# 💰 Millionaire Game (Powered by Gemini AI)

A console-based \*Who Wants to Be a Millionaire\* style quiz game where questions are generated on the fly using the Gemini AI model. Players keep answering multiple choice questions until they give a wrong answer!

## 📁 Project Structure

Millionaire-Game/  
├── getmodel.py # Initializes Gemini model using your API key  
├── millionaire.py # Main game logic  
├── README.md # You're here

## 🚀 Features

* 🤖 Auto-generated multiple choice questions using Gemini AI
* 📈 Gradual increase in question difficulty
* ⛔ Game stops if the user gives the wrong answer
* 🧠 General Knowledge based questions
* ✅ Clean console-based interface
* 🔒 API key-based authentication

## 🔧 Setup Instructions

### 1. Clone the Repo

git clone https://github.com/your-username/Millionaire-Game.git  
cd Millionaire-Game

### 2. Install Dependencies

pip install google-generativeai

### 3. Setup Your API Key

1. Get your API key from the Google AI Studio (https://makersuite.google.com/).  
2. Create a .env file or directly paste your key in getmodel.py.  
  
Example in getmodel.py:  
import google.generativeai as genai  
  
genai.configure(api\_key="YOUR\_API\_KEY\_HERE")  
  
model = genai.GenerativeModel("gemini-pro")

## 🧠 How It Works

### getmodel.py

This script initializes and returns a Gemini GenerativeModel instance using your API key.

### millionaire.py

- Sends a prompt to Gemini API to generate a list of 10 questions.  
- Parses them into Python format.  
- Asks questions one-by-one.  
- Ends the game if the player gives a wrong answer.  
- Tracks score.

## 📷 Sample Gameplay

Q1: What is the capital of Japan?  
1. Beijing  
2. Seoul  
3. Tokyo  
4. Bangkok  
Enter your answer (1-4): 3  
✅ Correct!  
  
Q2: Who wrote 'Hamlet'?  
1. Shakespeare  
2. Dickens  
3. Homer  
4. Twain  
Enter your answer (1-4): 1  
✅ Correct!  
  
Q3: What is the square root of 256?  
1. 12  
2. 14  
3. 16  
4. 18  
Enter your answer (1-4): 2  
❌ Wrong answer!  
  
🎯 Game Over! Your score: 2

## 📌 To Do / Ideas

* Add lifelines (50:50, audience poll)
* Add score saving and leaderboard
* Add GUI using Tkinter or PyQt
* Add topic-based quiz mode

## 📜 License

This project is for educational purposes. Customize and extend it as you wish!

## 👨‍💻 Author

\*\*Jai Kumar Mangla\*\*  
Student at Manav Rachna International University  
✨ Passionate about AI, products, and game development