Project: Loan Approval Analysis

In [1]:
```
# importing major libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px

# additional libraries
import warnings
warnings.filterwarnings('ignore')
```

In [2]:
```
#importing dataset
df = pd.read_csv('loan_sanction_test.csv')
```

In [3]:
```
# overview data
df.head()
```

Out[3]:

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	Applicant
0	LP001015	Male	Yes	0	Graduate	No	
1	LP001022	Male	Yes	1	Graduate	No	
2	LP001031	Male	Yes	2	Graduate	No	
3	LP001035	Male	Yes	2	Graduate	No	
4	LP001051	Male	No	0	Not Graduate	No	

In [4]:
```
#shape
df.shape
```

Out[4]:
```
(367, 12)
```

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

Out[5]:
```
Index(['Loan_ID', 'Gender', 'Married', 'Dependents', 'Education',
       'Self_Employed', 'ApplicantIncome', 'CoapplicantIncome', 'LoanAmount',
       'Loan_Amount_Term', 'Credit_History', 'Property_Area'],
      dtype='object')
```

📄 Data Card — Loan Approval Dataset

📌 Dataset Overview

- **Dataset Name:** Loan Approval Dataset
- **Domain:** Banking & Financial Services
- **Use Case:** Home Loan Approval Analysis
- **Data Type:** Structured (Tabular)
- **Primary Objective:**
  To analyze applicant demographics, financial background, and credit history to understand loan approval patterns.

📊 Dataset Structure

- **Total Records:** Not specified (sample shown)
- **Total Features:** 12
- **Identifier Column:** `Loan_ID`

🧾 Column Description

Column Name	Data Type	Description
Loan_ID	Categorical (ID)	Unique identifier for each loan application
Gender	Categorical	Gender of the applicant (Male/Female)
Married	Categorical	Marital status of the applicant
Dependents	Categorical	Number of dependents of the applicant
Education	Categorical	Education level (Graduate / Not Graduate)
Self_Employed	Categorical	Employment status of the applicant
ApplicantIncome	Numerical	Monthly income of the applicant
CoapplicantIncome	Numerical	Monthly income of the co-applicant
LoanAmount	Numerical	Loan amount requested (in thousands)
Loan_Amount_Term	Numerical	Loan repayment term (in months)
Credit_History	Numerical (Binary)	Credit history record (1 = Good, 0 = Bad)
Property_Area	Categorical	Location of property (Urban / Semiurban / Rural)

In [6]:
```
# Seaking Information
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 367 entries, 0 to 366
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Loan_ID            367 non-null    object 
 1   Gender             356 non-null    object 
 2   Married            367 non-null    object 
 3   Dependents         357 non-null    object 
 4   Education          367 non-null    object 
 5   Self_Employed      344 non-null    object 
 6   ApplicantIncome    367 non-null    int64  
 7   CoapplicantIncome  367 non-null    int64  
 8   LoanAmount         362 non-null    float64
 9   Loan_Amount_Term   361 non-null    float64
 10  Credit_History     338 non-null    float64
 11  Property_Area      367 non-null    object 
dtypes: float64(3), int64(2), object(7)
memory usage: 34.5+ KB
```

Data Quality Observation:

The dataset contains 367 records with 12 well-defined features. A few columns such as `Gender`, `Dependents`, `Self_Employed`, and `Credit_History` have missing values, with `Credit_History` showing the highest gaps. Income-related variables have no missing values, ensuring reliable financial analysis. Overall, the data quality is good and suitable for exploratory data analysis after basic imputation.

In [7]:
```
# Seeking description
df.describe()
```

Out[7]:

	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	C
count	367.000000	367.000000	362.000000	361.000000	
mean	4805.599455	1569.577657	136.132597	342.537396	
std	4910.685399	2334.232099	61.366652	65.156643	
min	0.000000	0.000000	28.000000	6.000000	
25%	2864.000000	0.000000	100.250000	360.000000	
50%	3786.000000	1025.000000	125.000000	360.000000	
75%	5060.000000	2430.500000	158.000000	360.000000	
max	72529.000000	24000.000000	550.000000	480.000000	

Accuracy Issues Identified:

The dataset shows extreme outliers in `ApplicantIncome` and `CoapplicantIncome`, which can distort average values. Several numerical columns (`LoanAmount`, `Loan_Amount_Term`, `Credit_History`) have missing records, affecting statistical accuracy. Income and loan distributions are highly skewed, making mean values less representative than medians. These issues require careful handling during analysis to avoid misleading insights.

In [8]:
```
df.isnull().sum()
```

Out[8]:
```
Loan_ID               0
Gender               11
Married               0
Dependents           10
Education             0
Self_Employed        23
ApplicantIncome       0
CoapplicantIncome     0
LoanAmount            5
Loan_Amount_Term      6
Credit_History       29
Property_Area         0
dtype: int64
```

In [11]:
```
# Numerical → Median
num_cols = df.select_dtypes(include=np.number).columns
df[num_cols] = df[num_cols].fillna(df[num_cols].median())
```

In [12]:
```
# Categorical → Mode
cat_cols = df.select_dtypes(include='object').columns
df[cat_cols] = df[cat_cols].fillna(df[cat_cols].mode().iloc[0])
```

