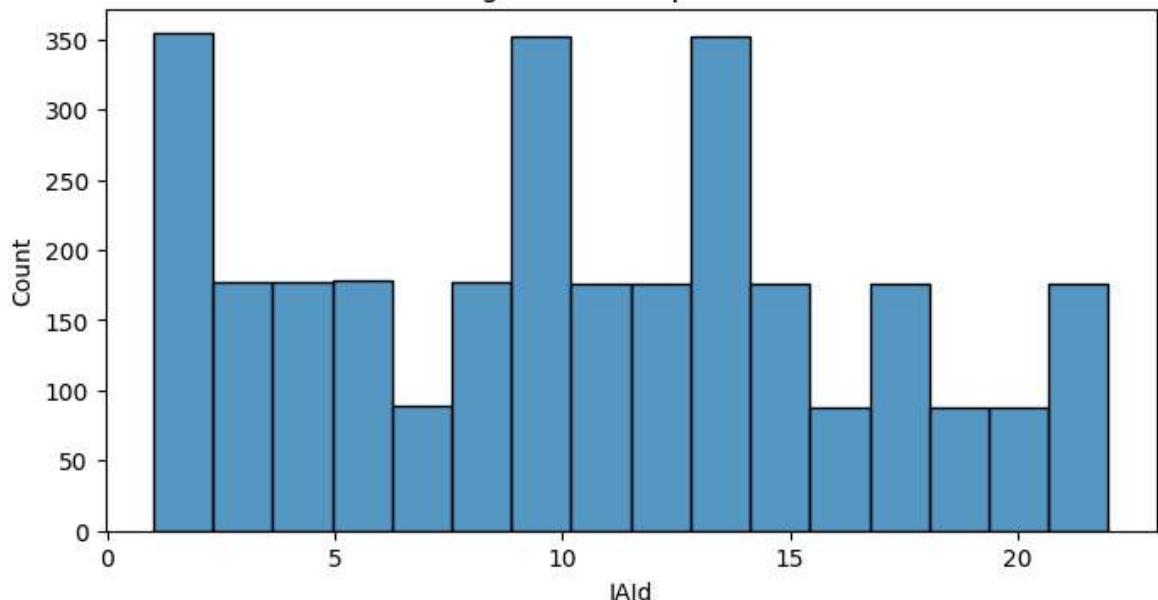
```
In [13]: # Histogram of value counts for different Occupation

for col in categorical_cols:
    if col == "Occupation":
        continue
    plt.figure(figsize=(8,4))
    sns.histplot(df[col])
    plt.title('Histogram of Occupation Count')
```

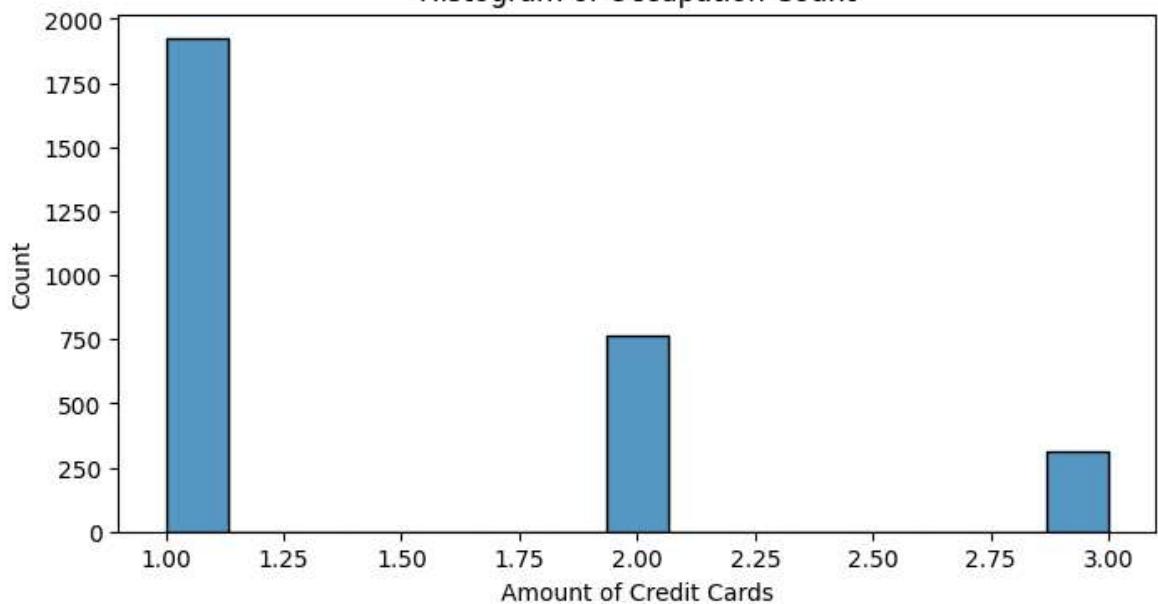
```
plt.xlabel(col)
plt.ylabel("Count")
plt.show()
```



