Citizen AI: Intelligent Citizen Engagement Platform

Generative AI with IBM

Team members

Karthikeyan.M
Kailashnathan.T
Mohammed Faizal.S
Mukesh.A

Introduction of Citizen AI

Citizen AI refers to the idea of developing artificial intelligence systems that act responsibly, ethically, and as productive members of society, much like a good citizen. The concept goes beyond just building powerful AI tools; it emphasizes embedding values such as fairness, accountability, transparency, and inclusivity into AI systems so they can contribute positively to human life.

With the increasing role of AI in everyday decision-making—whether in healthcare, education, governance, or business—there is a pressing need to ensure that these systems behave in ways that align with human rights, laws, and societal norms. Citizen AI encourages organizations, governments, and developers to treat AI not merely as a technological tool but as a social entity that should uphold civic responsibilities.

The main goals of Citizen AI include:

- Promoting trust and fairness in AI decisions.
- Avoiding biases and discrimination.
- Ensuring transparency in how AI works.
- Supporting responsible innovation for the benefit of society.

In short, Citizen AI is about making AI not just smart but also responsible, ensuring it acts as a positive force for individuals, communities, and the world at large.

Project Description:

Citizen AI uses the Granite model from Hugging Face to give quick, helpful answers about government services and civic issues. It tracks public sentiment and shows simple dashboards for officials to see feedback. This project will be deployed in Google Colab using Granite for easy, low-cost setup and reliable performance.

Pre-requisites:

1. Gradio Framework Knowledge: Gradio Documentation
2. IBM Granite Models (Hugging Face): IBM Granite models
3. Python Programming Proficiency: Python Documentation
4. Version Control with Git: Git Documentation
5. Google Collab's T4 GPU Knowledge: Google collab

Project Workflow:

Activity-1: Exploring Naan Mudhalavan Smart Interz Portal.

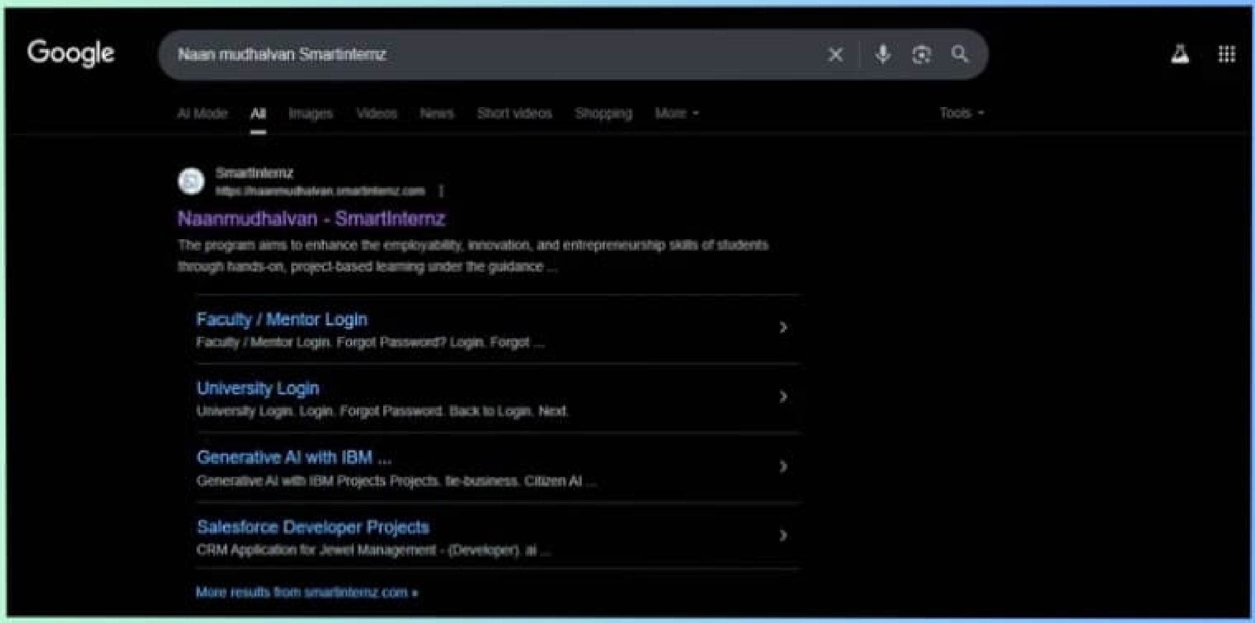
Activity-2: Choosing a IBM Granite Model From Hugging Face.

Activity-3: Running Application In Google Colab. Activity-4:

Upload your Project in Github.

Activity-1: Exploring Naan Mudhalavan Smart Interz Portal.

- Search for “Naan Mudhalavan Smart Interz” Portal in any Browser.



Smartinternz
https://naanmudhalvan.smartinternz.com

Naanmudhalvan - SmartInternz

The program aims to enhance the employability, innovation, and entrepreneurship skills of students through hands-on, project-based learning under the guidance ...

Faculty / Mentor Login
Faculty / Mentor Login. Forgot Password? Login. Forgot ...

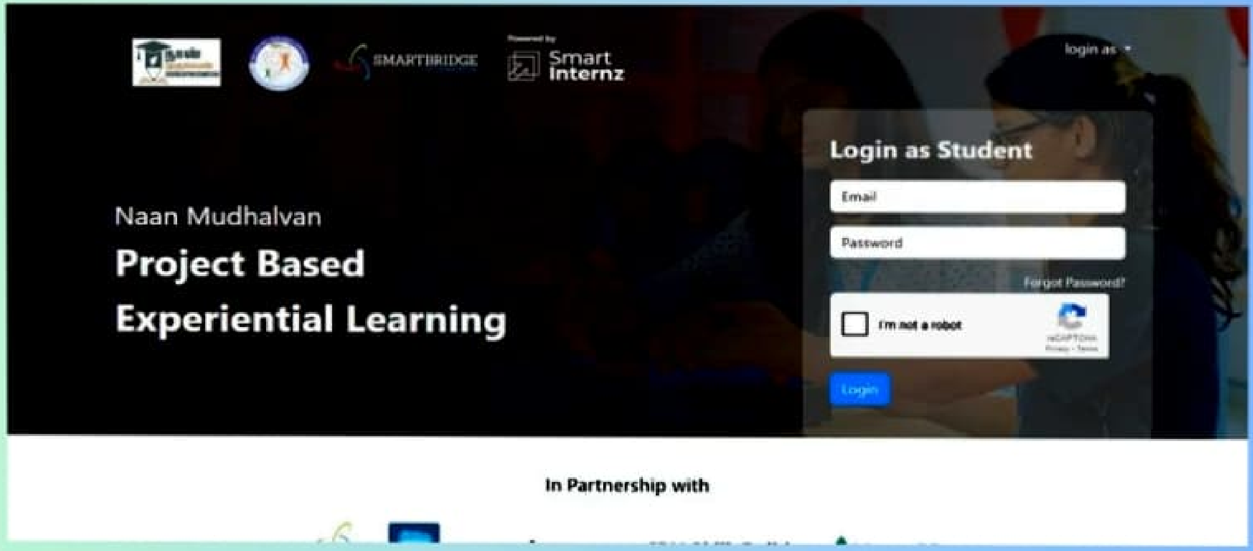
University Login
University Login. Login. Forgot Password. Back to Login. Next.

Generative AI with IBM ...
Generative AI with IBM Projects Projects. tie-business. Citizen AI ...

Salesforce Developer Projects
CRM Application for Jewel Management - (Developer). ai ...

More results from smartinternz.com »

- Then Click on the first link. ([Naanmudhalvan Smartinternz]) Then login with your details.



Naan Mudhalvan

Project Based

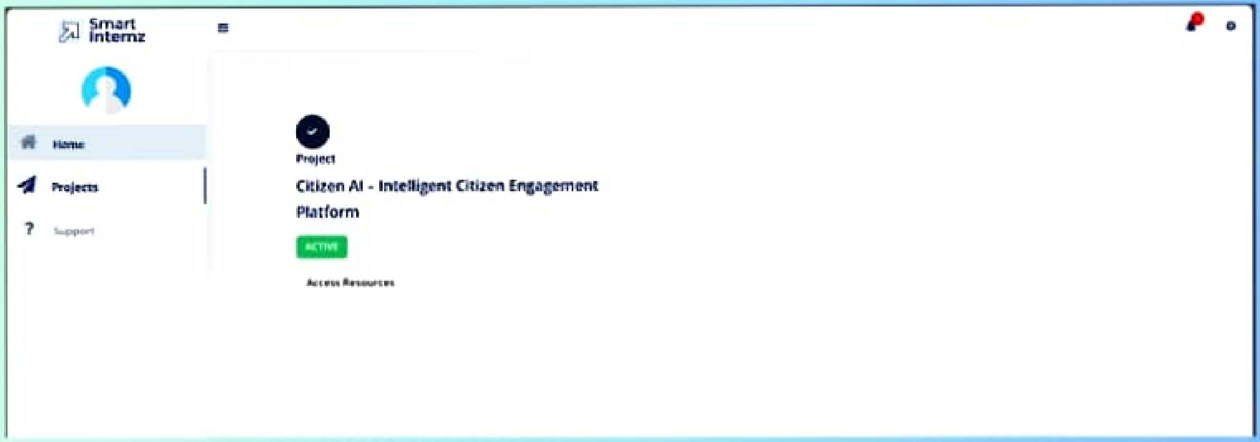
Experiential Learning

Login as Student

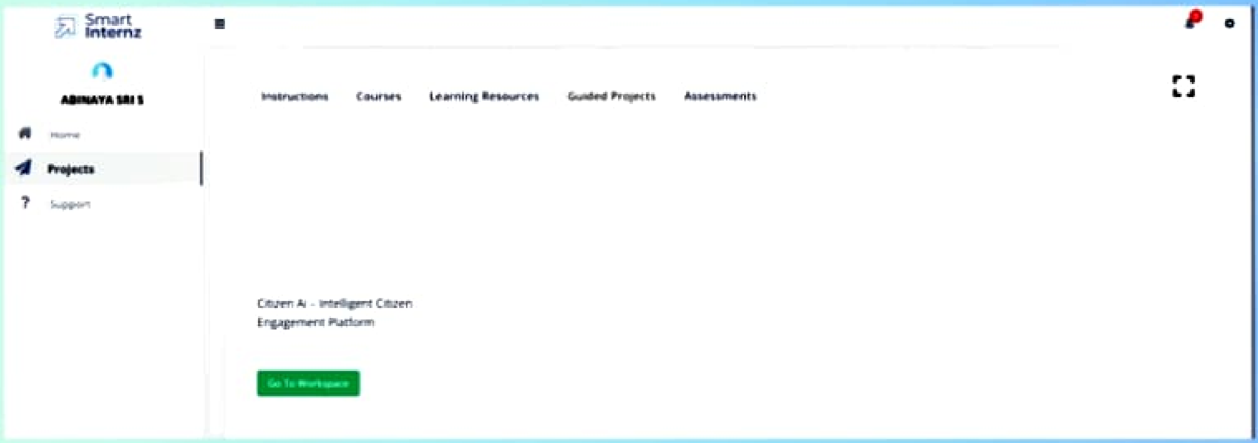
In Partnership with

- Then you will be redirected to your account then click on “Projects”

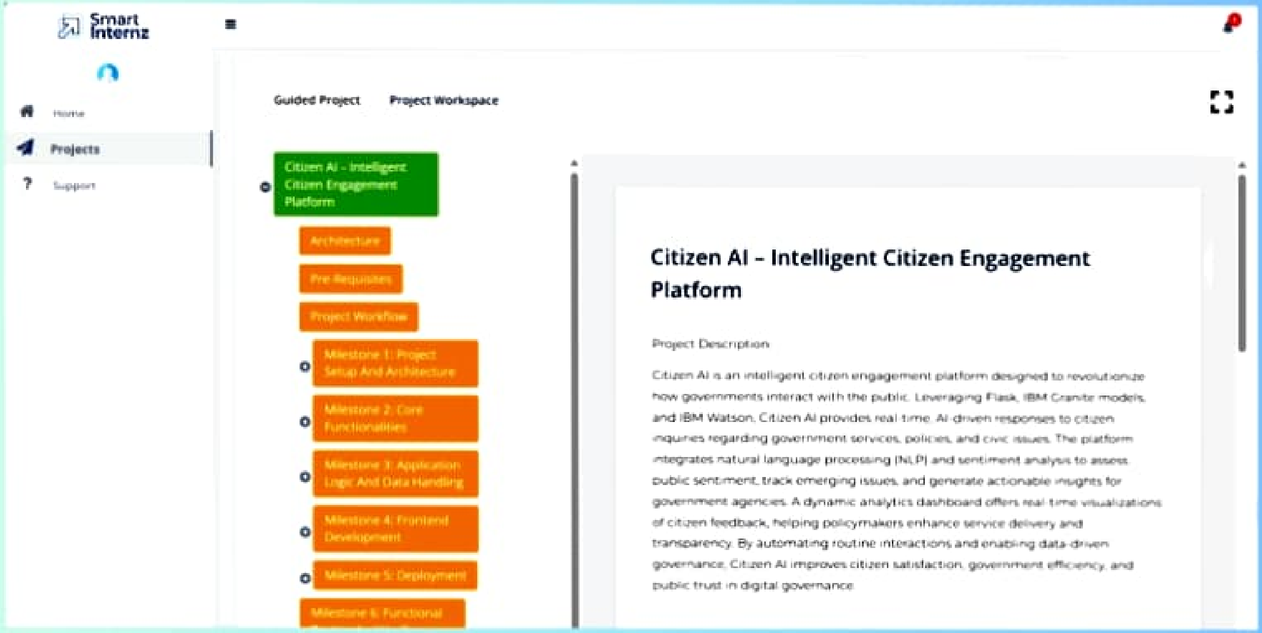
Section. There you can see which project you have enrolled in here it is "Citizen AI".



- Then click on "Access Resources" and go to the "Guided Project" Section.



- Click on the "Go to workspace" section. Then you can find the detailed explanation of Generative AI Project using IBM Watsonx API key.

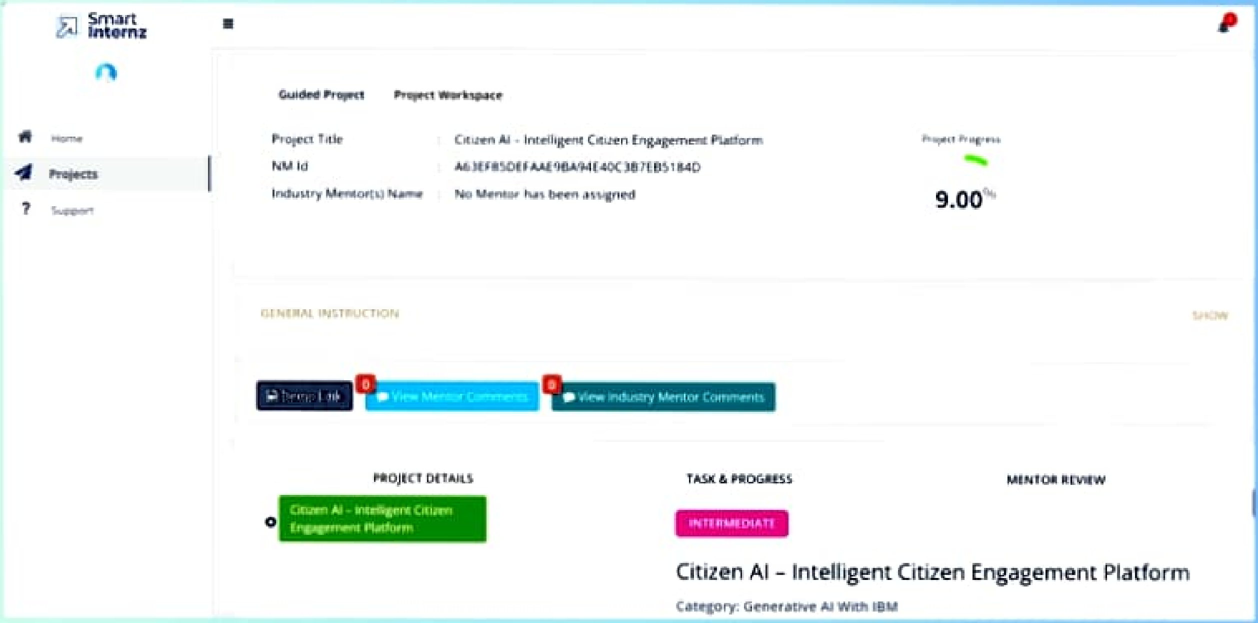

Smart Internz
Home
Projects
Support
Guided Project
Project Workspace
Citizen AI - Intelligent Citizen Engagement Platform
Architecture
Pre Requisites
Project Workflow
Milestone 1: Project Setup And Architecture
Milestone 2: Core Functionalities
Milestone 3: Application Logic And Data Handling
Milestone 4: Frontend Development
Milestone 5: Deployment
Milestone 6: Functional

Citizen AI – Intelligent Citizen Engagement Platform

Project Description

Citizen AI is an intelligent citizen engagement platform designed to revolutionize how governments interact with the public. Leveraging Flask, IBM Granite models, and IBM Watson, Citizen AI provides real-time, AI-driven responses to citizen inquiries regarding government services, policies, and civic issues. The platform integrates natural language processing (NLP) and sentiment analysis to assess public sentiment, track emerging issues, and generate actionable insights for government agencies. A dynamic analytics dashboard offers real-time visualizations of citizen feedback, helping policymakers enhance service delivery and transparency. By automating routine interactions and enabling data-driven governance, Citizen AI improves citizen satisfaction, government efficiency, and public trust in digital governance.

- Click on “Project Workspace”, there you can find your project progress and Place to upload “Demo link”.



Project Title	:	Citizen AI - Intelligent Citizen Engagement Platform
NM Id	:	A63EF85DEFAAE9BA94E40C3B7EB5184D
Industry Mentor(s) Name	:	No Mentor has been assigned

9.00%

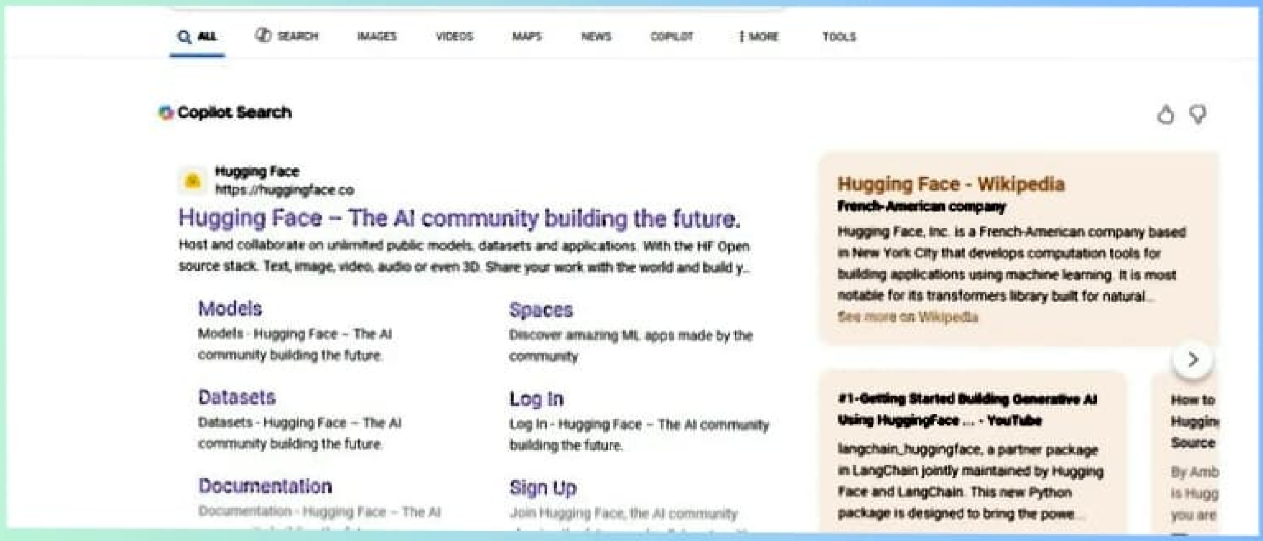
Citizen AI – Intelligent Citizen Engagement Platform

Category: Generative AI With IBM

- Now we have gone through portal understanding, now lets find a IBM granite model from hugging face to integrate in our project.

Activity-2: Choose a IBM Granite model From Hugging Face.

- Search for “Hugging face” in any browser.



Hugging Face
https://huggingface.co
Hugging Face – The AI community building the future.
Host and collaborate on unlimited public models, datasets and applications. With the HF Open source stack. Text, image, video, audio or even 3D. Share your work with the world and build y...

Models
Models · Hugging Face – The AI community building the future.

Spaces
Discover amazing ML apps made by the community

Datasets
Datasets · Hugging Face – The AI community building the future.

Log In
Log In · Hugging Face – The AI community building the future.

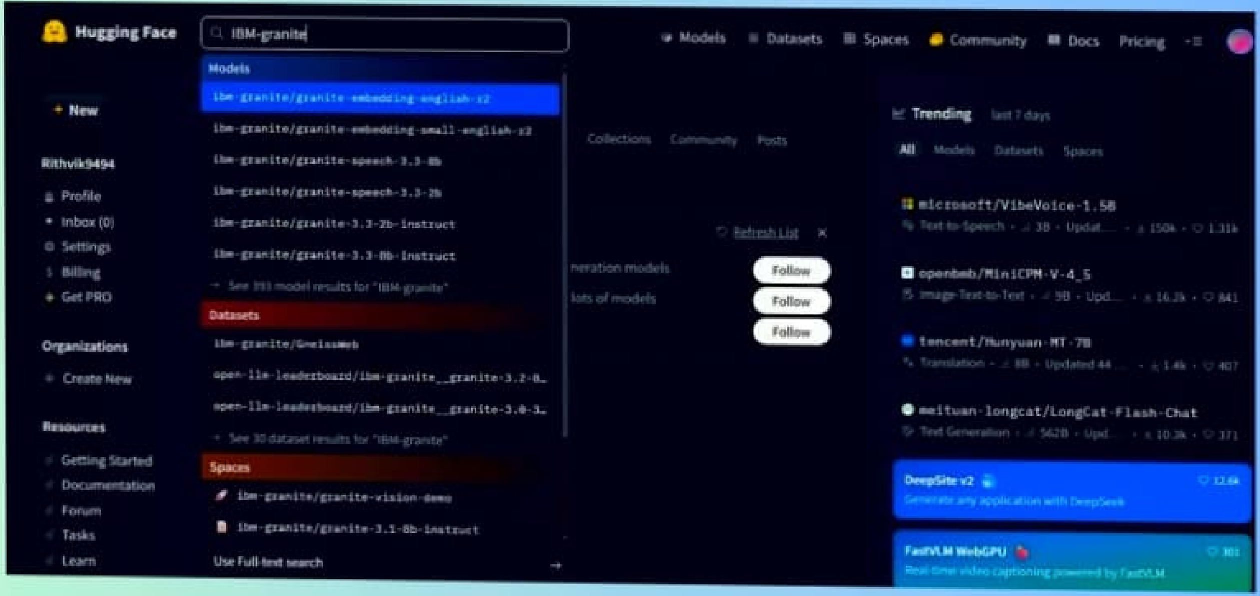
Documentation
Documentation · Hugging Face – The AI

Sign Up
Join Hugging Face, the AI community

Hugging Face - Wikipedia
French-American company
Hugging Face, Inc. is a French-American company based in New York City that develops computation tools for building applications using machine learning. It is most notable for its transformers library built for natural...
See more on Wikipedia

#1-Getting Started Building Generative AI Using HuggingFace ... - YouTube
langchain_huggingface, a partner package in LangChain jointly maintained by Hugging Face and LangChain. This new Python package is designed to bring the powe...

- 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.



IBM-granite

Models
ibm-granite/granite-embedding-english-r2
ibm-granite/granite-embedding-small-english-r2
ibm-granite/granite-speech-3.3-8b
ibm-granite/granite-speech-3.3-2b
ibm-granite/granite-3.3-2b-instruct
ibm-granite/granite-3.3-8b-instruct
See 393 model results for "IBM-granite"

Datasets
ibm-granite/GneissWeb
open-llm-leaderboard/ibm-granite__granite-3.2-8...
open-llm-leaderboard/ibm-granite__granite-3.0-3...
See 30 dataset results for "IBM-granite"

Spaces
ibm-granite/granite-vision-demo
ibm-granite/granite-3.1-8b-instruct

Use Full-text search

microsoft/VibeVoice-1.5B
Text-to-Speech · 3B · Updat... · 150k · 1.31k

openbmb/MiniCPM-V-4_5
Image-Text-to-Text · 9B · Upd... · 16.2k · 841

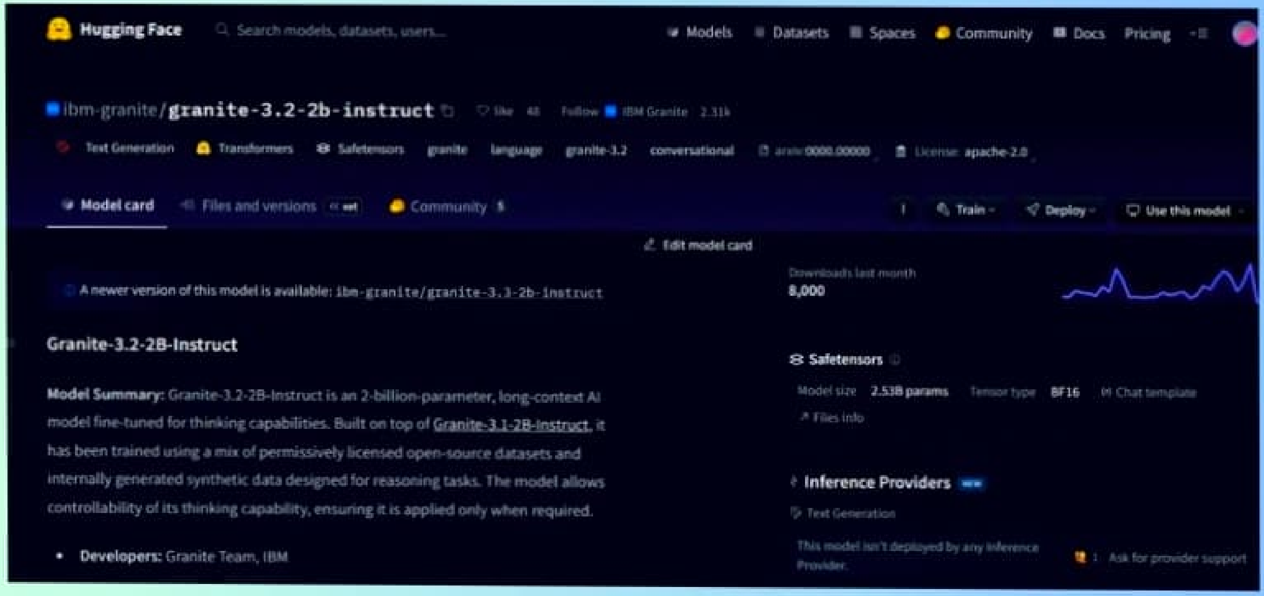
tencent/Hunyuan-MT-7B
Translation · 8B · Updated 44 ... · 1.4k · 407

meituan-longcat/LongCat-Flash-Chat
Text Generation · 562B · Upd... · 10.3k · 371

DeepSite v2
Generate any application with DeepSeek
12.6k

FastVLM WebGPU
Real-time video captioning powered by FastVLM
301

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



A newer version of this model is available: ibm-granite/granite-3.3-2b-instruct

Granite-3.2-2B-Instruct

Model Summary: Granite-3.2-2B-Instruct is an 2-billion-parameter, long-context AI model fine-tuned for thinking capabilities. Built on top of Granite-3.1-2B-Instruct, it has been trained using a mix of permissively licensed open-source datasets and internally generated synthetic data designed for reasoning tasks. The model allows controllability of its thinking capability, ensuring it is applied only when required.

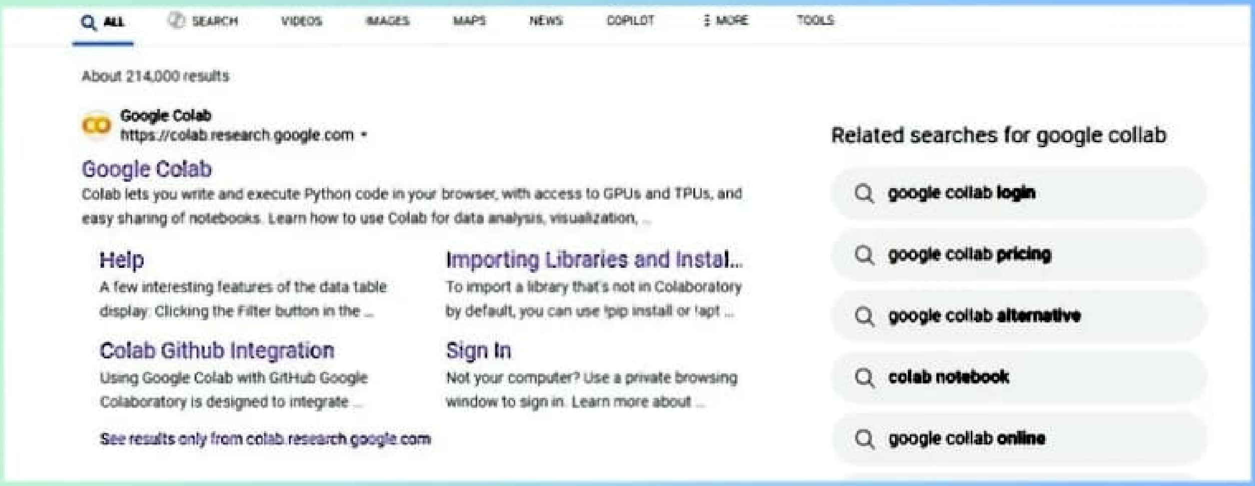
- **Developers:** Granite Team, IBM

This model isn't deployed by any Inference Provider.

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

Activity-3: Running Application in Google Collab.

- Search for "Google collab" in any browser.



About 214,000 results

Google Colab
https://colab.research.google.com

Google Colab
Colab lets you write and execute Python code in your browser, with access to GPUs and TPUs, and easy sharing of notebooks. Learn how to use Colab for data analysis, visualization, ...

Help
A few interesting features of the data table display: Clicking the Filter button in the ...

Importing Libraries and Instal...
To import a library that's not in Colaboratory by default, you can use !pip install or !apt ...

Colab Github Integration
Using Google Colab with GitHub Google Colaboratory is designed to integrate ...

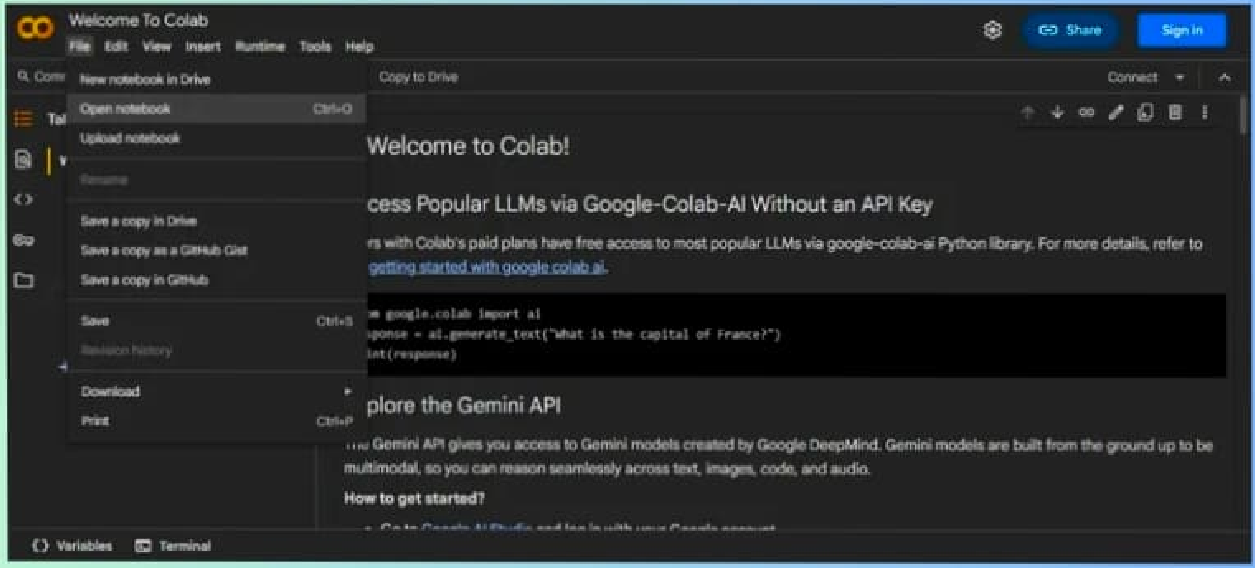
Sign In
Not your computer? Use a private browsing window to sign in. Learn more about ...

See results only from colab.research.google.com

Related searches for google collab

- google collab login
- google collab pricing
- google collab alternative
- colab notebook
- google collab online

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



Welcome to Colab!

cess Popular LLMs via Google-Colab-AI Without an API Key

rs with Colab's paid plans have free access to most popular LLMs via google-colab-ai Python library. For more details, refer to getting started with google colab ai.

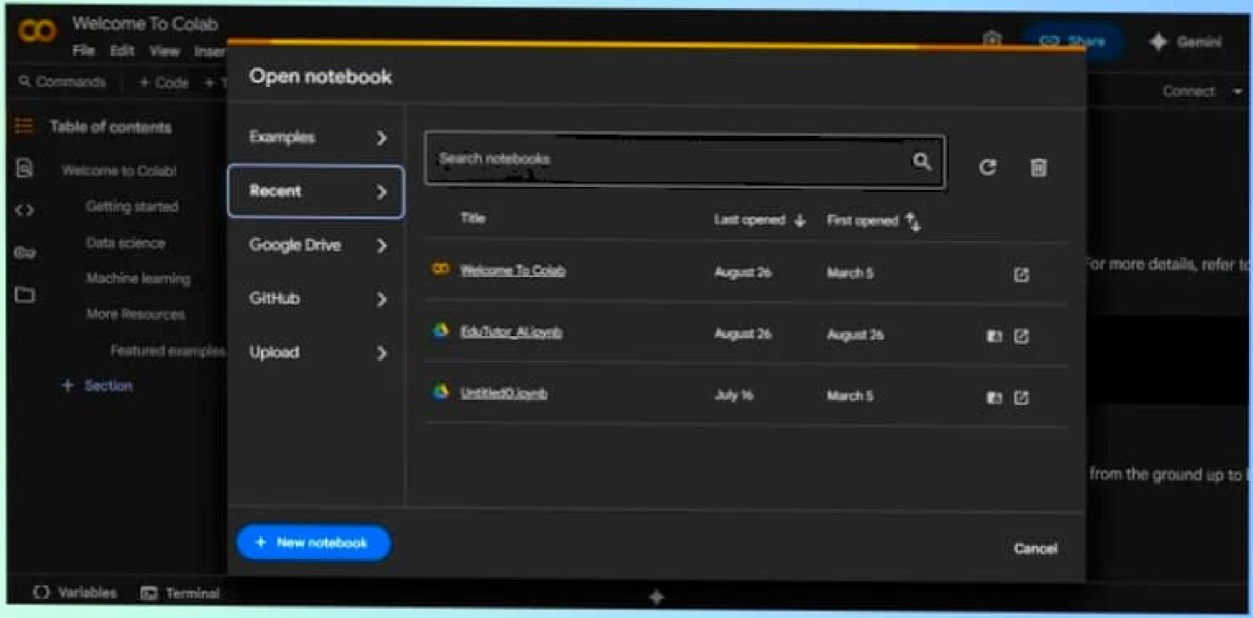
```
m google.colab import ai
sponse = ai.generate_text("What is the capital of France?")
int(response)
```

plore the Gemini API

The Gemini API gives you access to Gemini models created by Google DeepMind. Gemini models are built from the ground up to be multimodal, so you can reason seamlessly across text, images, code, and audio.

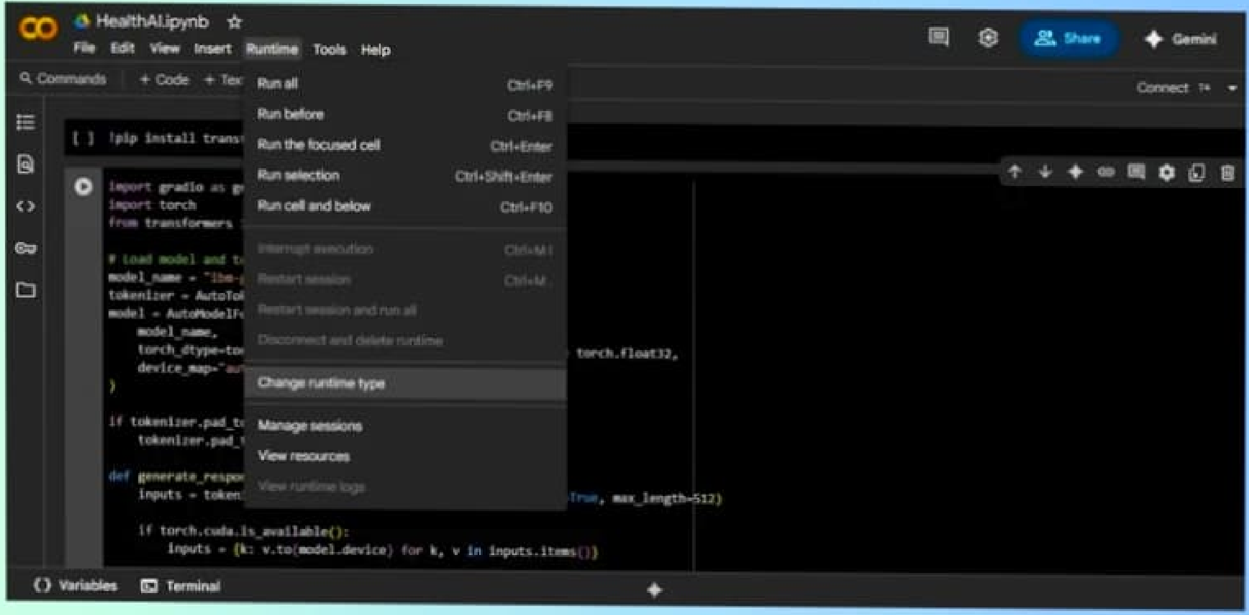
How to get started?

- Click on “New Notebook”



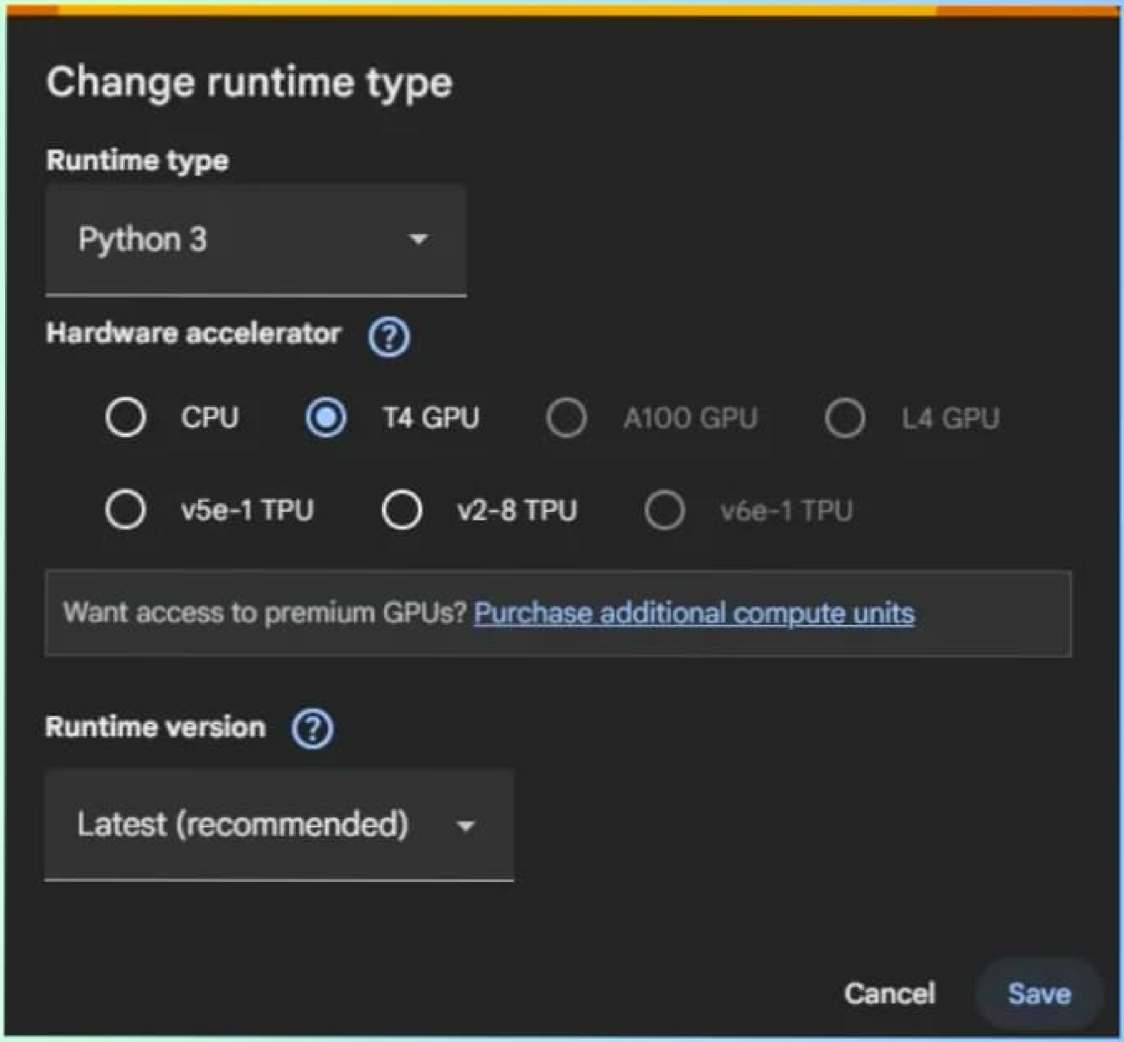
Title	Last opened	First opened
Welcome To Colab	August 26	March 5
EduTutor_AI.ipynb	August 26	August 26
Untitled0.ipynb	July 16	March 5

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


HealthAI.ipynb
File Edit View Insert Runtime Tools Help
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 trans

import gradio as g
import torch
from transformers

# Load model and t
model_name = "ibm-
tokenizer = AutoTo
model = AutoModelF
    model_name,
    torch_dtype=to                          torch.float32,
    device_map="au
)

if tokenizer.pad_t
    tokenizer.pad_

def generate_respo
    inputs = token                          True, max_length=512)

    if torch.cuda.is_available():
        inputs = {k: v.to(model.device) for k, v in inputs.items()}
```

- Choose “T4 GPU” and click on “Save”


Change runtime type

Runtime type

Python 3

Hardware accelerator

CPU [✓ T4 GPU] A100 GPU L4 GPU

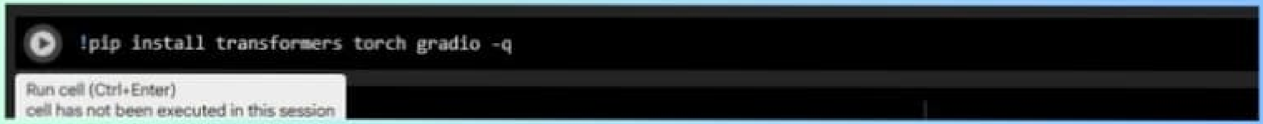
v5e-1 TPU v2-8 TPU v6e-1 TPU

Want access to premium GPUs? Purchase additional compute units

Runtime version

Latest (recommended)

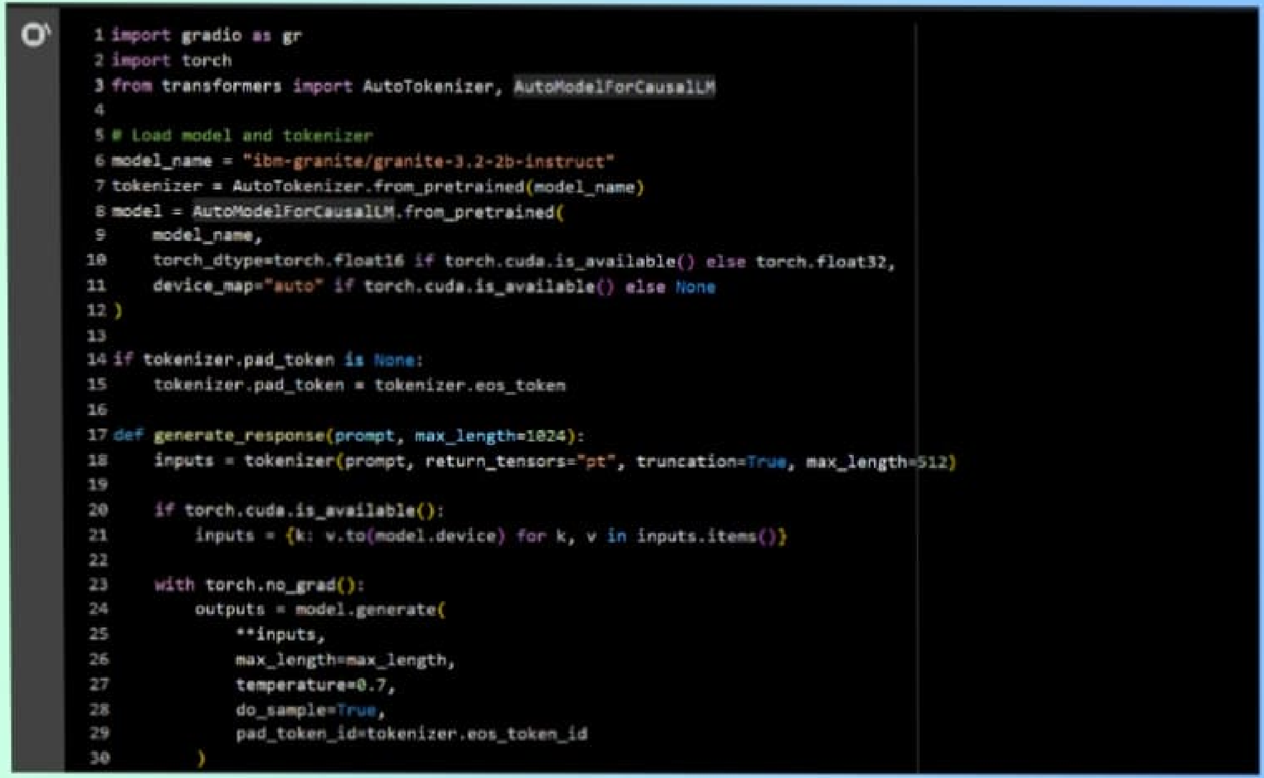
- Then run this command in the first cell “!pipinstalltransformerstorch gradio -q”. To install the required libraries to run our application.



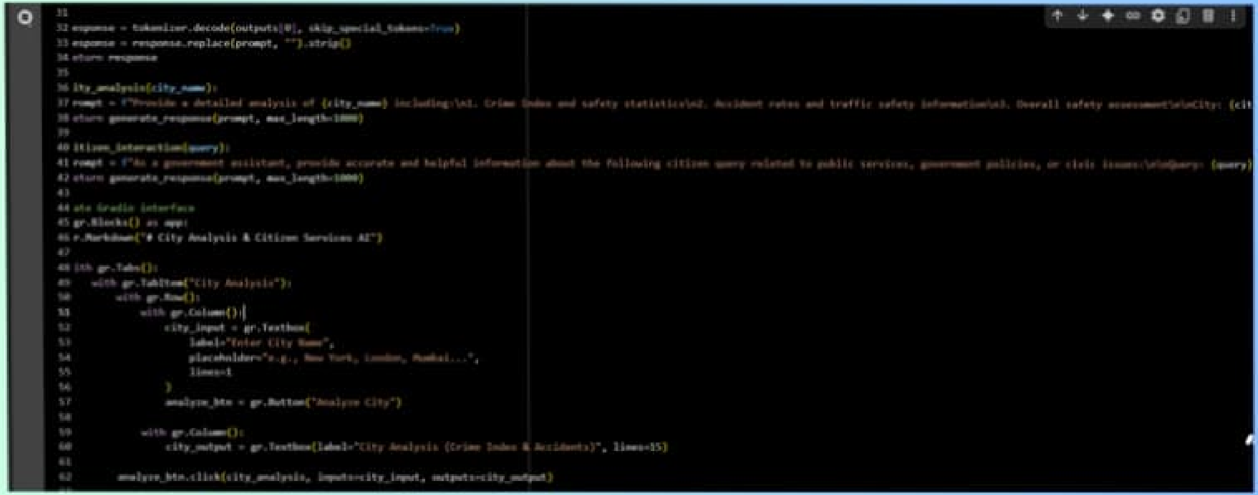
```
!pip install transformers torch gradio -q
```

Run cell (Ctrl+Enter)
cell has not been executed in this session

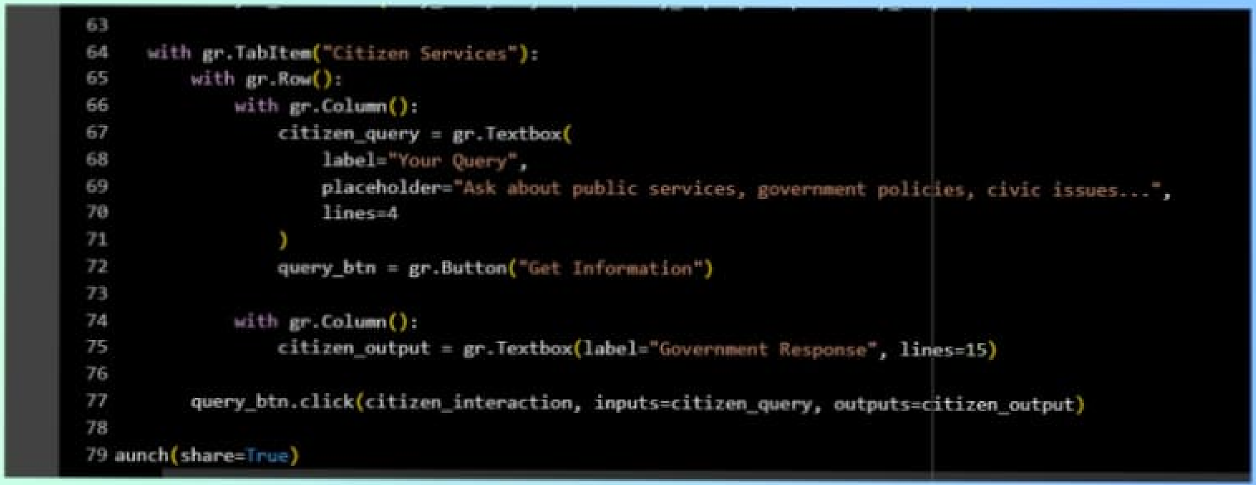
- Then run the rest of the code in the next cell.



```python
import gradio as gr
import torch
from transformers import AutoTokenizer, AutoModelForCausalLM

# Load model and tokenizer
model_name = "ibm-granite/granite-3.2-2b-instruct"
tokenizer = AutoTokenizer.from_pretrained(model_name)
model = AutoModelForCausalLM.from_pretrained(
    model_name,
    torch_dtype=torch.float16 if torch.cuda.is_available() else torch.float32,
    device_map="auto" if torch.cuda.is_available() else None
)

if tokenizer.pad_token is None:
    tokenizer.pad_token = tokenizer.eos_token

def generate_response(prompt, max_length=1024):
    inputs = tokenizer(prompt, return_tensors="pt", truncation=True, max_length=512)

    if torch.cuda.is_available():
        inputs = {k: v.to(model.device) for k, v in inputs.items()}

    with torch.no_grad():
        outputs = model.generate(
            **inputs,
            max_length=max_length,
            temperature=0.7,
            do_sample=True,
            pad_token_id=tokenizer.eos_token_id
        )
```



```python

esponse = tokenizer.decode(outputs[0], skip_special_tokens=True)
esponse = response.replace(prompt, "").strip()
eturn response

ity_analysis(city_name):
rompt = f"Provide a detailed analysis of {city_name} including:\n1. Crime Index and safety statistics\n2. Accident rates and traffic safety information\n3. Overall safety assessment\n\nCity: {cit
eturn generate_response(prompt, max_length=1000)

itizen_interaction(query):
rompt = f"As a government assistant, provide accurate and helpful information about the following citizen query related to public services, government policies, or civic issues:\n\nQuery: {query}
eturn generate_response(prompt, max_length=1000)

ate Gradio interface
gr.Blocks() as app:
r.Markdown("# City Analysis & Citizen Services AI")

ith gr.Tabs():
    with gr.TabItem("City Analysis"):
        with gr.Row():
            with gr.Column():
                city_input = gr.Textbox(
                    label="Enter City Name",
                    placeholder="e.g., New York, London, Mumbai...",
                    lines=1
                )
                analyze_btn = gr.Button("Analyze City")

            with gr.Column():
                city_output = gr.Textbox(label="City Analysis (Crime Index & Accidents)", lines=15)

        analyze_btn.click(city_analysis, inputs=city_input, outputs=city_output)
```

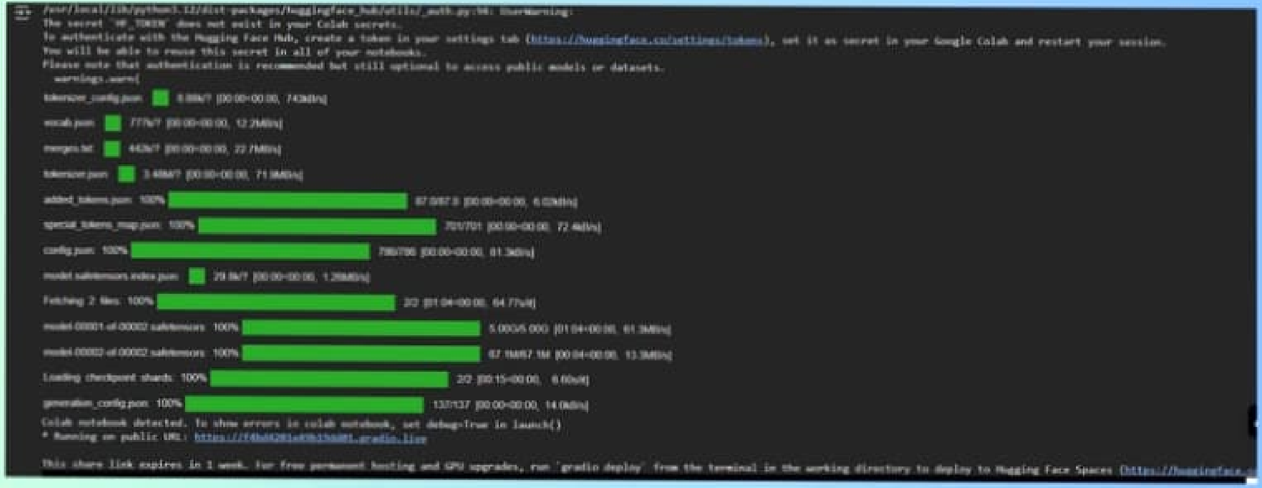


```python
    with gr.TabItem("Citizen Services"):
        with gr.Row():
            with gr.Column():
                citizen_query = gr.Textbox(
                    label="Your Query",
                    placeholder="Ask about public services, government policies, civic issues...",
                    lines=4
                )
                query_btn = gr.Button("Get Information")

            with gr.Column():
                citizen_output = gr.Textbox(label="Government Response", lines=15)

        query_btn.click(citizen_interaction, inputs=citizen_query, outputs=citizen_output)

aunch(share=True)
```

- You can find the code here in this link: [CitizenAI Code]()

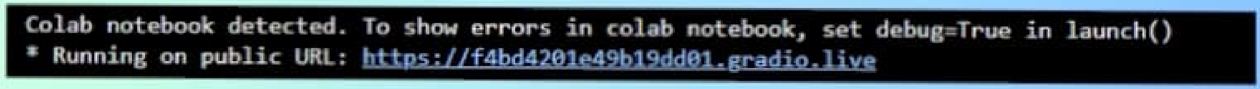
Output:

- Now you can see our model is being Downloaded and application is running.



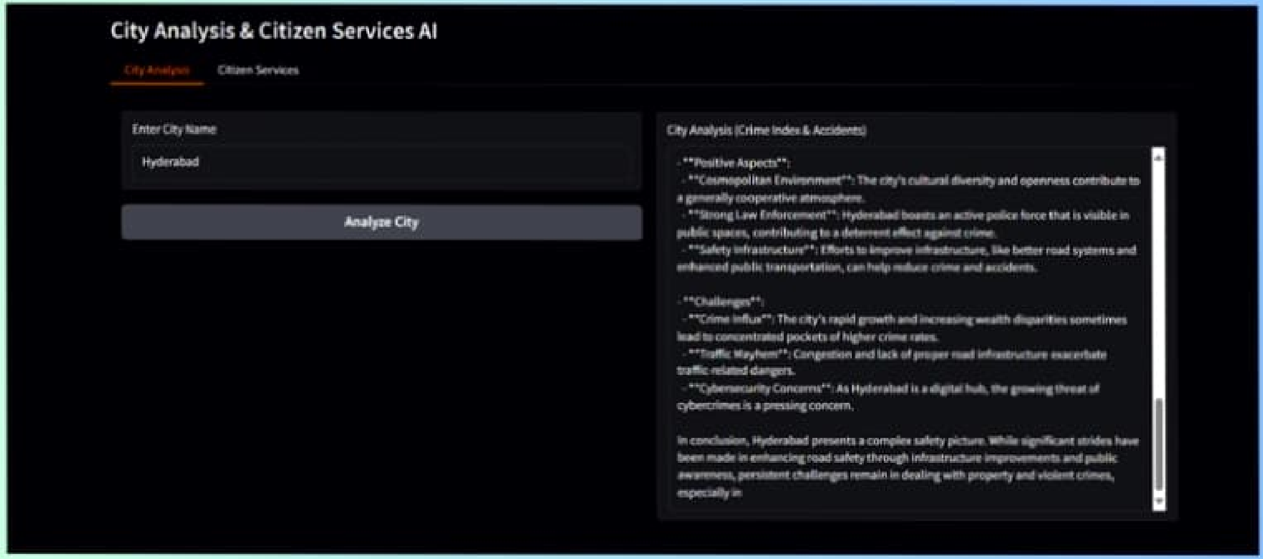
```
/usr/local/lib/python3.12/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as secret in your Google Colab and restart your session.
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
  warnings.warn(
tokenizer_config.json:  6.88k/? [00:00<00:00, 743kB/s]
vocab.json:  777k/? [00:00<00:00, 12.2MB/s]
merges.txt:  442k/? [00:00<00:00, 22.7MB/s]
tokenizer.json:  3.48M/? [00:00<00:00, 71.9MB/s]
added_tokens.json: 100%  87.0/87.0 [00:00<00:00, 6.02kB/s]
special_tokens_map.json: 100%  701/701 [00:00<00:00, 72.4kB/s]
config.json: 100%  786/786 [00:00<00:00, 81.3kB/s]
model.safetensors.index.json:  29.8k/? [00:00<00:00, 1.28MB/s]
Fetching 2 files: 100%  2/2 [01:04<00:00, 64.77s/it]
model-00001-of-00002.safetensors: 100%  5.00G/5.00G [01:04<00:00, 61.3MB/s]
model-00002-of-00002.safetensors: 100%  67.1M/67.1M [00:04<00:00, 13.3MB/s]
Loading checkpoint shards: 100%  2/2 [00:15<00:00, 6.60s/it]
generation_config.json: 100%  137/137 [00:00<00:00, 14.0kB/s]
Colab notebook detected. To show errors in colab notebook, set debug=True in launch()
* Running on public URL: https://f4bd4201e49b19dd01.gradio.live

This share link expires in 1 week. For free permanent hosting and GPU upgrades, run `gradio deploy` from the terminal in the working directory to deploy to Hugging Face Spaces (https://huggingface.
```

- Click on the URl to open the Gradio Application click on the link.

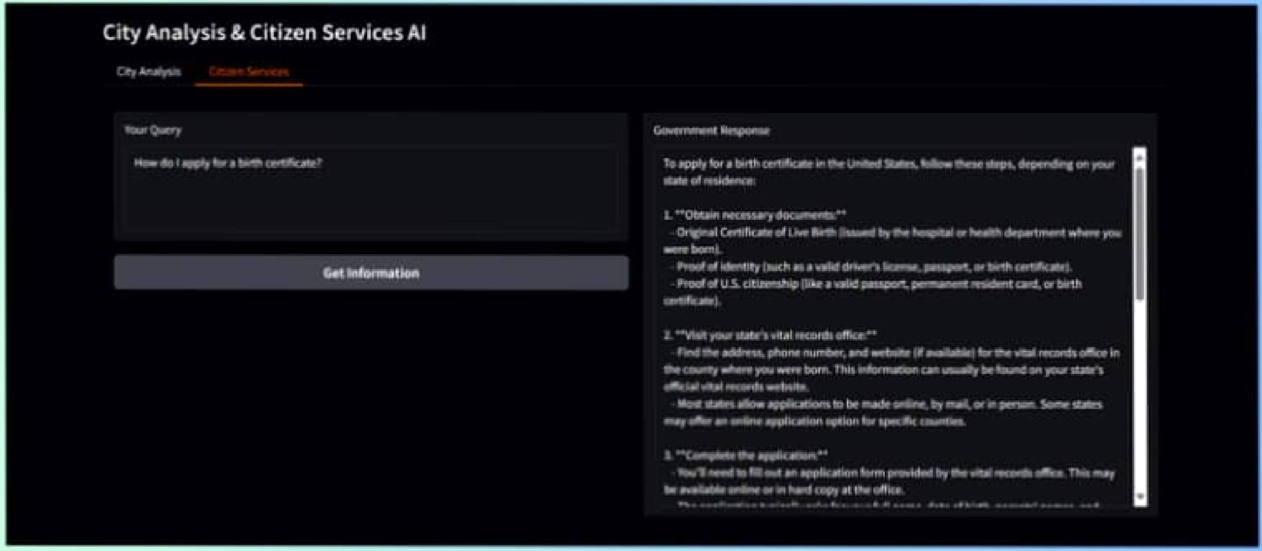


```
Colab notebook detected. To show errors in colab notebook, set debug=True in launch()
* Running on public URL: https://f4bd4201e49b19dd01.gradio.live
```

- You can View the Application is the running in the other tab.


City Analysis & Citizen Services AI
City Analysis
Citizen Services
Enter City Name
Hyderabad
Analyze City
City Analysis (Crime Index & Accidents)

```
- **Positive Aspects**:
  - **Cosmopolitan Environment**: The city's cultural diversity and openness contribute to a generally cooperative atmosphere.
  - **Strong Law Enforcement**: Hyderabad boasts an active police force that is visible in public spaces, contributing to a deterrent effect against crime.
  - **Safety Infrastructure**: Efforts to improve infrastructure, like better road systems and enhanced public transportation, can help reduce crime and accidents.

- **Challenges**:
  - **Crime Influx**: The city's rapid growth and increasing wealth disparities sometimes lead to concentrated pockets of higher crime rates.
  - **Traffic Mayhem**: Congestion and lack of proper road infrastructure exacerbate traffic-related dangers.
  - **Cybersecurity Concerns**: As Hyderabad is a digital hub, the growing threat of cybercrimes is a pressing concern.

In conclusion, Hyderabad presents a complex safety picture. While significant strides have been made in enhancing road safety through infrastructure improvements and public awareness, persistent challenges remain in dealing with property and violent crimes, especially in
```

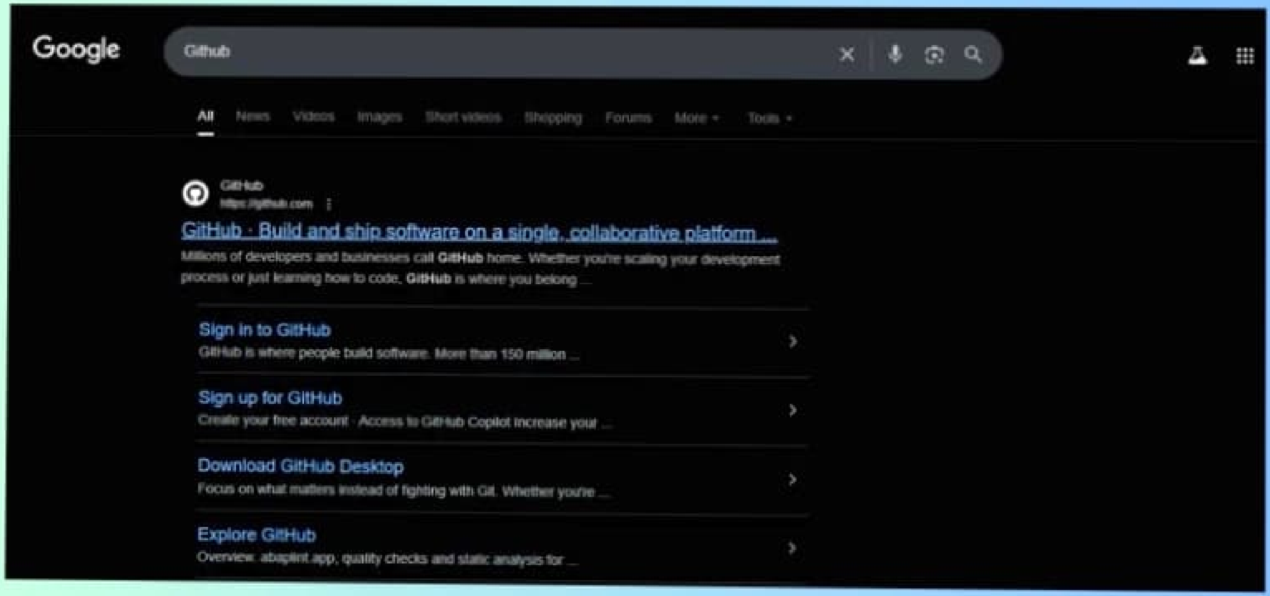


How do I apply for a birth certificate?

```
To apply for a birth certificate in the United States, follow these steps, depending on your state of residence:

1. **Obtain necessary documents:**
   - Original Certificate of Live Birth (issued by the hospital or health department where you were born).
   - Proof of identity (such as a valid driver's license, passport, or birth certificate).
   - Proof of U.S. citizenship (like a valid passport, permanent resident card, or birth certificate).

2. **Visit your state's vital records office:**
   - Find the address, phone number, and website (if available) for the vital records office in the county where you were born. This information can usually be found on your state's official vital records website.
   - Most states allow applications to be made online, by mail, or in person. Some states may offer an online application option for specific counties.

3. **Complete the application:**
   - You'll need to fill out an application form provided by the vital records office. This may be available online or in hard copy at the office.
```

Activity-4: Upload Your Project in GitHub.

- Search for "GitHub" in any browser, then click on the first link (GitHub).



Google

Github

All News Videos Images Short videos Shopping Forums More Tools

GitHub
https://github.com

[GitHub · Build and ship software on a single, collaborative platform ...]

Millions of developers and businesses call **GitHub** home. Whether you're scaling your development process or just learning how to code, **GitHub** is where you belong ...

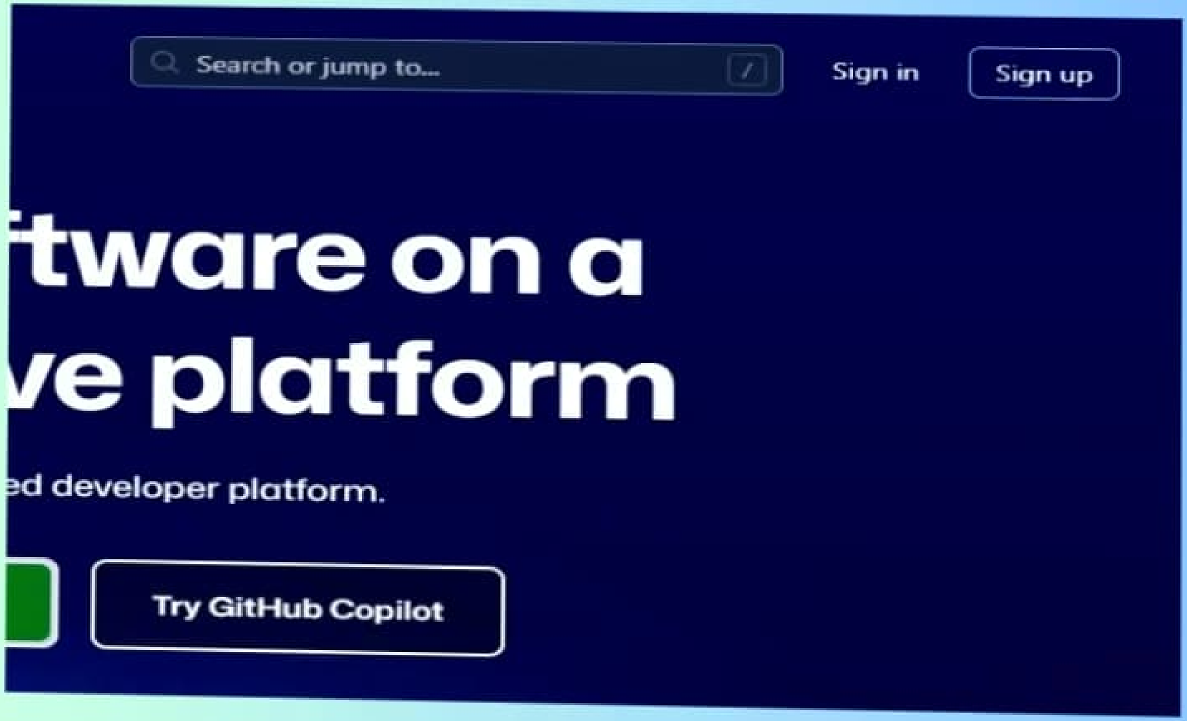
Sign in to GitHub
GitHub is where people build software. More than 150 million ...

Sign up for GitHub
Create your free account · Access to GitHub Copilot Increase your ...

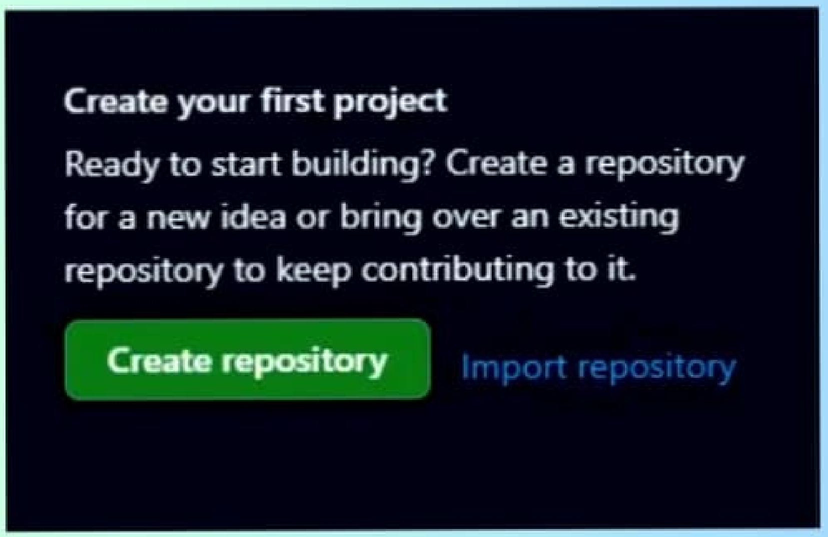
Download GitHub Desktop
Focus on what matters instead of fighting with Git. Whether you're ...

Explore GitHub
Overview. abaplint.app, quality checks and static analysis for ...

- Then click on "Signup" and create your own account in GitHub. If you already have an account click on "Sign in"



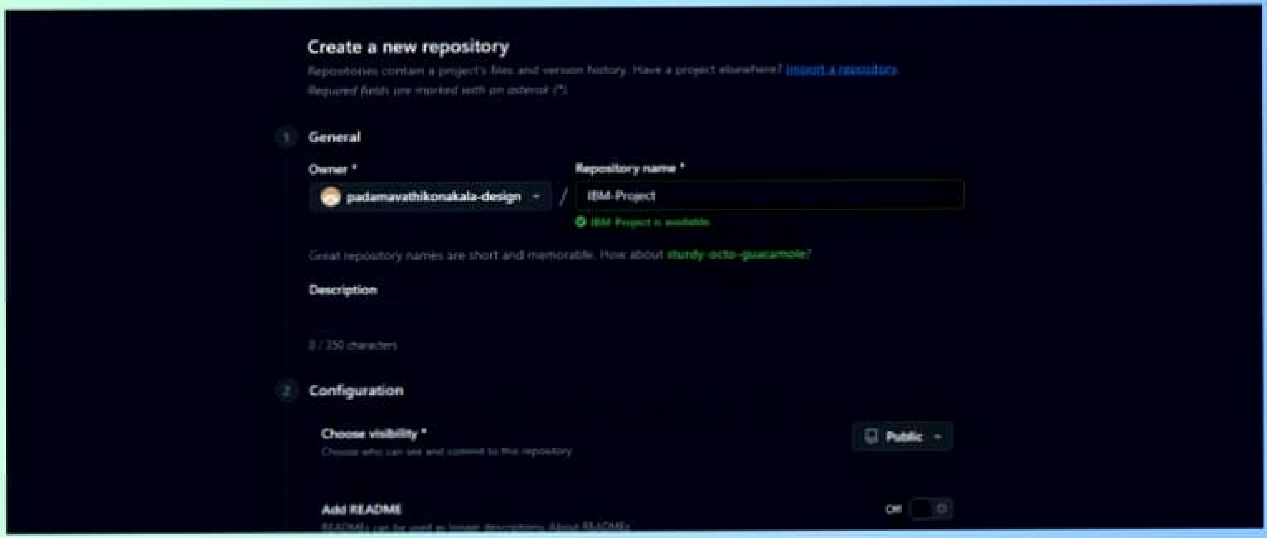
- Click on "Create repository".



Create your first project

Ready to start building? Create a repository for a new idea or bring over an existing repository to keep contributing to it.

- In "General" Name your repo. (Here I have given "IBM-Project" as my repo name and it is available)



Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? Import a repository.
Required fields are marked with an asterisk (*).

General

Owner * padamavathikonakala-design / Repository name * IBM-Project

IBM-Project is available.

Great repository names are short and memorable. How about sturdy-octo-guacamole?

Description

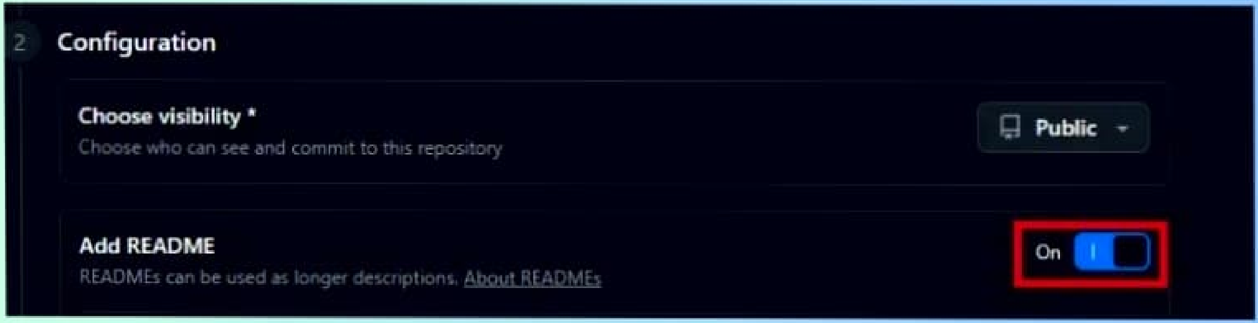
0 / 350 characters

Configuration

Choose visibility * Public
Choose who can see and commit to this repository

Add README Off
READMEs can be used as longer descriptions. About READMEs

- In "Configurations" Turn On "Add readme" file Option.


2 Configuration

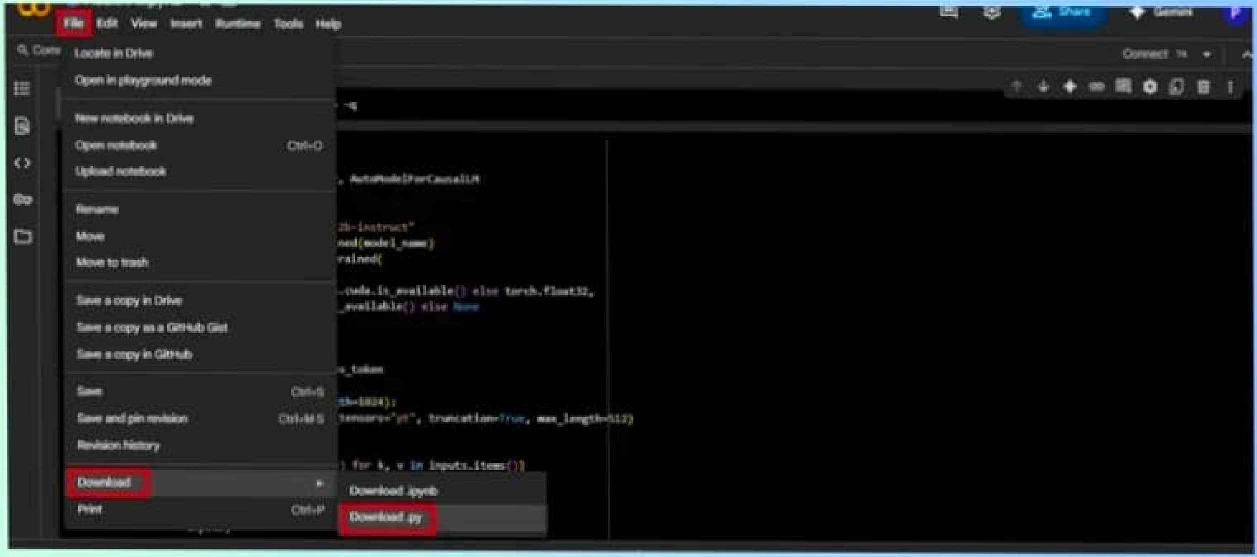
Choose visibility *
Choose who can see and commit to this repository

Public

Add README
READMEs can be used as longer descriptions. About READMEs

On

- Now Download your code from Google collab by Clicking on "File", then Goto "Download" then download as ".py".

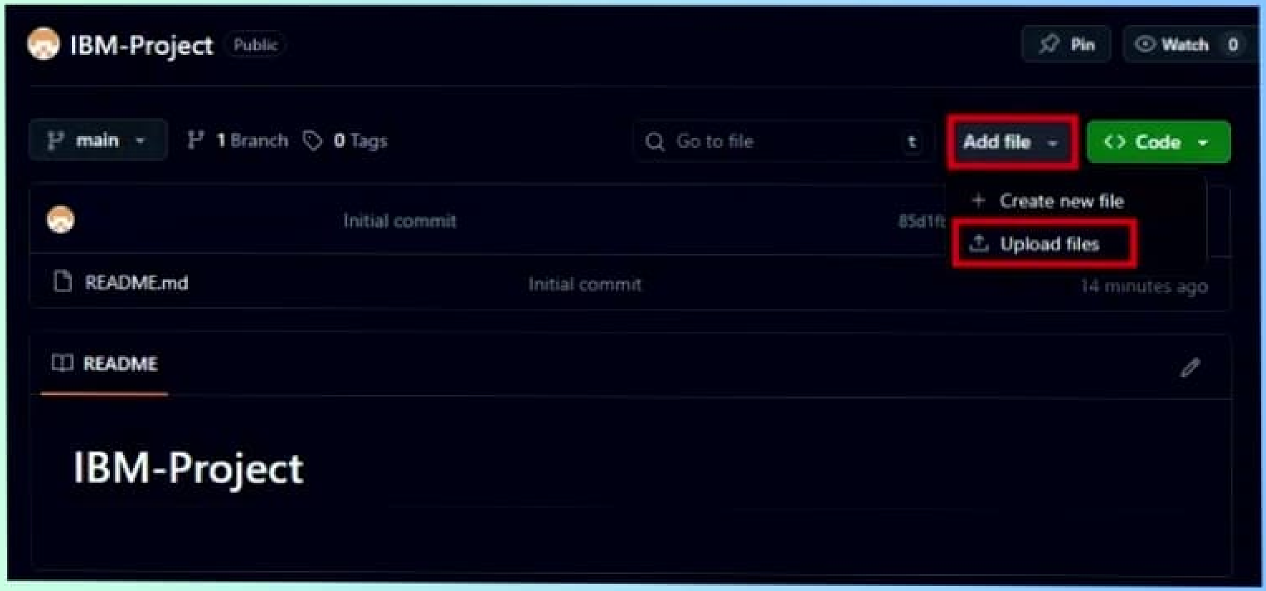


```
, AutoModelForCausalLM

2b-instruct"
ned(model_name)
rained(

.cuda.is_available() else torch.float32,
_available() else None

s_token

th=1024):
tensors="pt", truncation=True, max_length=512)

) for k, v in inputs.items()}
```

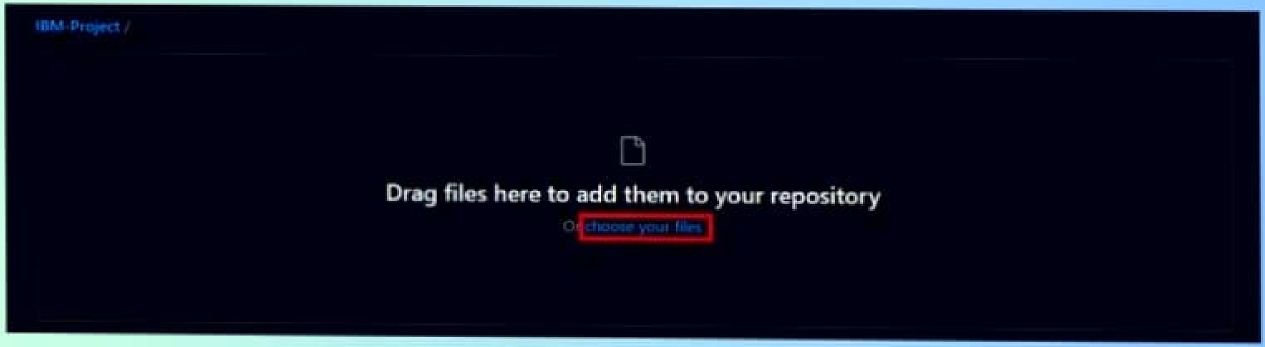
Then your repository is created, then Click on "Add file" Option. Then Click

-

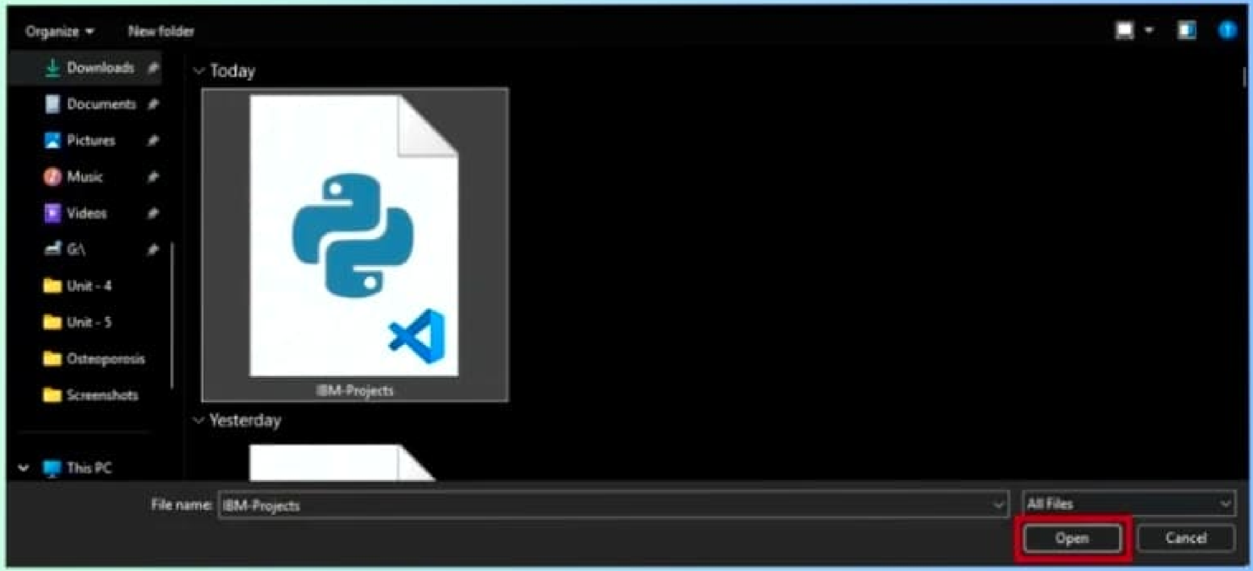
"Upload files" to upload your files.



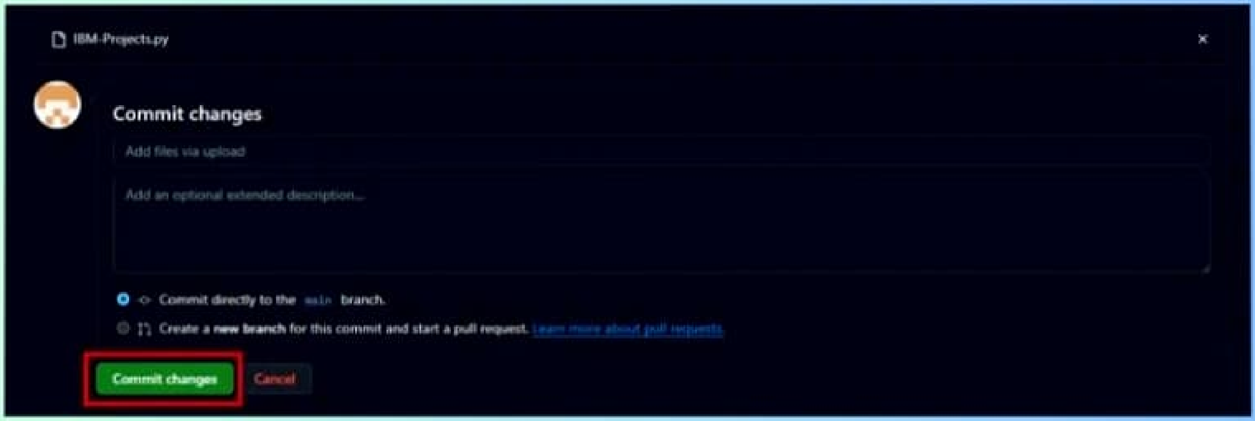
- Click on "choose your files".



- Choose your project file and click on "Open".



- After your file has Uploaded Click on "Commit changes".


IBM-Projects.py
Commit changes
Add files via upload
Add an optional extended description...
Commit directly to the main branch.
Create a new branch for this commit and start a pull request. Learn more about pull requests.
Commit changes
Cancel