Streamlit Steps to Create Loan Prediction App

This document provides a step-by-step guide to creating a loan prediction app using Streamlit. The app will load a pre-trained machine learning model, accept user input, and predict whether a loan will be approved or rejected based on the given data.  
  
### Prerequisites:  
1. Python installed on your machine.  
2. Streamlit library installed.  
3. Trained model in a `.pkl` file.  
4. A basic understanding of Python and machine learning concepts.  
  
Let's go through the process step by step.

# Step 1: Install Streamlit

Before you start, you need to install Streamlit on your local machine. You can do this using pip:  
  
```  
pip install streamlit  
```  
This command will install Streamlit and its dependencies.

# Step 2: Prepare Model File

Since your model is saved as a `.pkl` file, download it from your cloud storage (Google Drive or another location) to your local machine.  
  
For example, if your file is `loan\_model.pkl`, save it in the same folder as your Streamlit app (`loan\_prediction\_app.py`).

# Step 3: Create the Streamlit Python File

Create a new Python file for your Streamlit app, e.g., `loan\_prediction\_app.py`, and write the code to load the trained model and make predictions based on user input.  
  
Here is an example code snippet:  
  
```python  
import streamlit as st  
import pickle  
import pandas as pd  
  
# Load the trained model from .pkl file  
model = pickle.load(open('loan\_model.pkl', 'rb'))  
  
# Streamlit UI  
st.title('Loan Approval Prediction')  
st.write('Enter the following details to predict loan approval.')  
  
# Input fields for prediction  
no\_of\_dependents = st.number\_input('No of Dependents', min\_value=0, max\_value=10)  
education = st.selectbox('Education', ['Graduate', 'Not Graduate'])  
self\_employed = st.selectbox('Self Employed', ['Yes', 'No'])  
income\_annum = st.number\_input('Annual Income (in ₹)', min\_value=0)  
loan\_amount = st.number\_input('Loan Amount (in ₹)', min\_value=0)  
loan\_term = st.number\_input('Loan Term (in years)', min\_value=1)  
cibil\_score = st.number\_input('Cibil Score', min\_value=300, max\_value=900)  
residential\_assets\_value = st.number\_input('Residential Assets Value (in ₹)', min\_value=0)  
commercial\_assets\_value = st.number\_input('Commercial Assets Value (in ₹)', min\_value=0)  
luxury\_assets\_value = st.number\_input('Luxury Assets Value (in ₹)', min\_value=0)  
bank\_asset\_value = st.number\_input('Bank Asset Value (in ₹)', min\_value=0)  
  
# Convert categorical inputs to numeric values (if needed)  
education = 1 if education == 'Graduate' else 0  
self\_employed = 1 if self\_employed == 'Yes' else 0  
  
# Prepare input data for prediction  
input\_data = pd.DataFrame({  
 'no\_of\_dependents': [no\_of\_dependents],  
 'education': [education],  
 'self\_employed': [self\_employed],  
 'income\_annum': [income\_annum],  
 'loan\_amount': [loan\_amount],  
 'loan\_term': [loan\_term],  
 'cibil\_score': [cibil\_score],  
 'residential\_assets\_value': [residential\_assets\_value],  
 'commercial\_assets\_value': [commercial\_assets\_value],  
 'luxury\_assets\_value': [luxury\_assets\_value],  
 'bank\_asset\_value': [bank\_asset\_value]  
})  
  
# Predict button  
if st.button('Predict Loan Approval'):  
 prediction = model.predict(input\_data)  
 result = 'Approved' if prediction[0] == 1 else 'Rejected'  
 st.write(f'Loan Status: {result}')  
```  
  
This code creates a simple UI with inputs for the various features and uses the trained model to predict whether the loan is approved or rejected.

# Step 4: Running the Streamlit App

Once the Python file (`loan\_prediction\_app.py`) is ready, it's time to run the app.  
  
To start the Streamlit app, open a terminal or command prompt and navigate to the folder where the `loan\_prediction\_app.py` file is stored.  
  
Run the following command:  
  
```  
streamlit run loan\_prediction\_app.py  
```  
  
Streamlit will start the app and provide you with a local URL (e.g., `http://localhost:8501`) to access it in your browser.

# Step 5: Interacting with the App

Now that the app is running, open your browser and go to the provided local URL (e.g., `http://localhost:8501`).  
  
Enter the values for the input fields (e.g., no\_of\_dependents, income\_annum, etc.) and click the "Predict Loan Approval" button to see if the loan is approved or rejected based on the model's prediction.

# Conclusion

This document provided a step-by-step guide to creating a Loan Prediction App using Streamlit. By following the steps outlined above, you can create an interactive web app to predict loan approval based on user inputs. If you face any issues or need further assistance, feel free to reach out for more support.