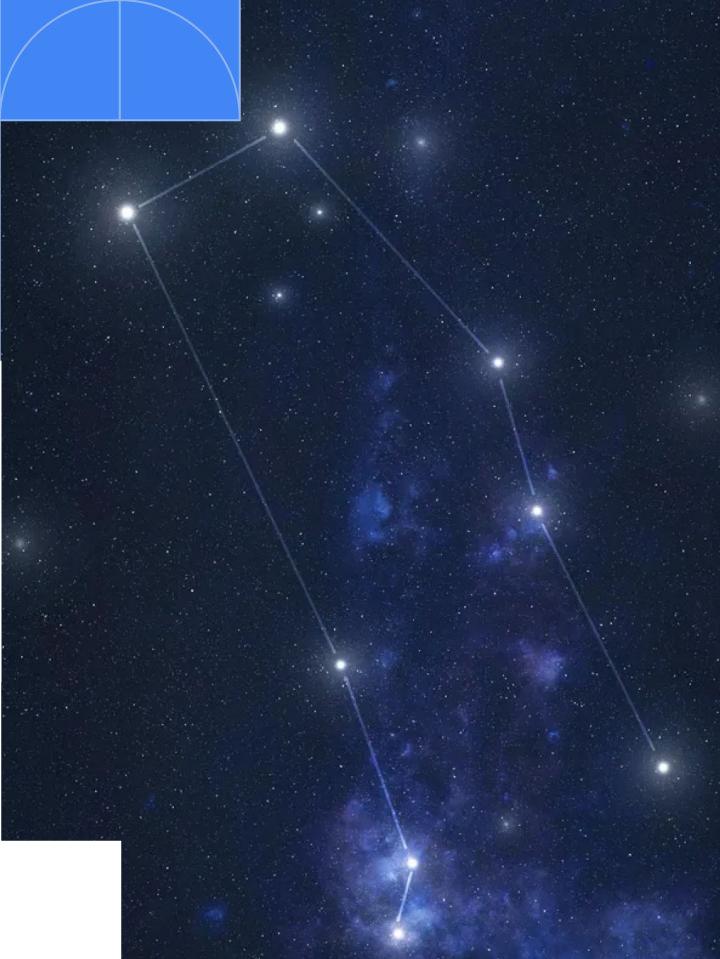
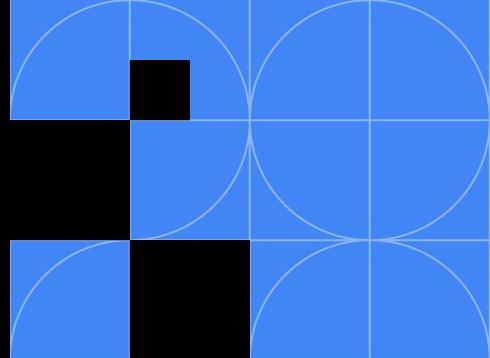


Build with AI



Intro to AI + Gemini 101

 Google Developer Groups
Capital Region

Who am I?

ChengCheng Tan

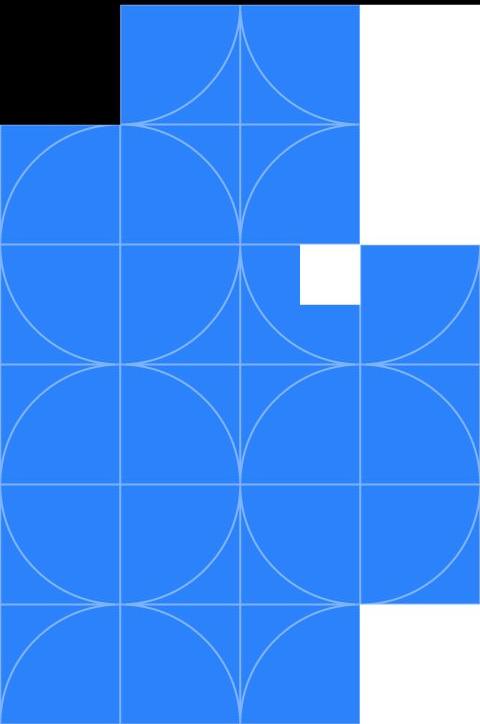
- BA Linguistics & CS, **UCLA**
- MS CS HCI, **Stanford**
- **LLM + AI Safety**
FAR AI Communications
AISafety.info + chatbot
- Google WTM Ambassador

* DISCLAIMER ideas presented here are my own



temperature Reinforcement frontier
RLHF Learning models
Pretrained AIML Supervised video
Gemini Generative AI Learning
Claude LLM vision Neural Networks
GPT tokens Multimodal
API prompts Conversation Finetuned
keys SDK Instruction Tuned language NLP

Build with AI



Google Developer Groups
Capital Region

Intro to AI

AI Map

Artificial Intelligence (AI)

Create machines that can perform tasks with human-like abilities: reasoning, learning & problem-solving.

AI Map

AI

GOFAI

Expert Systems

Planning Systems

Fuzzy Logic

Machine Learning (ML)

Learn patterns from data,
without explicit programming.

AI Map

AI

ML

Deep Learning
Complex patterns with
neural networks.