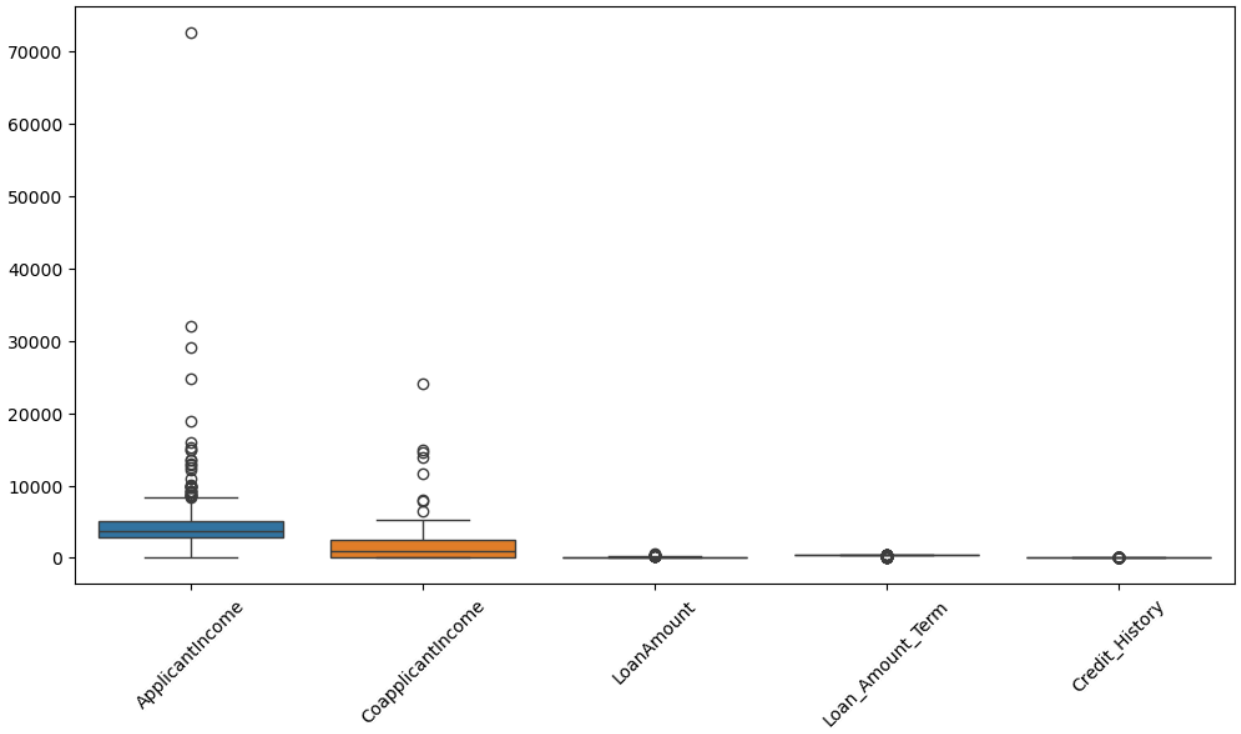
In [13]:
```
df.describe()
```

Out[13]:

	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	C
count	367.000000	367.000000	367.000000	367.000000	
mean	4805.599455	1569.577657	135.980926	342.822888	
std	4910.685399	2334.232099	60.959739	64.658402	
min	0.000000	0.000000	28.000000	6.000000	
25%	2864.000000	0.000000	101.000000	360.000000	
50%	3786.000000	1025.000000	125.000000	360.000000	
75%	5060.000000	2430.500000	157.500000	360.000000	
max	72529.000000	24000.000000	550.000000	480.000000	

Univariate Analysis

In [15]:
```
# Boxplots (Outlier Detection)
plt.figure(figsize=(12,6))
sns.boxplot(data=df[num_cols])
plt.xticks(rotation=45)
plt.show()
```

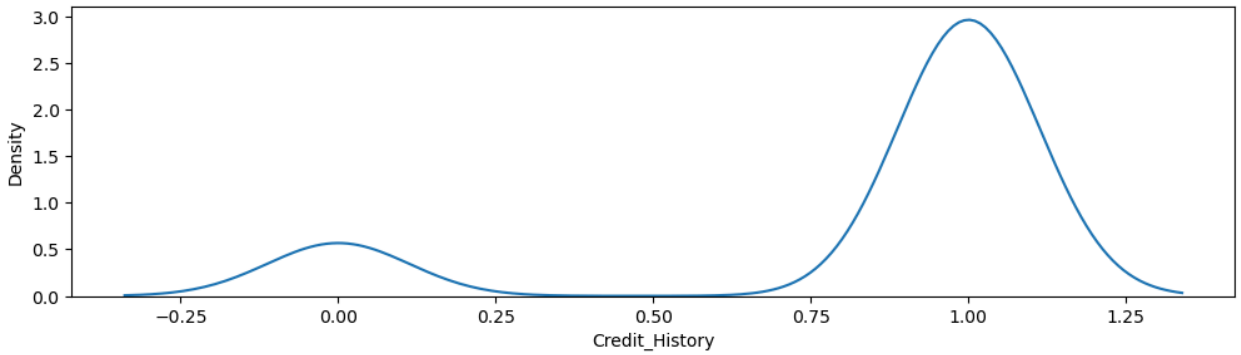


Insight

- Income-related variables contain extreme outliers
- These outliers may represent high-net-worth applicants

