

Here is the step-by-step blueprint on how to train your AI assistant to handle these eligibility rules.

Step 1: Write a Strict "System Prompt"

Lending Rules you MUST adhere to:

Max EMI Ratio: A client's total monthly loan payments cannot exceed 60% of their net monthly income.

Interest Rate: Assume a default interest rate of 8.0% annually.

LTV (Loan-to-Value): We only fund up to 80% of the property's cost.

Tenure: Assume a 20-year repayment period unless the user specifies otherwise.

Your Task:

Greet the user and ask for their Monthly Net Income, any Existing EMIs, and the Total Property Cost. Do not overwhelm them; ask naturally.

Once you have these three numbers, DO NOT calculate the math yourself.

Once the tool returns the result, explain the final approved loan amount to the user. Explicitly state whether their loan was restricted by their Income (EMI rule) or the Property Value (LTV rule)."

Step 2: Train it to Gather the Right Inputs

An untrained AI might ask generic questions like "Tell me about your finances." You need to train it to hunt for the specific variables required by your calculator.

Provide it with Few-Shot Examples (example conversations) so it learns how to steer the chat:

User: "I want to buy a house."

AI: "I'd love to help you figure out your budget! To see how much the bank can sanction, could you tell me your total monthly take-home salary and the approximate cost of the property you are eyeing?"

Step 3: Give the AI a "Calculator Tool" (Function Calling)

This is the most critical step. Since AI struggles with the Present Value of Annuity formulas, you must use Function Calling (available in OpenAI, Gemini, Claude, etc.).

You tell the AI: "Whenever you have the Income, EMIs, and Property Cost, stop talking and send that data to this JSON format."

JSON

```
{
  "income": 100000,
  "existing_emis": 0,
  "property_cost": 5000000
}
```

Step 4: Train it on Empathetic Delivery

Based on the buffer/bottleneck logic we discussed earlier, train the AI on how to deliver "bad" news if the client doesn't qualify for the full amount.

If restricted by LTV (Property): "Great news! Based on your income, you could comfortably afford a much larger loan. However, since the bank only finances 80% of the property value, your maximum approved loan for this specific ₹50 Lakh flat is ₹40 Lakhs. You'll need a down payment of ₹10 Lakhs."

If restricted by FOIR (Income): "You have a great property in mind! However, to ensure you aren't overburdened by monthly payments, bank rules cap your new EMI at 60% of your salary. Therefore, the maximum loan we can sanction right now is ₹25 Lakhs. We can look at extending the tenure or finding a co-applicant to increase this limit!"

Are you building this AI assistant using a specific framework (like OpenAI API, Dialogflow, or an open-source model) so I can give you the exact technical integration code for the calculator?