Requirements:

pip install gspread oauth2client

**🔐 Step-by-Step: Google Sheets API Access Setup (with Full Explanation)**

**🔹 Step 1: Go to Google Cloud Console**

* Visit: https://console.cloud.google.com/
* Sign in with your Google account.
* You'll land on the **Dashboard**.

**🔹 Step 2: Create a New Project**

1. In the top bar, click on the project name dropdown (next to the Google Cloud logo).
2. Click **"New Project"**.
3. Give your project a name like MathQuizProject.
4. Click **Create**.
5. Wait a few seconds until it’s ready, then **select** it.

**🔹 Step 3: Enable Google Sheets API & Google Drive API**

You need both APIs:

* **Sheets API** – to write to Google Sheets.
* **Drive API** – to access and open Sheets.

**➤ Enable Sheets API:**

1. In the left sidebar, go to **“APIs & Services > Library”**.
2. Search for **Google Sheets API**.
3. Click it → click **Enable**.

**➤ Enable Drive API:**

1. Still in **Library**, search for **Google Drive API**.
2. Click it → click **Enable**.

**🔹 Step 4: Create Service Account Credentials**

A service account is like a robot user your Python code uses to access Google Sheets.

1. Go to **“APIs & Services > Credentials”**.
2. Click **“+ Create Credentials”** → Choose **“Service Account”**.
3. Give it a name, like math-quiz-bot, and click **Create and Continue**.
4. For roles, **you can skip** (leave blank or choose “Editor” if needed) → Click **Continue**, then **Done**.

**🔹 Step 5: Create and Download JSON Key File**

1. After creating the service account, you’ll be back on the **Credentials** page.
2. Under the **"Service Accounts"** section, find your account (you may need to scroll).
3. Click the **email** (it will look like math-quiz-bot@your-project-id.iam.gserviceaccount.com).
4. Go to the **“Keys”** tab.
5. Click **“Add Key” > “Create New Key”** → Choose **JSON** → Click **Create**.
6. A .json file will download — **keep it safe**, and **do not share it**.

📝 **Place this JSON file in your Python project folder**. Rename it if you like (e.g., credentials.json).

**🔹 Step 6: Share Google Sheet with the Service Account**

Now your bot has a “login”, but it needs access to your Google Sheet.

1. Create a Google Sheet manually at sheets.google.com.
2. Give it a name like MathQuizData.
3. Click **Share** (top-right).
4. In the **“Add people and groups”** box, paste the **service account email**  
   (e.g., math-quiz-bot@your-project-id.iam.gserviceaccount.com)
5. Set role to **Editor**, then click **Send**.

**📄 Create a Google Sheet**

* Name it something like MathQuizData
* Make sure it has a first row with headers like:

Question | Correct Answer | User Answer | Result

**Final Python Code**

python

CopyEdit

import random

import gspread

from oauth2client.service\_account import ServiceAccountCredentials

# Authenticate and connect to Google Sheet

scope = ["https://spreadsheets.google.com/feeds", "https://www.googleapis.com/auth/drive"]

creds = ServiceAccountCredentials.from\_json\_keyfile\_name("your-credentials.json", scope)

client = gspread.authorize(creds)

sheet = client.open("MathQuizData").sheet1 # Use your actual sheet name

def generate\_question():

while True:

n1 = random.randint(1, 10)

n2 = random.randint(1, 10)

op = random.choice(['+', '-', '\*', '//', '%'])

if op in ['//', '%'] and n2 == 0:

continue

que = f"{n1} {op} {n2}"

ans = eval(que)

return que, ans

while True:

q, ans = generate\_question()

print(f"\nQuestion: {q}")

user\_input = input("Enter your answer (or type 'exit' to quit): ")

if user\_input.lower() == 'exit':

print("Thanks for playing!")

break

try:

user\_ans = int(user\_input)

result = "Correct" if user\_ans == ans else "Wrong"

print("🎉 7 Crore!!" if result == "Correct" else f"💸 0 Crore!! Correct answer was: {ans}")

# Store in Google Sheet

sheet.append\_row([q, ans, user\_ans, result])

except ValueError:

print("⚠️ Invalid input! Please enter a number or 'exit'.")