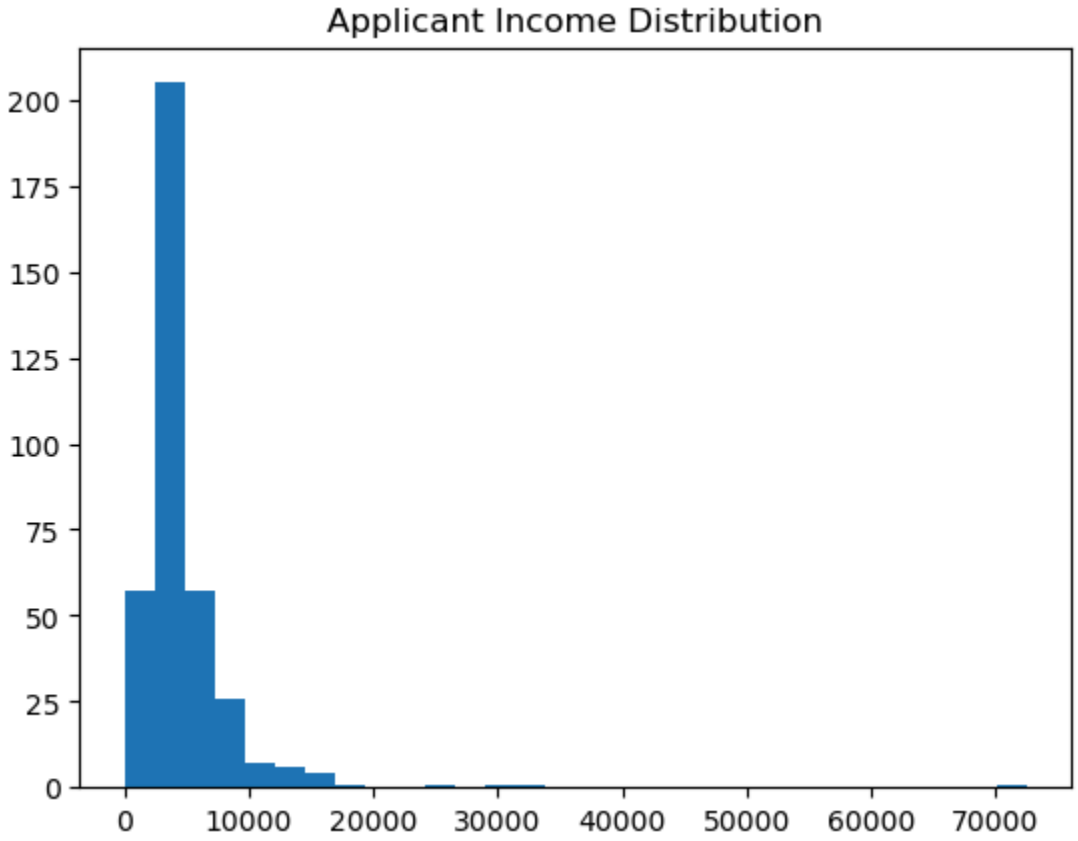
In [18]:
```
# Visualisation
plt.figure(figsize=(12,3))
sns.kdeplot(data=df,x='Credit_History')
plt.show()

print('skewness',df.Credit_History.skew())
```



skewness -1.8547214446428353

In [19]:
```
#Applicant Income – Histogram
plt.hist(df['ApplicantIncome'], bins=30)
plt.title('Applicant Income Distribution')
plt.show()
```

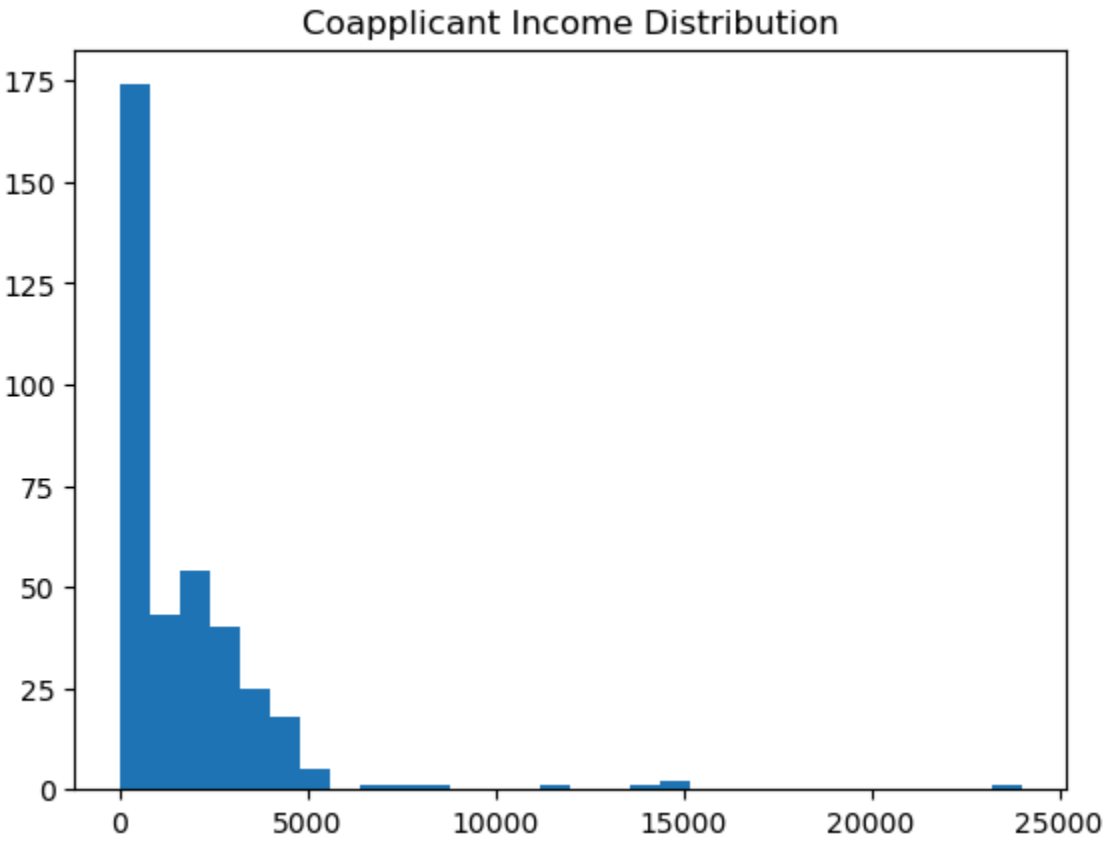


Insight:

- Income distribution is **right-skewed**
- Majority applicants belong to **low to mid income groups**
- A few high-income outliers indicate premium borrowers

In [20]:
```
#Coapplicant Income – Histogram
```

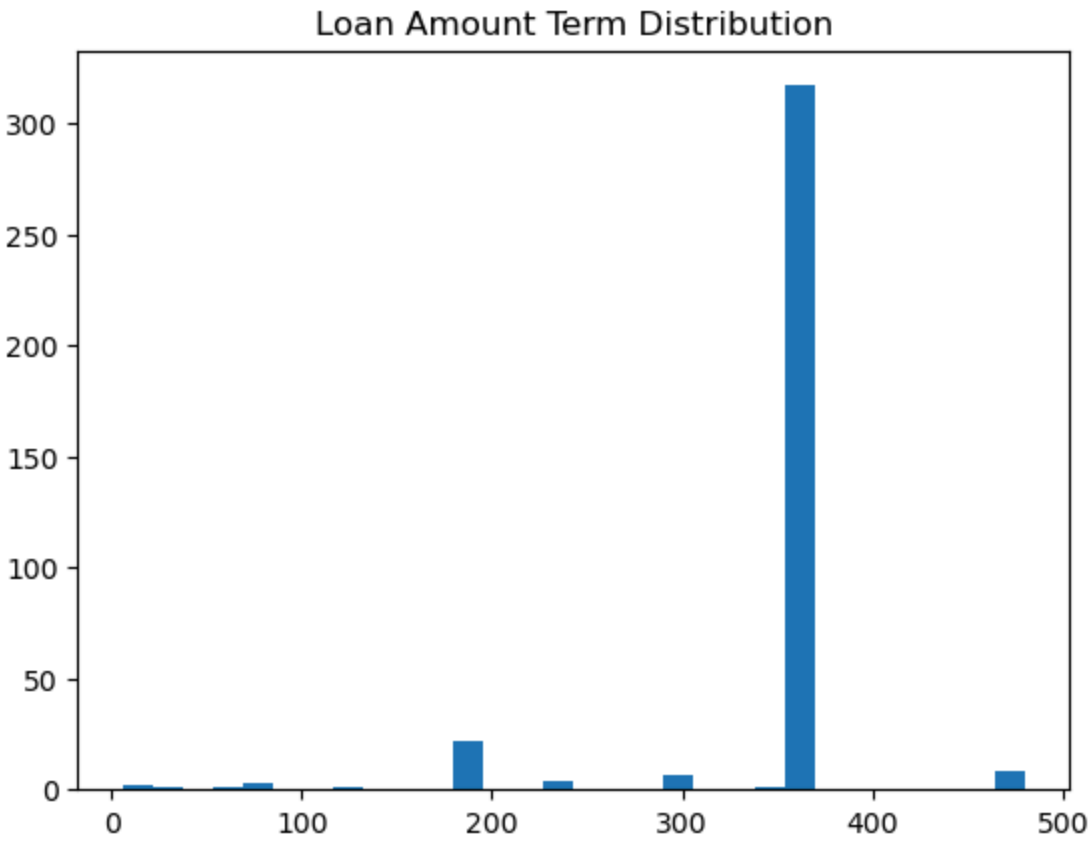
```
plt.hist(df['CoapplicantIncome'], bins=30)
plt.title('Coapplicant Income Distribution')
plt.show()
```



Insight:

- Many applicants have **zero coapplicant income**
- Indicates a high number of **single-income households**
- Coapplicant income plays a secondary role in approvals

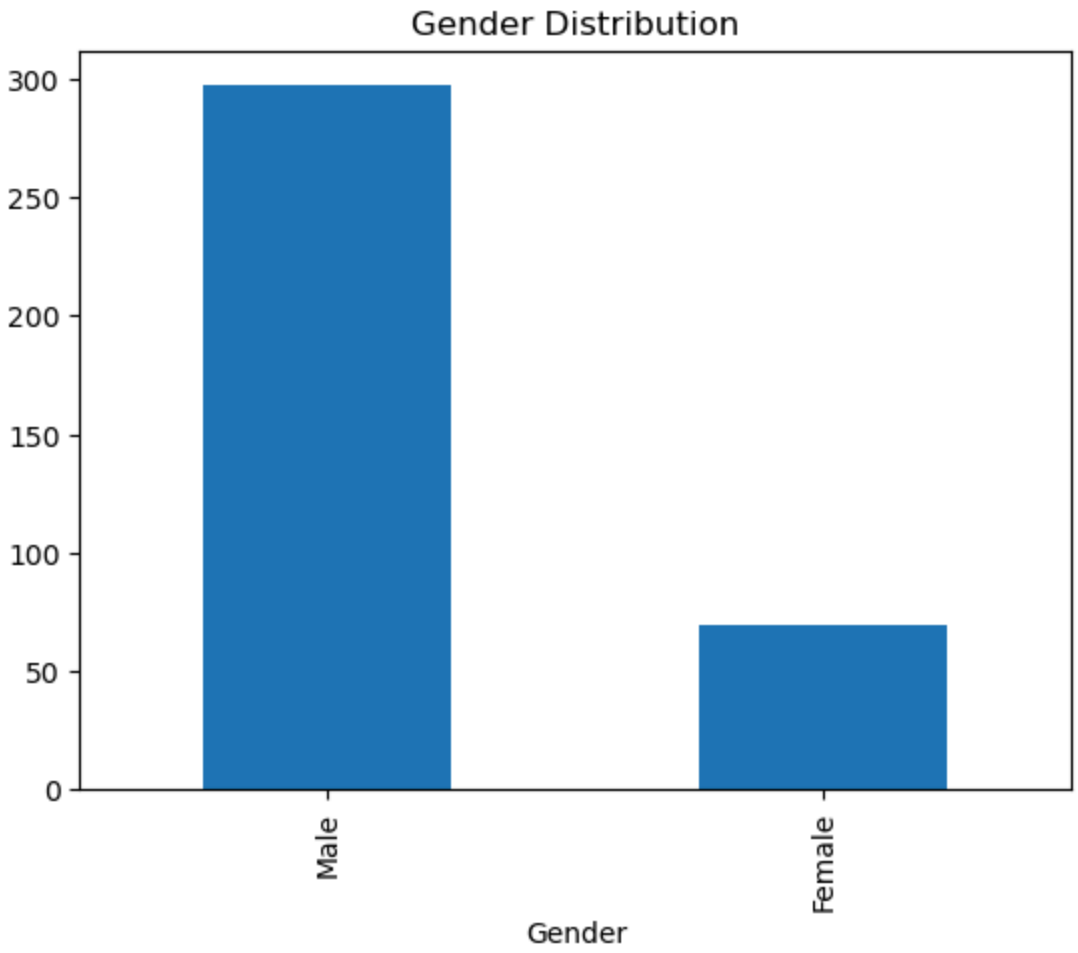
In [23]:
```
# Loan Amount Term – Histogram
plt.hist(df['Loan_Amount_Term'], bins=30)
plt.title('Loan Amount Term Distribution')
plt.show()
```



