**EduTutor-AI: Intelligent Educational AI Assistant**

**Project Documentation**

**Team ID : NM2025TMID03663**

**Team Size : 4**

**Team Leader : POOJA J**

**Team member : PERCY JANET T S**

**Team member : PAVITHRA R**

**Team member : PABITHA G**

**Category:** Educational AI Application Development  
**Skills Required:** **Python, Hugging Face Transformers, PyTorch, Gradio**

## Project Description

**EduTutor-AI** is an intelligent educational assistant that leverages **IBM Granite LLM models** to provide students with **interactive and personalized learning support**. It enhances academic understanding through **detailed concept explanations** and **automatically generated quizzes** for practice and assessment.

The platform includes:

* **Concept Explanation:** Offers in-depth explanations of academic concepts with examples.
* **Quiz Generator:** Creates diverse quiz questions (**MCQs, True/False, Short Answers**) with answers.
* **Interactive Learning:** Engages students with **conversational AI** for personalized guidance.
* **Easy-to-use Interface:** Powered by **Gradio**, making it accessible and user-friendly.

**Scenarios**

**Concept Learning**

* **Action:** Student enters a concept (e.g., *machine learning*).
* **Outcome:** EduTutor-AI explains the concept with **examples** for better understanding.

**Quiz Practice**

* **Action:** Student requests a quiz on a topic (e.g., *physics*).
* **Outcome:** EduTutor-AI generates **multiple question types** with **correct answers**.

**Interactive Assistance**

* **Action:** Student asks **subject-related queries** via the interface.
* **Outcome:** EduTutor-AI provides **detailed, conversational, and personalized responses**.

**Technical Architecture**

**Prerequisites**

* **Python (3.8+)**
* **Hugging Face Transformers**
* **PyTorch**
* **Gradio** (for frontend UI)
* **GPU recommended** for faster inference
* **Internet connection** for model downloads

**Project Structure:**

* **edututor.py** (main app)
* **models/** (Hugging Face model checkpoints)
* **static/** (optional for styling, images)

**Project Setup & Architecture**

**Model & Libraries Selection**

* **IBM Granite 3.2 2B Instruct model**
* **Hugging Face Transformers & PyTorch**
* **Gradio** for interactive UI

**System Design** **Input → AI inference → Processed output → UI Display**

**Core Functionalities**

* **generate\_response():** Core function for AI inference
* **concept\_explanation():** Provides detailed concept explanations with examples
* **quiz\_generator():** Creates quizzes with different question types and answers

**Data Handling & Logic**

* **User inputs** are processed via the model and returned as **formatted text**.
* **Session-based interactions** supported by Gradio.
* **Lightweight storage** possible for quiz history or student progress tracking.

**Frontend Development (Gradio)**

* **Tabs** for Concept Explanation and Quiz Generator.
* **Text input fields** for student queries.
* **Interactive buttons** to trigger responses.
* **Output areas** displaying explanations or quiz sets.

**Integration & Testing**

* **Local Run:** Execute edututor.py to start Gradio app.
* **Test Flows:**
  + Enter concept → **Explanation output**
  + Enter topic → **Quiz generation**
* **Debug & refine** model prompts and UI interactions.

**Deployment**

* **Containerization:** Use **Docker** with Python base image.
* **Hosting:** Deploy on cloud platforms (**AWS, Azure, Hugging Face Spaces**).
* **SSL & Security:** Ensure **HTTPS** and API key protection.
* **Monitoring:** Track errors, performance, and usage analytics.

**Documentation & Handover**

* **README:** Setup, usage, API references.
* **User Guide:** Screenshots and descriptions of features.
* **Demo Video:** Walkthrough of EduTutor-AI functionalities.

**EduTutor-AI provides an end-to-end intelligent learning assistant, enabling students to grasp complex concepts easily and practice with AI-generated quizzes, fostering a deeper and more engaging educational experience. In addition, it promotes personalized learning by adapting explanations and quizzes to different subject areas and complexity levels. It encourages self-paced study, allowing learners to revisit concepts and practice until mastery is achieved. The system also enhances critical thinking and problem-solving skills through diverse quiz formats, while its interactive design keeps students motivated and engaged. Furthermore, EduTutor-AI can serve as a valuable support tool for teachers, helping them create supplementary learning materials quickly and efficiently.**