Decision Trees

Random Forests

Gradient Boost

Naive Bayes

SVM

KNN

AI Map

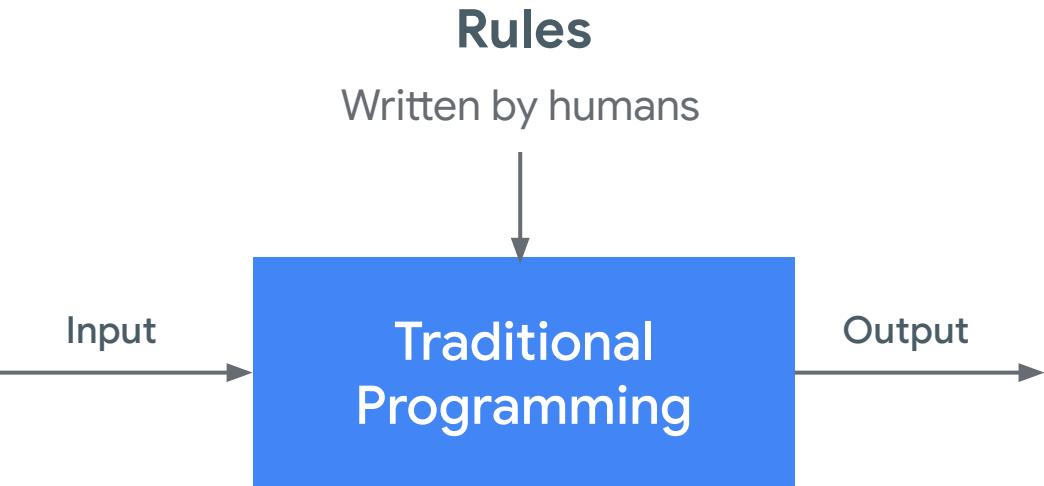
AI

ML

Deep
Learning

Generative AI

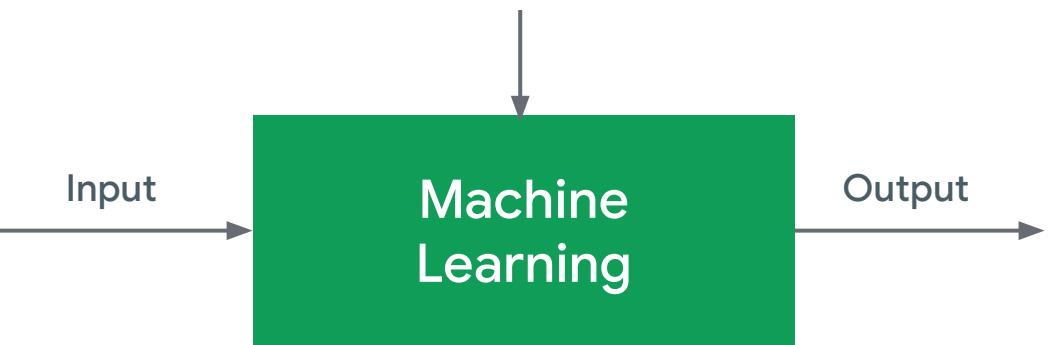
Old GOFAI Way



New ML Way

Examples

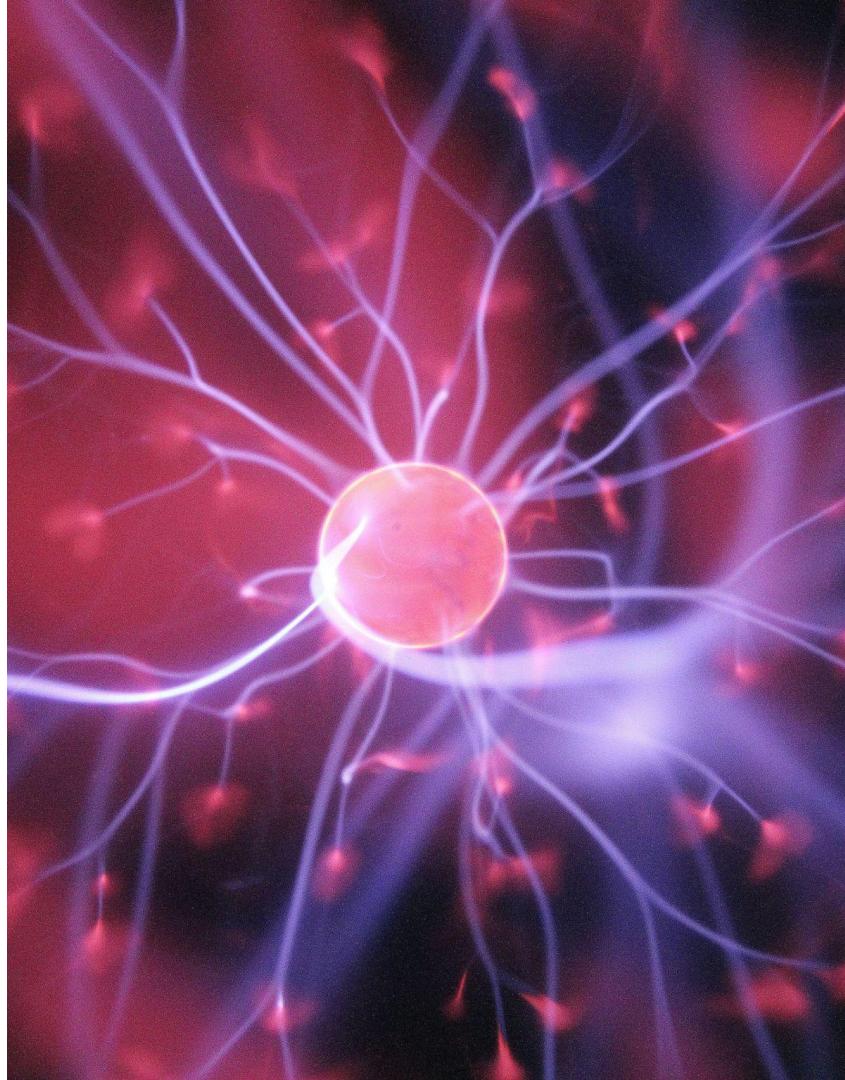
Computer learns rules



Neural Networks

Artificial neural networks (ANN)
are inspired by connections in the
[human brain](#).

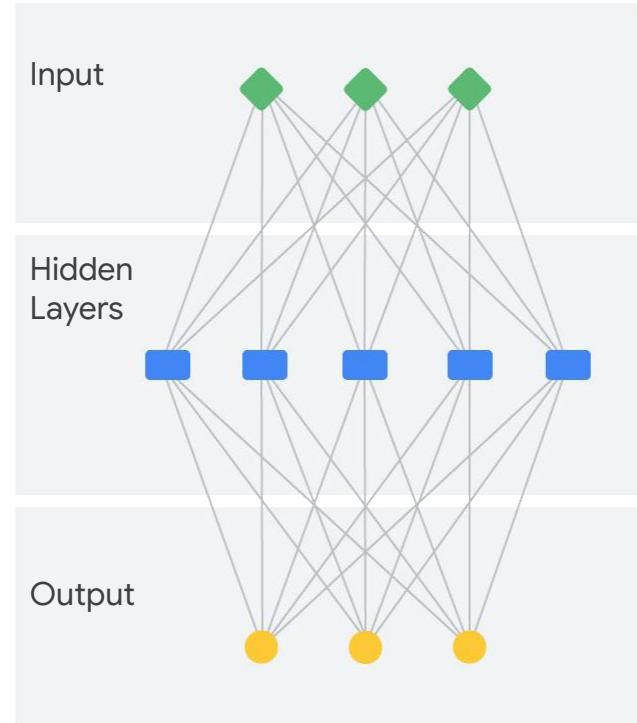
Can have one or more hidden layers.