Insight:

- Loan amounts are concentrated in the **lower range**
- Distribution shows **positive skewness**
- Higher loan values are comparatively rare

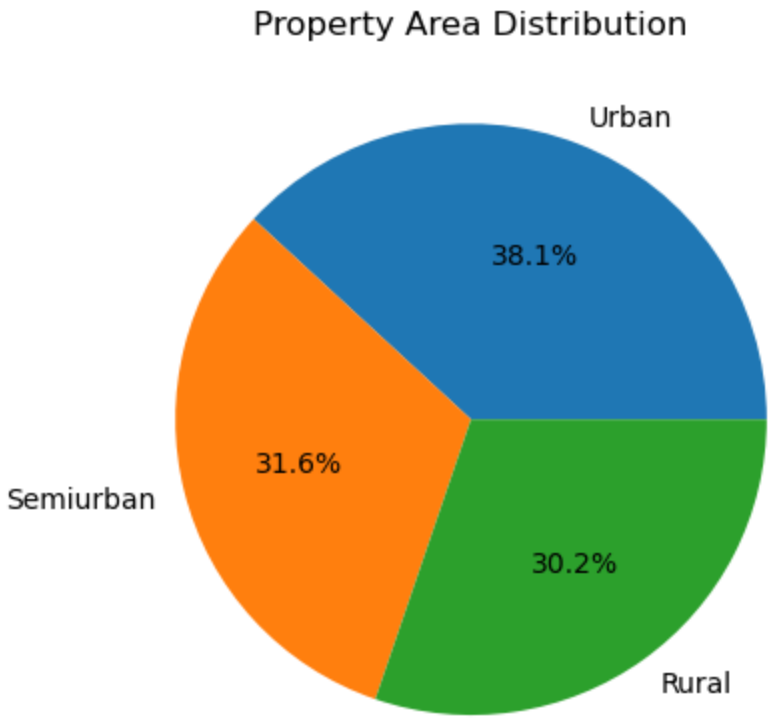
In [21]:
```
# Gender – Bar Chart
df['Gender'].value_counts().plot(kind='bar')
plt.title('Gender Distribution')
plt.show()
```



Insight:

- Male applicants dominate the dataset
- Female participation is lower but consistent
- Approval pattern is fairly balanced across genders

In [22]:
```
# Property Area – Pie Chart
df['Property_Area'].value_counts().plot(kind='pie', autopct='%1.1f%%')
plt.title('Property Area Distribution')
plt.ylabel('')
plt.show()
```

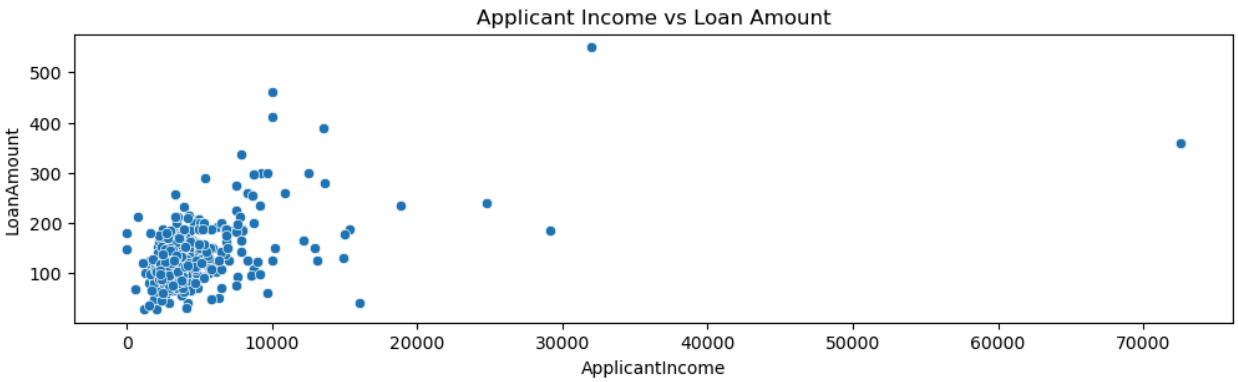


Insight:

- Semiurban areas contribute the **highest number of applications**
- Rural and urban areas show moderate participation