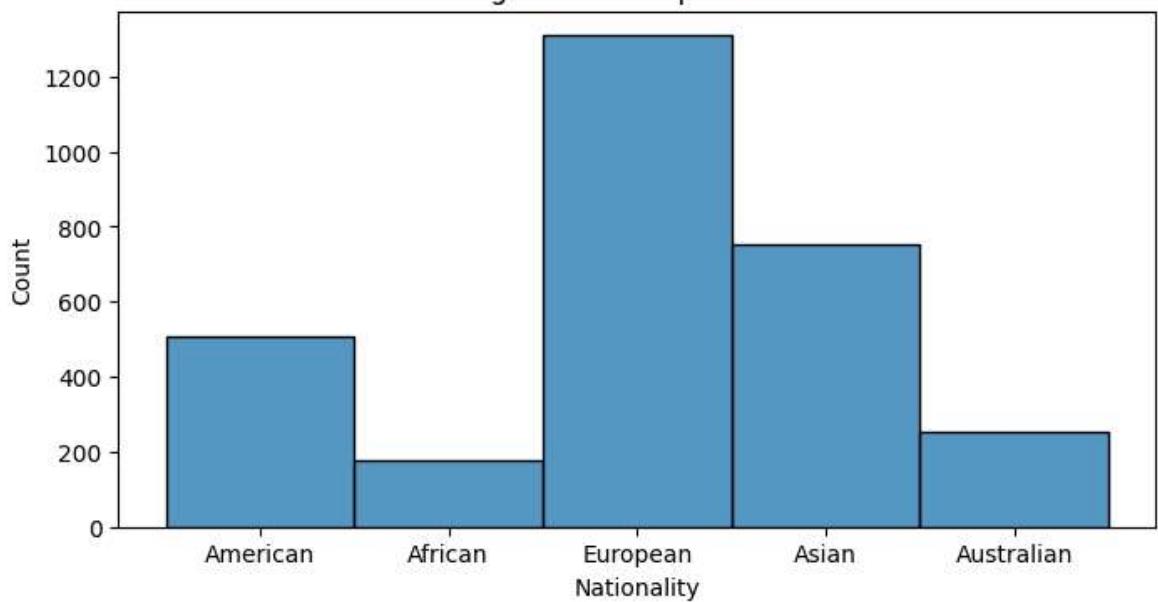
Histogram of Occupation Count



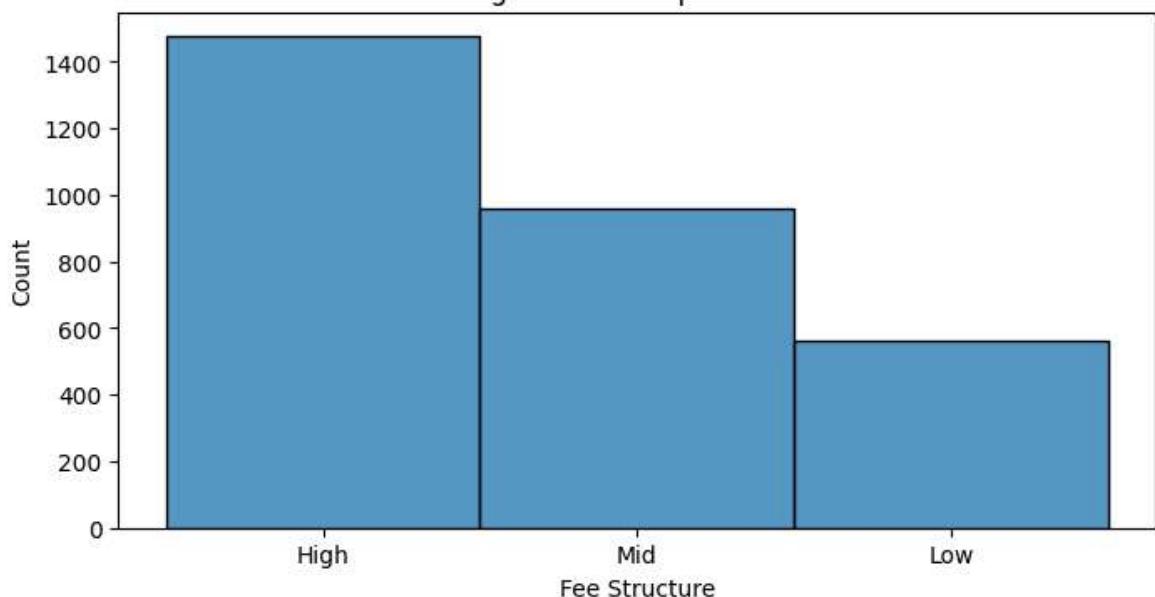
Histogram of Occupation Count



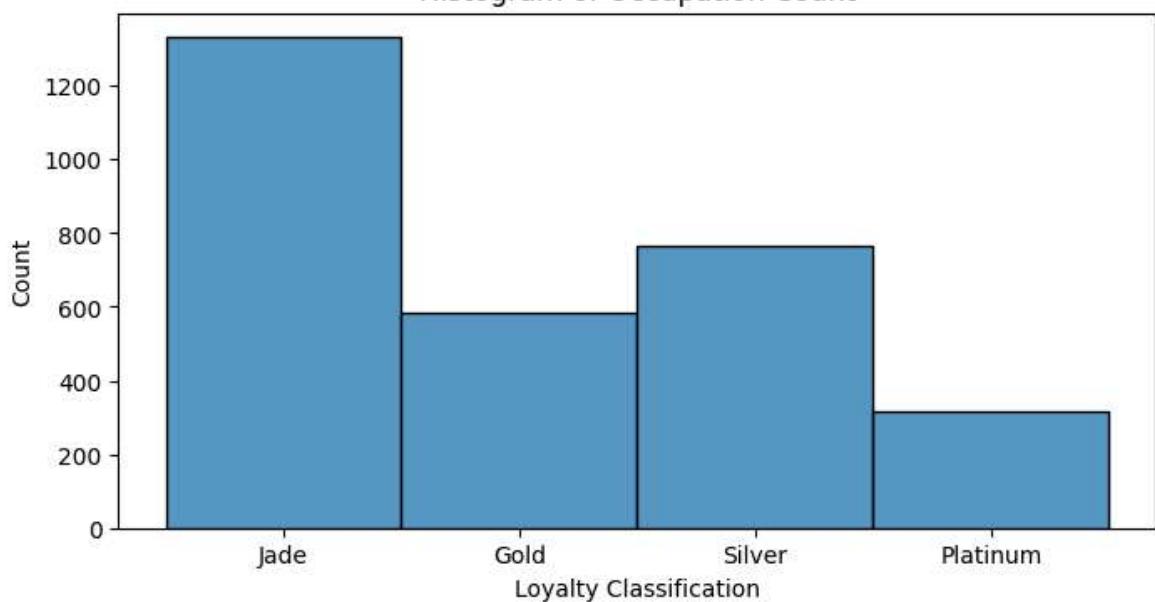
Histogram of Occupation Count



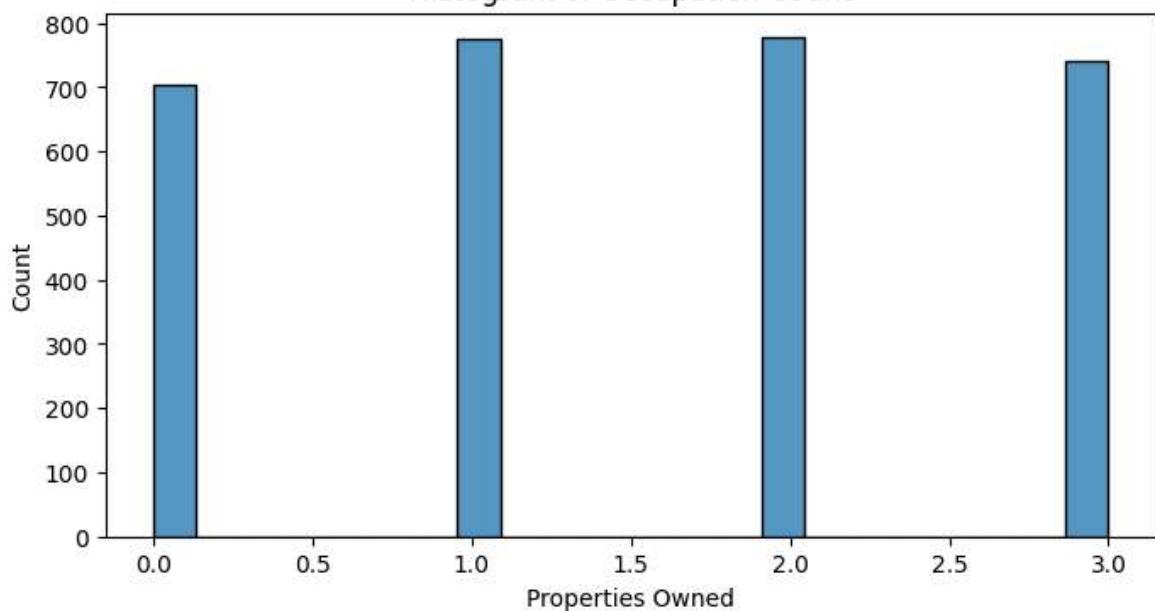
Histogram of Occupation Count

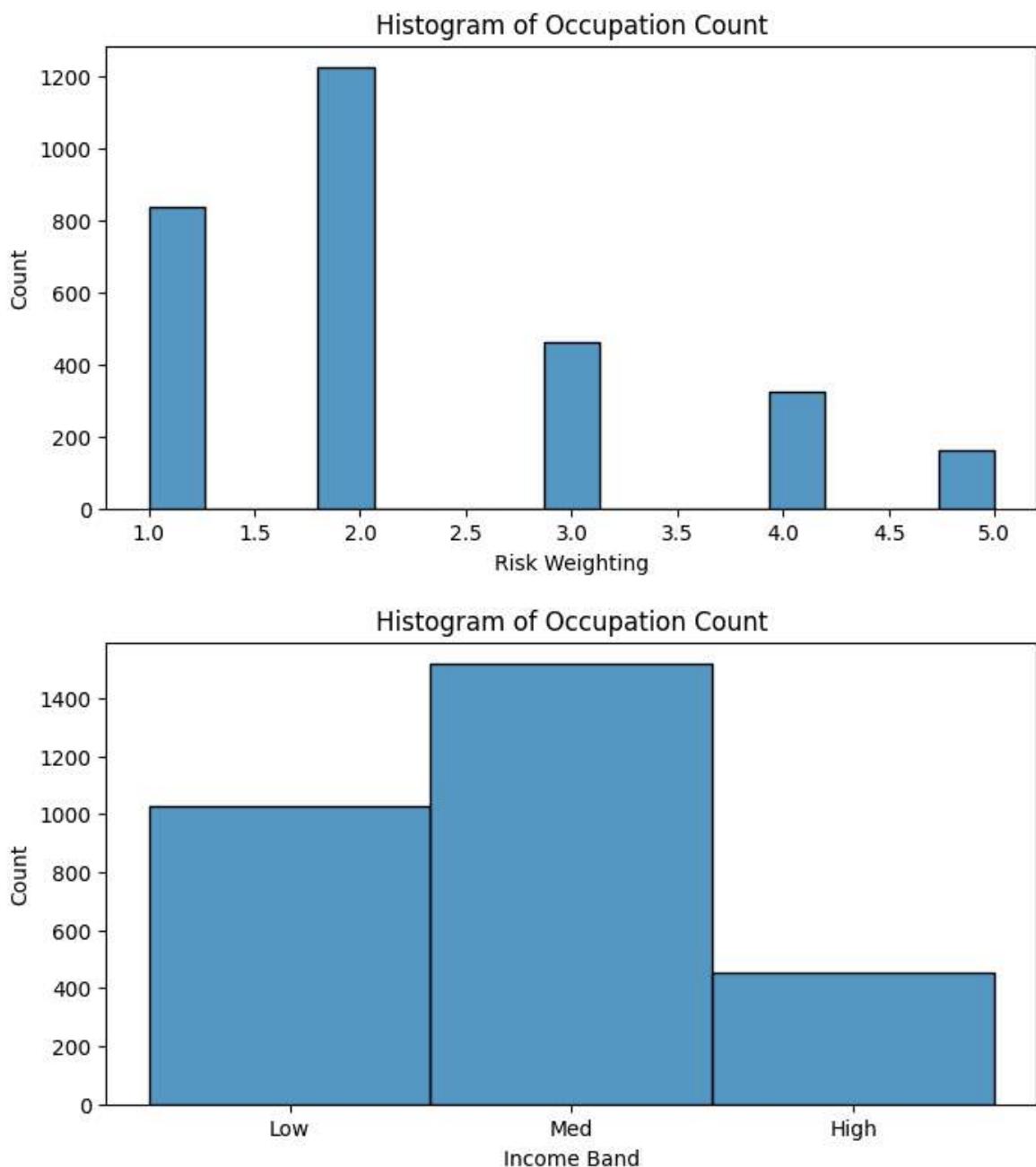


Histogram of Occupation Count



Histogram of Occupation Count

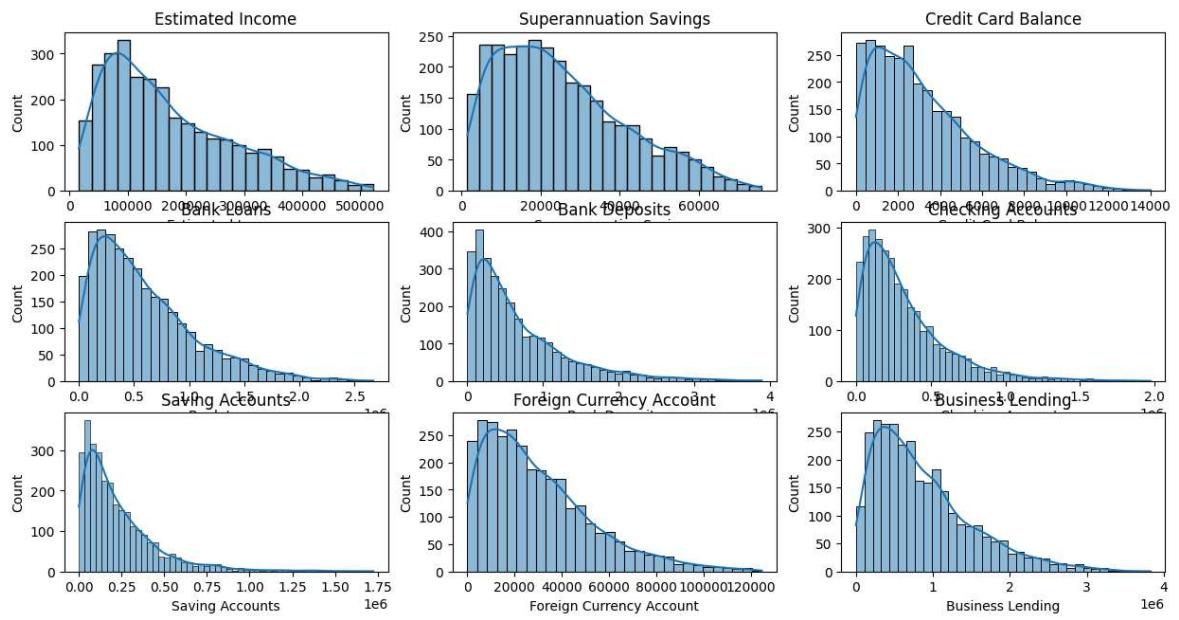




Numerical Analysis

```
In [14]: numerical_cols = ['Estimated Income', 'Superannuation Savings', 'Credit Card Bal

# Univariate analysis and visualization
plt.figure(figsize=(15,10))
for i,col in enumerate(numerical_cols):
    plt.subplot(4,3,i+1)
    sns.histplot(df[col],kde=True)
    plt.title(col)
plt.show()
```



Heatmaps

```
In [15]: numerical_cols = ['Estimated Income', 'Superannuation Savings', 'Credit Card Balance']

correlation_matrix = df[numerical_cols].corr()

plt.figure(figsize=(12,12))
sns.heatmap(correlation_matrix, annot=True, cmap='crest', fmt=".2f")
plt.title("Correlation Matrix")
plt.show()
```



```
In [5]: import pandas as pd
import numpy as np
```

```
In [6]: df = pd.read_csv('Banking.csv')
df.head()
```

Out[6]:

	Client ID	Name	Age	Location ID	Joined Bank	Banking Contact	Nationality	Occupation	Stru
0	IND81288	Raymond Mills	24	34324	06-05-2019	Anthony Torres	American	Safety Technician IV	
1	IND65833	Julia Spencer	23	42205	10-12-2001	Jonathan Hawkins	African	Software Consultant	
2	IND47499	Stephen Murray	27	7314	25-01-2010	Anthony Berry	European	Help Desk Operator	
3	IND72498	Virginia Garza	40	34594	28-03-2019	Steve Diaz	American	Geologist II	
4	IND60181	Melissa Sanders	46	41269	20-07-2012	Shawn Long	American	Assistant Professor	

5 rows × 25 columns

◀ ▶

```
In [7]: # Check the shape of the DataFrame
print("Shape of the DataFrame:", df.shape)

# Get a concise summary of the DataFrame
print("\nDataFrame Info:")
df.info()
```

Shape of the DataFrame: (3000, 25)

DataFrame Info:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000 entries, 0 to 2999
Data columns (total 25 columns):
 #   Column           Non-Null Count Dtype  
 ---  -- 
 0   Client ID        3000 non-null  object  
 1   Name              3000 non-null  object  
 2   Age               3000 non-null  int64   
 3   Location ID      3000 non-null  int64   
 4   Joined Bank       3000 non-null  object  
 5   Banking Contact  3000 non-null  object  
 6   Nationality       3000 non-null  object  
 7   Occupation        3000 non-null  object  
 8   Fee Structure     3000 non-null  object  
 9   Loyalty Classification 3000 non-null  object  
 10  Estimated Income  3000 non-null  float64 
 11  Superannuation Savings 3000 non-null  float64 
 12  Amount of Credit Cards 3000 non-null  int64   
 13  Credit Card Balance 3000 non-null  float64 
 14  Bank Loans        3000 non-null  float64 
 15  Bank Deposits    3000 non-null  float64 
 16  Checking Accounts 3000 non-null  float64 
 17  Saving Accounts   3000 non-null  float64 
 18  Foreign Currency Account 3000 non-null  float64 
 19  Business Lending  3000 non-null  float64 
 20  Properties Owned  3000 non-null  int64   
 21  Risk Weighting    3000 non-null  int64   
 22  BRId              3000 non-null  int64   
 23  GenderId          3000 non-null  int64   
 24  IAId              3000 non-null  int64 

dtypes: float64(9), int64(8), object(8)
memory usage: 586.1+ KB
```

In [8]: `df["Estimated Income"]`

```
Out[8]: 0      75384.77
1      289834.31
2      169935.23
3      356808.11
4      130711.68
...
2995    297617.14
2996    42397.46
2997    48339.88
2998    107265.87
2999    56826.53

Name: Estimated Income, Length: 3000, dtype: float64
```

In [9]: `# Define income band boundaries`

```
bins = [0, 100000, 300000, float('inf')]
