GOVERNMENT ARTS AND SCIENCE COLLEGE

POONTHOTAM STREET, THERADI, THIRUVOTTIYUR

CHENNAI – 600 019

Edu Tutor AI: Personalized Learning.

A NAAN MUDHALVAN PROJECT

Submitted in Partial Fulfillment for the Award of

BATCHELOR OF COMPUTER APPLICATION

BY

Team Leader: Karthika.V
Team member: Ranjani.M
Team member: Mohana.M
Team member: Priyadharshini.A

2025 - 2026

GOVERNMENT ARTS AND SCIENCE COLLEGE

POONTHOTAM STREET, THERADI, THIRUVOTTIYUR,

CHENNAI – 600019

BONAFIDE CERTIFICATE

Certified that this project report title is Edu Tutor AI: Personalized Learning.
a Bonafide record of name with register number **KARTHIKA V** of **BCA Degree** course in **Government Arts and Science College, Thiruvottiyur**, Chennai-600019, during the academic year 2025-2026.

“NAAN MUDHALVAN PROJECT”

Acknowledgement

We would like to express our deepest gratitude to everyone who has contributed to the development of this project. From the designers who created the user-friendly interface, to the developers who wrote the code and implemented the functionality, and to the testers who provided valuable feedback and helped to identify and fix bugs - your hard work and dedication have made this project a success.

We would also like to thank the Flask community for creating and maintaining the Flask framework, which provided the foundation for this project. Your contributions and support have been invaluable.

We are personally indebted to a number of people who gave us their useful insights to aid in our overall progress for this project. A complete acknowledgement would therefore be encyclopedic. First of all, we would like to give our deepest gratitude to our parents for permitting us to take up this course.

We extend our sincere gratitude to our respected **Principle Dr.V. VIJAYA M.s.c, M Phil, PHD Government Arts and Science College**, Thiruvottiyur for providing facilities in the college premises for carrying out this project work.

We express our sincere thanks **to Mrs.S.RAJALAKSHMI M.C.A, Head of the Department,Computer Application**, for her encouragement to do this project.

We express our thanks to our Internal Guide **Mrs.S.RAJALAKSHMI M.C.A**, for her encouragement and valuable guidance to complete the project successfully.

1.Introduction

Project Title : Edu Tutor AI: Personalized Learning

Team Members: Team Leader: Karthika.V
Team member: Ranjani.M
Team member: Mohana.M
Team member: Priyadharshini.A

2. Project Overview

Purpose:

EduTutor AI provides a personalized learning experience using IBM Granite models from Hugging Face. It generates concept explainers, quizzes, and other AI-powered learning tools.

Features:

Concept explanation using AI

Automatic quiz generation

Deployment via Google Colab for easy setup

Integration with Hugging Face Granite model

3. Architecture

Component Structure:

Frontend (Gradio-based interface for learners)

Backend (Granite-3.2-2b-instruct model running on Google Colab)

Data Flow: User input → Granite Model → AI-generated content → Display in UI

State Management:
Managed at runtime in Colab, with no persistent global state.

Routing:
Not applicable (single-page Gradio app).

4. Setup Instructions

Prerequisites:

Python 3

Gradio

Transformers, Torch

GitHub account

Google Colab with T4 GPU access

Installation:

1. Clone project from GitHub

2. Install dependencies:

```
!pip install transformers torch gradio -q
```

3. Configure Google Colab runtime (T4 GPU)

5. Folder Structure

Client: Gradio UI files

Utilities: Python scripts for model interaction, helper functions

6. Running the Application

Frontend:
Run the notebook in Google Colab → Open Gradio URL

7. Component Documentation

Key Components:

Gradio Interface (UI input/output)

Model Loader (Granite model from Hugging Face)

Reusable Components:

Quiz generator module

Concept explanation module

8. State Management

Global State: None (stateless interactions in Gradio.

Local State: Temporary session data for each user query.

9. User Interface

Gradio app with input text box and output display (screenshots can be added here).