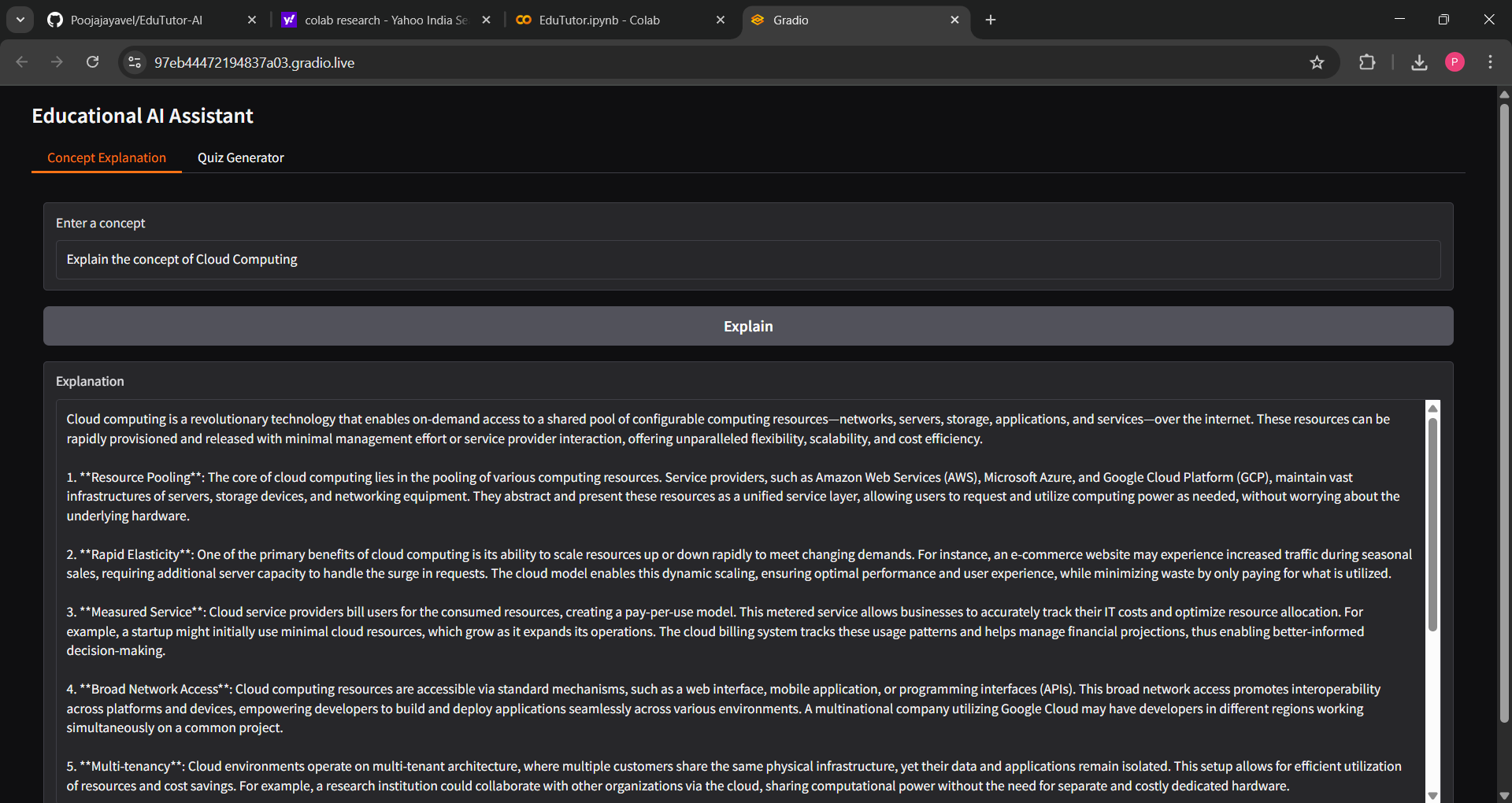
**Project Output**

Below are examples of the EduTutor-AI project outputs:

**1. Concept Explanation Tab**

**Input: *Explain the concept of Cloud Computing***

**Output:  
 The system provides a detailed explanation with key points such as Resource Pooling, Rapid Elasticity, Measured Service, Broad Network Access, and Multi-tenancy. It also highlights On-Demand Self-Service, where users can provision computing resources automatically without human intervention, and Cost Efficiency, since the pay-as-you-go pricing model reduces upfront infrastructure costs and supports scalable usage.**

