Bank Customer Churn Analysis

The objective of this analysis is to investigate and gain insights into the major factors impacting the bank's customer churn rate. Subsequently, the study aims to provide recommendations to the bank, thereby facilitating a reduction in churn rate and improving customer retention.

In [2]:
```python
#Importing libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
```

In [3]:
```python
#Importing the dataset
bank_data = pd.read_csv("Customer_Churn_Records.csv")
bank_data.head()
```

Out[3]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsA
0	1	15634602	Hargrave	619	France	Female	42	2	0.00	1	1	
1	2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	
2	3	15619304	Onio	502	France	Female	42	8	159660.80	3	1	
3	4	15701354	Boni	699	France	Female	39	1	0.00	2	0	
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	

In [4]:
```python
bank_data.tail()
```

Out[4]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard
9995	9996	15606229	Obijiaku	771	France	Male	39	5	0.00	2	1
9996	9997	15569892	Johnstone	516	France	Male	35	10	57369.61	1	1
9997	9998	15584532	Liu	709	France	Female	36	7	0.00	1	0
9998	9999	15682355	Sabbatini	772	Germany	Male	42	3	75075.31	2	1
9999	10000	15628319	Walker	792	France	Female	28	4	130142.79	1	1

In [5]:
```python
bank_data.shape
```

Out[5]:
```
(10000, 18)
```

In [6]:
```python
bank_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 18 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   RowNumber           10000 non-null  int64  
 1   CustomerId          10000 non-null  int64  
 2   Surname             10000 non-null  object 
 3   CreditScore         10000 non-null  int64  
 4   Geography           10000 non-null  object 
 5   Gender              10000 non-null  object 
 6   Age                 10000 non-null  int64  
 7   Tenure              10000 non-null  int64  
 8   Balance             10000 non-null  float64
 9   NumOfProducts       10000 non-null  int64  
 10  HasCrCard           10000 non-null  int64  
 11  IsActiveMember      10000 non-null  int64  
 12  EstimatedSalary     10000 non-null  float64
 13  Exited              10000 non-null  int64  
 14  Complain            10000 non-null  int64  
 15  Satisfaction Score  10000 non-null  int64  
 16  Card Type           10000 non-null  object 
 17  Point Earned        10000 non-null  int64  
dtypes: float64(2), int64(12), object(4)
memory usage: 1.4+ MB
```

In [7]:
```python
bank_data.describe()
```

Out[7]:

	RowNumber	CustomerId	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiv
count	10000.00000	1.000000e+04	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	10000.00000	1000
mean	5000.50000	1.569094e+07	650.528800	38.921800	5.012800	76485.889288	1.530200	0.70550	
std	2886.89568	7.193619e+04	96.653299	10.487806	2.892174	62397.405202	0.581654	0.45584	
min	1.00000	1.556570e+07	350.000000	18.000000	0.000000	0.000000	1.000000	0.00000	
25%	2500.75000	1.562853e+07	584.000000	32.000000	3.000000	0.000000	1.000000	0.00000	
50%	5000.50000	1.569074e+07	652.000000	37.000000	5.000000	97198.540000	1.000000	1.00000	
75%	7500.25000	1.575323e+07	718.000000	44.000000	7.000000	127644.240000	2.000000	1.00000	
max	10000.00000	1.581569e+07	850.000000	92.000000	10.000000	250898.090000	4.000000	1.00000	

In [8]:

```
bank_data.isnull().sum()
```

Out[8]:

```
RowNumber             0
CustomerId            0
Surname               0
CreditScore           0
Geography             0
Gender                0
Age                   0
Tenure                0
Balance               0
NumOfProducts         0
HasCrCard             0
IsActiveMember        0
EstimatedSalary       0
Exited                0
Complain              0
Satisfaction Score    0
Card Type             0
Point Earned          0
dtype: int64
```

In [9]:

```
bank_data.duplicated().sum()
```

Out[9]:

```
0
```

There are no duplicates or null values.

In [10]:

```
#Dropping columns that are irrelavant for the analysis
df = bank_data.drop(columns={'RowNumber', 'Surname'})
df.head()
```

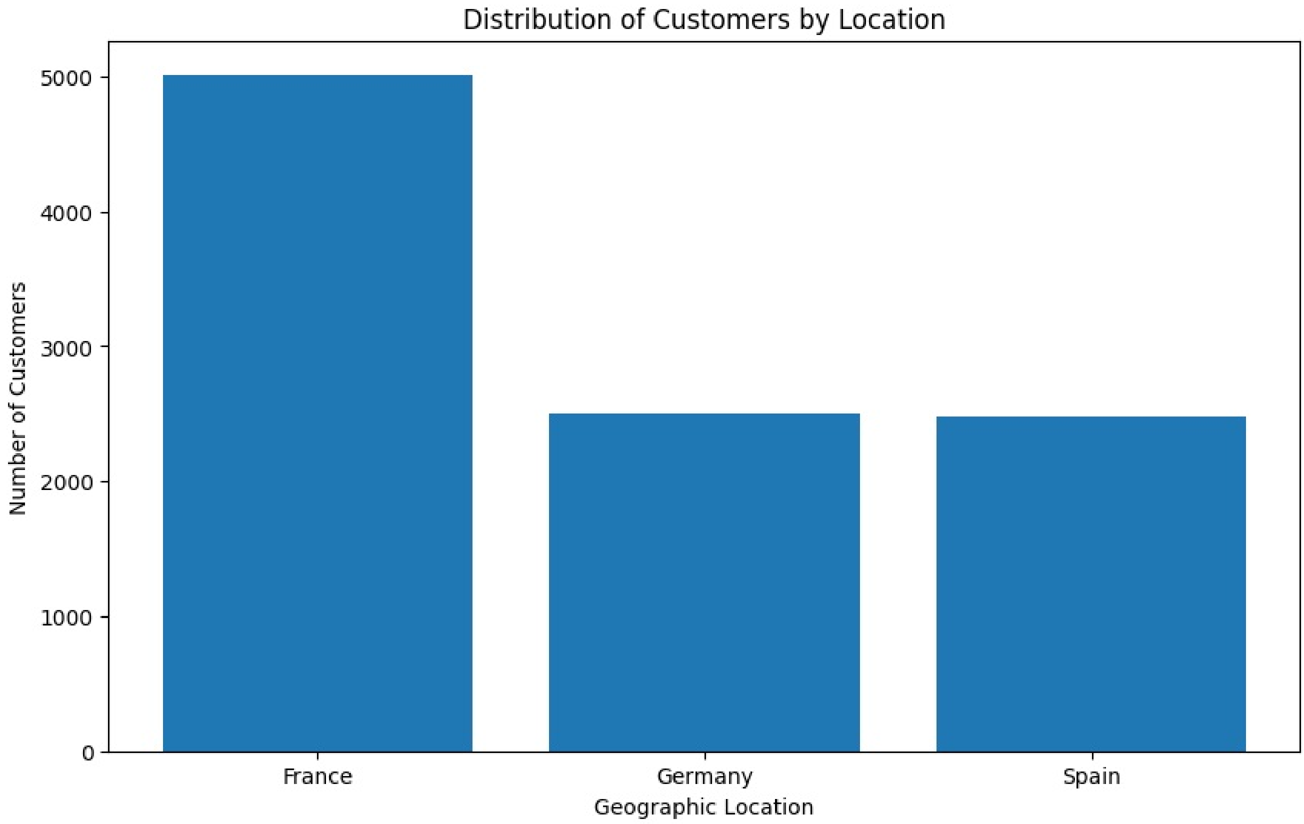
Out[10]:

	CustomerId	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	Estimated
0	15634602	619	France	Female	42	2	0.00	1	1	1	101
1	15647311	608	Spain	Female	41	1	83807.86	1	0	1	112
2	15619304	502	France	Female	42	8	159660.80	3	1	0	113
3	15701354	699	France	Female	39	1	0.00	2	0	0	93
4	15737888	850	Spain	Female	43	2	125510.82	1	1	1	79

Performing Exploratory Data Analysis

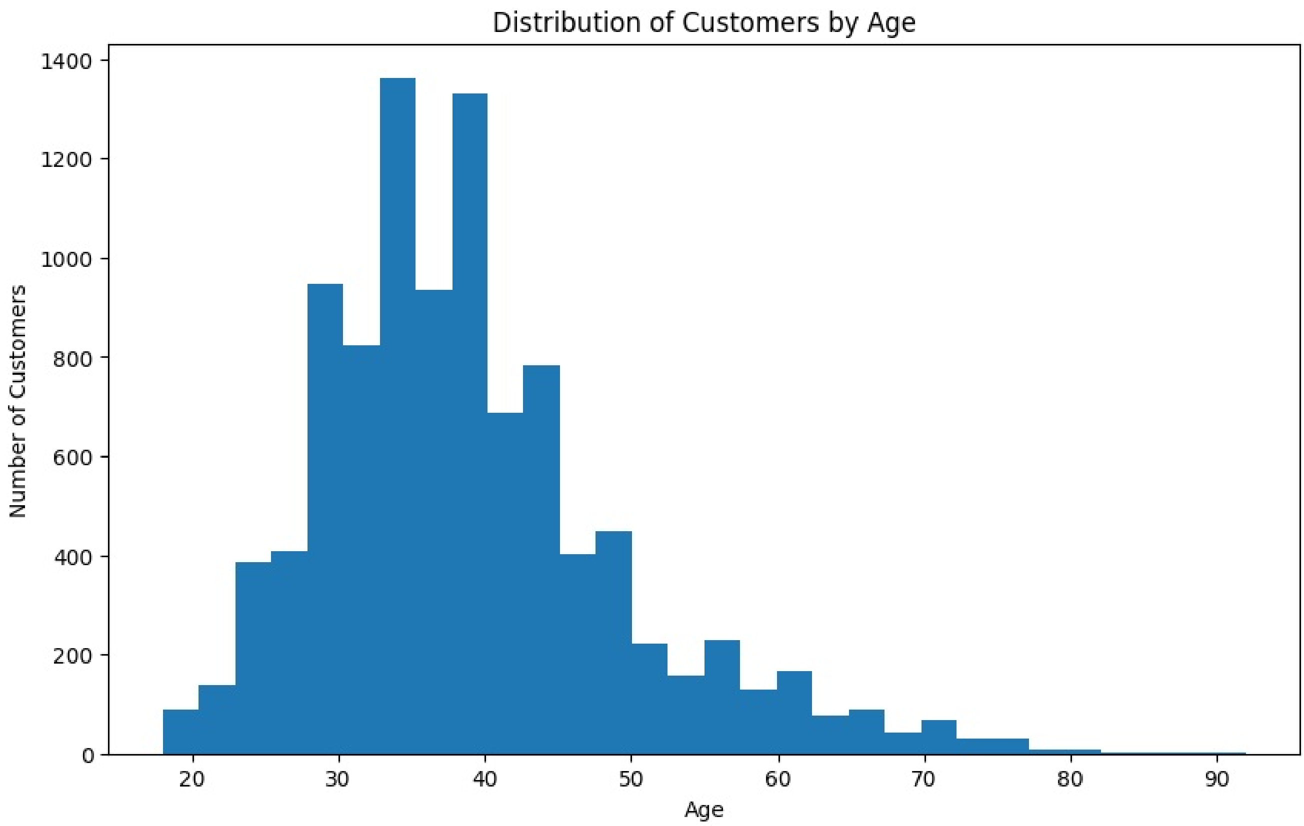
In [11]:

```
# Plotting a bar chart for the geography attribute
plt.figure(figsize =(10,6))
geography_counts = df['Geography'].value_counts()
plt.bar(geography_counts.index, geography_counts.values)
plt.xlabel('Geographic Location')
plt.ylabel('Number of Customers')
plt.title('Distribution of Customers by Location')
plt.show()
```



Majority of the bank customers are from France.

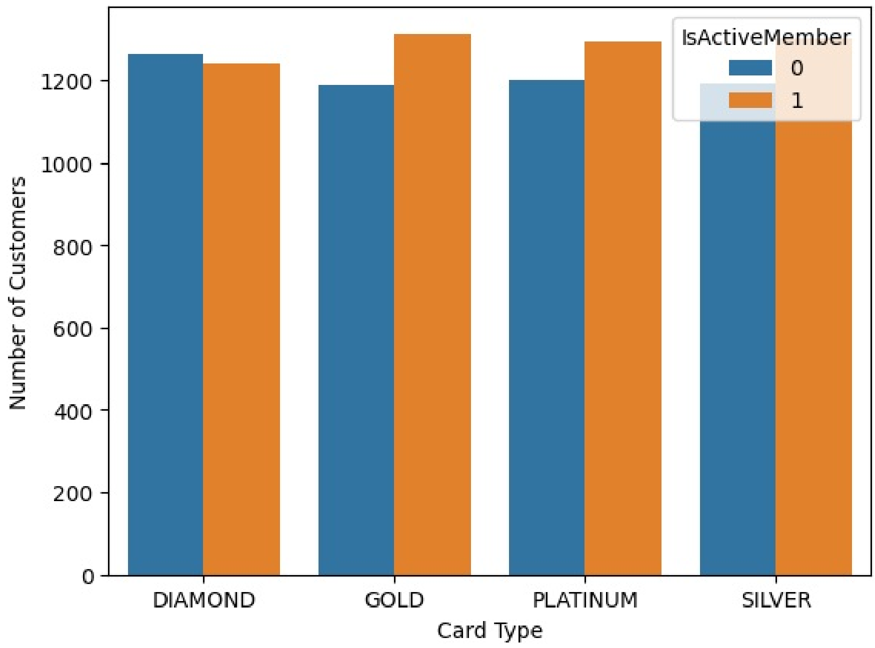
In [12]:
```
# Plotting a histogram for the age distribution
plt.figure(figsize =(10,6))
plt.hist(df['Age'], bins=30)
plt.xlabel('Age')
plt.ylabel('Number of Customers')
plt.title('Distribution of Customers by Age')
plt.show()
```



Majority of the customers are between the ages 24-50. This is valid as people above the age of 24 start working and hence need a bank account.