10. Styling

CSS Frameworks/Libraries: Default Gradio styling

Theming: Light theme UI

11. Testing

Testing Strategy:

Manual testing on Google Colab

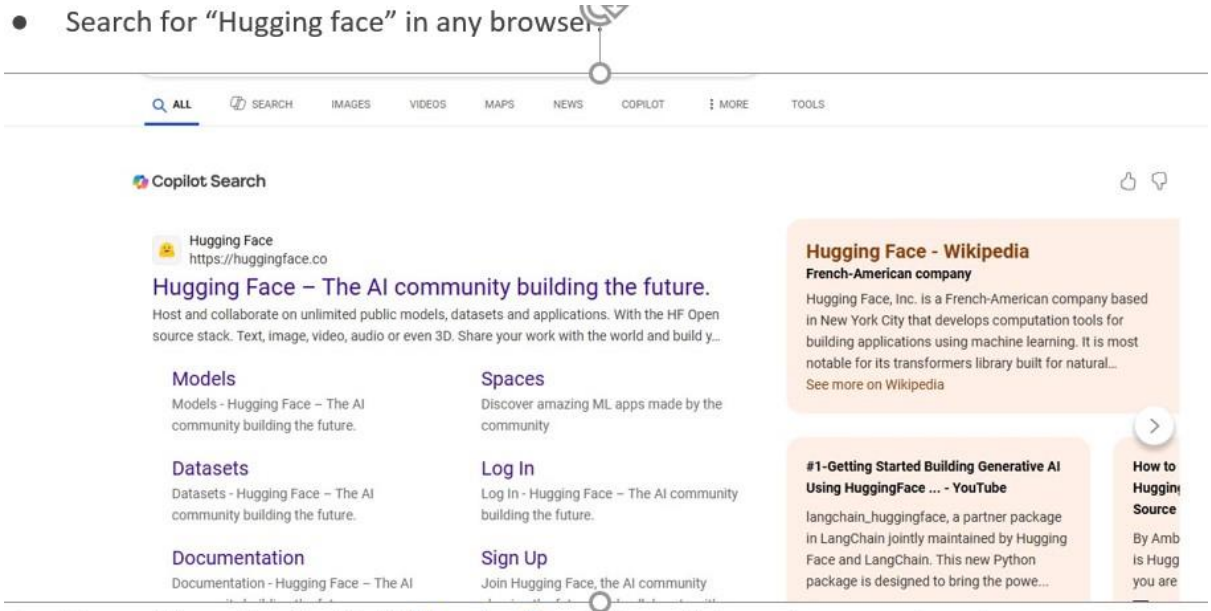
Unit testing for Python modules

Code Coverage: Limited, focused on functional testing

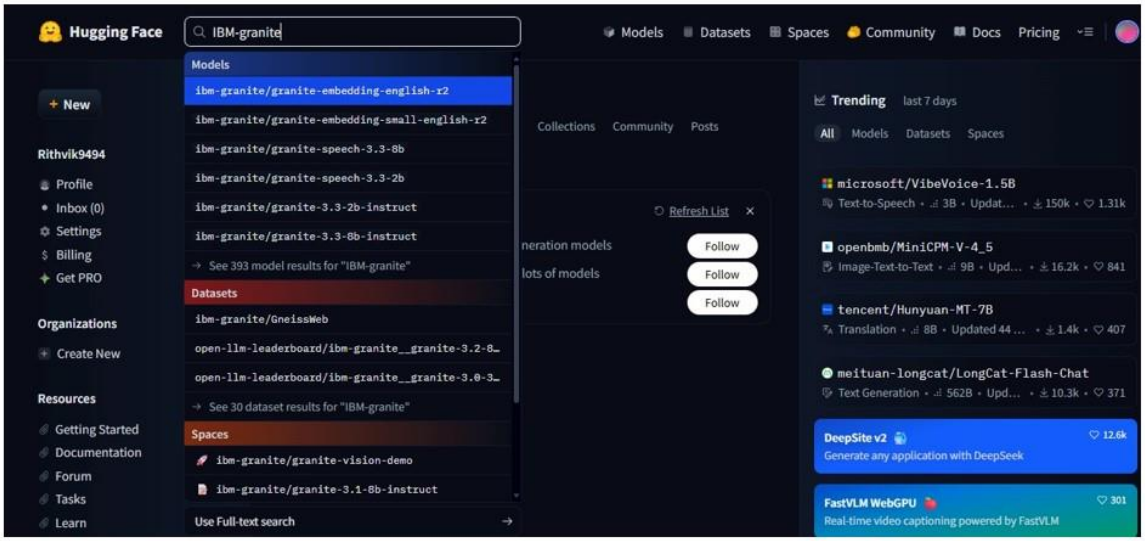
12. Screenshots or Demo