Bivariate Analysis

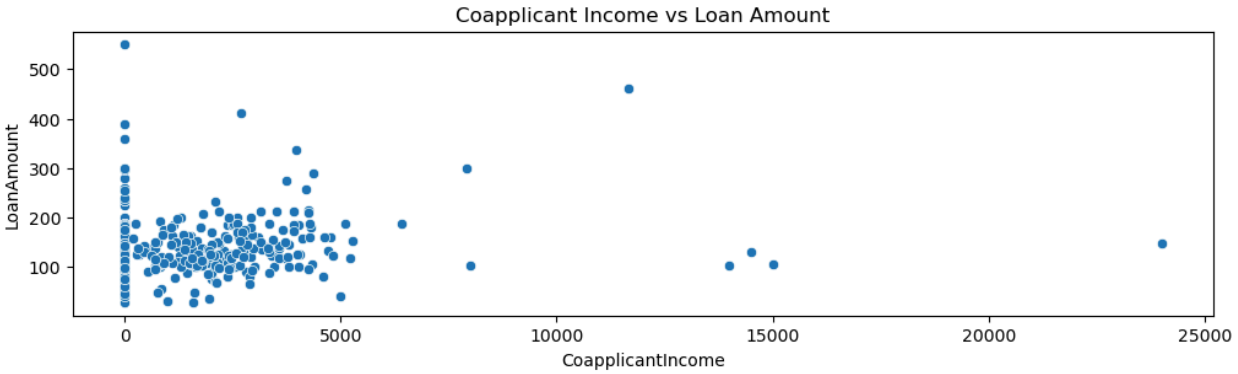
In [26]:
```
plt.figure(figsize=(12,3))
sns.scatterplot(data=df, x='ApplicantIncome', y='LoanAmount')
plt.title('Applicant Income vs Loan Amount')
plt.show()
```



Insight:

Higher applicant income generally allows higher loan amounts, though income alone does not guarantee larger loans.

In [27]:
```
plt.figure(figsize=(12,3))
sns.scatterplot(data=df, x='CoapplicantIncome', y='LoanAmount')
plt.title('Coapplicant Income vs Loan Amount')
plt.show()
```



Insight:

Coapplicant income supports loan eligibility but shows weaker influence compared to applicant income.

In [28]:
```
fig = px.scatter(df,x='ApplicantIncome',y='LoanAmount',color='Credit_History',
    height=500,title='Applicant Income vs Loan Amount by Credit History')
fig.show()
```

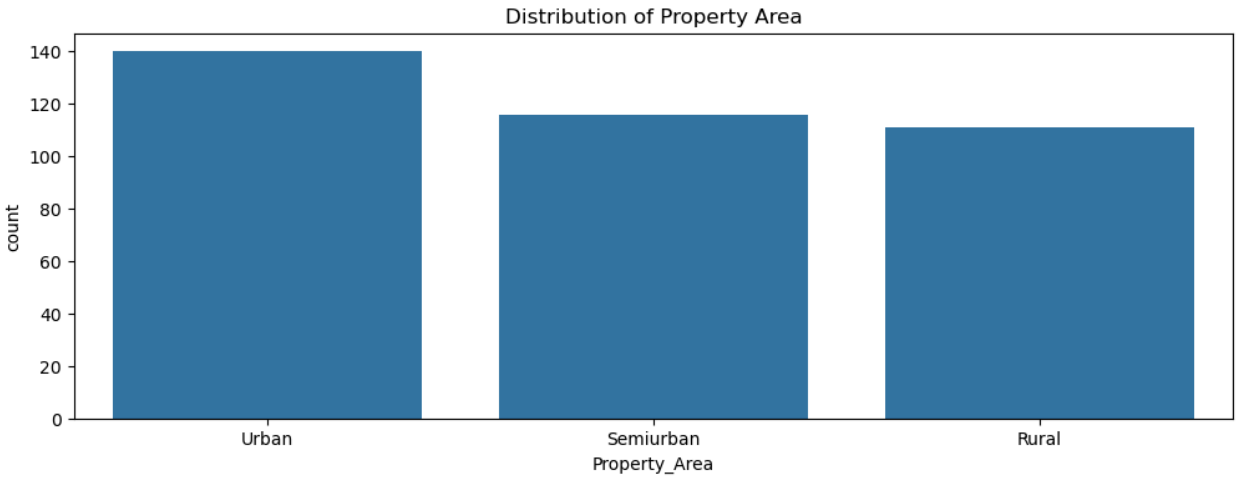
Insight:

Applicants with a good credit history (`Credit_History = 1`) are more likely to receive higher loan amounts across different income levels. While loan amount generally increases with applicant income, strong credit history clearly strengthens loan eligibility. Applicants with poor or missing credit history show limited access to higher loan amounts.

In [29]:
```
fig = px.scatter(df,x='ApplicantIncome',y='Loan_Amount_Term',color='Education'
    hover_data=['ApplicantIncome', 'Loan_Amount_Term', 'LoanAmount'],
    height=500,title='Income vs Loan Term by Education')
fig.show()
```

Insight:
Loan repayment terms remain largely consistent across income levels, with most applicants opting for standard long-term durations. Both graduate and non-graduate applicants show similar loan tenure patterns, indicating that **education level has minimal impact on loan term selection**. Income primarily affects loan eligibility rather than repayment duration.

