AI Productivity and AI API Integration for Developers

+ Partnerships Hacktiv8

This Program is part of the AI Opportunity Fund: Asia Pacific, in collaboration with AVPN and supported by Google.org and the Asian Development Bank.

HACKTIV8

Introduction to AI & Boosting Productivity with AI

Chapter 1

HACKTIV8

Chapters

Introduction to AI & Boosting Productivity with AI

Part 1

Introduction to AI & Boosting Productivity with AI

- AI, generative AI concepts & ethics
- Prompt engineering for LLM (Large Language Models)
- AI tools for productivity
  - Gemini Code Assist
  - Copilot
  - Cursor
- Introduction to NotebookLM

Part 2

Building a Website Using AI

- Introduction to Gemini Canvas & V0 to build a website
- Explore Gemini Canvas & V0's Features
- Use Effective Prompts
- Leverage Gemini Canvas & V0 Effectively

Part 3

Implementing Gemini AI

- Introduction to Gemini AI API
- Initialize model Gemini AI
- Configuration parameter
  - temp
  - top_p
  - top_k
- Text generation, input:
  - image
  - audio
  - text
  - document
- File API

Part 4

Implementing Gemini AI Part. 2

- Develop Chatbot
- Implementing Gemini AI in Chatbot

HACKTIV8

Introduction to AI & Boosting Productivity with AI

Objective

- What is AI, Generative AI, and LLM ?
- Understanding AI Ethics
- Introduction to Prompt Engineering
- Prompting Techniques
- AI Productivity Tools Overview
- How to use Gemini Code Assist, Github Copilot and Cursor AI
- Introduction NotebookLM

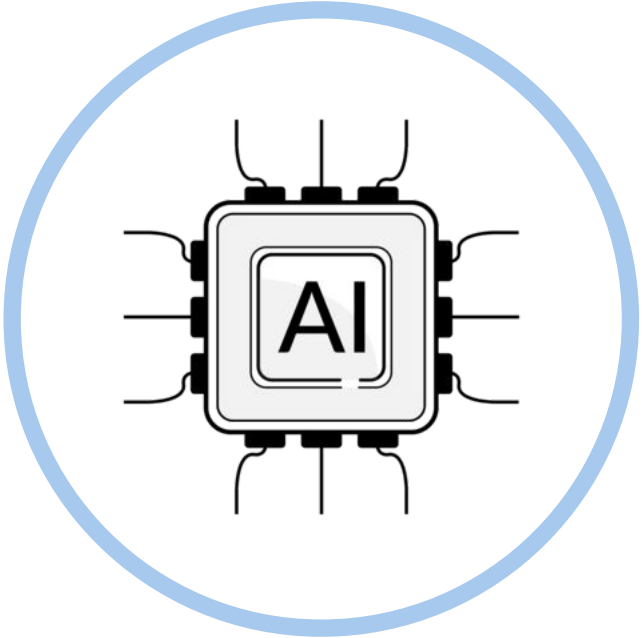
Introductions

What is AI ?

AI is the ability of a computer system to simulate human intelligence processes—such as reasoning, learning, decision-making, and problem-solving.

Analogy: Imagine AI as a very fast and obedient assistant. You give it a task like sorting emails, analyzing spreadsheets, or suggesting solutions—and it performs it efficiently using patterns it has learned.

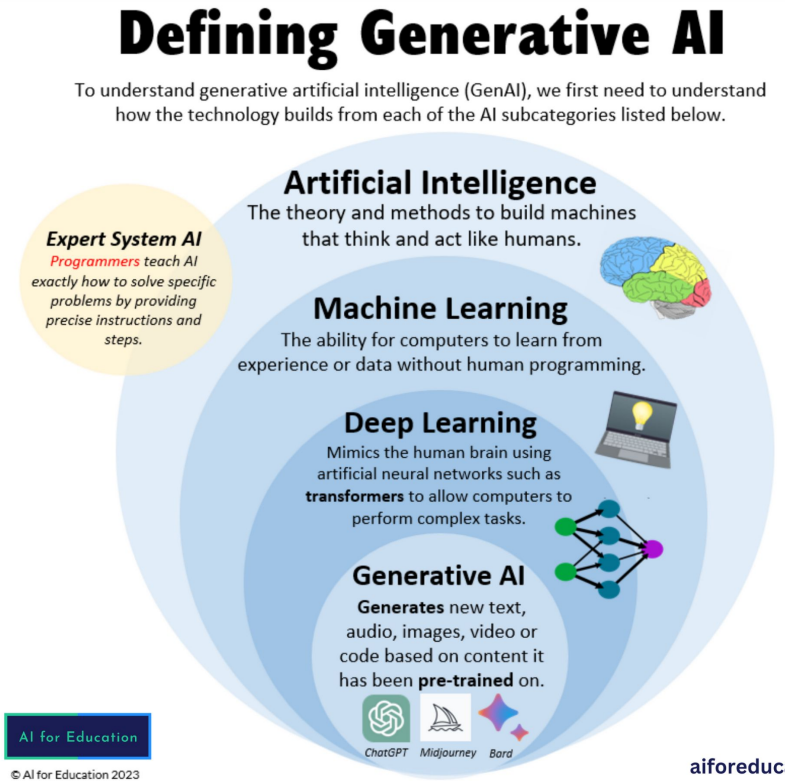
Example: Google Translate uses AI to detect languages and translate them in real time.



What is Generative AI ?

Generative AI is a type of artificial intelligence that can create new content, such as text, images, music, or even videos, based on previously learned data.

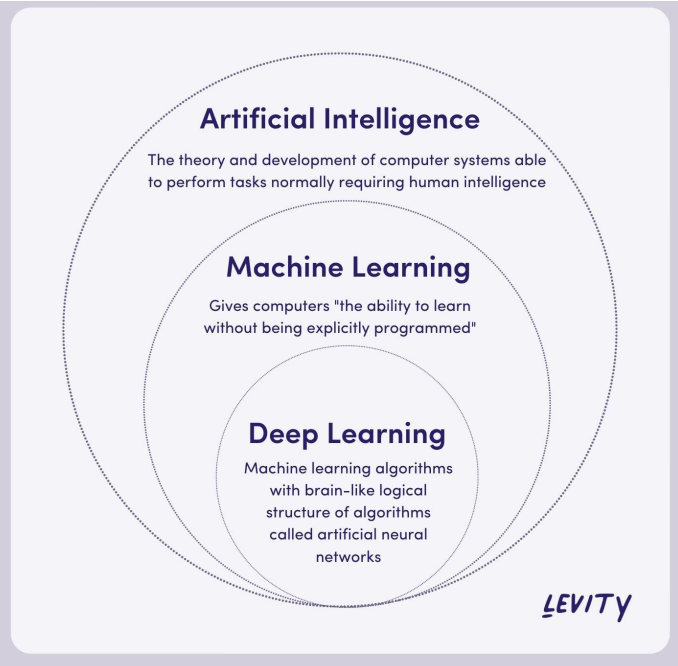
Generative AI works by learning patterns or structures from existing data. Several techniques are used for this, one of which is Deep Learning, a method where the AI model is trained using very large neural networks.



What is Machine Learning ?

Machine Learning is a field of study that gives computers the ability to learn without being explicitly programmed. (Introduced by *Arthur Samuel, 1959*)

In machine learning, instead of telling the computer the exact steps to follow, we provide data and let the algorithm discover patterns and make decisions on its own.



What is Large Language Model(LLM) ?

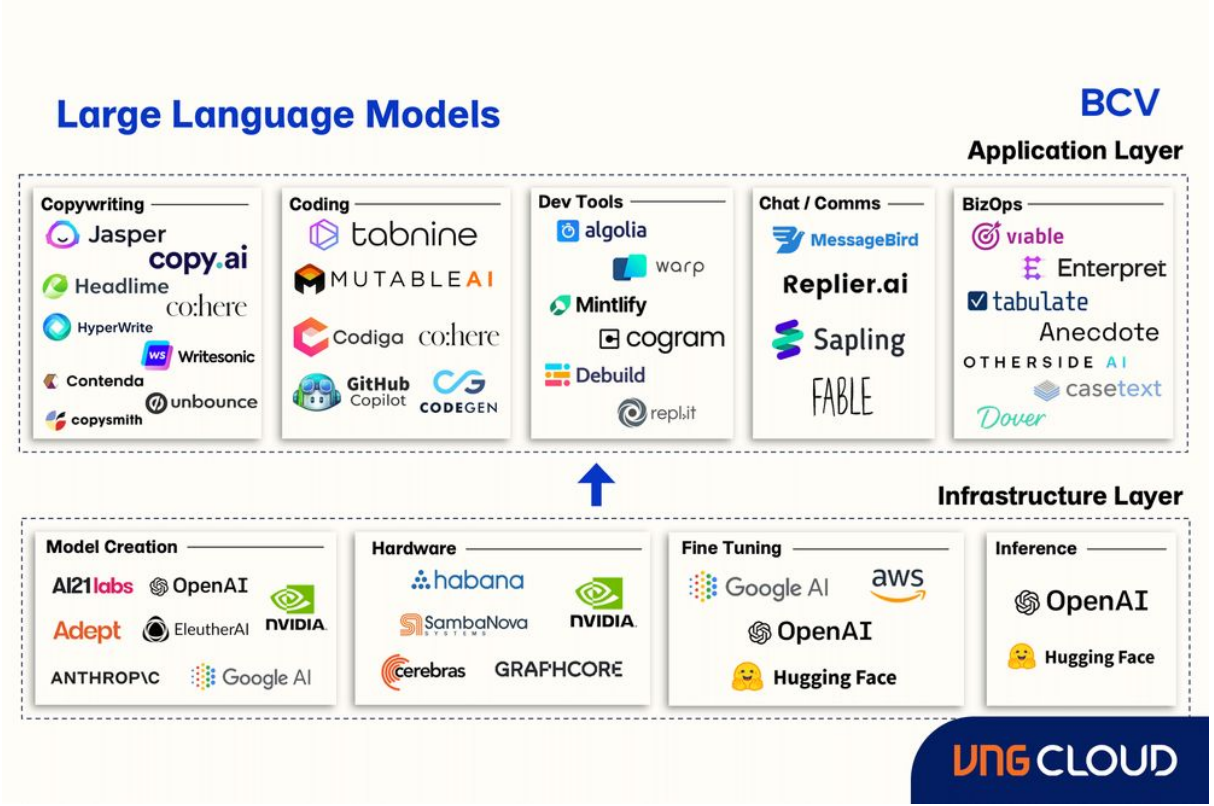
LLM (Large Language Model) is a computer program that learns and generates human-like language using a transformer architecture trained on massive amounts of data.

LLMs are deep learning models (neural networks) designed to process and understand natural human language
An LLM can be built by applying a transformer, which is a type of neural network architecture consisting of encoders and decoders equipped with self-attention capabilities.

LLMs are highly flexible—one model can perform entirely different tasks, such as answering questions, summarizing documents, translating languages, and completing sentences.

LLMs can be used in **Generative AI** applications, such as generating content based on human-language input prompts.

What is Large Language Model(LLM) ?



Insights

Artificial Intelligence (AI) serves as the broad foundation—referring to computer systems that mimic human intelligence to perform tasks such as reasoning, decision-making, and learning. Within AI, **Machine Learning (ML)** is a crucial subset that enables systems to learn from data without being explicitly programmed. ML gives AI the "learning" capability by allowing algorithms to detect patterns and improve over time through experience.

Building further on ML, **Generative AI** specializes in *creating* new content—text, images, music, or video—based on what it has learned. This is achieved using deep learning models that understand and replicate complex data patterns.

One of the most powerful forms of these deep learning models is the **Large Language Model (LLM)**. LLMs use transformer architectures to process and generate human-like language at scale. They represent a significant advancement in Generative AI, allowing a single model to perform multiple tasks—such as summarizing documents, translating languages, answering questions, and even coding—just from natural language prompts.

HACKTIV8

Understanding of AI Ethics

What is AI Ethics ?

AI Ethics refers to the **moral principles** and **guidelines that govern** how artificial intelligence should be developed, deployed, and used. It addresses the responsibilities of AI creators and users to ensure that AI is safe, fair, and beneficial to humanity.

At its core, AI Ethics seeks to answer one important question:

"Just because we can build powerful AI systems—should we? And if so, how do we make sure they help, not harm?"

Why is AI Ethics Important ?

AI systems are not conscious—they do not understand right or wrong. They make decisions based on data and instructions given by humans. If the data is biased or incomplete, the AI will learn and amplify those errors.

Here’s where ethics become vital:

- AI can make decisions that affect real lives—job applications, loan approvals, medical diagnoses, etc.
- AI can be biased, especially when trained on biased historical data.
- AI can invade privacy, especially if it's collecting and analyzing personal data without consent.
- AI can mislead, such as generating deepfakes or false information confidently.

Core Principles of AI Ethics ?

Principle	Explanation	Analogy
Transparency	The AI system should be understandable—users should know what it's doing and why.	Like a calculator: you know the formula behind it.
Fairness	AI must treat all users equally—regardless of gender, race, or background.	A fair referee in a game: applies rules equally.
Accountability	Humans—not machines—should be responsible for AI's outcomes.	Like a chef is responsible for the dish—not the knife.
Privacy	Personal data used by AI should be protected and only used with consent.	Like a locked diary: only accessible to people you trust.
Safety	AI must not cause harm—intentionally or by accident.	Like designing a car with brakes and airbags.

AI Ethics Concern Samples

Scenario	Ethical Concern
AI decides who gets hired	**Bias and transparency**
AI used in school exams	**Fairness and misuse**
AI scraping social media data	**Privacy violation**
AI suggesting medical treatments	**Accountability and risk**

Summary

Think of AI like Neo in *The Matrix* — equipped with extraordinary powers in a world built from data. But just like Neo had to choose between control and conscience, we too must ensure that the AI we create not only acts with precision but with principles.

In this digital age, with great algorithmic power comes an even greater ethical responsibility. Teaching machines what's right is just as important as teaching them what works."

HACKTIV8

Introduction to Prompt Engineering 101

What is Prompt Engineering ?

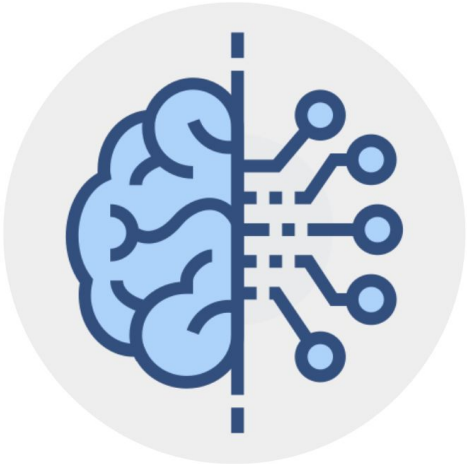
Prompt engineering is the art of crafting inputs (*prompts*) to guide AI into producing useful outputs.

Why it matters:
AI doesn't "think" like humans. It needs clear, context-rich instructions.

Analogy: Talking to AI is like ordering food at a restaurant. If you say,

> *“Give me something good.”*
> you might get a random dish. But if you say,
>
> *“I'd like a spicy vegetarian curry, no peanuts.”*
> you get what you want. Prompting works the same way.

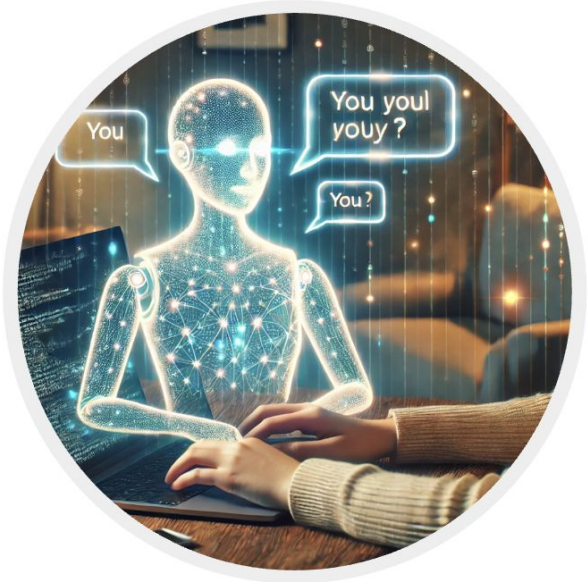
Elements of Prompt



- **Instructions**
- **Questions**
- **Additional input data / context**
- **Examples how to answer the questions**
- **Desired output format**

A prompt can have those 5 elements , but actually none of the elements have to be present, you can also start a sentence with auto completion of the model. But in order to have a good prompt, at least one **instruction** or **questions** should be present.

Use Cases of Prompting



- **Summarization**
- **Classification**
- **Translation**
- **Text generation / completion**
- **Question / answering**
- **Coaching**
- **Image generation**

This just some an idea for prompting so we know what we can do with prompts.

General Tips

#1 Tips: **Use instruction and clear questions** : The more precise your prompt, the better the AI can understand and respond to your needs.

Instead of: "Tell me about typescript"

Try: "**Explain the key differences between javascript and typescript, focusing on the cons and pros for each language**"

#2 Tips: **Provide context** : Give the AI relevant background information to frame your request.

Instead of: "Suggest 3 best features for E-commerce application"

Try: "**I'm currently making an application about E-commerce using react, please give me 3 best features to be implemented in the application so my application meet the general standard requirement of E-commerce apps**"

#3 Tips: **Be specific** : Specify the desired output format. This will increase the chances we get what we want to see.

Instead of: "Please give me a learning material for react'"

Try: **"Please give me a learning material for react, answer it with day-to-day learning phase in one week and please include the video references"**

#4 Tips: **Encourage the model to be factual** : Give the AI encouragement to answer with reliable sources.

Instead of: "How to use react redux toolkit?"

Try: **"How to use react redux toolkit? Please answer only using reliable documentation and include the link of the documentation"**

Prompting Techniques to Control the Output

HACKTIV8

#1 Technique: **Length Control** : Specify desired output length

Try: **"Give me a 50 data of popular movie of all time with name, year of release and director"**

#2 Technique: **Style Control** : Specify desired output style

Try: **"Give me a data of movie with name, year of release and director in array of object javascript"**

#3 Technique: **Audience Control** : Specify desired output based on the audience

Try: **"Give me an explanation of what is next js to 5 year old kid"**

#4 Technique: **Scenario Based Guiding**: Specify desired output by specify the custom scenario

Try: **"Give me an explanation of data type in javascript when you being the senior developer"**

#5 Technique: **Break Complex Task Into Subtask**

Try: **"Please help me make an simple portfolio website with react by following these steps :**

1. Make an endpoint using react router that has 3 pages , Landing page , About Me, and Portfolio page
2. While making the landing page, please make it stylish and colorful
3. While making the portfolio page, please make the style that highlight my techstack and application features
"

Summary

- **Keep the elements of a prompting in mind**
- **Apply general tips**
- **Apply multiple prompting techniques to control the output**
- **Iterate!**

"By understanding the importance of good prompting, you can maximize the potential of AI for various purposes."

HACKTIV8

AI Productivity Tools Overview

Introductions

AI tools are no longer just experimental—they are powerful co-pilots in modern work. Whether you're writing code, drafting reports, analyzing documents, or managing tasks, AI can save hours and boost quality if you know how to use it.

AI productivity tools are designed to automate repetitive tasks, enhance creativity, and assist in technical workflows especially in fields like software development, content creation, research, and project management.

Imagine you had a team of invisible assistants:

- One writes your boilerplate code.
- Another explains APIs in real time.
- One searches documents and summarizes key ideas.
- Another gives live suggestions while you're typing.

That’s essentially what these AI productivity tools do.

Tools

Gemini Code Assist

Github Copilot

Cursor AI

NotebookLM

Gemini Code Assist

Gemini Code Assist

Gemini Code Assist

A code-generation and code-assist tool powered by Google's Gemini LLM (Large Language Model). It allows developers to generate code, understand snippets, fix bugs, and build web components all from prompts.

Core Features:

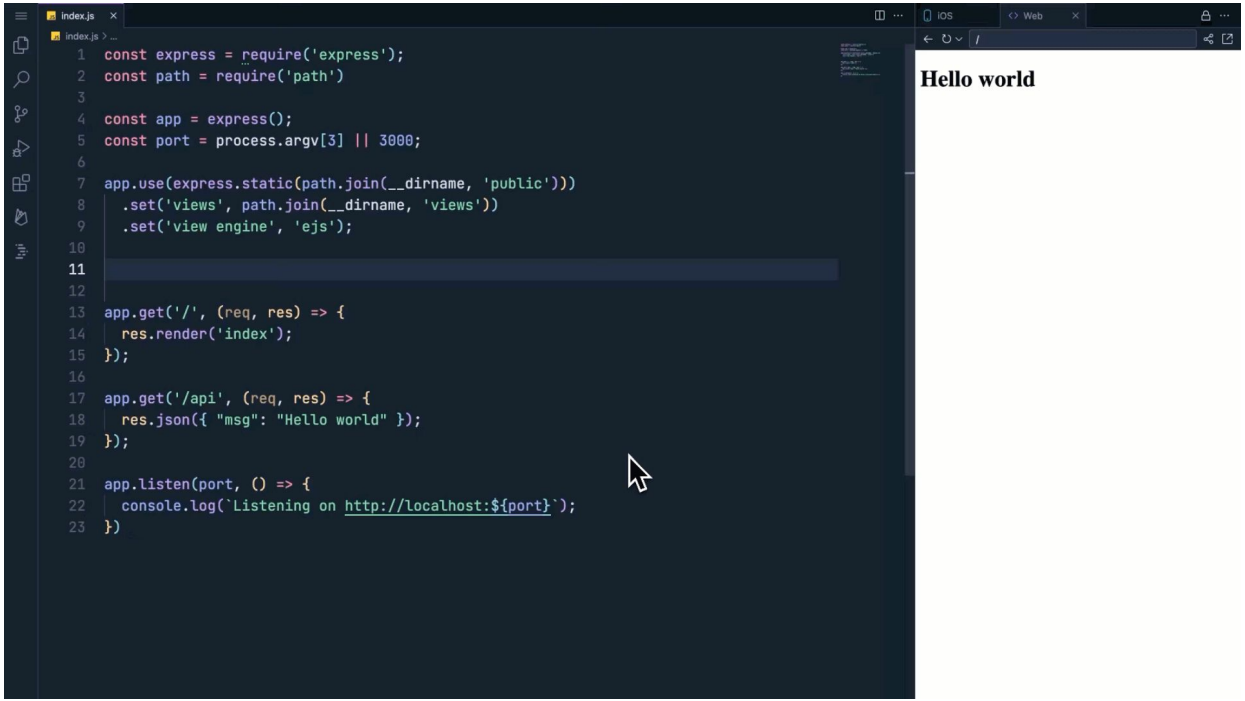
- Works with multi-modal inputs (code, text, images, documents)
- Understands complex prompts
- Can generate full HTML pages, REST APIs, or app logic

To do :

1. Do register, installation and configuration on your vscode

Reference :
https://developers.google.com/gemini-code-assist/docs/set-up-gemini

Gemini Code Assist



Source : https://firebase.studio/blog/article/build-with-gemini-in-idx

Github Copilot

Github Copilot

Github Copilot

A coding assistant that lives inside your IDE (VS Code, JetBrains, etc.). It suggests complete code lines, functions, and logic based on your intent and coding patterns.

Core Features:

- Real-time auto-completion for code
- Context-aware: adapts to the current file and project
- Supports dozens of languages: JavaScript, Python, Java, etc.

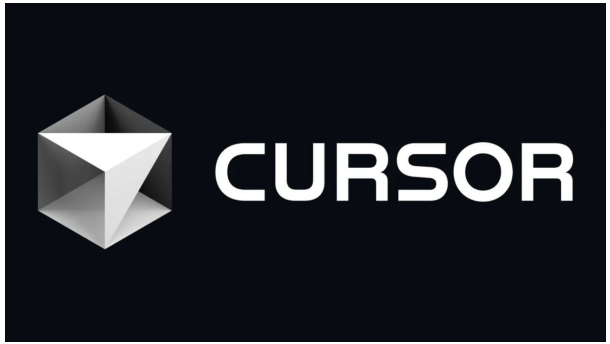
***Alternative (Optional)**
To do :
1. Do register, installation and configuration on your vscode

Reference : https://docs.github.com/en/copilot/quickstart

Github Copilot

Cursor AI

Cursor AI



Cursor AI

A full-featured AI-native code editor, where you can chat directly with your code. It combines the power of a text editor with conversational AI features.

Core Features:

- “Explain this code” within the editor
- Suggest code edits and improvements
- Debugging assistance
- Contextual chat about your current file

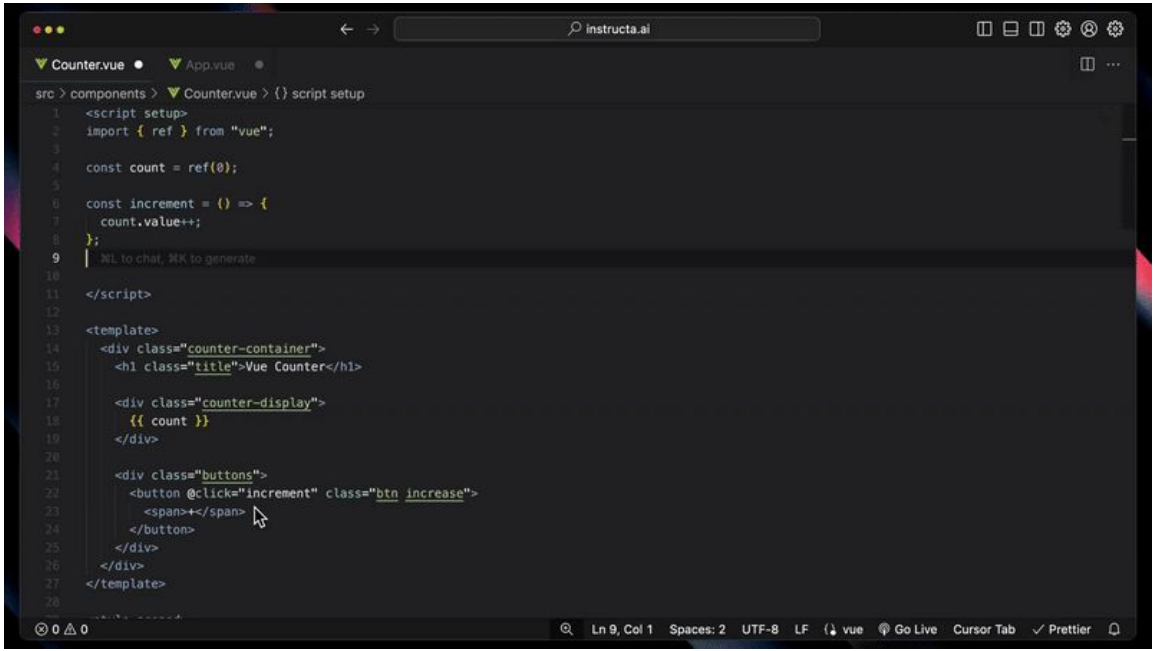
***Alternative (Optional)**

To do :

1. Do register, installation and configuration on your vscode

Reference : https://docs.cursor.com/get-started/installation

Cursor AI



Source : https://docs.cursor.com/get-started/installation

Summary AI Productivity Tools

Summary

Modern AI tools are built to boost developer and workplace productivity:

Tools	Main Function	Best For
Gemini Code Assist	Generate code and explain snippets	Web/App Developers
GitHub Copilot	Inline code suggestions in your IDE	Software Engineers
Cursor AI	AI-native code editor with chat-like flow	AI-powered development workflow

NotebookLM

NotebookLM

NotebookLM

NotebookLM is a **document-based AI assistant** from Google. It reads and understands your uploaded files and answers questions directly from them.

How it works:

1. Upload your documents (PDF, Google Docs, etc.)
2. Ask questions like:
   - “What are the main findings in section 3?”
   - “Summarize the results for Q2 2023.”

Use Cases:

- Research analysis
- Policy interpretation
- Preparing reports from large text sources

Analogy: It’s like having a personal researcher who’s read your files and never forgets anything.

Summary

Conclusion

Conclusions

In this session, we have built a solid foundation on how AI works, how we interact with it using prompts, and how modern tools can enhance productivity in real-world tasks.

Key Takeaways:

- AI helps machines perform human-like tasks. Generative AI creates new content. LLMs power tools like Gemini and ChatGPT.
- Ethics are essential to prevent AI misuse and protect people.
- Prompt engineering is crucial to getting the right response from AI.
- Tools like Gemini, Copilot, Cursor AI, and NotebookLM offer hands-on benefits for real development and documentation tasks.

Next, you'll start building your own simple website or chatbot using these AI tools.

HACKTIV8

Thank You

(021) 8067 5787
halo@hacktiv8.com

Hacktiv8id

Hacktiv8id

Hacktiv8

Hacktiv8 Indonesia