Screenshot of running Gradio

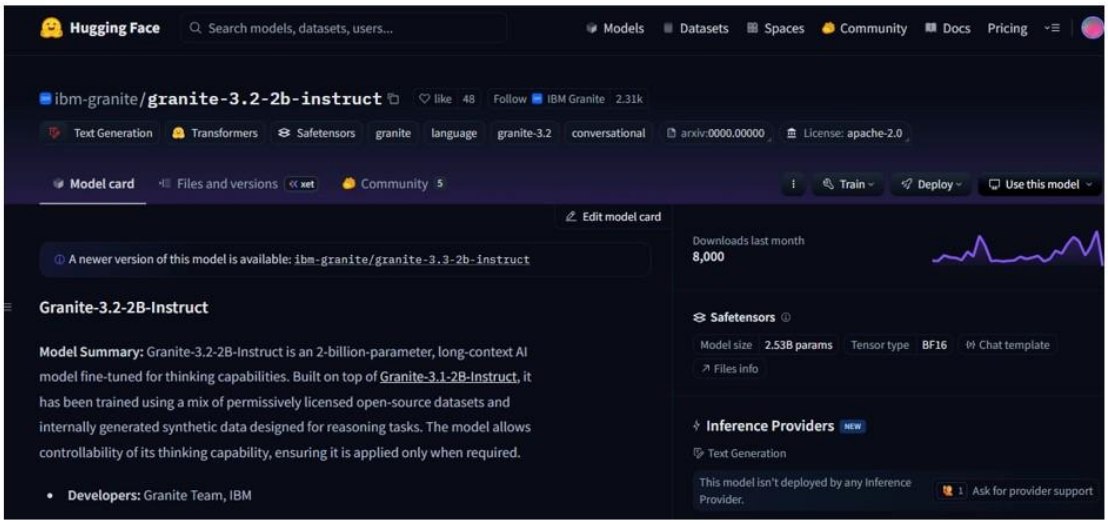
- Search for “Hugging face” in any browser.



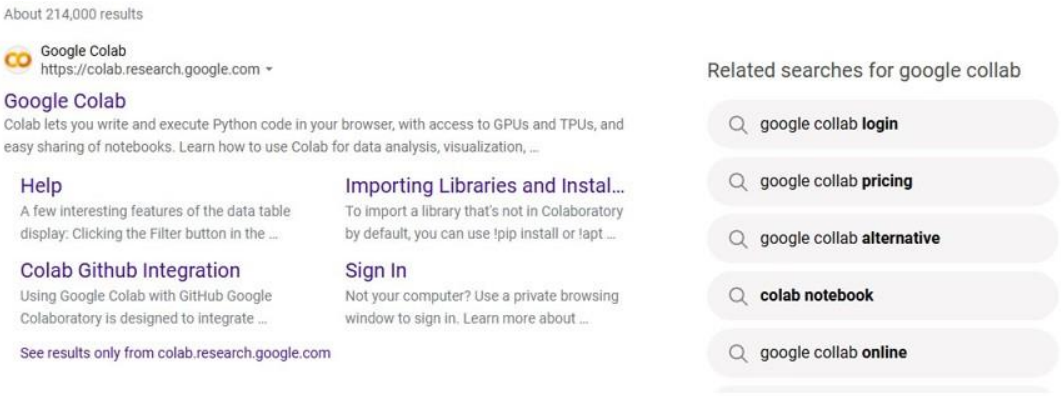
- Then click on the first link (Hugging Face), then click on signup and create your own account in Hugging Face. Then search for “IBM-Granite models” and choose any model.