Neural Networks

Artificial neural networks (ANN)
are inspired by connections in the
[human brain](#).

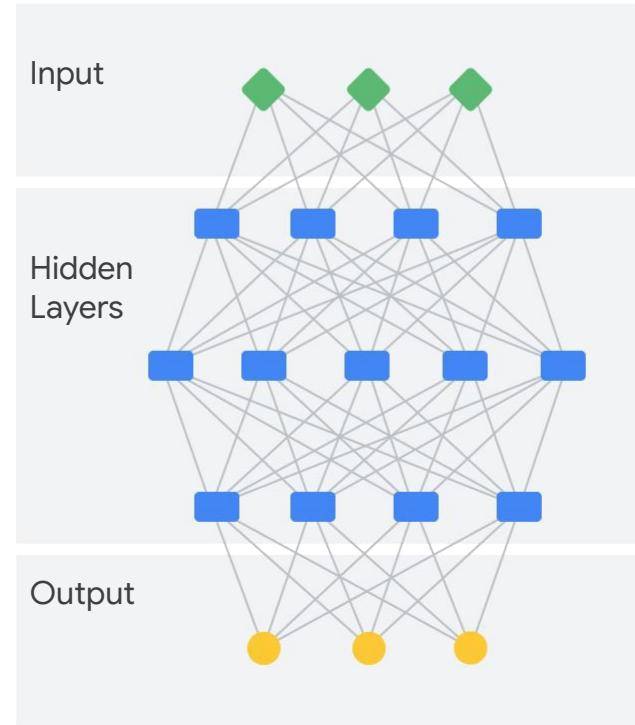
Can have one or more hidden layers.
Each node is a neuron or parameter.



Deep Learning

A **deep** neural network (DNN)
with multiple hidden layers

Large foundational models have
billions of neurons or parameters.



AI Model Types



Discriminative
Simple Output



Generative
Complex Output

Generative AI (GenAI)

AI that **generates content** for you.