labels = ['Low', 'Mid', 'High']

# Create the 'Income Band' column using pd.cut
df['Income Band'] = pd.cut(df['Estimated Income'], bins=bins, labels=labels, inc
```

In [10]: `# Examine the distribution of unique categories in categorical columns`

```
categorical_cols = df[['Risk Weighting', 'Nationality', 'Occupation', 'Fee Structur
```

```
for col in categorical_cols:
    # if col in ["Client ID", "Name", "Joined Bank"]:
    #     continue
    print(f"\nValue Counts for '{col}':")
    display(df[col].value_counts())
```

Value Counts for 'Risk Weighting':

Risk Weighting

2	1222
1	836
3	460
4	322
5	160

Name: count, dtype: int64

Value Counts for 'Nationality':

Nationality

European	1309
Asian	754
American	507
Australian	254
African	176

Name: count, dtype: int64

Value Counts for 'Occupation':

Occupation

Associate Professor	28
Structural Analysis Engineer	28
Recruiter	25
Account Coordinator	24
Human Resources Manager	24

..

Office Assistant IV	8
Automation Specialist I	7
Computer Systems Analyst I	6
Developer III	5
Senior Sales Associate	4

Name: count, Length: 195, dtype: int64

Value Counts for 'Fee Structure':

Fee Structure

High	1476
Mid	962
Low	562

Name: count, dtype: int64

Value Counts for 'Loyalty Classification':

Loyalty Classification

Jade	1331
Silver	767
Gold	585
Platinum	317

Name: count, dtype: int64

Value Counts for 'Properties Owned':

Properties Owned

2	777
1	776
3	742
0	705

Name: count, dtype: int64

Value Counts for 'Risk Weighting':

```
Risk Weighting
2    1222
1     836
3     460
4     322
5     160
Name: count, dtype: int64
Value Counts for 'Occupation':
Occupation
Associate Professor      28
Structural Analysis Engineer 28
Recruiter                25
Account Coordinator      24
Human Resources Manager  24
                           ..
Office Assistant IV      8
Automation Specialist I   7
Computer Systems Analyst I 6
Developer III             5
Senior Sales Associate    4
Name: count, Length: 195, dtype: int64
Value Counts for 'Income Band':
Income Band
Mid      1517
Low      1027
High     456
Name: count, dtype: int64
```

```
In [11]: # Generate descriptive statistics for numerical columns
print("\nDescriptive Statistics for Numerical Columns:")
display(df.describe())
```

Descriptive Statistics for Numerical Columns:

	Age	Location ID	Estimated Income	Superannuation Savings	Amount of Credit Cards	Credit C Bala
count	3000.000000	3000.000000	3000.000000	3000.000000	3000.000000	3000.000
mean	51.039667	21563.323000	171305.034263	25531.599673	1.463667	3176.206
std	19.854760	12462.273017	111935.808209	16259.950770	0.676387	2497.094
min	17.000000	12.000000	15919.480000	1482.030000	1.000000	1.170
25%	34.000000	10803.500000	82906.595000	12513.775000	1.000000	1236.630
50%	51.000000	21129.500000	142313.480000	22357.355000	1.000000	2560.805
75%	69.000000	32054.500000	242290.305000	35464.740000	2.000000	4522.632
max	85.000000	43369.000000	522330.260000	75963.900000	3.000000	13991.990



```
In [12]: # Check for missing values
missing_values = df.isnull().sum()
print("Missing values per column:\n", missing_values)
```

```
Missing values per column:
Client ID           0
Name                0
Age                 0
Location ID         0
Joined Bank         0
Banking Contact    0
Nationality         0
Occupation          0
Fee Structure       0
Loyalty Classification 0
Estimated Income   0
Superannuation Savings 0
Amount of Credit Cards 0
Credit Card Balance 0
Bank Loans          0
Bank Deposits      0
Checking Accounts   0
Saving Accounts     0
Foreign Currency Account 0
Business Lending    0
Properties Owned   0
Risk Weighting      0
BRIId               0
GenderId            0
IAId                0
Income Band         0
dtype: int64
```