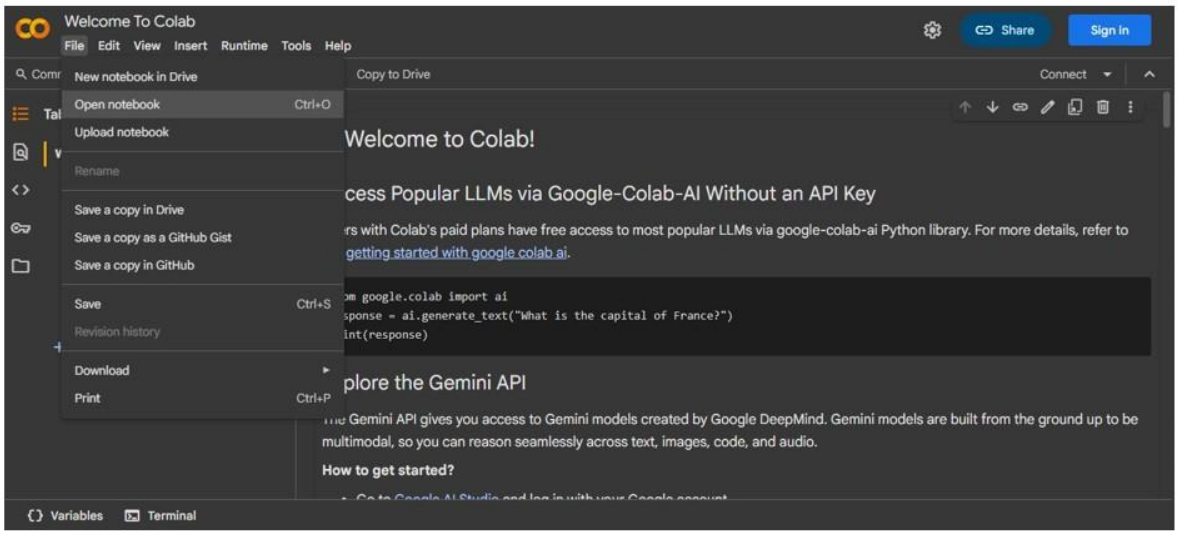
- Here for this project we are using “granite-3.2-2b-instruct” which is compatible fast and light weight.

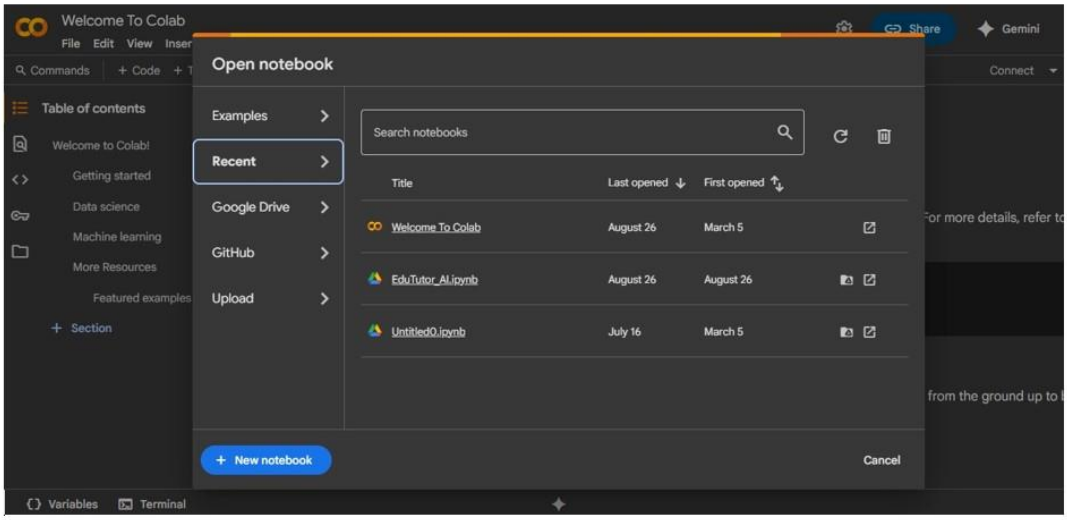


- Now we will start building our project in Google collab.