Text



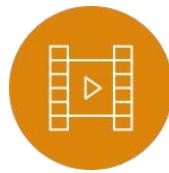
Code



Image



Speech



Video

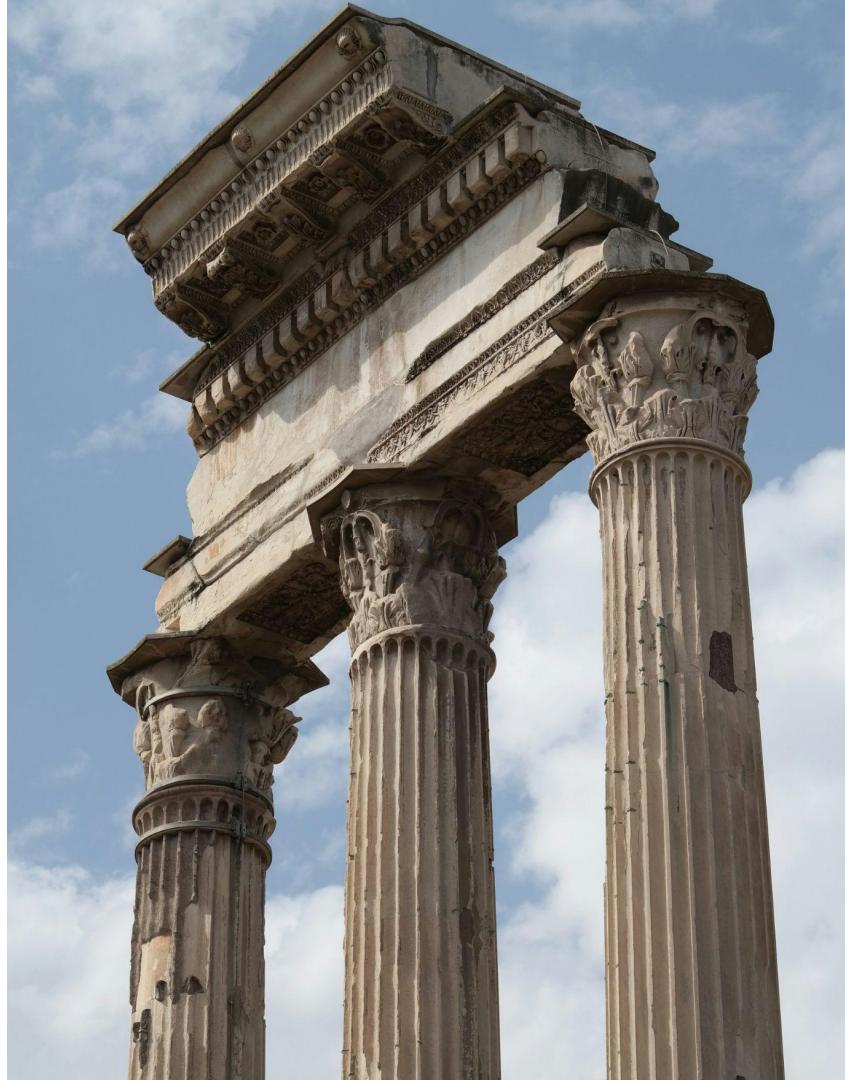


3D

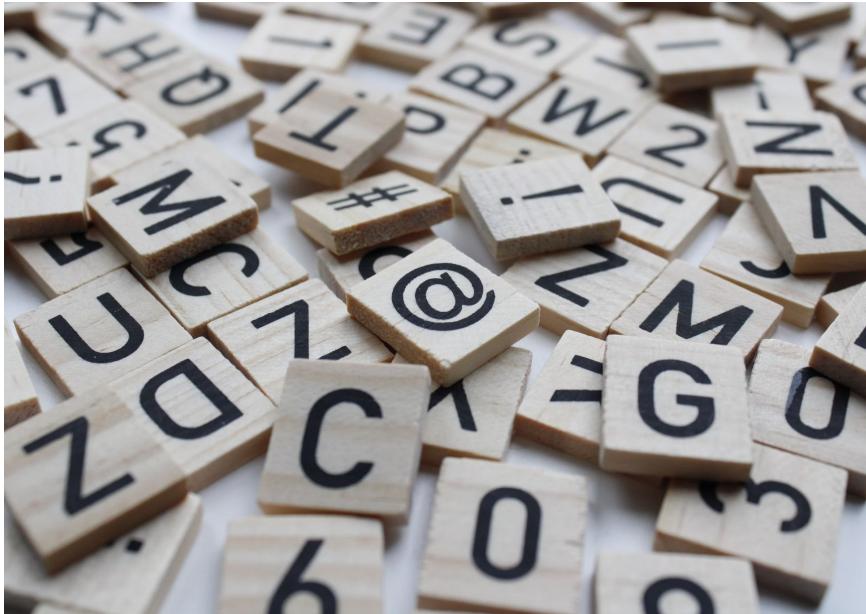
Why now?

Converging Forces

- Data
- Algorithms
- GPU Compute



How does it work?



Language Models

Next word prediction



Image Generation

Denoising images

Understanding the Excitement

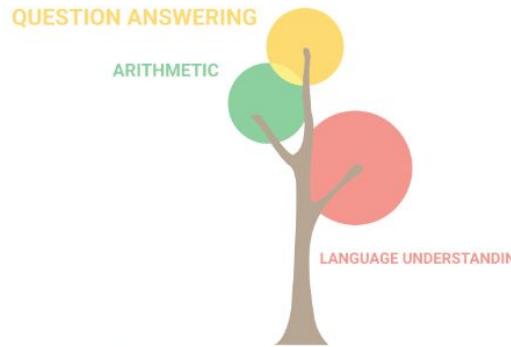


Narrow
Specialists



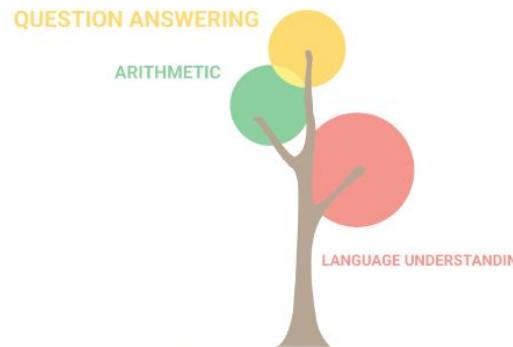
General
Artificial General Intelligence (AGI)