```
In [13]: df['Joined Bank'] = pd.to_datetime(df['Joined Bank'], format='%d-%m-%Y')
print(df['Joined Bank'].dtype)
```

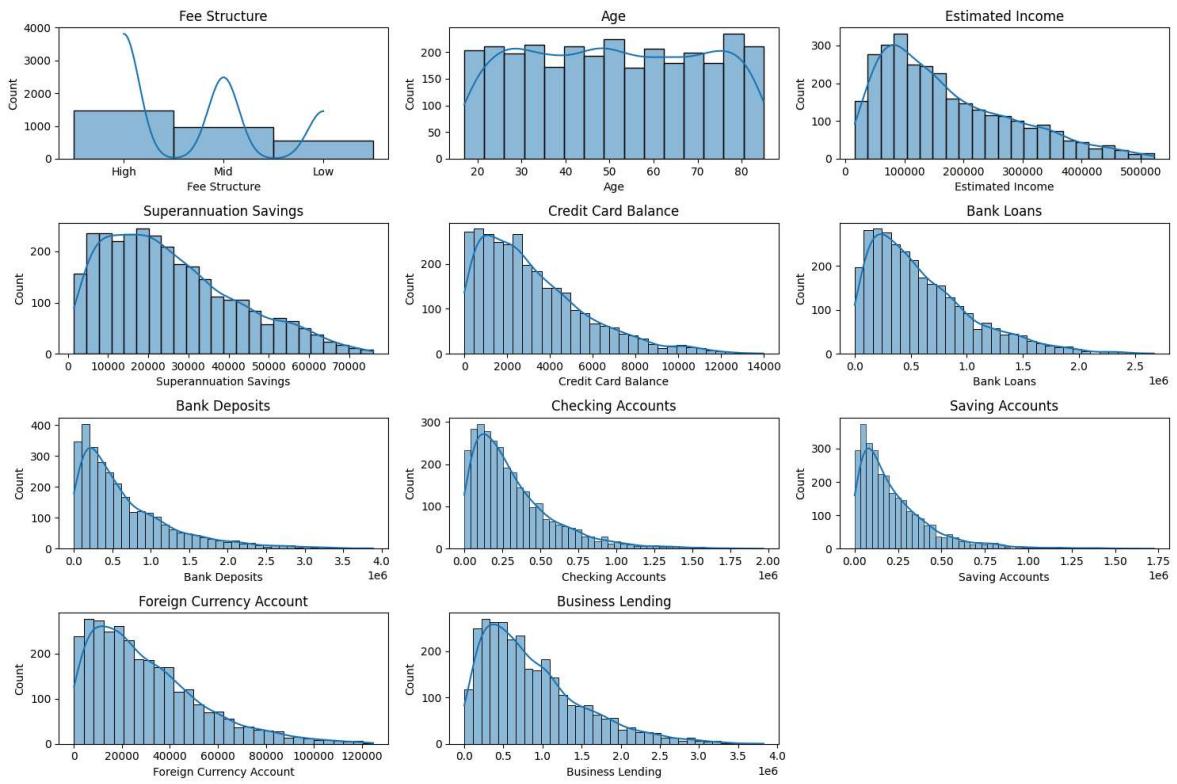
datetime64[ns]

```
In [14]: import matplotlib.pyplot as plt
import seaborn as sns

# Numerical analysis and exploration
numerical_cols = ['Fee Structure', 'Age', 'Estimated Income', 'Superannuation Savings']

# Univariate analysis and visualization
plt.figure(figsize=(15, 10))
for i, col in enumerate(numerical_cols):
    plt.subplot(4, 3, i + 1)
    sns.histplot(df[col], kde=True)
    plt.title(col)
plt.tight_layout()
plt.show()
```

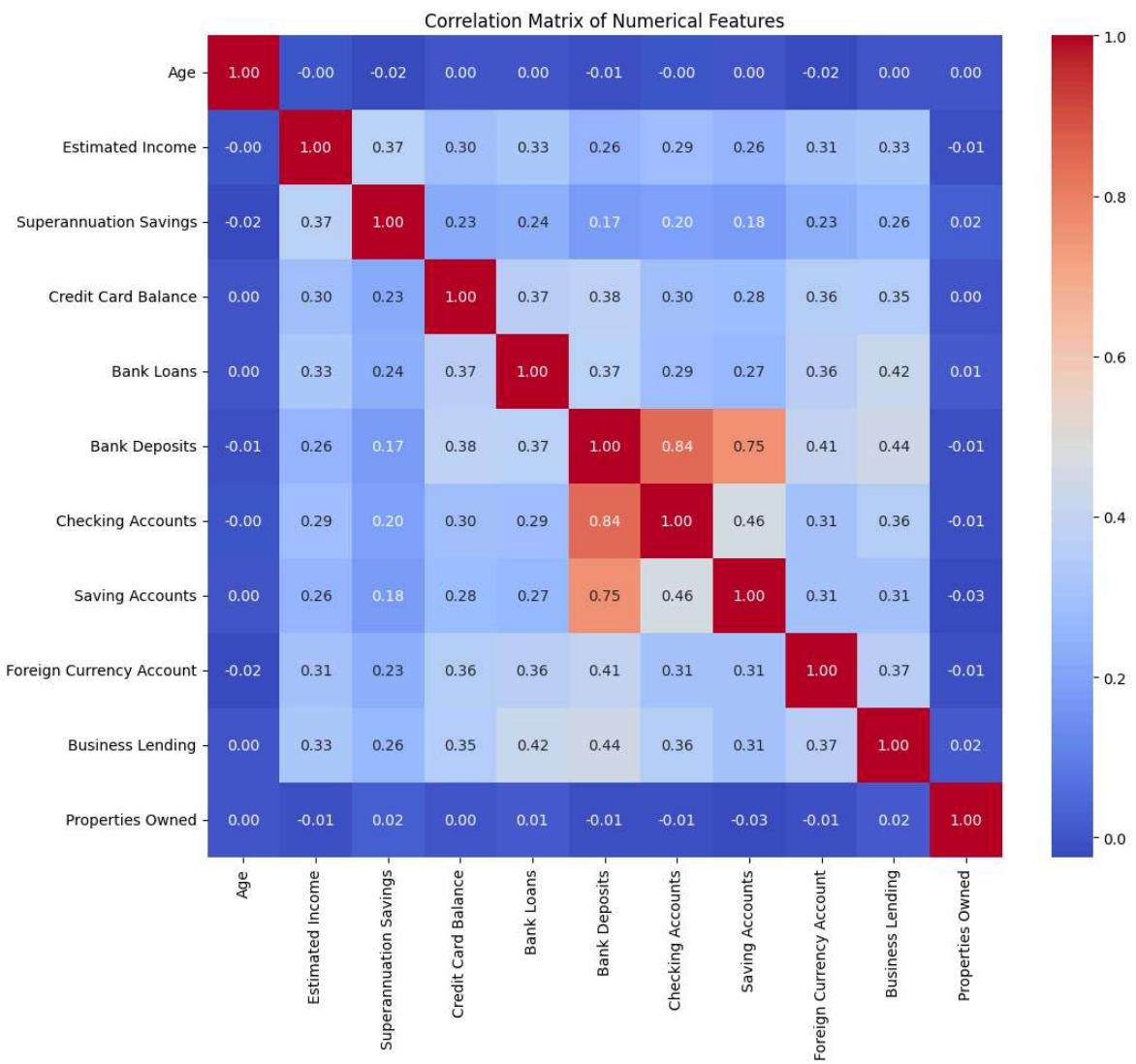
Bank_EDA_2



```
In [15]: # Select numerical columns for correlation analysis
numerical_cols = ['Age', 'Estimated Income', 'Superannuation Savings', 'Credit C
               'Bank Loans', 'Bank Deposits', 'Checking Accounts', 'Saving Ac
               'Foreign Currency Account', 'Business Lending', 'Properties Ow

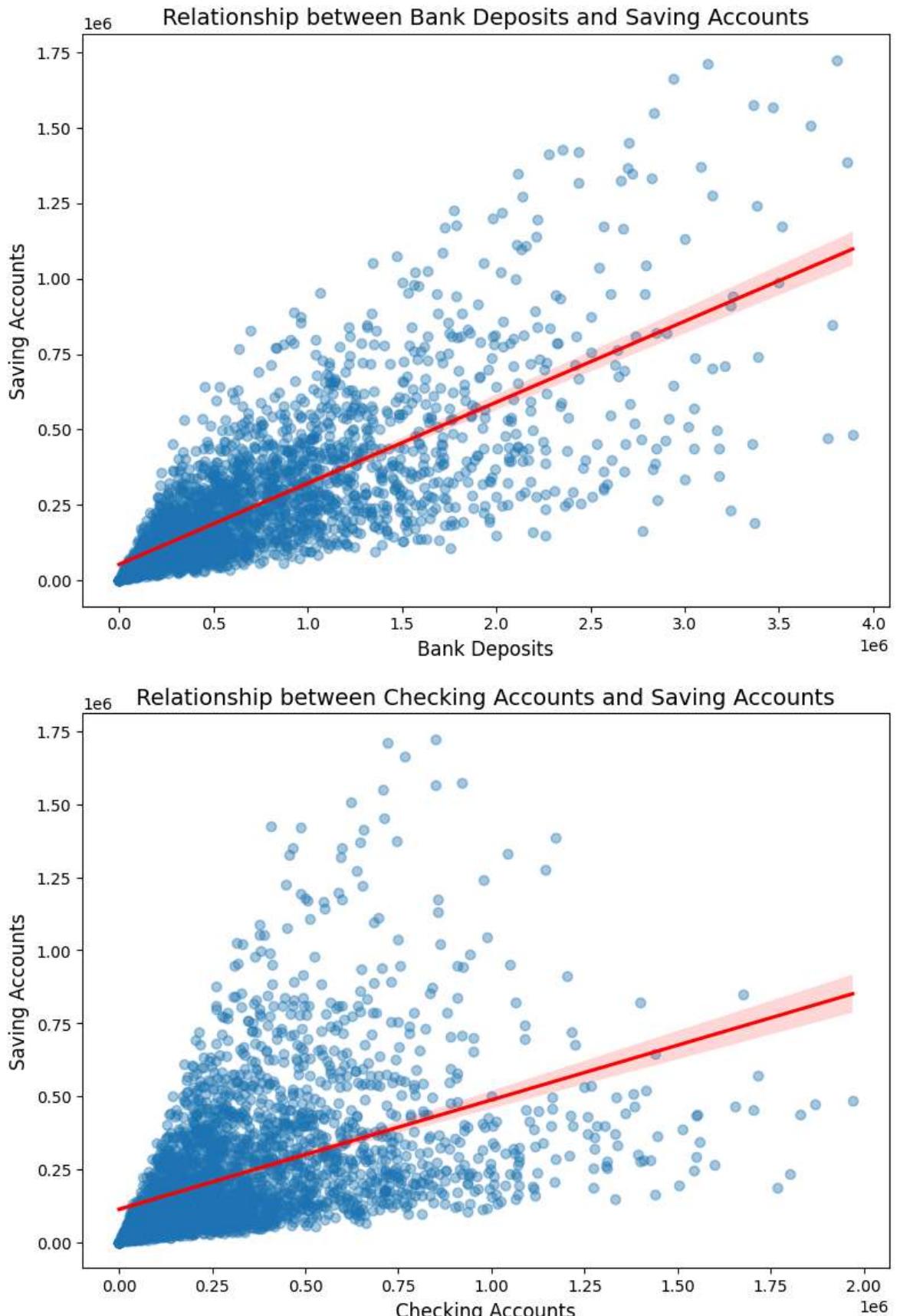
# Calculate the correlation matrix
correlation_matrix = df[numerical_cols].corr()

