**Library Initialization**

In [1]:

*#Required Libraries*

**import** numpy **as** np

**import** pandas **as** pd

**import** matplotlib.pyplot **as** plt

**import** seaborn **as** sns

**from** sklearn.preprocessing **import** LabelEncoder

**from** sklearn.preprocessing **import** MinMaxScaler

**from** sklearn.model\_selection **import** train\_test\_split

**from** sklearn.preprocessing **import** StandardScaler

In [2]:

*#Dataset path initialization*

df**=**pd**.**read\_csv('/content/Churn\_Modelling.csv')

**Dataset Summary**

In [3]:

df**.**head()

Out[3]:

|  | **RowNumber** | **CustomerId** | **Surname** | **CreditScore** | **Geography** | **Gender** | **Age** | **Tenure** | **Balance** | **NumOfProducts** | **HasCrCard** | **IsActiveMember** | **EstimatedSalary** | **Exited** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 1 | 15634602 | Hargrave | 619 | France | Female | 42 | 2 | 0.00 | 1 | 1 | 1 | 101348.88 | 1 |
| **1** | 2 | 15647311 | Hill | 608 | Spain | Female | 41 | 1 | 83807.86 | 1 | 0 | 1 | 112542.58 | 0 |
| **2** | 3 | 15619304 | Onio | 502 | France | Female | 42 | 8 | 159660.80 | 3 | 1 | 0 | 113931.57 | 1 |
| **3** | 4 | 15701354 | Boni | 699 | France | Female | 39 | 1 | 0.00 | 2 | 0 | 0 | 93826.63 | 0 |
| **4** | 5 | 15737888 | Mitchell | 850 | Spain | Female | 43 | 2 | 125510.82 | 1 | 1 | 1 | 79084.10 | 0 |

In [4]:

df**.**tail()

Out[4]:

|  | **RowNumber** | **CustomerId** | **Surname** | **CreditScore** | **Geography** | **Gender** | **Age** | **Tenure** | **Balance** | **NumOfProducts** | **HasCrCard** | **IsActiveMember** | **EstimatedSalary** | **Exited** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **9995** | 9996 | 15606229 | Obijiaku | 771 | France | Male | 39 | 5 | 0.00 | 2 | 1 | 0 | 96270.64 | 0 |
| **9996** | 9997 | 15569892 | Johnstone | 516 | France | Male | 35 | 10 | 57369.61 | 1 | 1 | 1 | 101699.77 | 0 |
| **9997** | 9998 | 15584532 | Liu | 709 | France | Female | 36 | 7 | 0.00 | 1 | 0 | 1 | 42085.58 | 1 |
| **9998** | 9999 | 15682355 | Sabbatini | 772 | Germany | Male | 42 | 3 | 75075.31 | 2 | 1 | 0 | 92888.52 | 1 |
| **9999** | 10000 | 15628319 | Walker | 792 | France | Female | 28 | 4 | 130142.79 | 1 | 1 | 0 | 38190.78 | 0 |

In [5]:

df**.**info

Out[5]:

In [6]:

df**.**shape

Out[6]:

(10000, 14)

In [7]:

df**.**isnull()**.**sum()

Out[7]:

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

dtype: int64

In [8]:

df**.**drop(['RowNumber','CustomerId','Surname'],axis**=**1, inplace**=True**)

In [13]:

*#Data visualization*

df**.**Geography**.**value\_counts()**.**plot(kind**=**'hist',color**=**"Purple")

df**.**Geography**.**value\_counts()

Out[13]:

France 5014

Germany 2509

Spain 2477

Name: Geography, dtype: int64



In [14]:

df**.**Age**.**describe()

Out[14]:

count 10000.000000

mean 38.921800

std 10.487806

min 18.000000

25% 32.000000

50% 37.000000

75% 44.000000

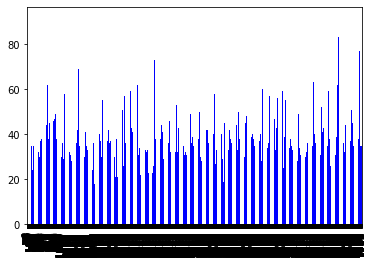
max 92.000000

Name: Age, dtype: float64

In [15]:

df**.**Age**.**plot(kind**=**'bar',color**=**"blue")

Out[15]:



In [17]:

df**.**IsActiveMember**.**value\_counts()**.**plot(kind**=**'bar',color**=**"pink")

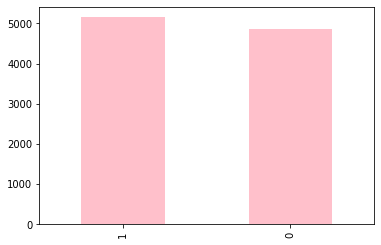
df**.**IsActiveMember**.**value\_counts()

Out[17]:

1 5151

0 4849

Name: IsActiveMember, dtype: int64



In [18]:

df**.**HasCrCard**.**value\_counts()**.**plot(kind**=**'bar',color**=**"Orange")

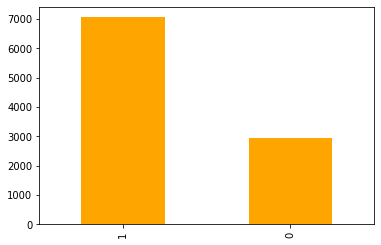
df**.**HasCrCard**.**value\_counts()

Out[18]:

1 7055

0 2945

Name: HasCrCard, dtype: int64



In [19]:

df**.**Tenure**.**value\_counts()**.**plot(kind**=**'bar',color**=**"SkyBlue");

df**.**Tenure**.**value\_counts()

Out[19]:

2 1048

1 1035

7 1028

8 1025

5 1012

3 1009

4 989

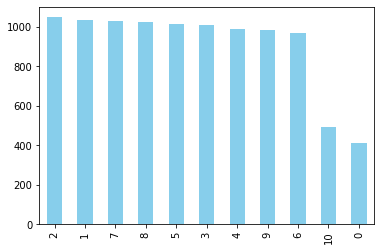
9 984

6 967

10 490

0 413

Name: Tenure, dtype: int64



In [20]:

df**.**NumOfProducts**.**value\_counts()**.**plot(kind**=**'bar',color**=**"black");

df**.**NumOfProducts**.**value\_counts()

Out[20]:

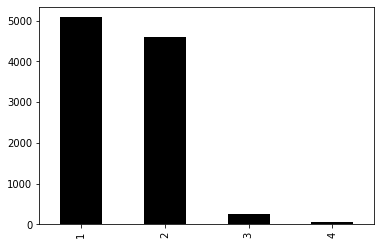
1 5084

2 4590

3 266

4 60

Name: NumOfProducts, dtype: int64



In [21]:

df**.**Exited**.**value\_counts()**.**plot(kind**=**'bar',color**=**"darkgreen");

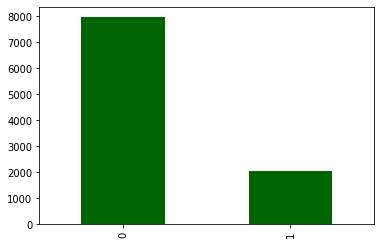
df**.**Exited**.**value\_counts()

Out[21]:

0 7963

1 2037

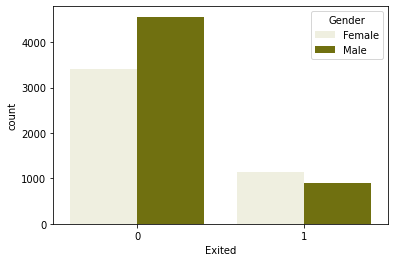
Name: Exited, dtype: int64



In [23]:

sns**.**countplot(x**=**df**.**Exited,hue**=**df**.**Gender,color**=**"Olive")

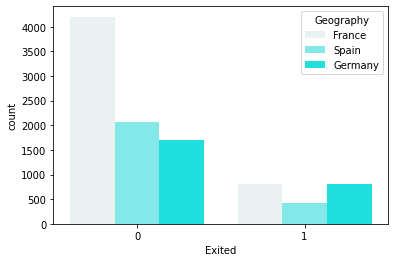
Out[23]:



In [24]:

sns**.**countplot(x**=**df**.**Exited,hue**=**df**.**Geography,color**=**"cyan")

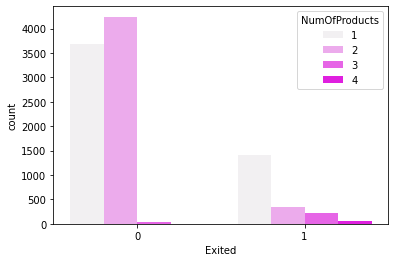
Out[24]:



In [30]:

sns**.**countplot(x**=**df**.**Exited,hue**=**df**.**NumOfProducts,color**=**"fuchsia")

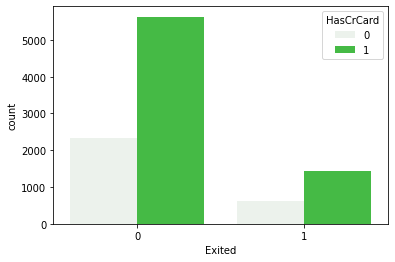
Out[30]:



In [32]:

sns**.**countplot(x**=**df**.**Exited,hue**=**df**.**HasCrCard,color**=**"limegreen")

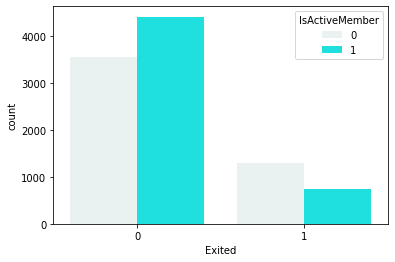
Out[32]:



In [33]:

sns**.**countplot(x**=**df**.**Exited,hue**=**df**.**IsActiveMember,color**=**"aqua")

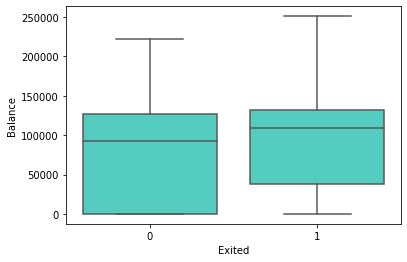
Out[33]:



In [36]:

sns**.**boxplot(x**=**df**.**Exited,y**=**df**.**Balance,color**=**"turquoise")

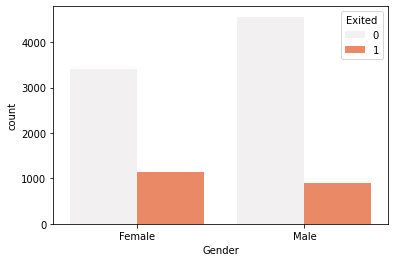
Out[36]:



In [37]:

sns**.**countplot(x**=**"Gender",hue**=**"Exited",data**=**df,color**=**"coral")

Out[37]:



In [38]:

df['Geography']**=**df['Geography']**.**map({'France':0,'Spain':1,'Germany':2})

In [39]:

X**=**df**.**iloc[:,:**-**1]**.**values

y**=**df**.**iloc[:,**-**1]**.**values

In [40]:

X**.**shape

Out[40]:

(10000, 10)

In [41]:

*#Feature Scaling of Data Set*

le**=**LabelEncoder()

X[:,2]**=**le**.**fit\_transform(X[:,2])

In [42]:

print(X)

[[619 0 0 ... 1 1 101348.88]

[608 1 0 ... 0 1 112542.58]

[502 0 0 ... 1 0 113931.57]

...

[709 0 0 ... 0 1 42085.58]

[772 2 1 ... 1 0 92888.52]

[792 0 0 ... 1 0 38190.78]]

In [43]:

scalerx **=** MinMaxScaler()

In [44]:

X **=** scalerx**.**fit\_transform(X)

In [45]:

X\_train, X\_test, y\_train, y\_test **=**train\_test\_split(X,y,test\_size**=**0.2, random\_state**=**0)

In [46]:

stdscaler **=** StandardScaler()

X\_train **=** stdscaler**.**fit\_transform(X\_train)

X\_test **=** stdscaler**.**transform(X\_test)