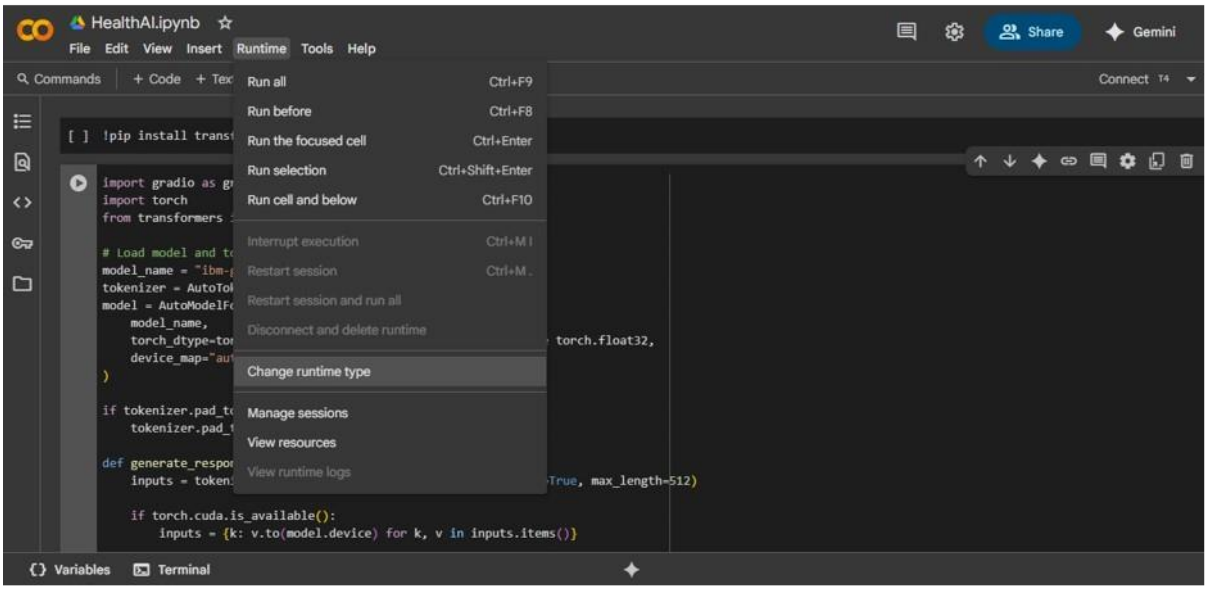


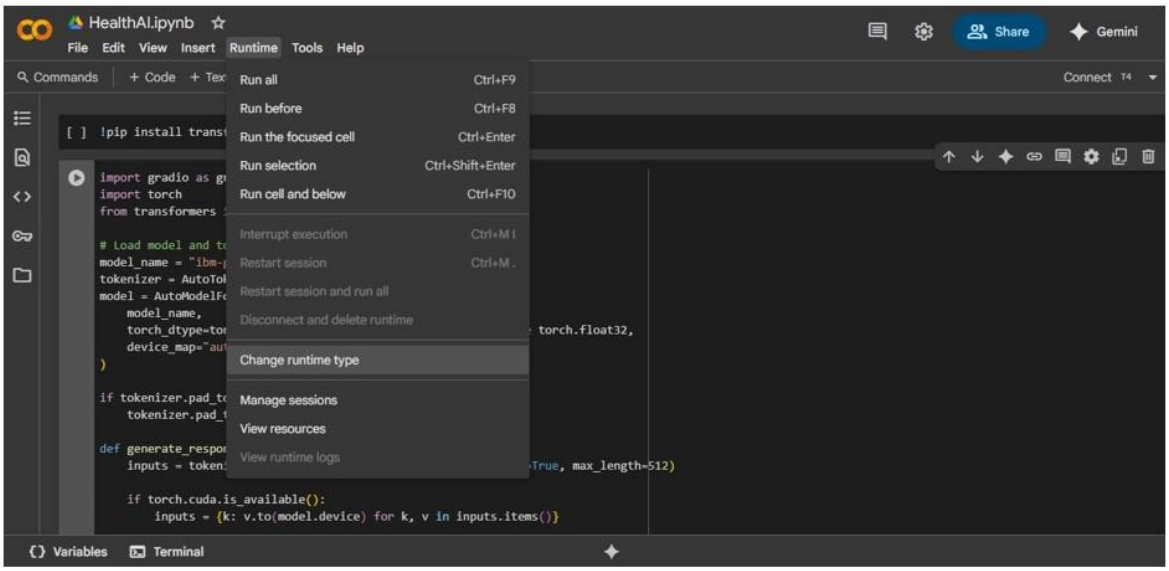
- Click on the first link (Google Colab), then click on “Files” and then “Open Notebook”.