In [32]:
```
# Countplot
plt.figure(figsize=(12,4))
sns.countplot(data=df, x='Property_Area')
plt.title('Distribution of Property Area')
plt.show()
# Pie chart
temp = df['Property_Area'].value_counts().reset_index()
temp.columns = ['Property_Area', 'count']
px.pie(
temp,names='Property_Area',values='count',height=400,title='Property Area Dist
```

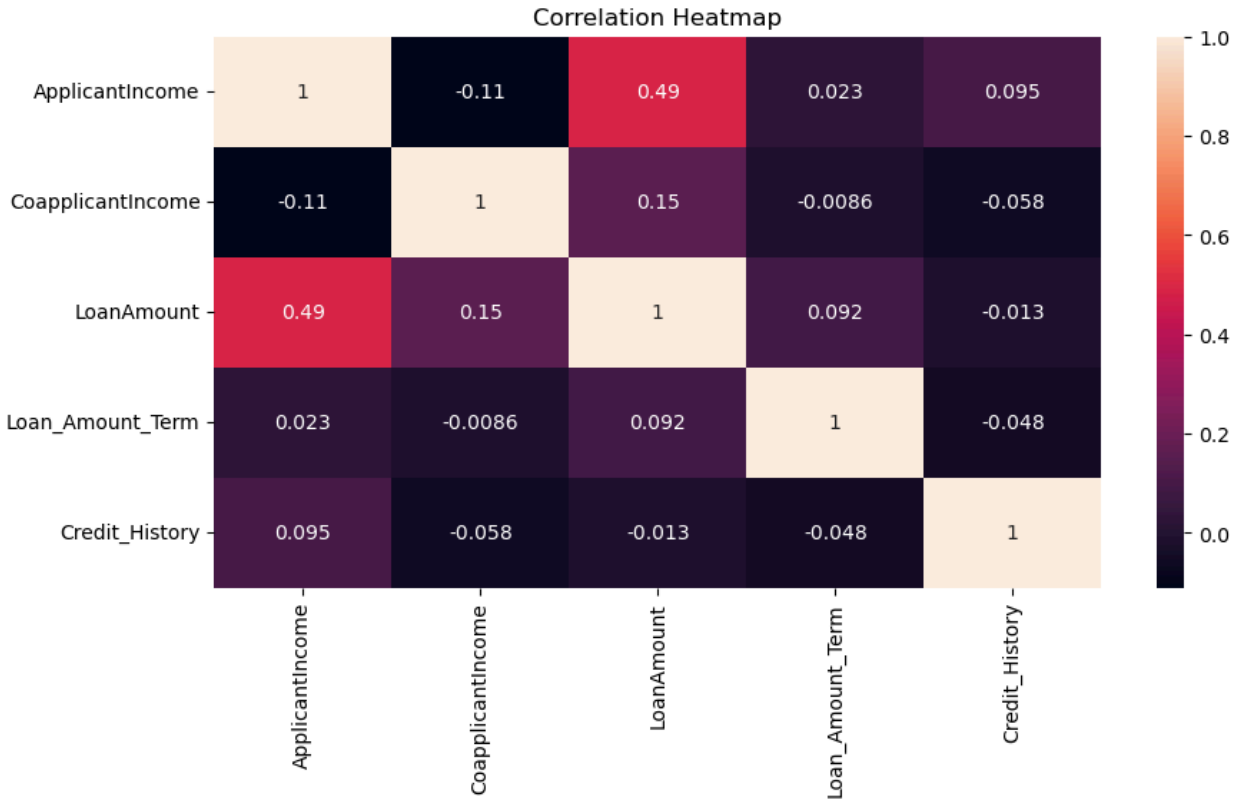


Insight:
Semiurban areas account for the highest number of loan applications, followed by urban regions, while rural areas contribute the least. This suggests that loan demand is stronger in semiurban and urban locations, possibly due to better access to financial institutions and housing development.