Emergent Abilities



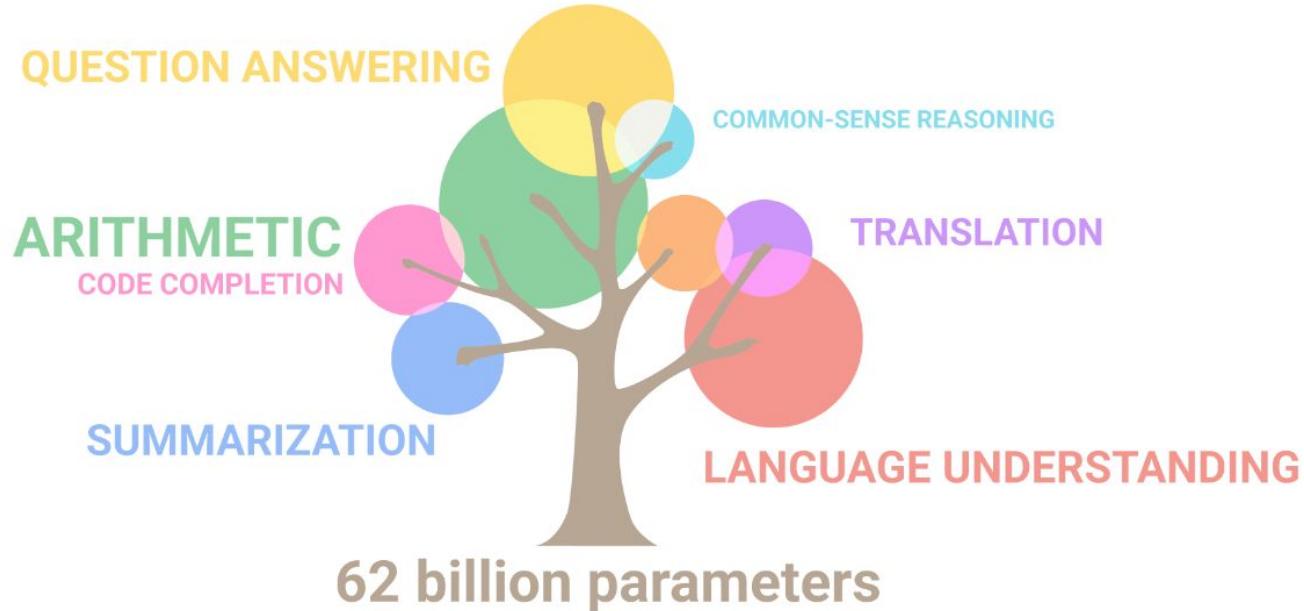
8 billion parameters

Emergent Abilities



8 billion parameters

Emergent Abilities



AI Map

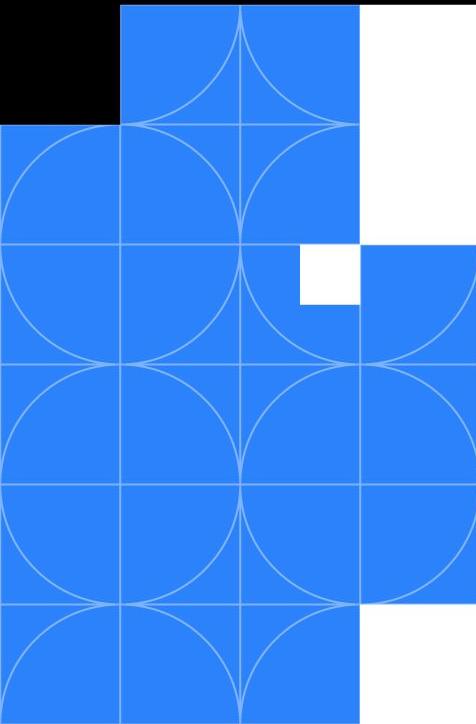
AI

ML

Deep
Learning

Generative AI

Build with AI

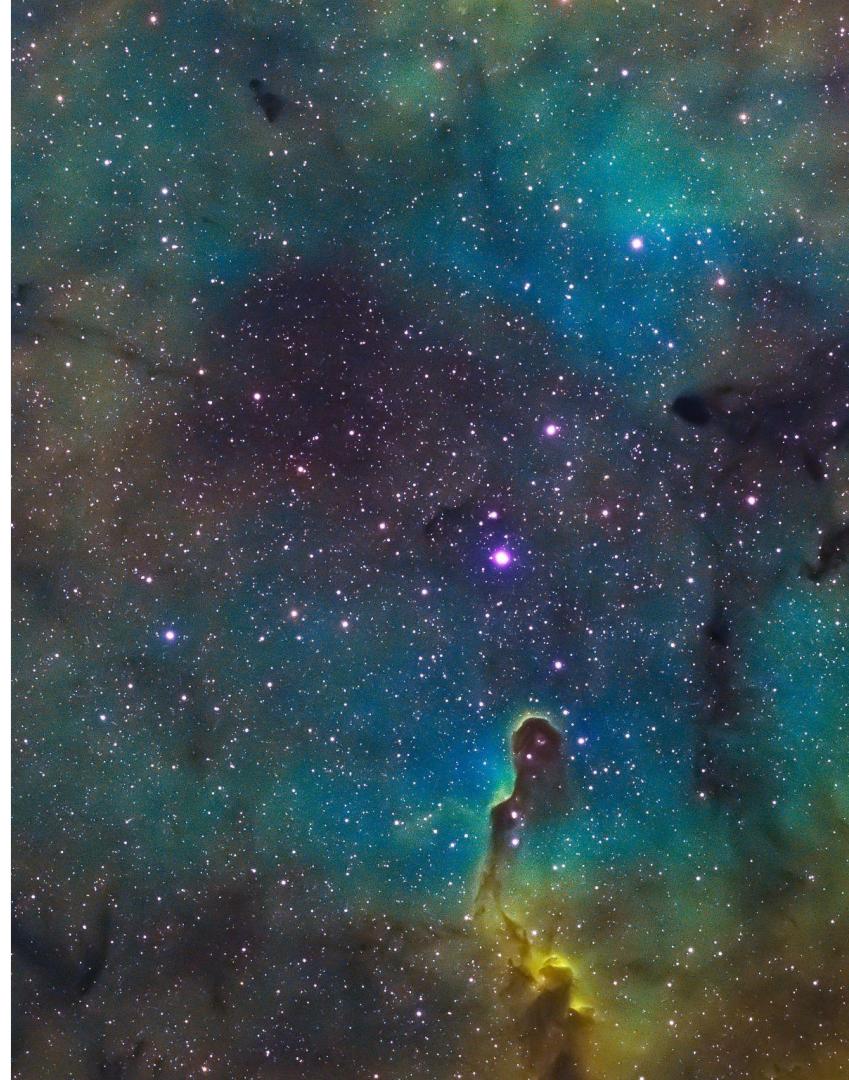


Google Developer Groups
Capital Region

Gemini 101

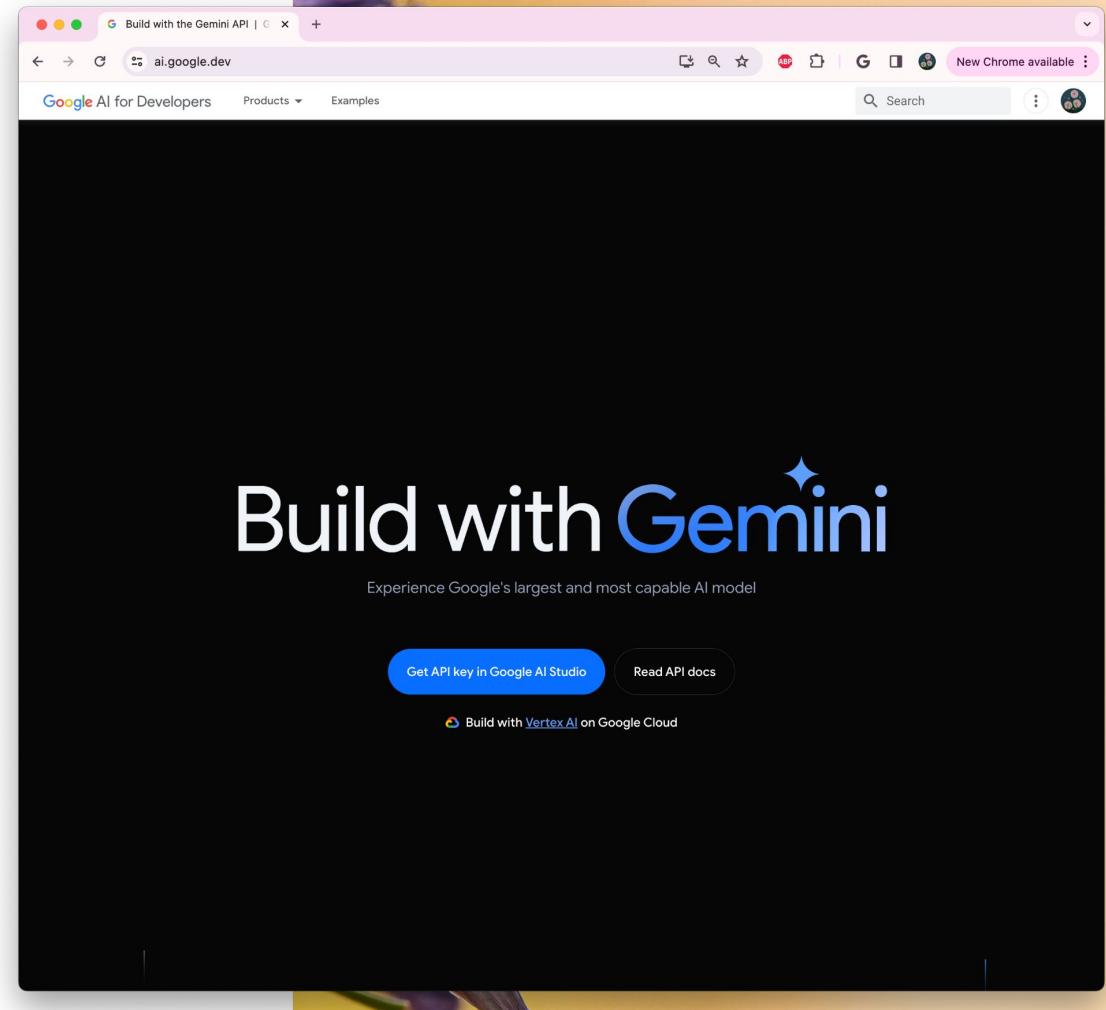


Generalized Multimodal
Intelligence Network



Build with Gemini

ai.google.dev

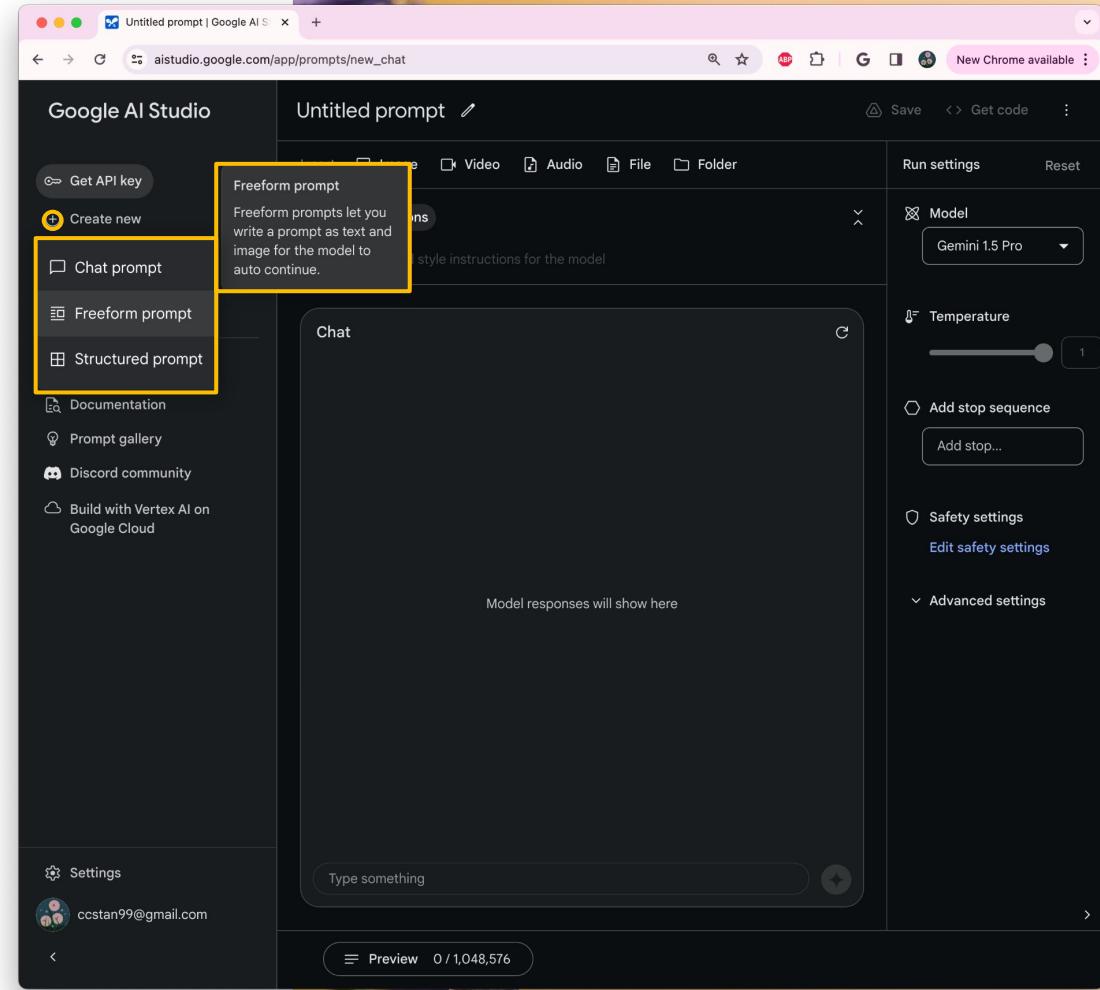


Prototyping with Google AI Studio

The screenshot shows the Google AI Studio interface. On the left, a sidebar lists various options: Get API key, Create new, New tuned model, My library (with 'No prompts yet'), Getting started, Documentation, Prompt gallery, Discord community, and Build with Vertex AI on Google Cloud. At the bottom of the sidebar are Settings and a user profile for ccstan99@gmail.com. The main area is titled 'Untitled prompt' with a 'Chat' section. Below the Chat section is a text input field with the placeholder 'Type something'. To the right of the Chat section are buttons for 'Run settings' and 'Reset'. A 'Model' section dropdown is set to 'Gemini 1.0 Pro'. Below it are sliders for 'Temperature' (set to 0.9), 'Add stop sequence' (with a 'Add stop...' button), and 'Safety settings' (with a 'Edit safety settings' link). At the bottom, there's a 'Preview' button followed by a file count of '-- / 30,720'.

Create new

- Freeform prompt
- Structured prompt
- Chat prompt



Freeform Prompt

Blog post creator

Write a prompt as text and image for the model to auto continue.