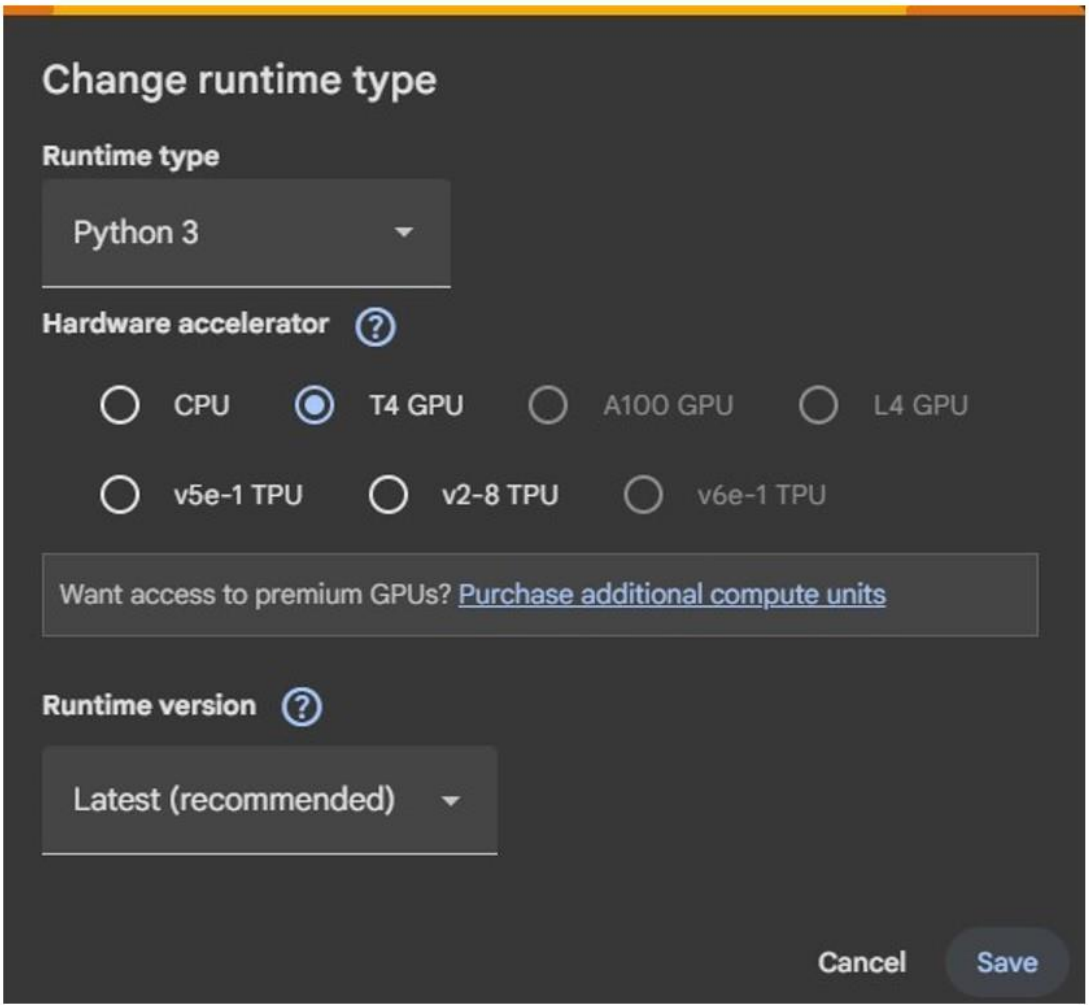




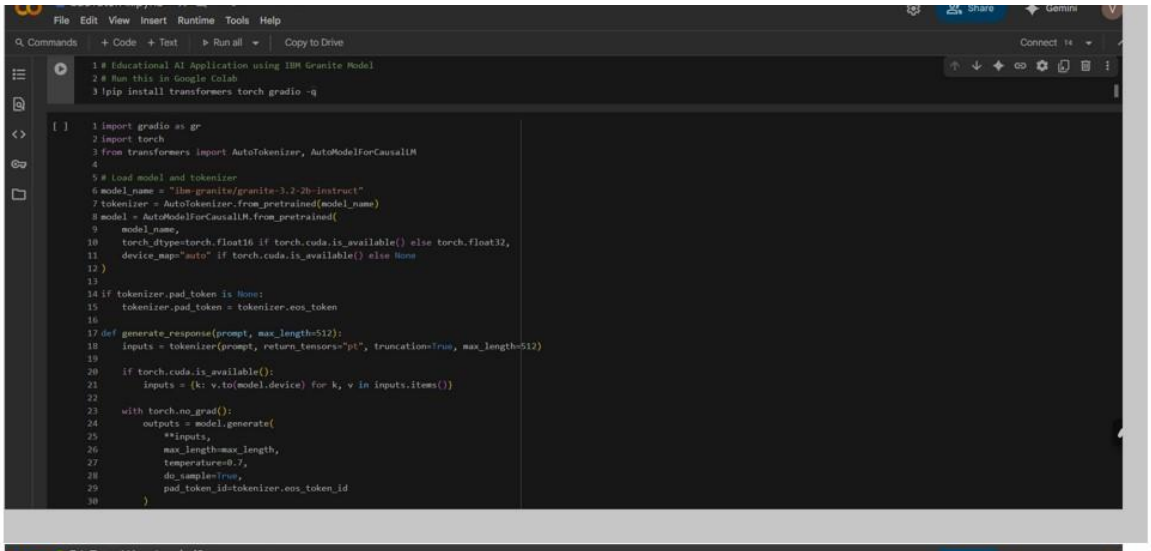
- Change the title of the notebook “Untitled” to “Health AI”. Then click on “Runtime”, then go to “Change Runtime Type”.