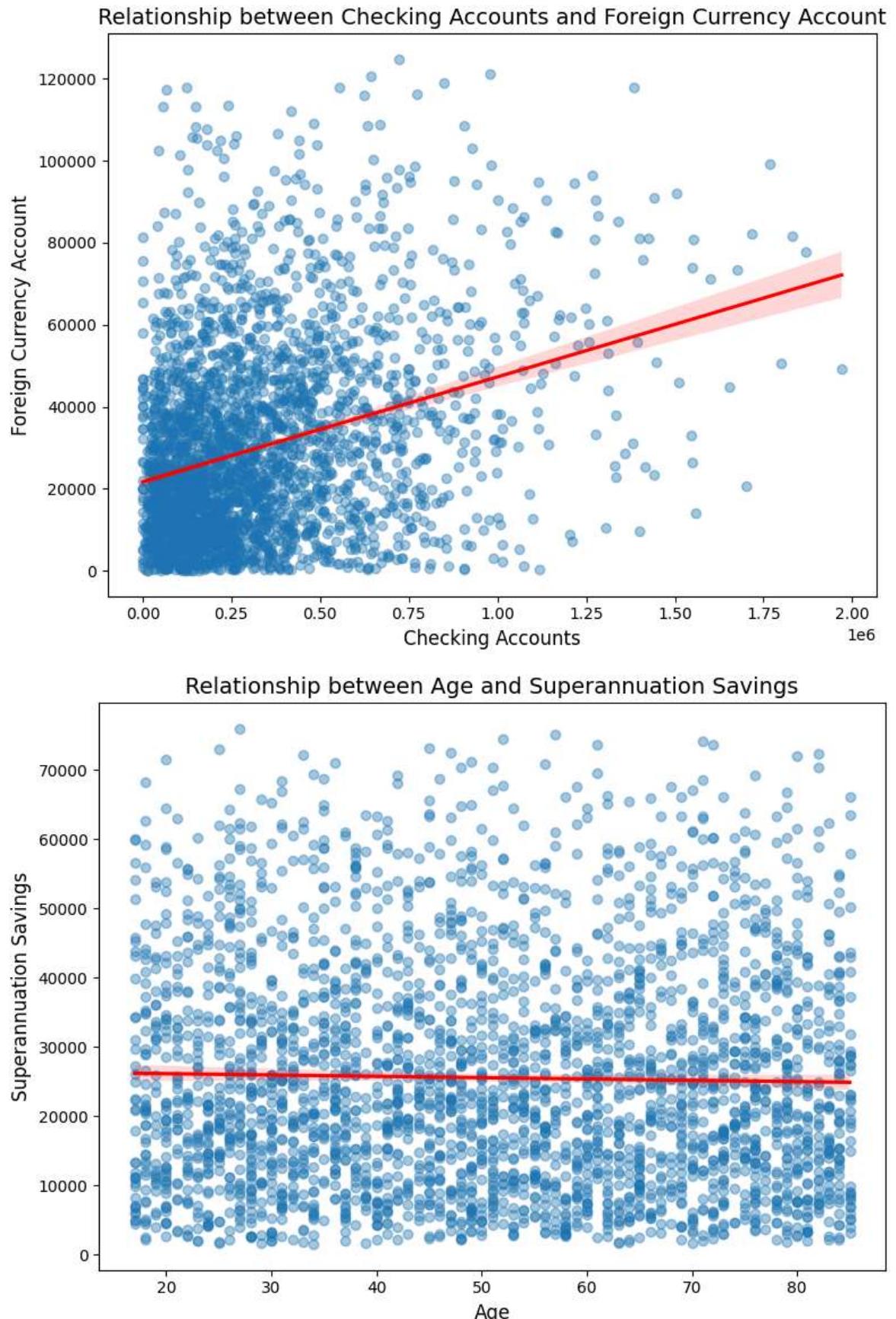
# Create a heatmap of the correlation matrix
plt.figure(figsize=(12, 10))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Correlation Matrix of Numerical Features')
plt.show()
```

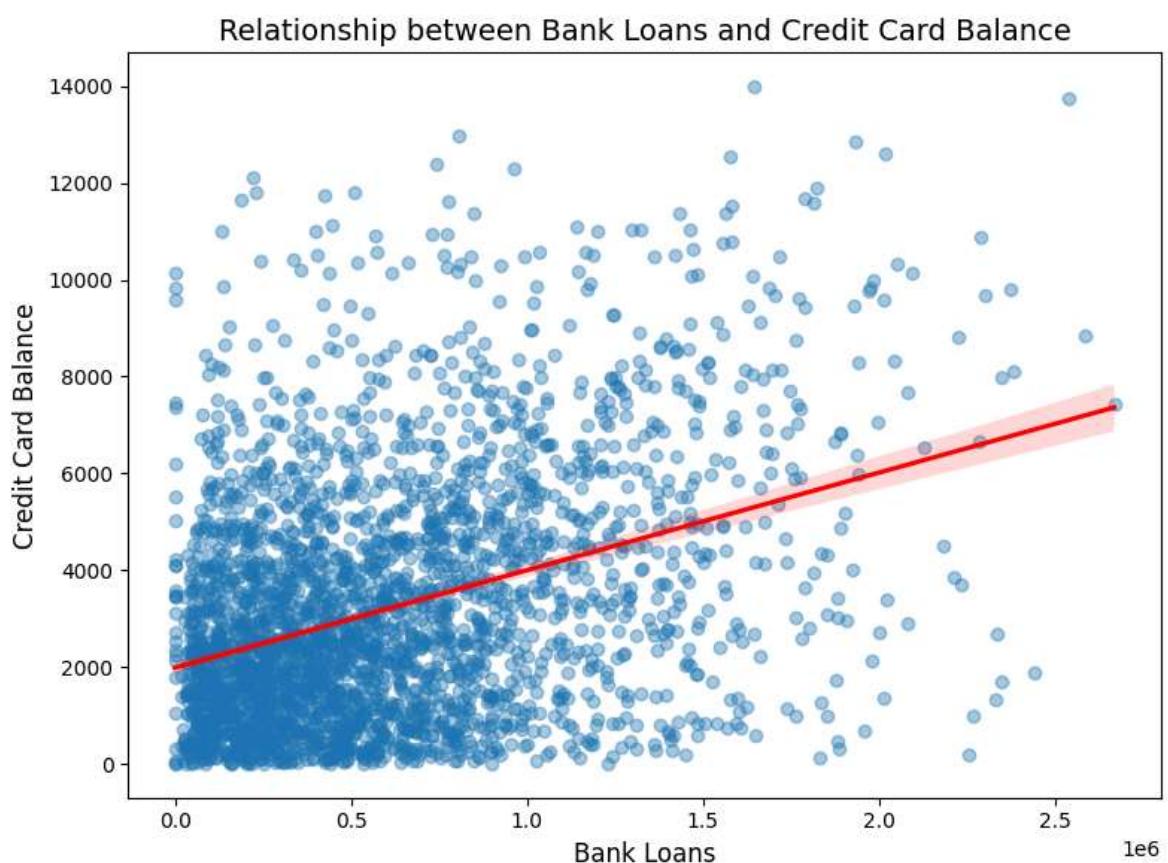
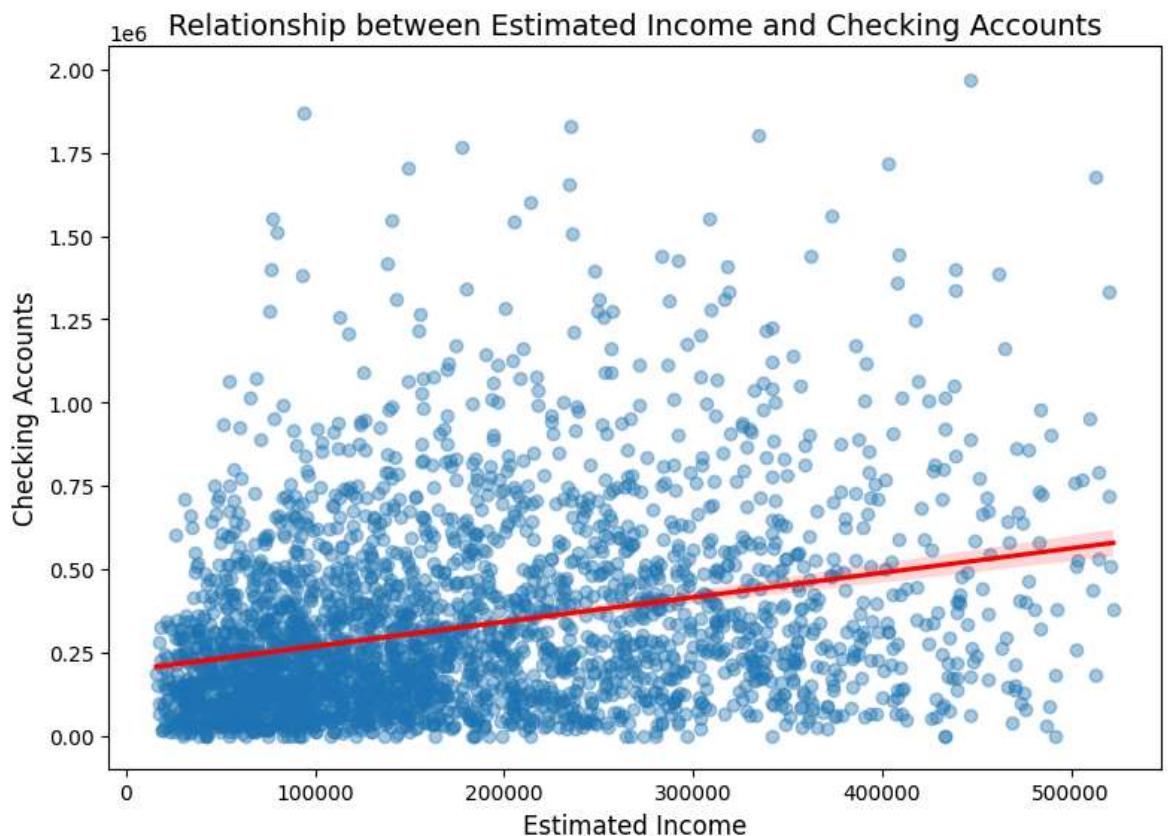


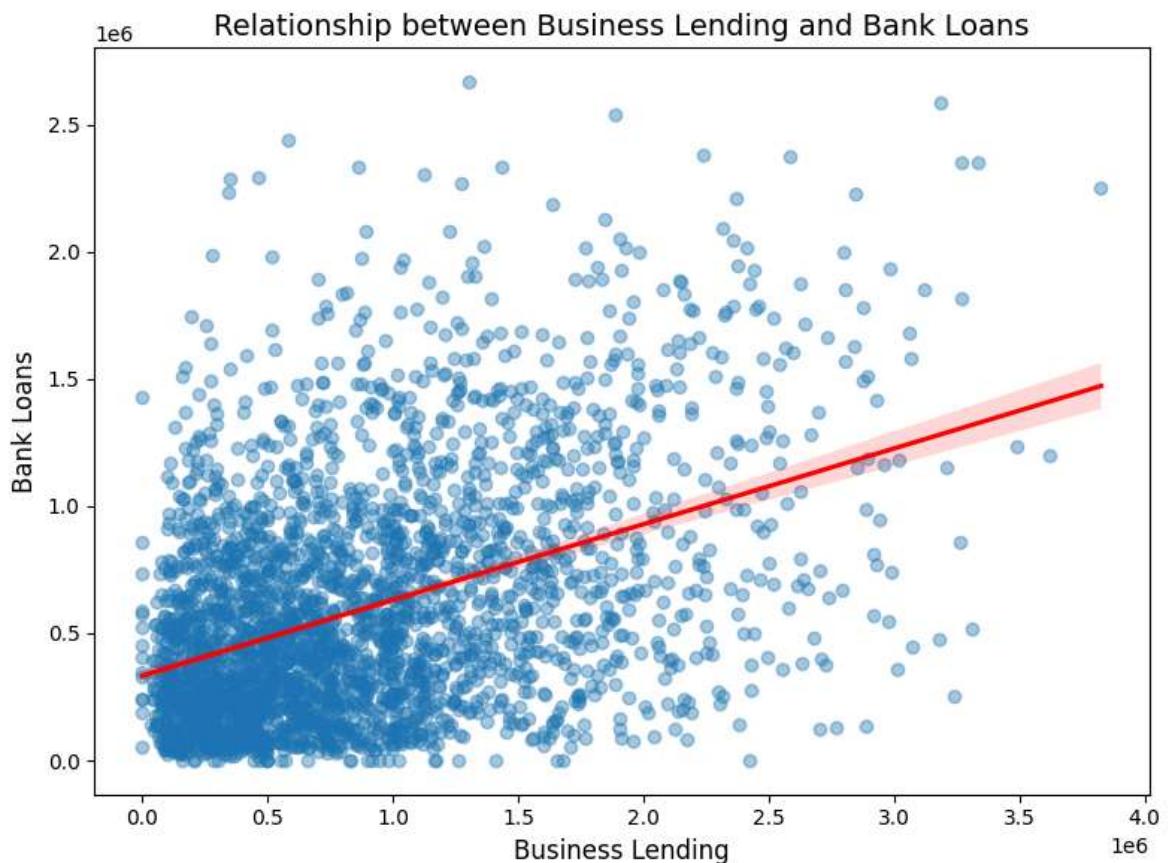
```
In [16]: pairs_to_plot = [
    ('Bank Deposits', 'Saving Accounts'),
    ('Checking Accounts', 'Saving Accounts'),
    ('Checking Accounts', 'Foreign Currency Account'),
    ('Age', 'Superannuation Savings'),
    ('Estimated Income', 'Checking Accounts'),
    ('Bank Loans', 'Credit Card Balance'),
    ('Business Lending', 'Bank Loans'),
]

for x_col, y_col in pairs_to_plot:
    plt.figure(figsize=(8, 6))
    sns.regplot(
        data=df,
        x=x_col,
        y=y_col,
        scatter_kws={'alpha': 0.4},           # semi-transparent points
        line_kws={'color': 'red'}            # best-fit line color
    )
    plt.title(f'Relationship between {x_col} and {y_col}', fontsize=14)
    plt.xlabel(x_col, fontsize=12)
    plt.ylabel(y_col, fontsize=12)
    plt.tight_layout()
    plt.show()
```









IAId

Investment Advisor

1

Victor Dean

2

Jeremy Porter

3

Ernest Knight

4

Eric Shaw

5

Kevin Kim

6

Victor Rogers

7

Eugene Cunningham

8

Joe Carroll

9

Steve Sanchez

10

Lawrence Sanchez

11

Peter Castillo

12

Victor Gutierrez

13

Daniel Carroll

14

Carl Anderson

15

Nicholas Ward

16

Fred Bryant

17

Ryan Taylor

18

Sean Vasquez

19

Nicholas Morrison

20

Jack Phillips

21

Juan Ramirez

22

Gregory Boyd

BRId

Banking Relationship

1

Retail

2

Institutional

3

Private Bank

4

Commercial

Gender	Gender
1	Male
2	Female

Global Employee Database Analysis - Q3 2023												
Client ID	Name	Age	Location ID	Joined Bank	Banking Contact	Nationality	Occupation	Fee Structure	Loyalty Classification	Estimated Income	Superannuation Savings	
Experience Level	Education Qualifications	Skills & Certifications	Performance Metrics	Training History	Work Experience	Project Management	Communication Skills	Problem Solving	Teamwork	Adaptability	Leadership Potential	
IND61268	Raymond Mills	24	34324	06-05-2019	Anthony Torres	American	Safety Technician IV	High	Jade	75384.77		
IND59333	Julia Spencer	23	42205	10-12-2001	Jonathan Hawkins	African	Software Consultant	High	Jade	289834.31		
IND47499	Stephen Murray	27	7314	25-01-2010	Anthony Berry	European	Help Desk Operator	High	Gold	169935.23		
IND72495	Virginia Garza	40	34694	28-03-2019	Steve Diaz	American	Geologist II	Mid	Silver	356810.11		
IND60181	Melissa Sanders	45	41269	20-07-2012	Shawn Long	American	Assistant Professor	Mid	Platinum	130711.68		
IND78532	Samuel Hudson	23	13204	07-02-2019	Douglas Tucker	American	Help Desk Technician	High	Silver	118326.96		
IND95683	Timothy Alexander	45	42910	02-08-2002	Douglas Tucker	Asian	Account Coordinator	High	Gold	57336.47		
IND40785	Carl Martin	78	6127	03-11-2010	Steve Diaz	European	Automation Specialist II	Mid	Gold	65125.80		
IND13570	Philip Day	67	32656	07-04-2015	Bruce Butler	Asian	Software Test Engineer II	High	Silver	67649.47		
IND53299	Jason Sims	51	28340	20-11-1995	Joe Price	European	Geologist III	Mid	Silver	65389.36		
IND76263	Amy Martinez	55	40459	18-10-2014	Adam Hernandez	European	Administrative Officer	High	Jade	74966.51		
IND64642	David Johnston	73	25663	12-09-2005	Chris Armstrong	American	Database Administrator II	Mid	Jade	111449.06		
IND28761	Wayne Foster	45	35687	17-03-2018	Joshua Ryan	African	Staff Scientist	Low	Silver	115637.19		
IND17897	Carlos Moore	44	19654	02-01-1995	Paul Larson	American	Programmer I	High	Jade	124434.78		
IND66325	Lisa Johnston	36	33368	05-06-2020	Mark Montgomery	Asian	Software Test Engineer I	High	Platinum	260795.34		
IND74197	Andrew Mills	55	27913	05-01-2021	Shawn Wallace	European	Analyst	Mid	Silver	70892.52		
IND28503	Jack Coleman	61	9605	22-08-2014	Ernest Rivera	Asian	Staff Accountant II	High	Silver	257120.22		
IND64538	Aaron Day	66	36323	19-01-2020	Gregory Simmons	Asian	Assistant Media Planner	Low	Jade	145704.66		
IND63604	Karen Weaver	43	6289	31-03-2019	Frank Brown	American	Staff Accountant I	Low	Jade	34484.16		
IND32064	Mary Fox	63	7694	09-03-2009	Adam Hernandez	Australian	Compensation Analyst	Mid	Jade	123060.89		
IND72934	Carlos Little	41	38321	03-04-2020	Jonathan Hawkins	American	Geologist I	Low	Jade	129981.31		
IND16101	Roger Boyd	68	12772	31-12-2015	Adam Hernandez	European	Web Designer II	High	Jade	123645.42		
IND93121	Aaron Marshall	26	28651	14-01-2013	Ernest Rivera	American	Media Manager IV	Mid	Jade	308162.53		
IND93310	Cheryl Stewart	21	8767	23-05-2021	Victor Martinez	European	Accounting Assistant II	High	Gold	122579.41		
IND71301	Annie Nguyen	41	14954	19-09-2021	Joshua Ryan	American	Programmer II	High	Jade	30542.07		
IND21279	Christopher Evans	36	24058	20-03-2013	Adam Hernandez	European	Administrative Assistant IV	High	Jade	50661.56		
IND99616	Maria Clark	34	8623	29-01-2014	Joe Hanson	Asian	Help Desk Operator	High	Silver	502309.09		
IND35589	Jimmy Simpson	52	38270	08-01-2018	George Lewis	European	Geological Engineer	High	Jade	87019.73		
IND40198	Louise Sanders	78	29119	24-05-2005	Shawn Cook	European	Junior Executive	Low	Jade	246615.84		
IND49516	Angela Alvarez	34	31338	22-09-2016	Patrick Graham	Australian	Biostatistician IV	Low	Jade	130023.35		
IND55475	Henry Grant	75	21778	05-04-2013	Anthony Simpson	European	Software Test Engineer I	Low	Jade	82703.81		
IND61272	Larry Foster	47	34773	10-12-2020	Raymond Alexander	European	Internal Auditor	High	Gold	137256.09		
IND33884	Carol GoPKRalaz	34	29287	29-05-2021	Carl Nguyen	European	Chemical Engineer	High	Platinum	224640.76		
IND53121	Timothy Stanley	79	29278	15-02-2021	Joshua Little	American	Accountant II	High	Jade	147292.91		
IND25680	Billy Wilson	73	99	25-09-2017	Dennis Morris	European	Registered Nurse	High	Platinum	263248.97		
IND65464	Kathleen Riley	60	20746	28-11-2017	Benjamin Kim	European	Cost Accountant	Low	Jade	193690.93		

IND66462	David Johnston	73	25663	12-09-2005	Chris Armstrong	American	Database Administrator II	Mid	Jade	111449.06
IND28766	Wayne Foster	45	35607	17-03-2018	Joshua Ryan	African	Staff Scientist	Low	Silver	115637.19
IND17397	Carlos Moore	44	19554	02-01-1996	Paul Larson	American	Programmer I	High	Jade	124434.78
IND66325	Lisa Johnston	36	33368	05-06-2020	Mark Montgomery	Asian	Software Test Engineer I	High	Platinum	260795.34
IND74197	Andrew Mills	56	27913	06-01-2021	Shawn Wallace	European	Actuary	Mid	Silver	78692.52
IND28503	Jack Coleman	61	9505	22-06-2014	Ernest Rivera	Asian	Staff Accountant II	High	Silver	257120.22
IND65539	Aaron Day	56	36232	19-01-2020	George Simmons	American	Assistant Media Planner	Low	Jade	145704.55
IND53804	Kevin Weaver	43	6299	31-03-2019	Frank Brown	American	Staff Accountant I	Low	Jade	34404.16
IND32064	Mary Fox	63	7684	09-03-2009	Adam Hernandez	Australian	Compensation Analyst	Mid	Jade	129060.98
IND72334	Carlos Little	41	38321	03-04-2020	Jonathan Hawkins	American	Geologist I	Low	Jade	129891.31
IND16101	Roger Boyd	58	12772	31-12-2015	Adam Hernandez	European	Web Designer II	High	Jade	123645.42
IND93121	Aaron Marshall	26	28661	14-01-2013	Ernest Rivera	American	Media Manager IV	Mid	Jade	308182.93