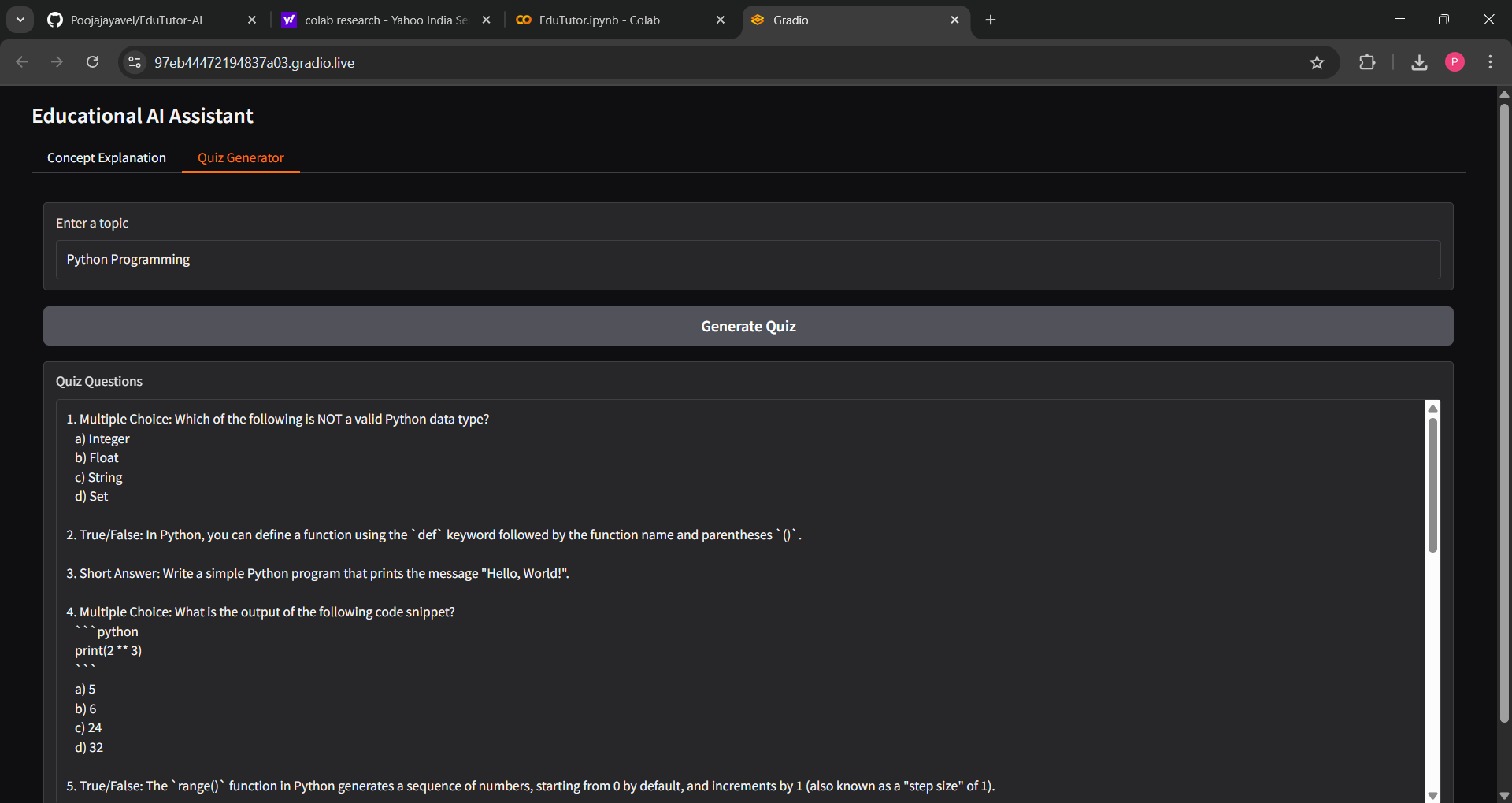
****

****

**2. Quiz Generator Tab**

**Input:** *Generate quiz on Python Programming* **Output:** The system generates **multiple question types** (MCQs, True/False, Short Answer). For example:

1. **Multiple Choice:** Which of the following is NOT a valid Python data type?  
    a) Integer b) Float c) String d) Set  
    **Answer: d) Set**
2. **True/False:** In Python, you can define a function using the def keyword.  
    **Answer: True**





**Conclusion**

The **EduTutor-AI project** has been an **enriching development experience**, allowing the integration of **advanced AI models** into an **educational setting**. By leveraging **IBM Granite LLM** and **Hugging Face Transformers**, the system successfully provides students with **detailed concept explanations** and **diverse quiz generation**, thereby **enhancing the learning process**.

Through this project, **key skills** in **Python programming, AI model integration, and user interface development with Gradio** were applied. The project demonstrates how **AI can be utilized** to create **interactive, engaging, and personalized educational tools** that benefit both learners and educators.

Overall, **EduTutor-AI contributes to the vision of AI-driven learning assistance**, offering a **scalable and user-friendly solution** to support students in **acquiring knowledge and practicing skills effectively**.