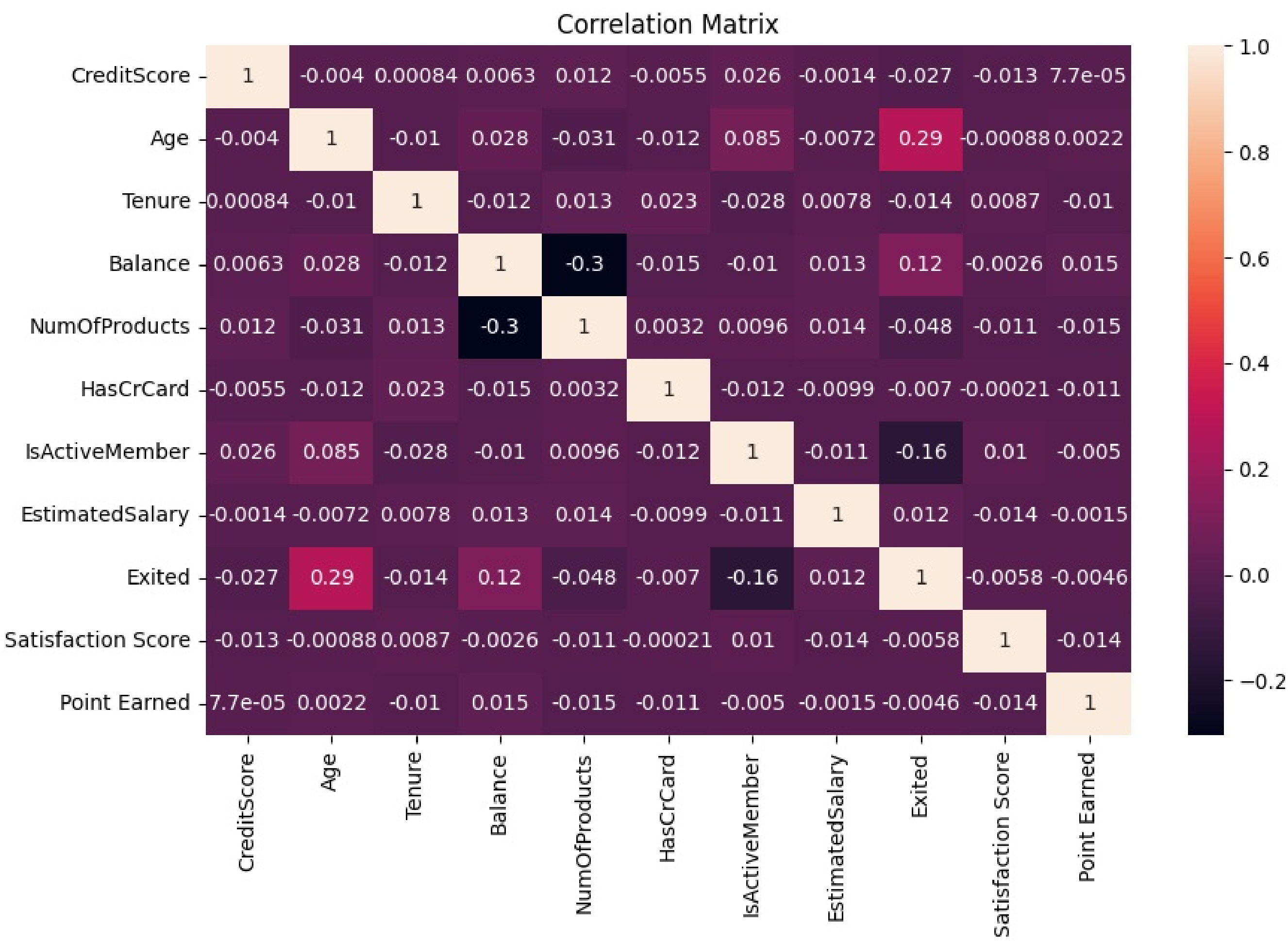
In [13]:
```
# Plotting a bar chart for card type and active status
active_card_member = df.groupby(['Card Type', 'IsActiveMember']).size().reset_index(name='Number of Customers')
sns.barplot(data= active_card_member, x= 'Card Type', y = 'Number of Customers', hue= 'IsActiveMember')
```

Out[13]: <Axes: xlabel='Card Type', ylabel='Number of Customers'>



This shows that only in the diamond category, there are more inactive members than active ones. Except for this, most members are active in other categories. This may be because the benefits offered in this card type are not what the customers are looking for.

In [14]:
```python
num_attributes = df[['CreditScore', 'Age', 'Tenure', 'Balance', 'NumOfProducts', 'HasCrCard', 'IsActiveMember',
correlation_matrix = num_attributes.corr()

#Plotting correlation matrix
plt.figure(figsize =(10,6))
sns.heatmap(correlation_matrix, annot=True)
plt.title('Correlation Matrix')
plt.show()
```



The correlation matrix shows that there is not a good linear relationship between attributes. However, there is a stronger correlation between the churn rate and age as well as churn rate and account balance which will be further explored.

- This could mean that there might be a combination of factors that are leading to the customer churn.

Analyzing Customer Churn Rate

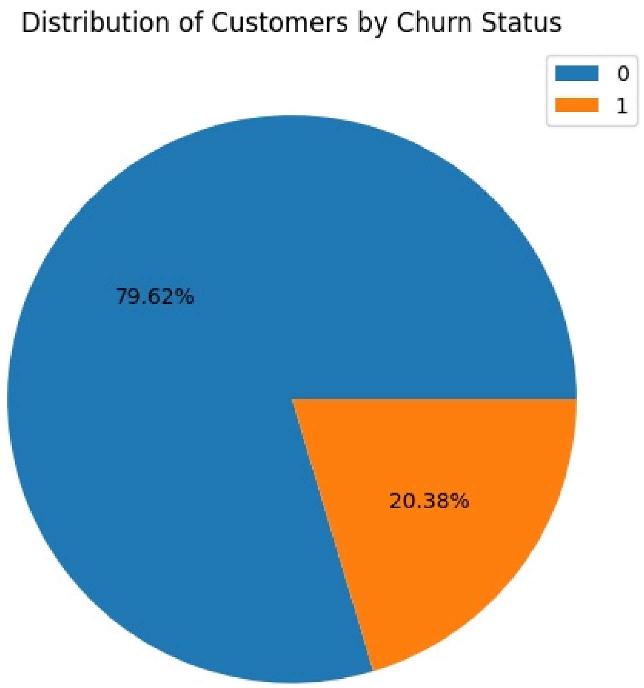
In [15]:
```
#Replacing 'Yes' and 'No' with 0 and 1's
df.Exited[df.Exited == 'No'] = 0
df.Exited[df.Exited == 'Yes'] = 1
df.head()
```

```
C:\Users\skash\AppData\Local\Temp\ipykernel_22104\653134756.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#retu
rning-a-view-versus-a-copy
  df.Exited[df.Exited == 'No'] = 0
C:\Users\skash\AppData\Local\Temp\ipykernel_22104\653134756.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#retu
rning-a-view-versus-a-copy
  df.Exited[df.Exited == 'Yes'] = 1
```

Out[15]:

	CustomerId	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	Estimated
0	15634602	619	France	Female	42	2	0.00	1	1	1	101
1	15647311	608	Spain	Female	41	1	83807.86	1	0	1	112
2	15619304	502	France	Female	42	8	159660.80	3	1	0	113
3	15701354	699	France	Female	39	1	0.00	2	0	0	93
4	15737888	850	Spain	Female	43	2	125510.82	1	1	1	79

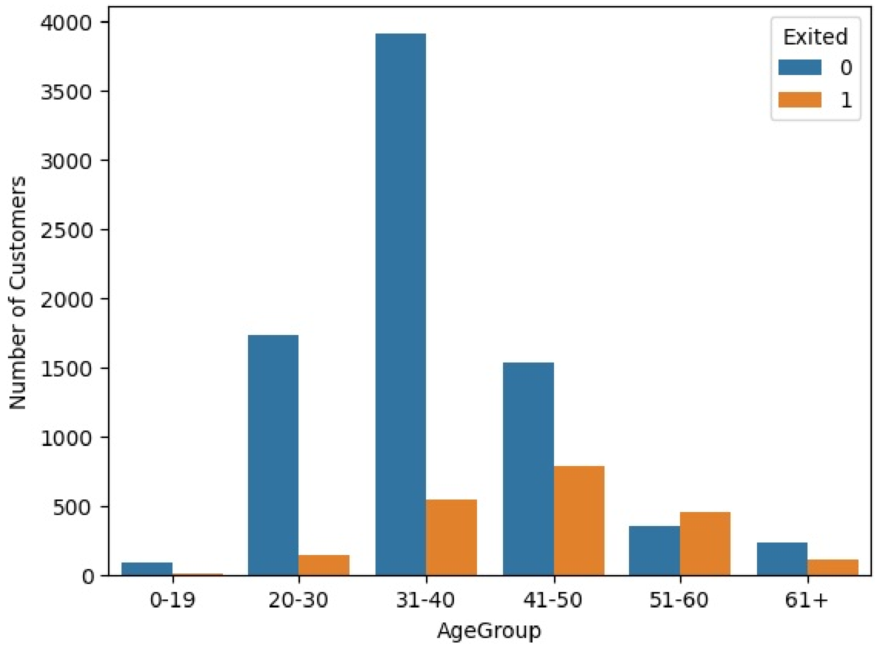
In [16]:
```
# Plotting the distribution of customers by churn status
plt.figure(figsize =(10,6))
churn_counts = df['Exited'].value_counts()
plt.pie(churn_counts.values, autopct='%1.2f%%')
plt.legend( loc = 'upper right', labels=churn_counts.index)
plt.title('Distribution of Customers by Churn Status')
plt.show()
```



This shows that 20% of the customers churned out. This could be due to various reasons that will be explored in the following analysis.

In [17]:
```
# Plotting a bar chart for the age attribute
# Creating age groups
age_bins = [0, 20, 30, 40, 50, 60, 70]
age_labels = ['0-19', '20-30', '31-40', '41-50', '51-60', '61+']
df['AgeGroup'] = pd.cut(df['Age'], bins=age_bins, labels=age_labels)

customer_count_age_exitstatus = df.groupby(['AgeGroup', 'Exited']).size().reset_index(name='Number of Customers
sns.barplot(data= customer_count_age_exitstatus, x= 'AgeGroup', y = 'Number of Customers', hue= 'Exited')
```

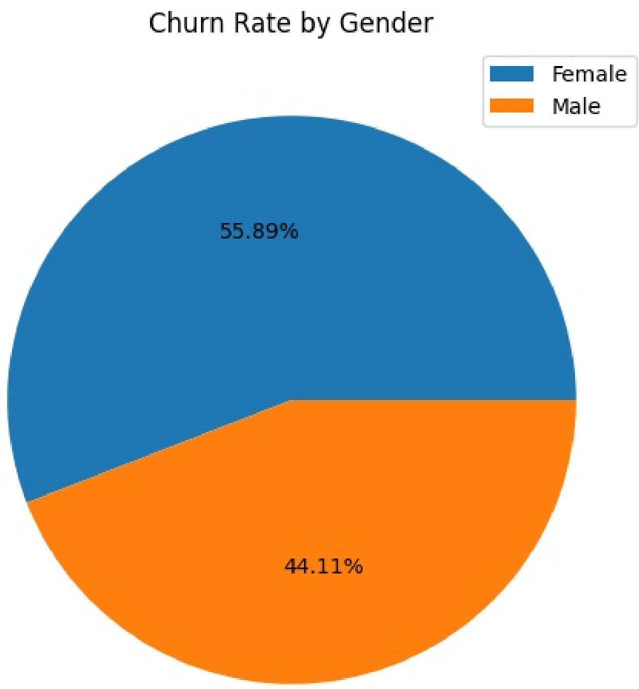
Out[17]: <Axes: xlabel='AgeGroup', ylabel='Number of Customers'>



Most of the customers that churned out are between the ages of 31 to 60 with the mojority being in the age group of 41-50. This could be due to various reasons such as:

- Changes in banking needs: Most people in this age group may be relocating, buying homes, have changes in financial needs and goals and hence may need different services than the current ones offered.
- Better offers in other banks: They may be attracted by other bank offers/promotions and services that may be more suitable to their needs now that they have higher incomes and more savings.
- Unsatisfactory services: If the customers feel dissatisfied due to issues like poor customer service, inconvenient branch locations, irrelevant benefits or poor quality products/services, customers are more likely to switch banks.

In [18]:
```
# Plotting the churn rate by gender
plt.figure(figsize =(10,6))
churn_rate_gender = df.groupby('Gender')['Exited'].sum()
plt.pie(churn_rate_gender.values, autopct='%1.2f%%')
plt.legend( loc = 'upper right', labels=churn_rate_gender.index)
plt.title('Churn Rate by Gender')
plt.show()
```

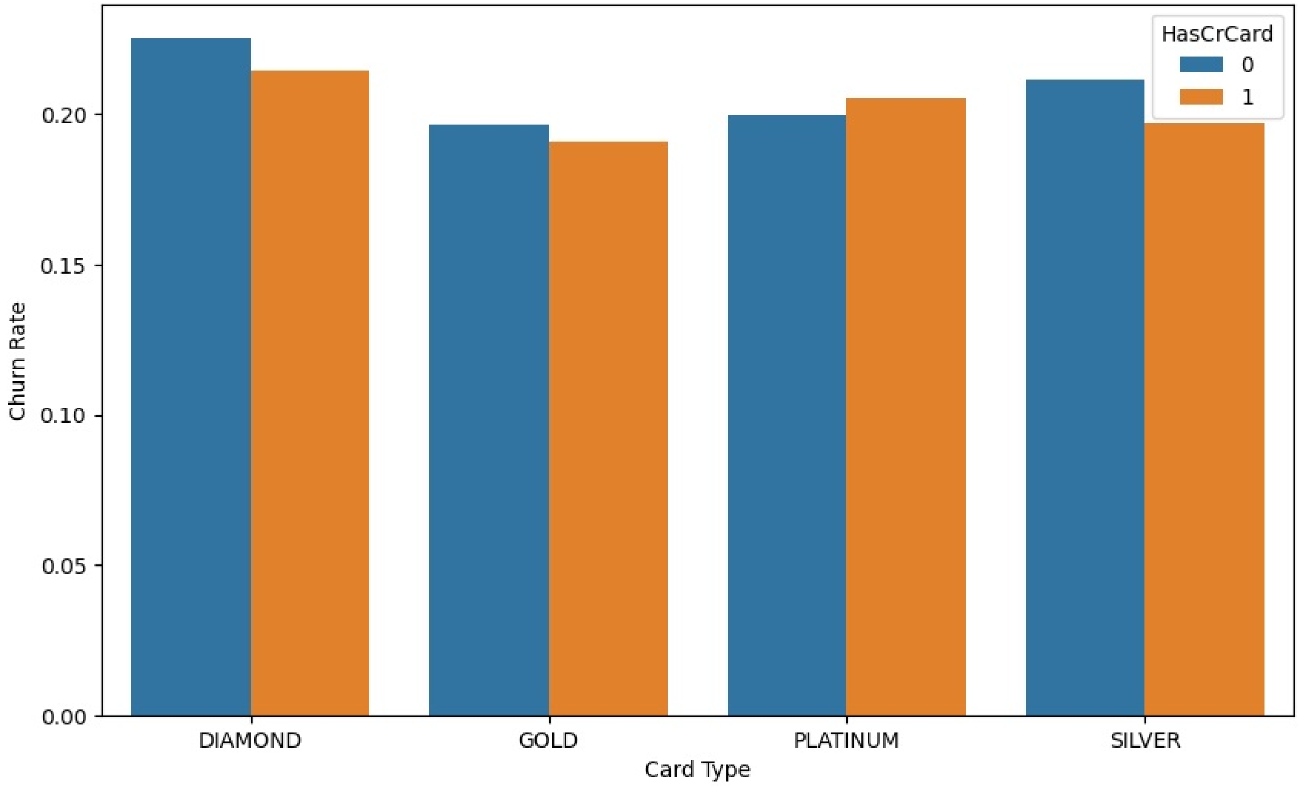


More than 50% of the churned customers are females.

- This could be due to various factors such as life transitions, inadequate trust and security offered by the bank, or dissatisfactory customer service experience.

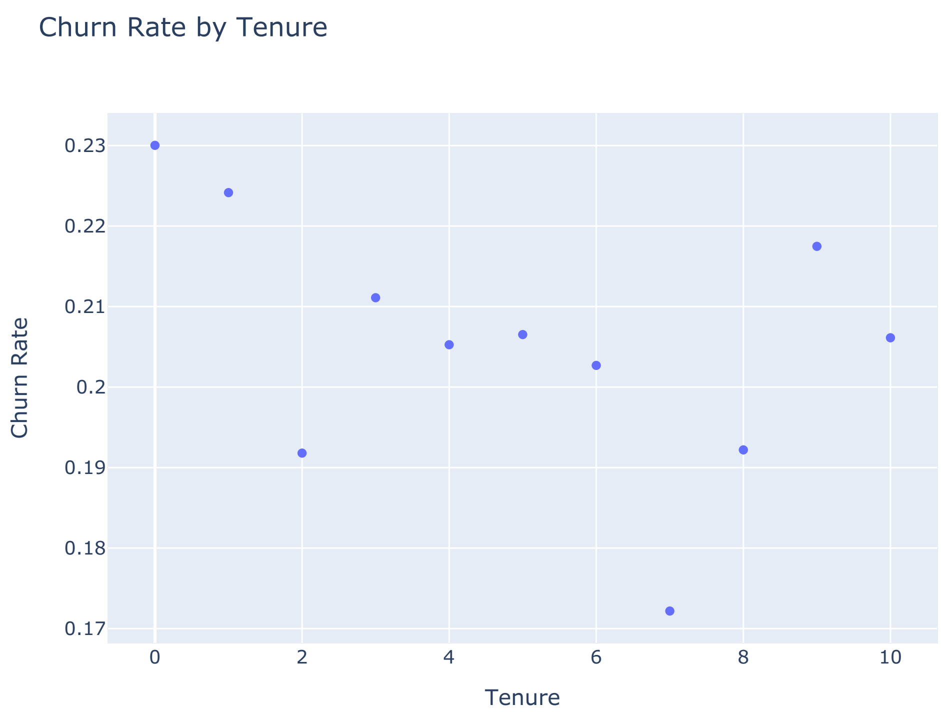
In [19]:
```
# Plotting the churn rate by card type
plt.figure(figsize =(10,6))
churn_rate_cardtype = df.groupby(['Card Type','HasCrCard'])['Exited'].mean().reset_index(name='Churn Rate')
sns.barplot(data= churn_rate_cardtype, x= 'Card Type', y = 'Churn Rate', hue= 'HasCrCard')
```

Out[19]: <Axes: xlabel='Card Type', ylabel='Churn Rate'>



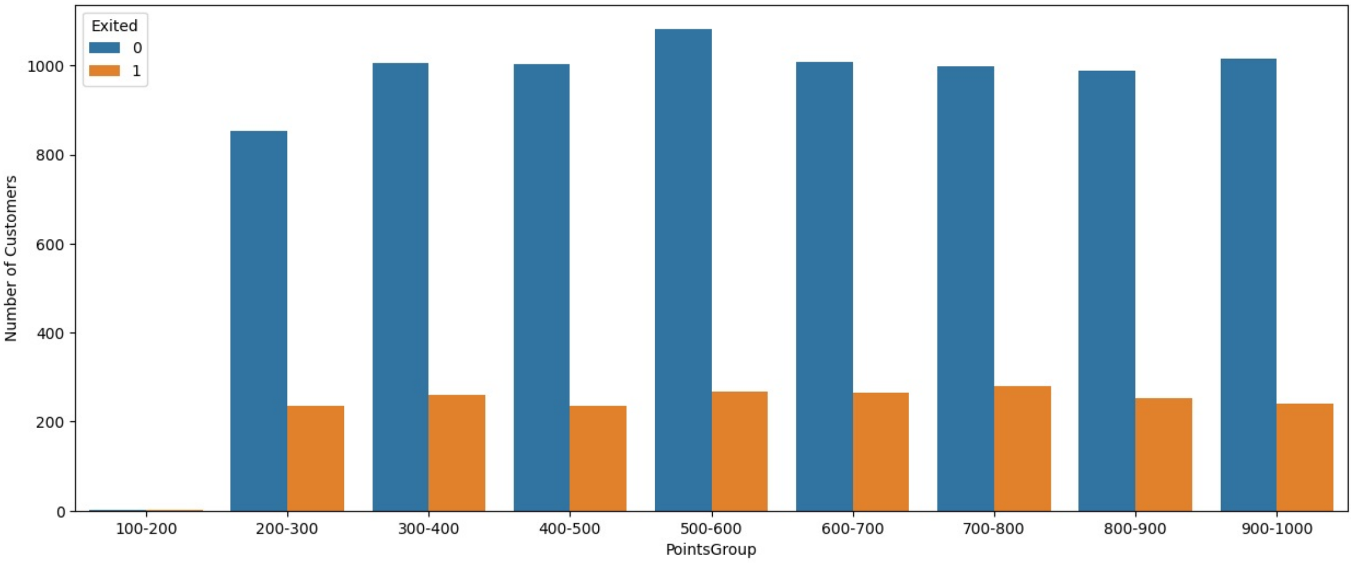
We can see that the Diamond Card type has the highest churn rate of just over 20% and the churn rate is higher for customers who do not have credit cards. This could be due to various reasons duch as high membership fees, benefits that may not be very attractive to customers, etc.

In [20]:
```
#Plotting churn rate by tenure
churn_rate_tenure = df.groupby(['Tenure'])['Exited'].mean().reset_index(name='Churn Rate')
fig = px.scatter(churn_rate_tenure, x='Tenure', y='Churn Rate', title='Churn Rate by Tenure')
fig.update_layout(xaxis_title='Tenure', yaxis_title='Churn Rate')
fig.show()
```



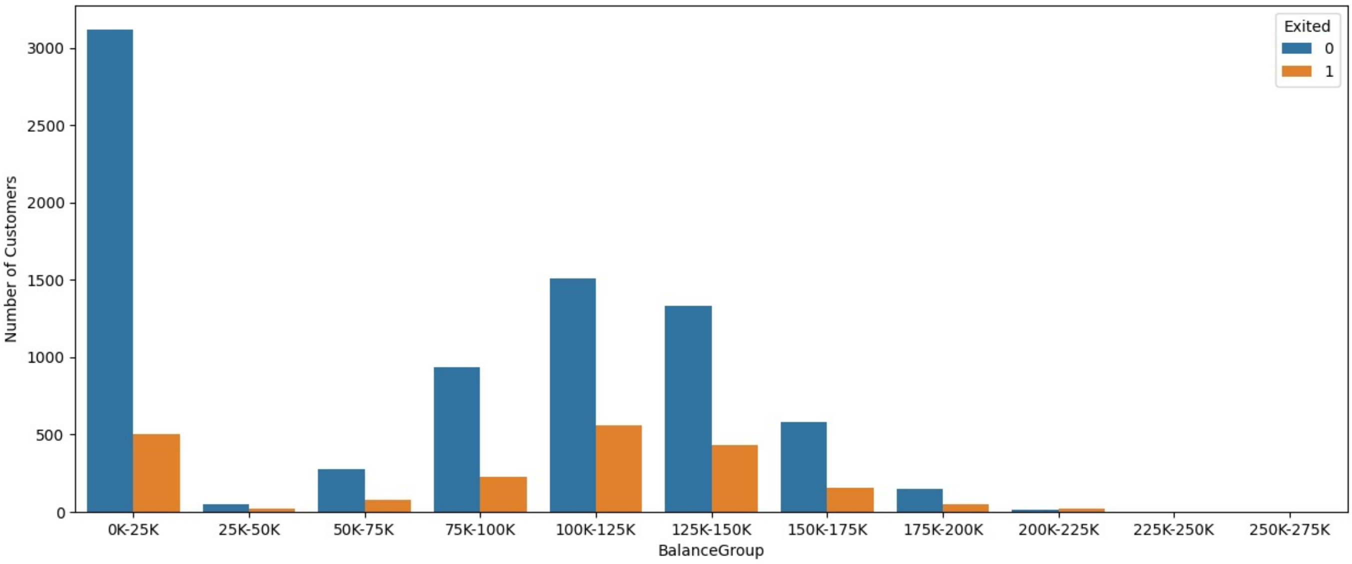
There is no significant relationship between the tenure and churn rate. Hence, this proves that tenure is not the reason for the customer churn. However, retaining existing customers is always beneficial.

In [21]:
```
# Plotting a bar chart for the points attribute
plt.figure(figsize=(15, 6))

# Creating points groups
points_bins = [100, 200, 300, 400, 500, 600, 700, 800, 900, 1001]
points_labels = ['100-200', '200-300', '300-400', '400-500', '500-600', '600-700', '700-800', '800-900', '900-1
df['PointsGroup'] = pd.cut(df['Point Earned'], bins=points_bins, labels=points_labels)

customer_count_points = df.groupby(['PointsGroup', 'Exited']).size().reset_index(name='Number of Customers')
sns.barplot(data= customer_count_points, x= 'PointsGroup', y = 'Number of Customers', hue= 'Exited')

plt.show()
```



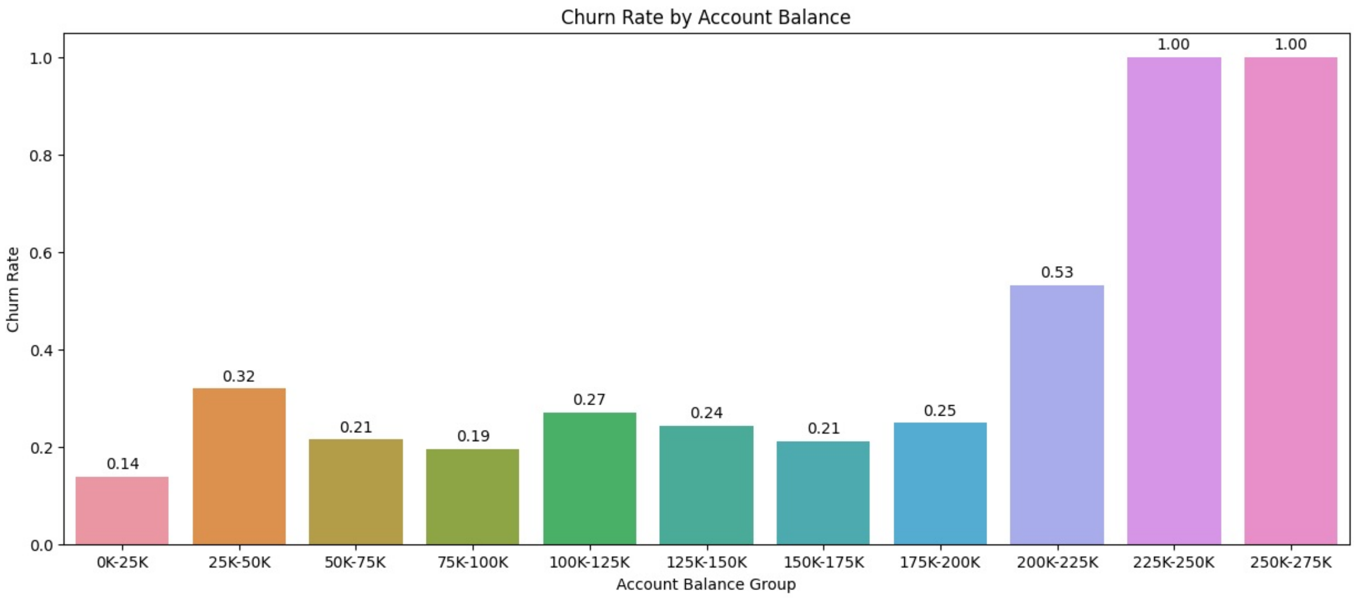
The analysis shows that there is no significant difference in the churn rate with respect to the points earned.

In [22]:
```
#Plotting churn rate by account balance
# Creating balance groups
plt.figure(figsize=(15, 6))
balance_bins = [-1, 25000,  50000, 75000, 100000, 125000, 150000, 175000, 200000, 225000, 250000, 275000]
balance_bins.sort()
balance_labels = ['0K-25K', '25K-50K', '50K-75K', '75K-100K', '100K-125K', '125K-150K', '150K-175K', '175K-200K
df['BalanceGroup'] = pd.cut(df['Balance'], bins=balance_bins, labels=balance_labels)

churn_rate_balance = df.groupby(['BalanceGroup', 'Exited']).size().reset_index(name='Number of Customers')
sns.barplot(data= churn_rate_balance, x= 'BalanceGroup', y = 'Number of Customers', hue= 'Exited')
plt.show()
```



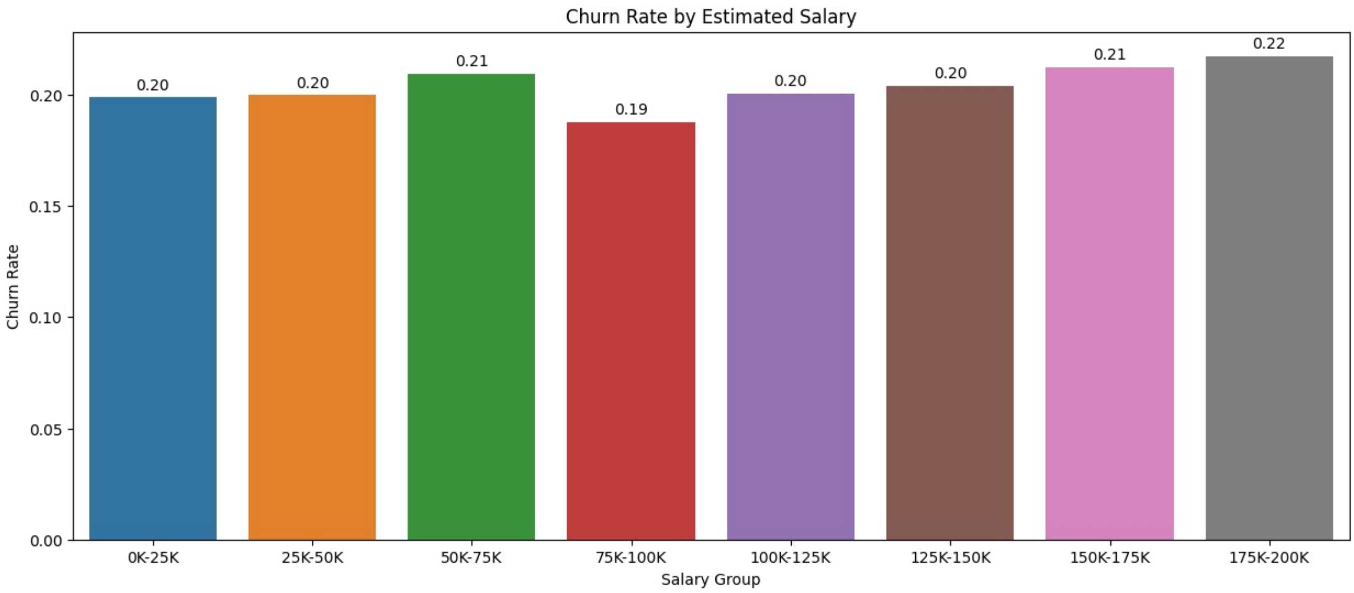
In [23]:
```
#Plotting churn rate by account balance
# Creating balance groups
plt.figure(figsize=(15, 6))
balance_bins = [-1, 25000,  50000, 75000, 100000, 125000, 150000, 175000, 200000, 225000, 250000, 275000]
balance_bins.sort()
balance_labels = ['0K-25K', '25K-50K', '50K-75K', '75K-100K', '100K-125K', '125K-150K', '150K-175K', '175K-200K
df['BalanceGroup'] = pd.cut(df['Balance'], bins=balance_bins, labels=balance_labels)
```

```
churn_rate_balance = df.groupby(['BalanceGroup'])['Exited'].mean().reset_index(name='Churn Rate')

plots = sns.barplot(data= churn_rate_balance, x= 'BalanceGroup', y = 'Churn Rate')

# Iterating over the bars one-by-one
for bar in plots.patches:
    plots.annotate(format(bar.get_height(), '.2f'),
                   (bar.get_x() + bar.get_width() / 2,
                    bar.get_height()), ha='center', va='center', xytext=(0, 8),
                   textcoords='offset points')

# Setting the label for x-axis
plt.xlabel("Account Balance Group")

# Setting the label for y-axis
plt.ylabel("Churn Rate")

# Setting the title for the graph
plt.title("Churn Rate by Account Balance")

# Finally showing the plot
plt.show()
```



There is no significant correlation between the account balance and churn rate.

In [24]:
```
#Plotting churn rate by Salary
# Creating salary groups
plt.figure(figsize=(15, 6))
salary_bins = [-1, 25000,  50000, 75000, 100000, 125000, 150000, 175000, 200000]
salary_bins.sort()
salary_labels = ['0K-25K', '25K-50K', '50K-75K', '75K-100K', '100K-125K', '125K-150K', '150K-175K', '175K-200K'
df['SalaryGroup'] = pd.cut(df['EstimatedSalary'], bins=salary_bins, labels=salary_labels)

churn_rate_salary = df.groupby(['SalaryGroup'])['Exited'].mean().reset_index(name='Churn Rate')

plots = sns.barplot(data= churn_rate_salary, x= 'SalaryGroup', y = 'Churn Rate')

# Iterating over the bars one-by-one
for bar in plots.patches:
    plots.annotate(format(bar.get_height(), '.2f'),
                   (bar.get_x() + bar.get_width() / 2,
                    bar.get_height()), ha='center', va='center', xytext=(0, 8),
                   textcoords='offset points')

# Setting the label for x-axis
plt.xlabel("Salary Group")

# Setting the label for y-axis
plt.ylabel("Churn Rate")

# Setting the title for the graph
plt.title("Churn Rate by Estimated Salary")

# Finally showing the plot
plt.show()
```

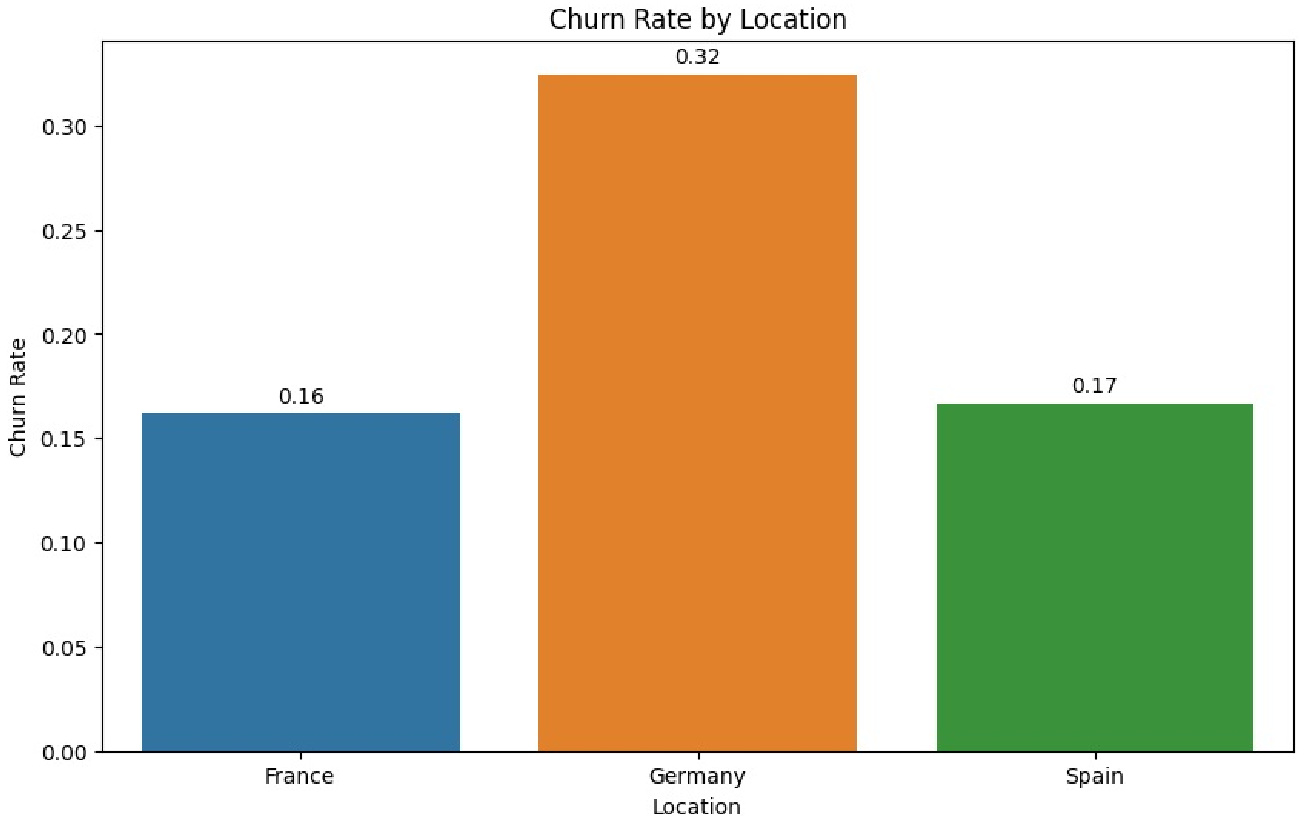


From the plot, it is clear that there isn't a significant correlation between estimated salary and churn rate. However, there is a marginal upward trend in the churn rate as the salaries increase. In order to gain a more comprehensive understanding of this relationship, it is imperative for the bank to conduct further investigations by speaking to the customers.

In [25]:
```
#Exporting updated dataframe to Excel file
import pandas as pd

# Specifying the filename for the Excel file
output_file = 'Updated_Customer_Churn_Records.xlsx'

# Exporting the DataFrame to Excel
df.to_excel(output_file, index=False)

print(f"DataFrame successfully exported to {output_file}.")
```

```
DataFrame successfully exported to Updated_Customer_Churn_Records.xlsx.
```

In [26]:
```
#Plotting churn rate by Location
plt.figure(figsize =(10,6))
churn_rate_location = df.groupby(['Geography'])['Exited'].mean().reset_index(name='Churn Rate')

plots = sns.barplot(data= churn_rate_location, x= 'Geography', y = 'Churn Rate')

# Iterating over the bars one-by-one
for bar in plots.patches:
    plots.annotate(format(bar.get_height(), '.2f'),
                   (bar.get_x() + bar.get_width() / 2,
                    bar.get_height()), ha='center', va='center', xytext=(0, 8),
                   textcoords='offset points')

# Setting the label for x-axis
plt.xlabel("Location")

# Setting the label for y-axis
plt.ylabel("Churn Rate")

# Setting the title for the graph
plt.title("Churn Rate by Location")

# Finally showing the plot
plt.show()
```



This shows that Germany has the highest churn rate. 32% of its customers have churned out. This could be due to the market condition in Germany or customer dissatisfaction.

Recommendations for the Bank:

- **Focus on retaining female customers:** Determine the factors contributing to the escalated churn rate observed among female customers, and devise effective solutions and customized marketing strategies to cater to their specific requirements.
- **Strengthen communication and informational programs:** Implement supplementary communication channels and educational initiatives aimed at enhancing customer knowledge regarding the comprehensive range of products and services provided. By fostering transparency and building trust, the bank can nurture customer loyalty and encourage retention.
- **Modify the marketing strategy and/or benefits of the products:** Optimize marketing strategies and strategically modify product benefits to effectively attract and engage the intended target customer segment.
- **Offer personalized financial plans that target the 40-65+ age group:** People in this age group have reached the peak of their financial goals and are nearing retirement. Hence, to retain this specific age group, it would be advantageous to offer personalized financial solutions such as investment plans, retirement planning techniques, etc. to help them accomplish their specific goals.
- **Improve customer satisfaction:** Establish a systematic approach of conducting regular feedback surveys to discern and promptly address customer issues and concerns. By proactively attending to their feedback, it is possible to mitigate churn rate and improve overall customer retention.

Implementing these strategies can improve customer satisfaction, bank and customer relationships and ultimtely customer retention.