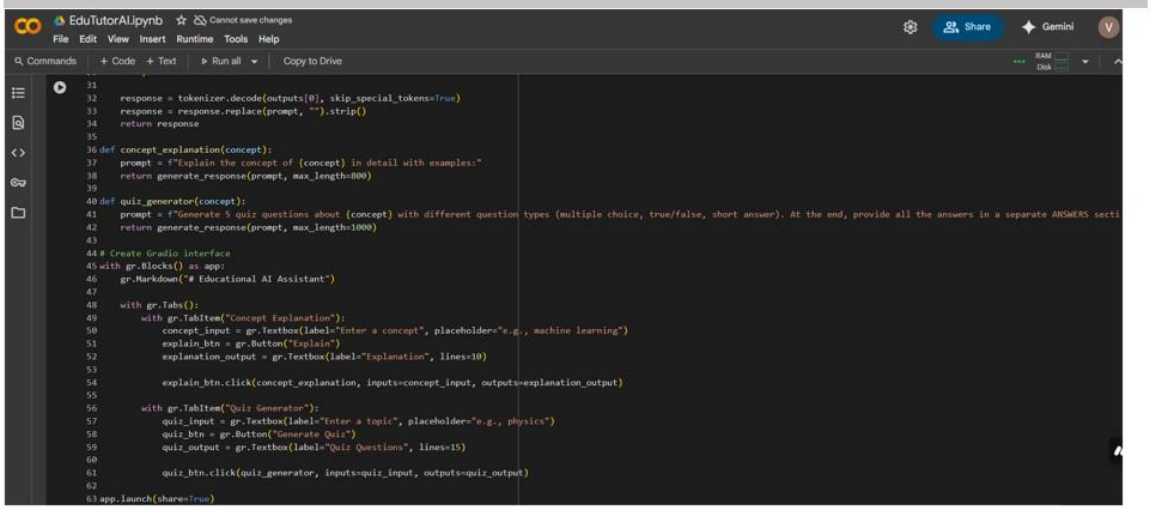



HealthAI.ipynb
File Edit View Insert Runtime Tools Help
Share
Gemini
Commands
+ Code
Connect
Run all Ctrl+F9
Run before Ctrl+F8
Run the focused cell Ctrl+Enter
Run selection Ctrl+Shift+Enter
Run cell and below Ctrl+F10
Interrupt execution Ctrl+M I
Restart session Ctrl+M .
Restart session and run all
Disconnect and delete runtime
Change runtime type
Manage sessions
View resources
View runtime logs
!pip install
import gradio as
import torch
from transformers
# Load model and
model_name = "ibm-
tokenizer = AutoTok
model = AutoModelFo
model_name,
torch_dtype=tor
device_map="au
)
if tokenizer.pad_t
tokenizer.pad_
def generate_respor
inputs = token
if torch.cuda.is_available():
inputs = {k: v.to(model.device) for k, v in inputs.items()}
torch.float32,
True, max_length=512)
Variables
Terminal