Untitled prompt

Insert: Image Video Audio File Folder {{ }} Test input

Run settings Reset

Model Gemini 1.0 Pro Visor

Temperature 0.4

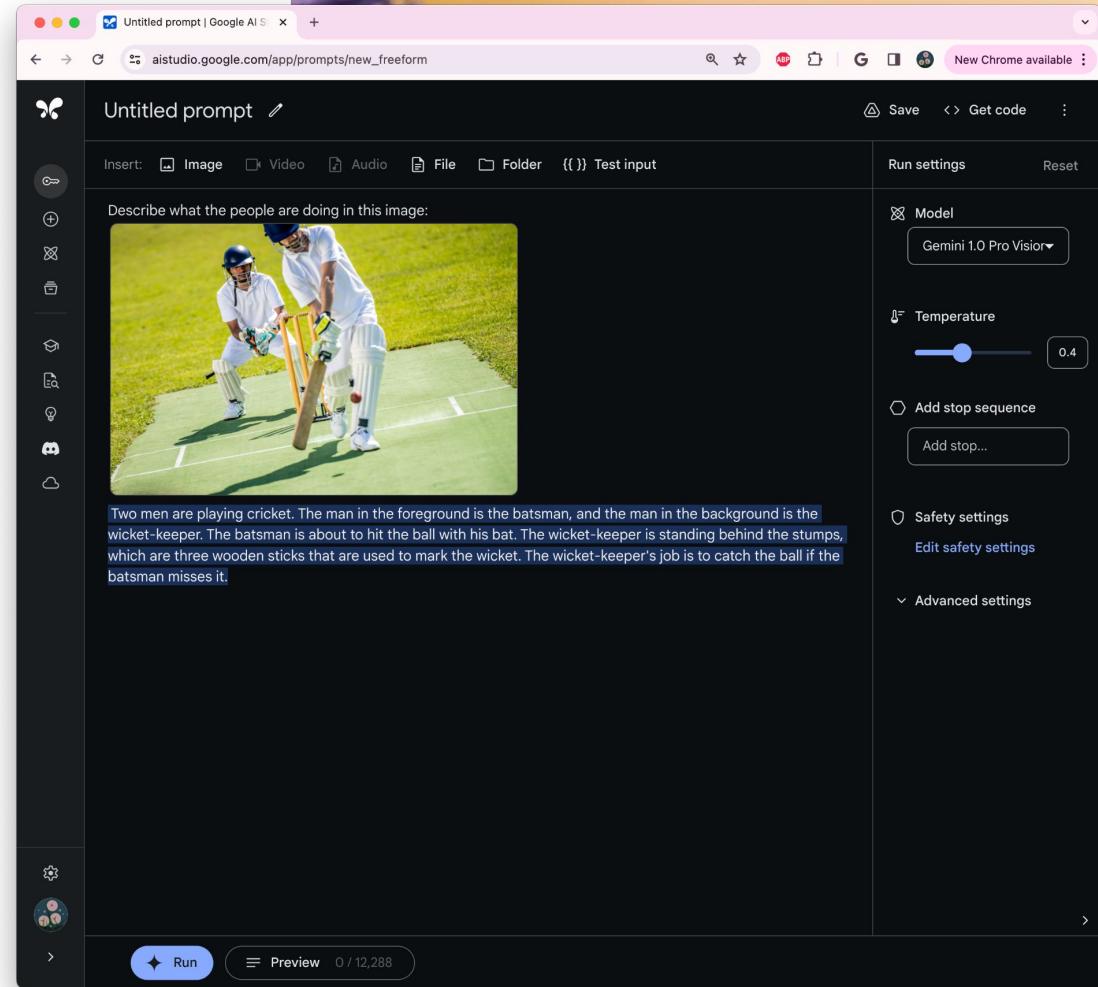
Add stop sequence Add stop...

Safety settings Edit safety settings

Advanced settings

Two men are playing cricket. The man in the foreground is the batsman, and the man in the background is the wicket-keeper. The batsman is about to hit the ball with his bat. The wicket-keeper is standing behind the stumps, which are three wooden sticks that are used to mark the wicket. The wicket-keeper's job is to catch the ball if the batsman misses it.

Run Preview 0 / 12,288



Structured Prompt

Marketing description

Table-based interface for
more complex model
priming and prompting

The screenshot shows the 'Marketing description writer' interface in Aistudio. The interface is divided into several sections:

- Header:** 'Marketing description writer' with a save icon, copy icon, and code icon.
- Left Sidebar:** Icons for Insert (Image), Key, Plus, X, Copy, Paste, Download, Find, Idea, Model, Cloud, and Settings.
- Main Content:** A table showing examples of marketing descriptions. The columns are 'INPUT Product Image:', 'INPUT Target Audience:', and 'OUTPUT Marketing Descri...'. Examples include:
 - A red sports car image for 'Mid-aged men': "Introducing the epitome of power and sophistication - the sleek and captivating sports car. It's more than just a car; it's a symbol of your passion for life and your unwavering commitment to excellence. Embrace the thrill and indulge in the ultimate driving pleasure."
 - A black bicycle image for 'Environmentalists': "Looking for a sustainable and eco-friendly way to get around? Look no further than this black bicycle. Biking is a great way to reduce your carbon footprint and improve your"
- Test Your Prompt:** A section where you can input a product image and target audience to generate a marketing description. An example is shown for a basketball and athletes.
- Run Settings Panel (highlighted with a yellow box):** Includes sections for Model (set to Gemini 1.0 Pro Visior), Temperature (set to 0.9), Add stop sequence (with a 'Add stop...' button), Safety settings (with a 'Edit safety settings' button), and Advanced settings.

Get Code

- Choose Language
- Open in Colab
- Copy to Editor

The screenshot shows the Google AI Studio interface with the "Marketing description writer" template selected. At the top right, there are buttons for "Save a copy" and "Get code". A yellow box highlights the "Get code" button. Below it, a modal window titled "Get code" is displayed, containing instructions to "Create your API key before using the code in your project" and "You can call this prompt from the Gemini API by copying the following code into your project". The "Python" tab is selected, showing the following code:

```
1 """
2 At the command line, only need to run once to install the package via pip:
3
4 $ pip install google-generativeai
5 """
6
7 from pathlib import Path
8 import google.generativeai as genai
9
10 genai.configure(api_key="YOUR_API_KEY")
11
12 # Set up the model
13 generation_config = {
14     "temperature": 0.9,
15     "top_p": 0.95,
```

Below the code, there is a preview of a marketing description for a basketball: "materials that can withstand even the most intense games. Whether you're a seasoned pro or just starting out, this basketball is perfect for anyone who loves the game."

