# EXPLORATORY DATA ANALYSIS FOR LOAN DEFAULT IN BANK PROJECT

#BY AGE CHECKING THE REPAYMENT AVERAGE

plt.figure(figsize=(15,6))

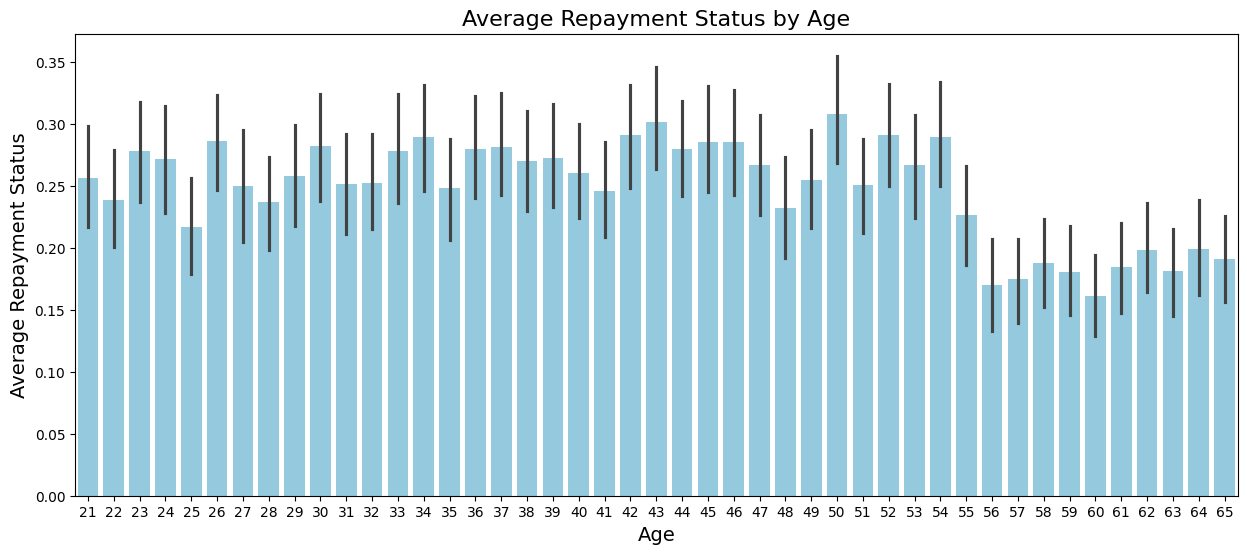
sns.barplot(x='age', y='repayment\_status', data=loan\_data, color='skyblue')

plt.title('Average Repayment Status by Age', fontsize=16)

plt.xlabel('Age', fontsize=14)

plt.ylabel('Average Repayment Status', fontsize=14)

plt.show()



#DISPLAY THE TOP 10 HIGH LOAN CUSTOMER

top\_10\_loan\_data = loan\_data[['name', 'loan\_amount', 'repayment\_status']].sort\_values(by='loan\_amount', ascending=False).head(10)

plt.figure(figsize=(25, 6))

sns.barplot(x='name', y='loan\_amount', hue='repayment\_status', data=top\_10\_loan\_data, palette='viridis')

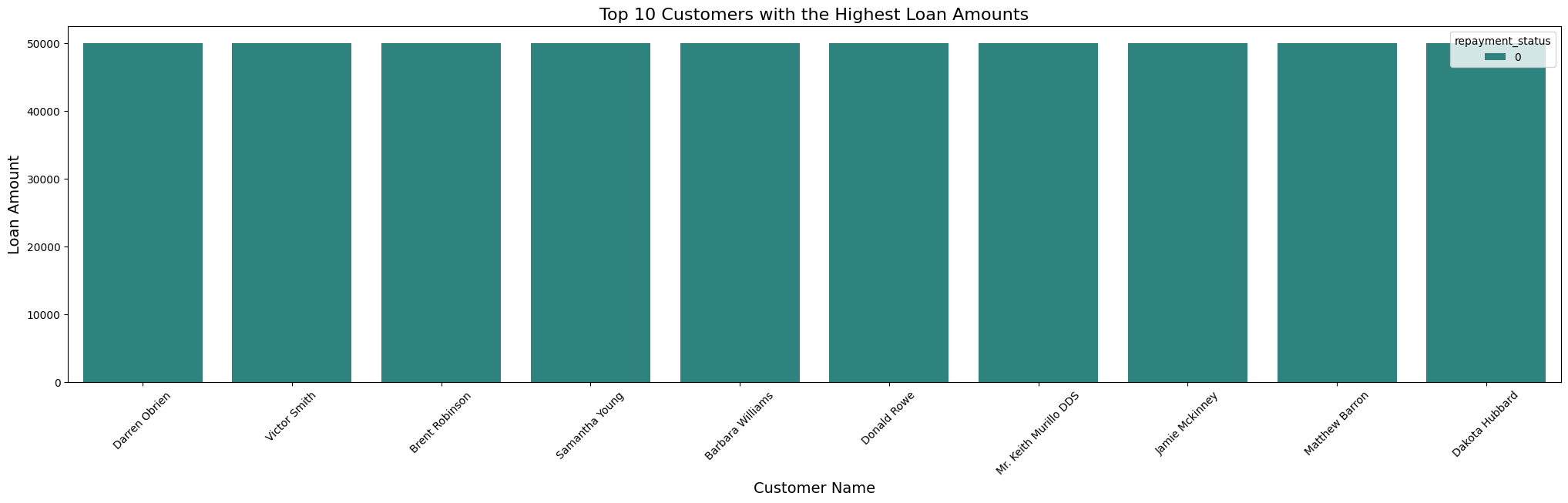
plt.title('Top 10 Customers with the Highest Loan Amounts', fontsize=16)

plt.xlabel('Customer Name', fontsize=14)

plt.ylabel('Loan Amount', fontsize=14)

plt.xticks(rotation=45)

plt.show()



#TOP 10 HIGH INCOME CUSTOMERS WITH REPAYMENT STATUS

top\_10\_loan\_data = loan\_data[['name', 'income', 'repayment\_status']].sort\_values(by='income', ascending=False).head(10)

plt.figure(figsize=(25, 6))

sns.barplot(x='name', y='income', hue='repayment\_status', data=top\_10\_loan\_data, palette='magma')

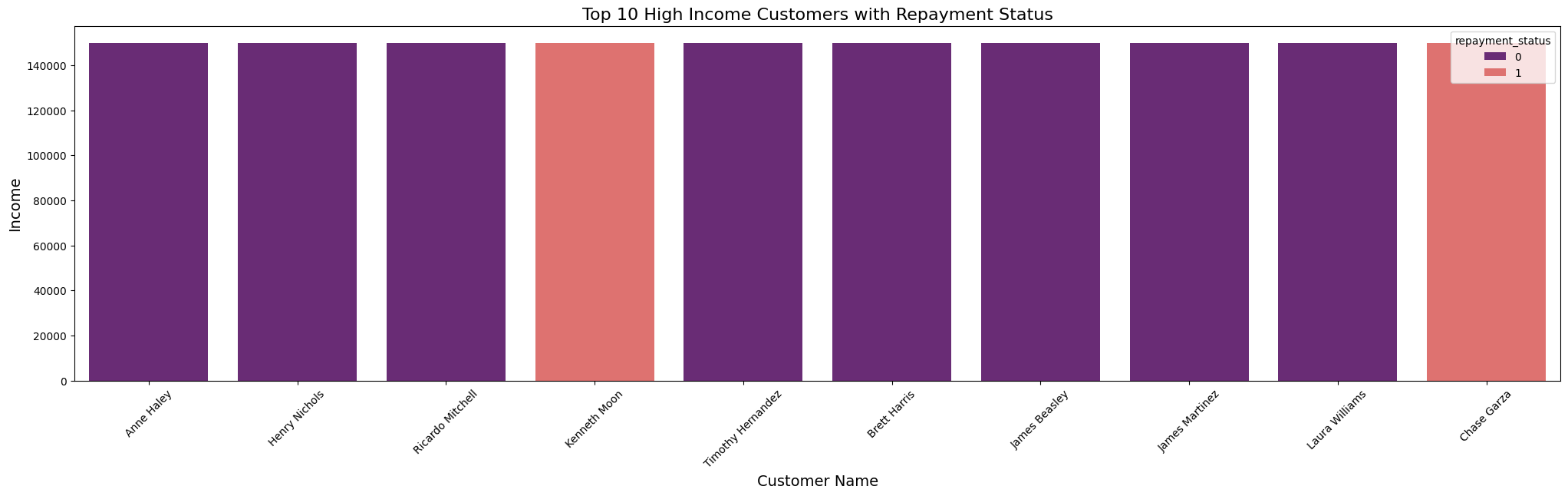
plt.title('Top 10 High Income Customers with Repayment Status', fontsize=16)

plt.xlabel('Customer Name', fontsize=14)

plt.ylabel('Income', fontsize=14)

plt.xticks(rotation=45)

plt.show()



#NO OF CUSTOMER COLLECTED LOAN BY AGE WISE

plt.figure(figsize=(13,6))

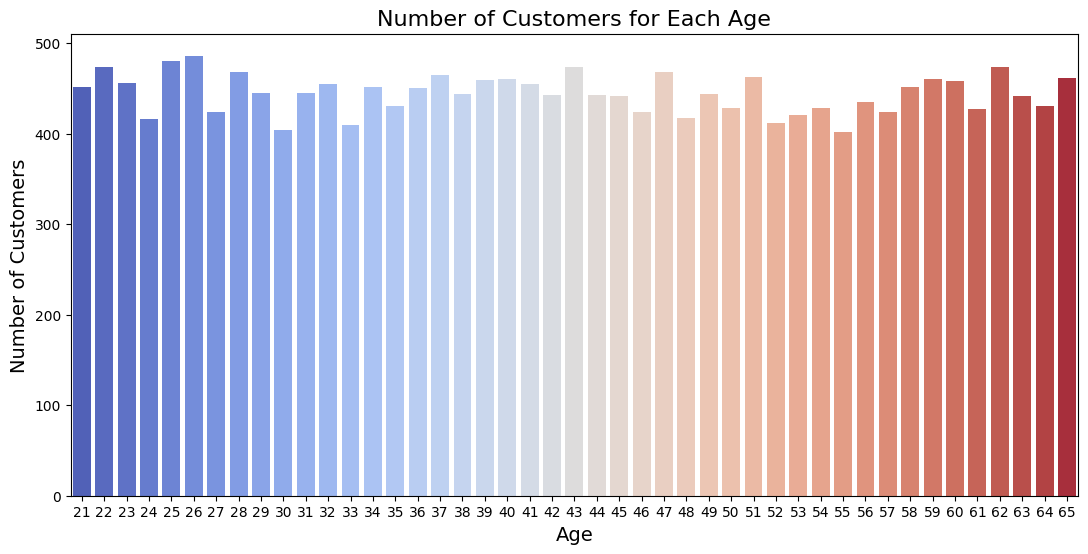
sns.countplot(x="age", data=loan\_data, palette='coolwarm')

plt.title('Number of Customers for Each Age', fontsize=16)

plt.xlabel('Age', fontsize=14)

plt.ylabel('Number of Customers', fontsize=14)

plt.show()



#CHECK THE CORRELATION FOR NUMERIC COLUMNS

correlation\_matrix = loan\_data[['age', 'income', 'credit\_score', 'loan\_term', 'loan\_amount', 'monthly\_installment', 'interest\_rate']].corr()

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

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

plt.title('Correlation Matrix', fontsize=16)

plt.show()