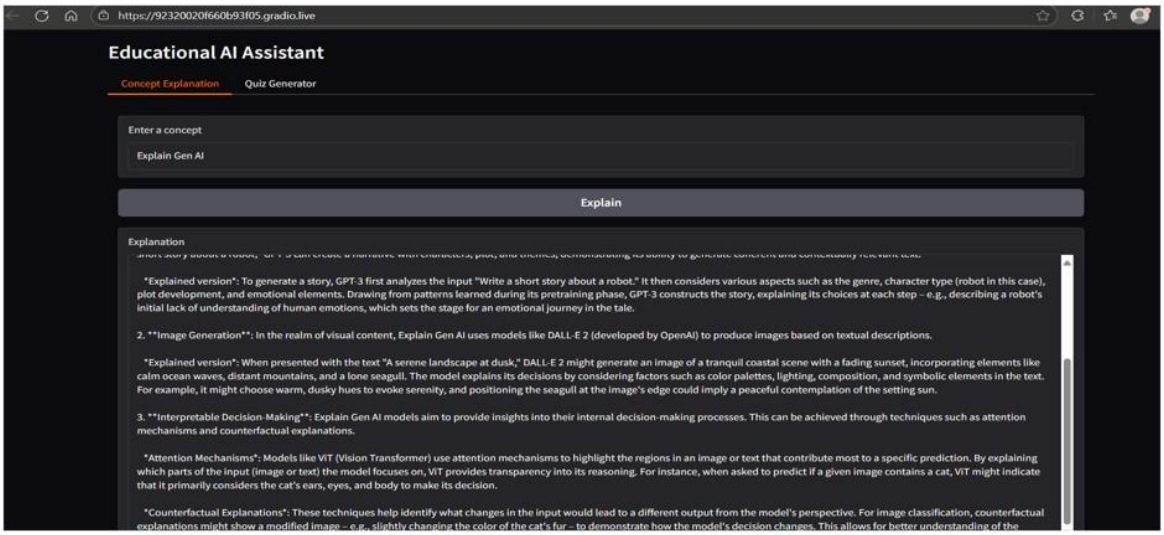

Change runtime type
Runtime type
Python 3
Hardware accelerator
CPU
T4 GPU
A100 GPU
L4 GPU
v5e-1 TPU
v2-8 TPU
v6e-1 TPU
Want access to premium GPUs? Purchase additional compute units
Runtime version
Latest (recommended)
Cancel
Save

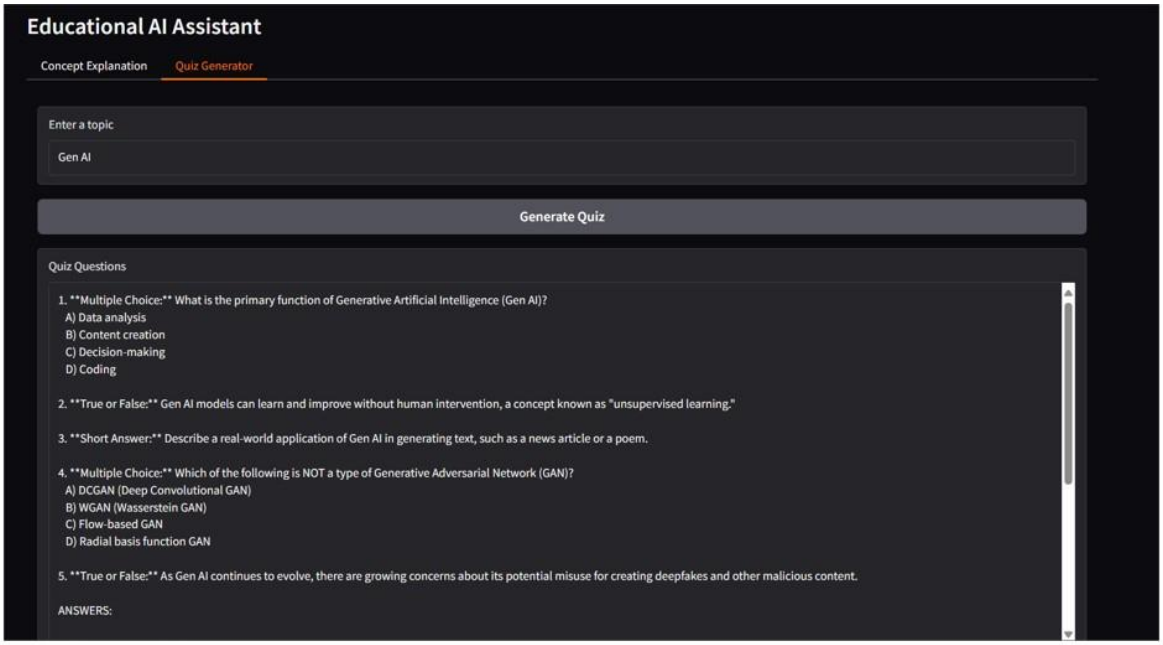
- Then run this command in the first cell "`!pip install transformers torch gradio -q`". To install the required libraries to run our application.





```
Colab notebook detected. To show errors in colab notebook, set debug=True in launch()
* Running on public URL: https://92320020f660b93f05.gradio.live
```





Demo link: https://github.com/Karthikavijayan2005/IBM-Project.git

13. Known Issues

Requires internet connection for Hugging Face and Colab

No offline deployment support

Limited UI customization in Gradio

14. Future Enhancements

Integration with LMS (Learning Management Systems)

Advanced analytics for tracking learner progress

Offline support using lightweight models

Improved UI/UX with custom styling