IND93119	Cheryl Stewart	21	8767	23-05-2021	Victor Martinez	European	Accounting Assistant II	High	Gold	122679.41
IND71301	Anne Nguyen	41	14954	19-09-2021	Joshua Ryan	American	Programmer II	High	Jade	30942.07
IND21279	Christopher Evans	36	24053	20-03-2013	Adam Hernandez	European	Administrative Assistant IV	High	Jade	50651.56
IND98818	Maria Clark	34	8623	29-01-2014	Joe Hanson	Asian	Help Desk Operator	High	Silver	502308.08
IND35569	Jimmy Simpson	52	38270	08-01-2018	George Lewis	European	Geological Engineer	High	Jade	87019.73
IND40198	Louise Sanders	78	29119	24-05-2005	Steve Cook	European	Junior Executive	Low	Jade	246515.84
IND49516	Angela Alvarez	34	31038	22-09-2015	Patrick Graham	Australian	Biostatistician IV	Low	Jade	130023.36
IND55475	Henry Grant	75	21778	06-04-2013	Anthony Simpson	European	Software Test Engineer I	Low	Jade	82703.81
IND61272	Larry Foster	47	34773	10-12-2020	Raymond Alexander	European	Internal Auditor	High	Gold	137256.09
IND33984	Carol GolPKRalez	34	29287	29-05-2021	Carl Nguyen	European	Chemical Engineer	High	Platinum	224640.75
IND51121	Timothy Stanley	79	29278	15-02-2021	Joshua Little	American	Accountant III	High	Jade	147292.91
IND25680	Billy Wilson	73	99	25-09-2017	Dennis Morris	European	Registered Nurse	High	Platinum	263248.97
IND65464	Kathleen Riley	80	20746	28-11-2017	Benjamin Kim	European	Cost Accountant	Low	Jade	193690.93
IND66454	Bobby Jenkins	73	15809	05-03-2014	Nicholas Cunningham	European	Developer IV	Low	Gold	91243.05
IND14252	Arthur Burns	51	30376	13-12-2005	Shawn Long	Asian	Accounting Assistant II	Low	Platinum	162607.39
IND67205	Lawrence Richardson	49	38915	13-01-2012	Anthony Simpson	European	Junior Executive	Mid	Gold	193729.94
IND50177	Julia Roberts	81	5659	08-11-2006	Victor Martinez	European	Software Engineer I	Mid	Platinum	131497.09
IND60970	Philip Fisher	27	28045	23-12-2001	Carl Nguyen	Asian	Accountant I	High	Gold	319205.91
IND24003	Jeffrey Taylor	56	16045	28-02-2020	Anthony Simpson	European	Automation Specialist IV	Mid	Silver	44089.51
IND87800	Mark McCoy	33	20816	24-01-2018	Patrick Graham	European	Biostatistician I	High	Silver	420683.86
IND75967	Timothy Cook	79	30025	11-12-2006	Chris Armstrong	European	Nurse	Low	Jade	32062.52
IND70063	Larry Green	56	12219	28-08-2016	Benjamin Kim	European	Human Resources Manager	Mid	Jade	92407.65
IND31444	Edward Gordon	55	33457	31-01-2020	Victor Martinez	European	Analyst Programmer	Mid	Gold	130572.51
IND70450	Edward Evans	77	31049	21-07-2021	Gregory Simmons	Australian	Database Administrator III	High	Gold	54753.75
IND47520	Joseph Morrison	49	3444	16-03-2020	George Lewis	Australian	Web Designer III	Mid	Silver	180214.79

ND87800	Mark McCoy	33	20816	24-01-2008	Patrick Graham	European	Biostatistician I	High	Silver	42063.66
ND75987	Timothy Cook	79	30025	11-12-2006	Chris Armstrong	European	Nurse	Low	Jade	32062.52
ND70003	Larry Green	56	12219	28-08-2016	Benjamin Kim	European	Human Resources Manager	Mid	Jade	92487.65
ND31444	Edward Gordon	55	32457	31-01-2020	Victor Martinez	European	Analyst Programmer	Mid	Gold	130572.51
ND70450	Edward Evans	77	31043	21-07-2021	Gregory Simmons	Australian	Database Administrator III	High	Gold	64753.76
ND47520	Joseph Morrison	49	3444	16-03-2020	George Lewis	Australian	Web Designer III	Mid	Silver	180214.79
ND69076	Kathy Wheeler	67	3711	12-11-1996	Anthony Tones	European	Senior Editor	High	Jade	52026.78
ND53190	Amy Sims	66	30998	02-01-2007	Joshua Bennett	European	Health Coach III	Low	Gold	43622.88
ND86356	Sharon Warren	64	3113	09-01-2018	Phillip Peters	African	Web Developer I	Mid	Silver	142641.26
ND33891	Nicholas Gilbert	49	30551	07-04-2009	Douglas Tucker	African	Systems Administrator III	High	Silver	90437.66
ND20399	Melissa Miller	65	16593	25-09-2008	Jason Duncan	Australian	Biostatistician III	High	Jade	32301.47
ND54314	Kevin Gomez	32	24168	01-01-2007	Jerry Green	European	Project Manager	Mid	Jade	293419.07
ND74308	Ronald Larson	29	25730	11-12-1998	Bobby Burton	Australian	Software Engineer III	High	Gold	55011.84
ND37195	Christine Williams	31	10528	16-04-2015	Joshua Bennett	European	Engineer III	High	Jade	77971.46
ND52609	Angela Roberts	22	30762	05-07-2005	Nicholas Cunningham	European	Health Coach I	High	Jade	84111.51
ND26283	Joshua Hughes	58	95	21-04-2006	Shawn Cook	European	Human Resources Assistant I	High	Jade	98369.29
ND61913	Joshua Stephens	76	10119	30-08-2012	Samuel Fowler	American	Software Test Engineer II	High	Silver	243490.73
ND35876	Evelyn Ramirez	18	28105	27-09-2006	Frank Brown	American	Web Designer II	Mid	Jade	72431.34
ND19011	Lillian Burns	47	32371	07-11-2003	Paul Larson	European	Sales Associate	Mid	Silver	456003.99
ND86703	Lillian Bell	41	11172	16-02-2018	Stephen Payne	European	Biostatistician I	High	Platinum	71422.04
ND23640	Philip Perez	36	16659	02-04-2018	Benjamin Kim	European	Accounting Assistant III	Low	Jade	39077.38
ND93407	Karen Morrison	84	18123	16-04-2010	Carl Nguyen	European	Dental Hygienist	High	Jade	274099.12
ND40136	John Cruz	26	27140	25-09-2019	Victor Martinez	European	Developer IV	Mid	Silver	260826.75
ND76215	Joseph Thomas	74	18521	07-01-1999	Jeremy Vasquez	American	Engineer IV	Low	Platinum	334331.81

All Time	Last 30 D	Last 90 D	Last 3 M	Last 6 M	Last 12 M	Last 24 M	CM	CQ	CY
Gender									
Female					Male				
Total Clients					Total Loan				
					\$4.38bn				
					Total Fees				
					\$158.19M				
Total Deposit					Total CC Amount				
					\$3.77bn				
					Saving Account Amount				
					\$698.73M				
					\$4.39K				

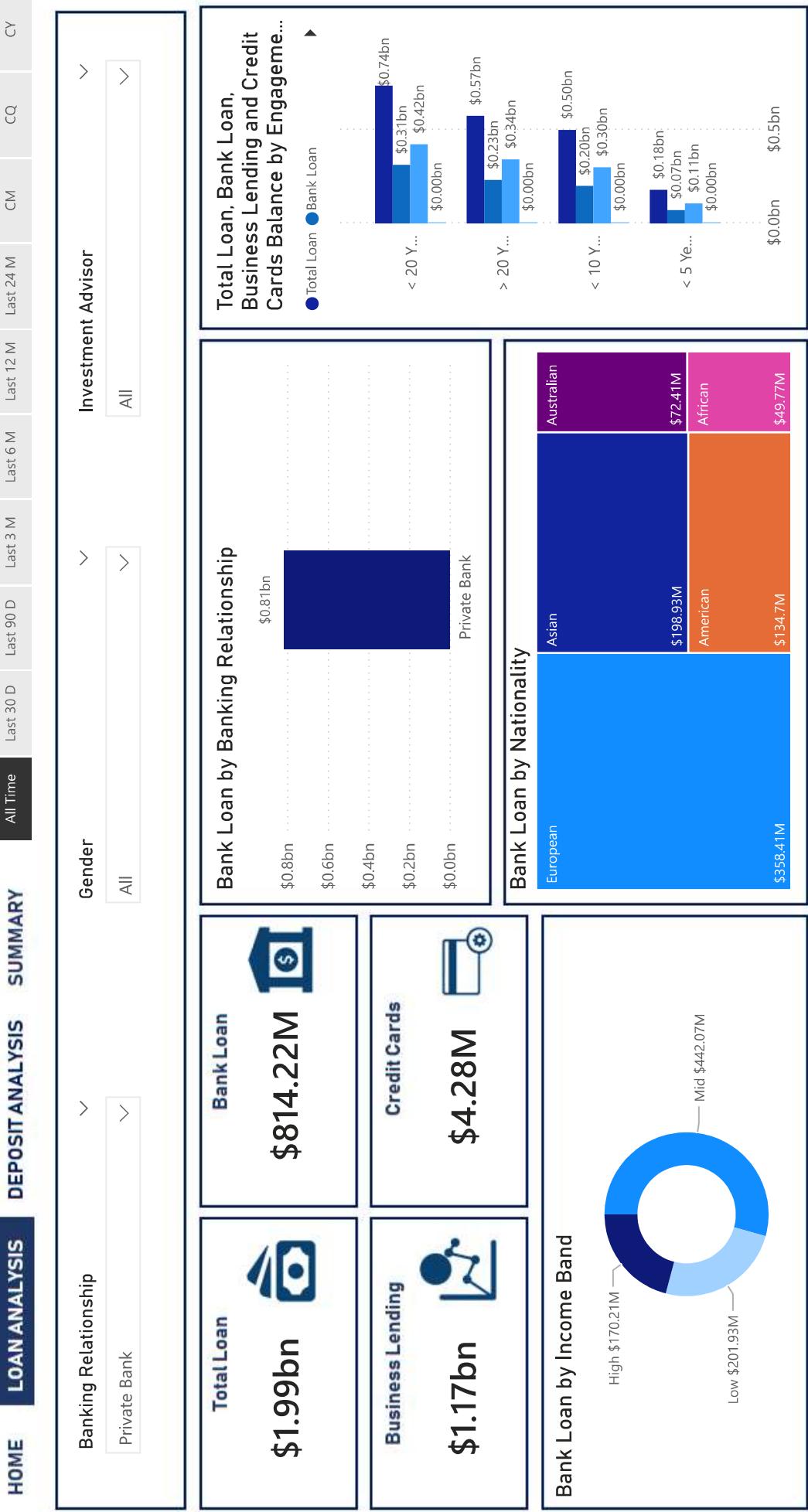


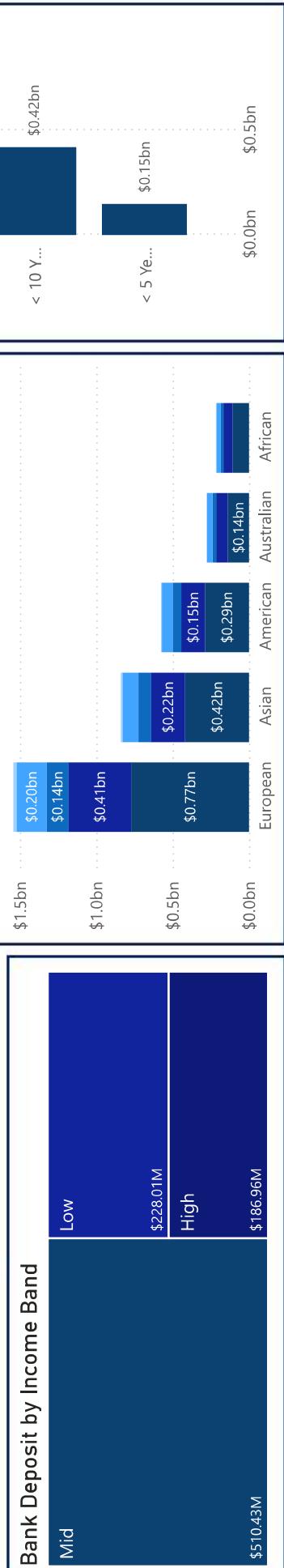
BANKING DASHBOARD

LOAN DEPOSIT

DEPOSIT ANALYSIS

SUMMARY





HOME **LOAN ANALYSIS** **DEPOSIT ANALYSIS** **SUMMARY**

Last 30 D Last 90 D Last 3 M Last 6 M Last 12 M Last 24 M CM CQ CY

Banking Relationship  Private Bank 

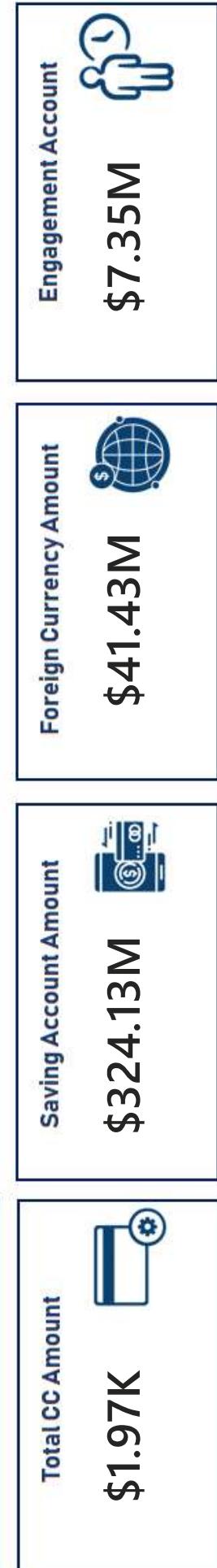
All Time  Gender 

All 

All 

All 

Banking Relationship  Private Bank 



Joining Year	2013	2018	2019	2020	2021
Gender	Female				
Male					
Total Clients	3000				
Total Loan	4.38bn				
Total Deposit	3.77bn				
Checking Accounts	963.28M				
Savings Accounts	698.73M				
Business Lending	2.60bn				



BANKING DASHBOARD

LOAN DEPOSIT **DEPOSIT ANALYSIS**

SUMMARY

HOME

LOAN ANALYSIS **DEPOSIT ANALYSIS** **SUMMARY**

Joining Year	2013	2018	2019	2020	2021
Gender					

Female					Male
--------	--	--	--	--	------

Total Clients	3000	
Total Loan	4.38bn	

Total Deposit	3.77bn	
Checking Accounts	963.28M	

Business Lending	2.60bn	
Savings Accounts	698.73M	



BANKING DASHBOARD

LOAN ANALYSIS **DEPOSIT ANALYSIS** **SUMMARY**