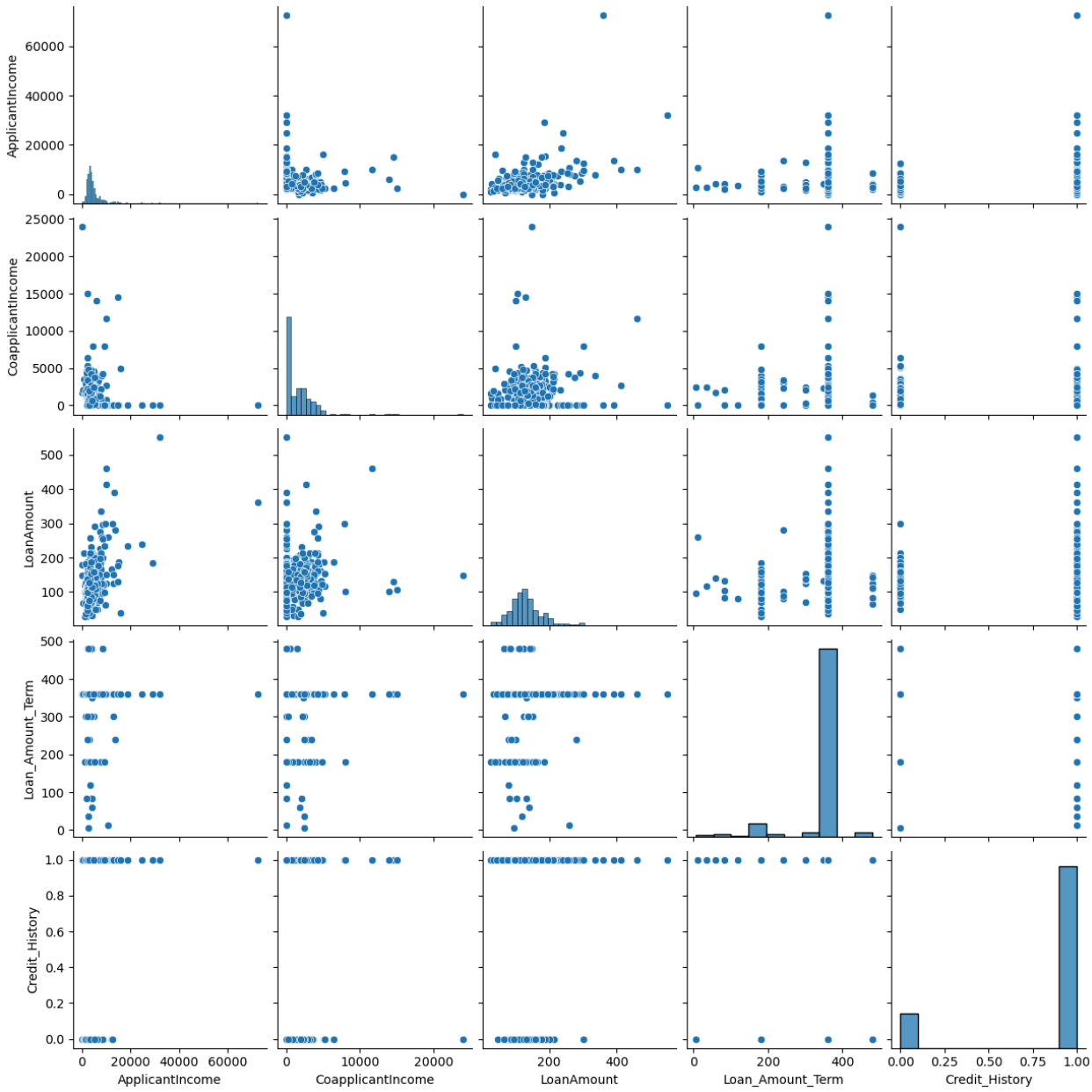
Multivariate Analysis

In [37]:
```
plt.figure(figsize=(10,5))
sns.heatmap(df.select_dtypes(include='number').corr(), annot=True)
plt.title('Correlation Heatmap')
```

```
plt.show()
```



In [39]:
```
# Pair Plot (Multiple Variables)
sns.pairplot(df)
plt.show()
```



In [40]:

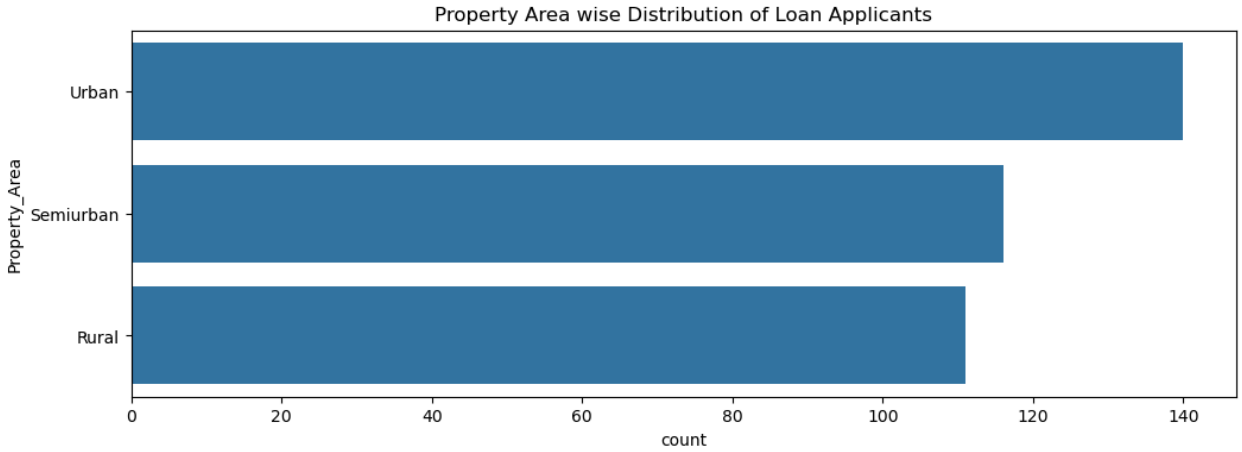
```
temp = df['Property_Area'].value_counts().reset_index()
temp.columns = ['Property_Area', 'count']

# Countplot (horizontal)
plt.figure(figsize=(12,4))
sns.countplot(
    data=df,
    y='Property_Area',
    order=temp['Property_Area']
)
plt.title('Property Area wise Distribution of Loan Applicants')
plt.show()

# Pie / Donut chart
px.pie(
    temp,
```

```
    values='count',
    names='Property_Area',
    height=400,
    hole=0.5,
    title='Distribution of Loan Applications by Property Area'
).show()

temp
```



Out[40]:

	Property_Area	count
0	Urban	140
1	Semiurban	116
2	Rural	111

Insight:

The majority of loan applications come from semiurban areas, followed by urban regions, while rural areas have the lowest representation. This indicates stronger loan demand in semiurban and urban locations compared to rural regions within the dataset.

In [41]:

```
temp = df['Education'].value_counts().reset_index()
temp.columns = ['Education', 'count']


# Pie / Donut chart
px.pie(
temp,
names='Education',
values='count',
title='Distribution of Education',
hole=0.4,
color_discrete_sequence=px.colors.sequential.RdBu
).show()


# Median LoanAmount by Education
temp2 = df.groupby('Education')['LoanAmount'].median().reset_index()


px.bar(
temp2,
x='LoanAmount',
y='Education',
title='Median Loan Amount by Education',
color_discrete_sequence=px.colors.sequential.RdBu
).show()


temp
```

Out[41]:

	Education	count
0	Graduate	283
1	Not Graduate	84

Insight:
Graduate applicants form the majority of loan applications in the dataset. The median loan amount for graduates is slightly higher than that of non-graduates, indicating better loan eligibility among applicants with higher education levels.

Conclusion

This exploratory data analysis highlights clear patterns in loan application behavior across demographic, financial, and regional factors. Applicant income, credit history, education level, and property area play an important role in understanding loan demand and eligibility trends. Most applications originate from semiurban and urban regions, with graduates generally qualifying for higher loan amounts.

Overall, the dataset is well-structured and suitable for exploratory analysis, providing meaningful insights into borrower profiles and lending patterns.

In [ ]: