# EXPLORATORY DATA ANALYSIS FOR CUSTOMER SEGMENTATION BY TRANSACTION DATA

# SELECT TOP 10 CUSTOMERS BASED ON TOTAL AMOUNT IN DESCENDING ORDER

top\_10 = customer\_data[["customer\_id","total\_no\_of\_transactions","total\_amount\_in\_dollar"]].sort\_values(by='total\_amount\_in\_dollar', ascending=False).head(10)

plt.figure(figsize=(13, 5))

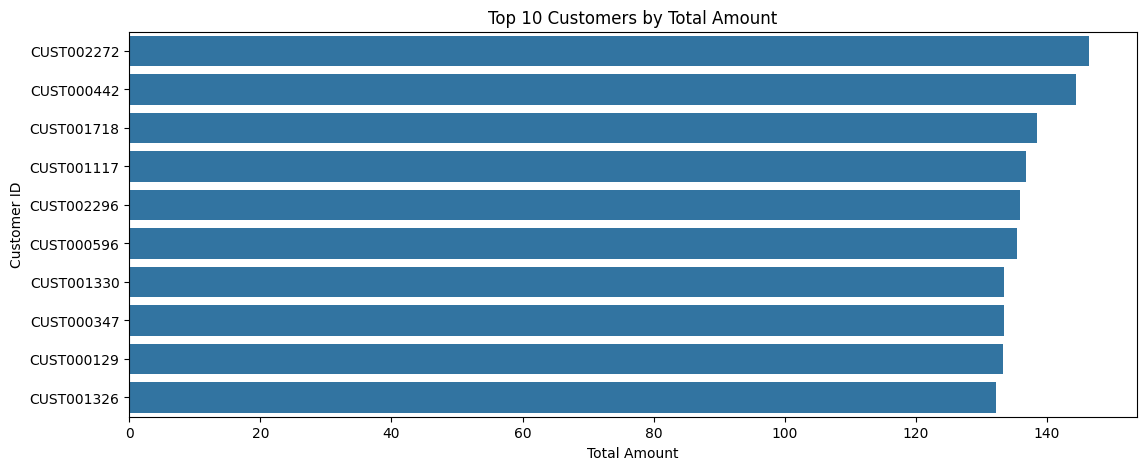
sns.barplot(y="customer\_id", x="total\_amount\_in\_dollar", data=top\_10, orient='h')

plt.xlabel("Total Amount")

plt.ylabel("Customer ID")

plt.title("Top 10 Customers by Total Amount")

plt.show()



#SELECT TOP 10 CUSTOMERS BASED ON WITHDRAWALS AMOUNT

top\_10 = customer\_data[["customer\_id","withdrawals\_amount\_in\_dollar"]].sort\_values(by='withdrawals\_amount\_in\_dollar', ascending=False).head(10)

plt.figure(figsize=(13, 5))

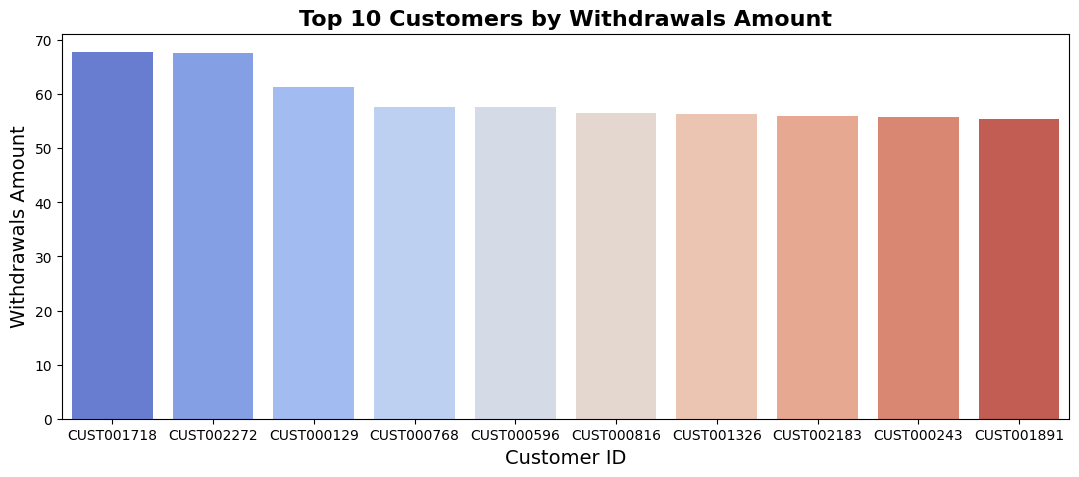
sns.barplot(x="customer\_id", y="withdrawals\_amount\_in\_dollar", data=top\_10, palette="coolwarm")

plt.title("Top 10 Customers by Withdrawals Amount", fontsize=16, fontweight="bold")

plt.xlabel("Customer ID", fontsize=14)

plt.ylabel("Withdrawals Amount", fontsize=14)

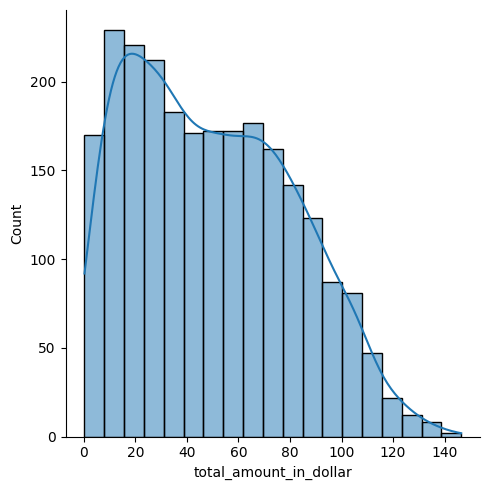
plt.show()



# CREATE A DISTRIBUTION PLOT FOR TOTAL AMOUNT WITH KDE

sns.displot(data=customer\_data, x="total\_amount\_in\_dollar", kde=True)

plt.show()



# SELECT TOP 10 CUSTOMERS BASED ON DEPOSITS AMOUNT

top\_10 = customer\_data[["customer\_id","deposits\_amount\_in\_dollar"]].sort\_values(by='deposits\_amount\_in\_dollar', ascending=False).head(10)

plt.figure(figsize=(13, 5))

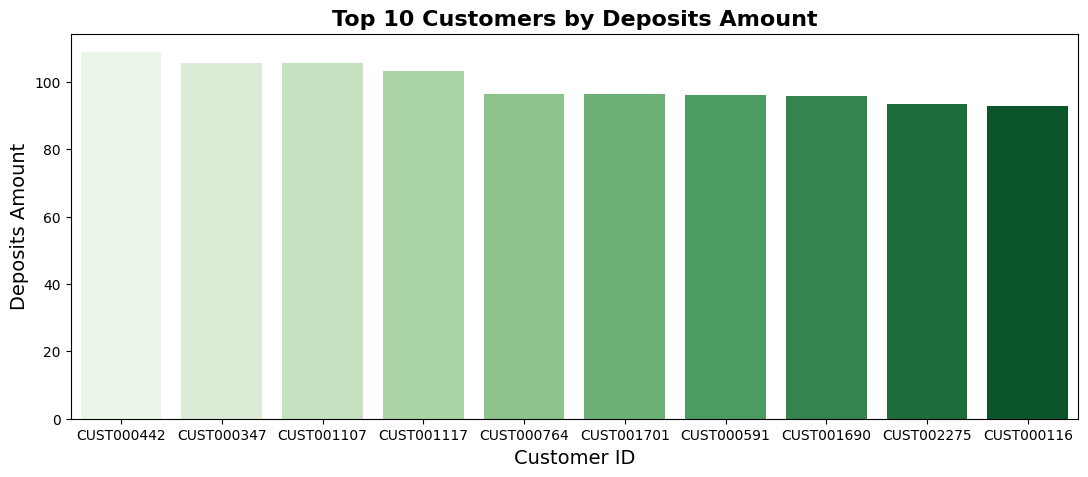
sns.barplot(x="customer\_id", y="deposits\_amount\_in\_dollar", data=top\_10, palette="Greens")

plt.title("Top 10 Customers by Deposits Amount", fontsize=16, fontweight="bold")

plt.xlabel("Customer ID", fontsize=14)

plt.ylabel("Deposits Amount", fontsize=14)

plt.show()



#CHECKING THE CORRELATION FOR NUMERIC COLUMNS USING HEATMAP

correlation\_matrix = customer\_data[["total\_no\_of\_transactions", "total\_amount\_in\_dollar",

"num\_of\_deposits", "num\_of\_withdrawals", "withdrawals\_amount\_in\_dollar", "deposits\_amount\_in\_dollar"]].corr()

plt.figure(figsize=(10, 8))

sns.heatmap(correlation\_matrix, annot=True, cmap='coolwarm', fmt='.2f')

plt.title("Correlation Matrix")

plt.show()