Get API Key

Treat like password

The screenshot shows the Google AI Studio interface with a dark theme. On the left sidebar, there's a 'Get API key' button highlighted with a yellow box. Below it are other options: 'Create new', 'New tuned model', 'My library', 'No prompts yet', 'Getting started', 'Documentation', 'Prompt gallery', 'Discord community', and 'Build with Vertex AI on Google Cloud'. At the bottom of the sidebar are 'Settings' and the user email 'ccstan99@gmail.com'. The main content area is titled 'Get API key' and contains a sub-section titled 'API keys'. It includes a note about creating projects and using Gemini API terms of service. A modal window titled 'Create API key' is open, showing a search bar with 'Generative Language Client' and a result 'gen-lang-client-0598043526'. Below the search bar is a button 'Create API key in existing project' with a tooltip explaining the command: '-X POST "https://generativelanguage.googleapis.com/v1beta/models/gemini-pro:generateContent?key=YOUR_API_KEY'.

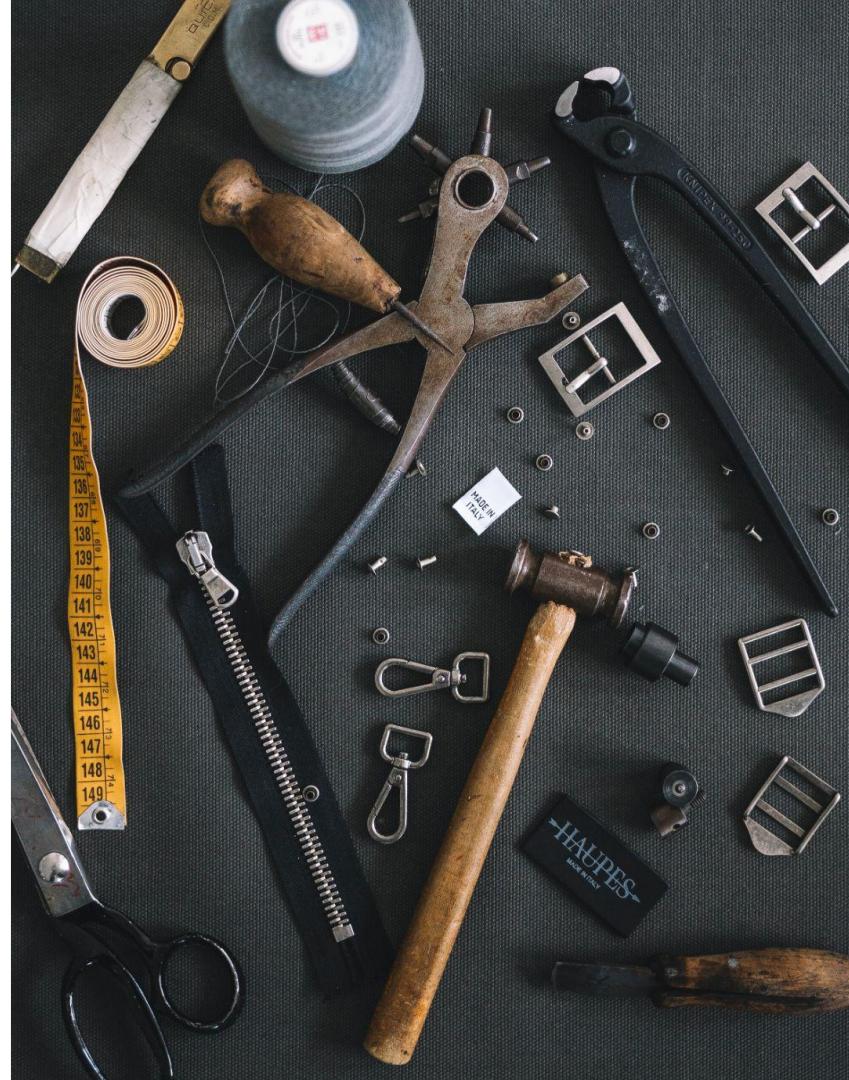
Settings

Tokens

- Words or subwords
- Different LLM tokenizers
- Training data, context window

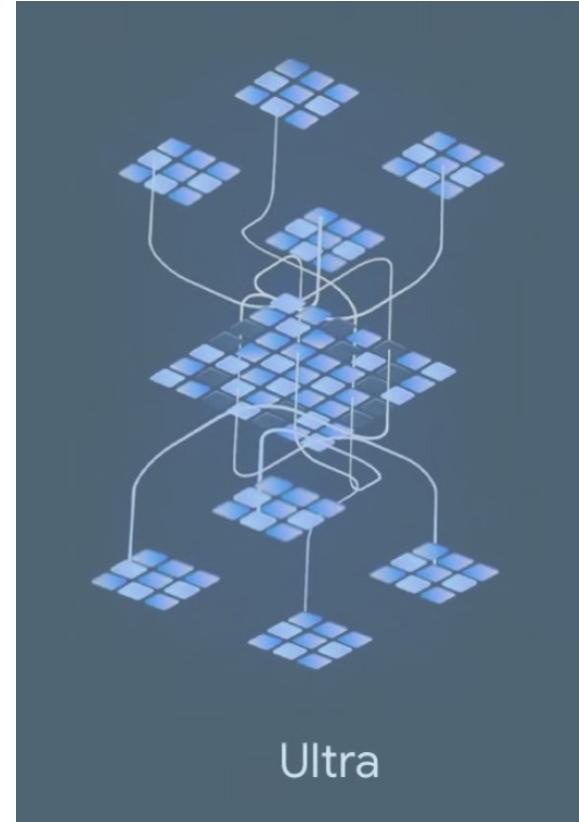
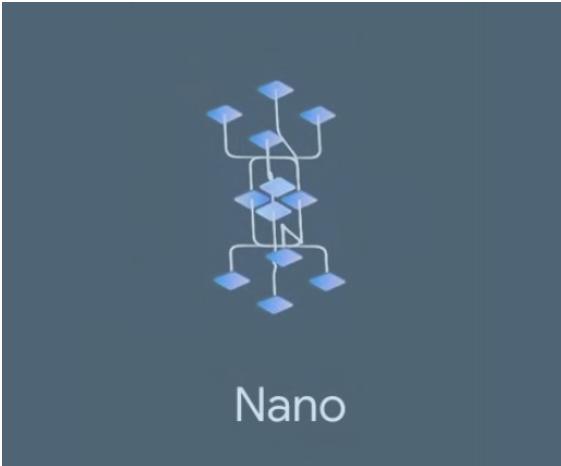
Temperature

- Selected by probability
- Between 0 to 1.0
- Diversity or “creativity”



Settings

Model Sizes

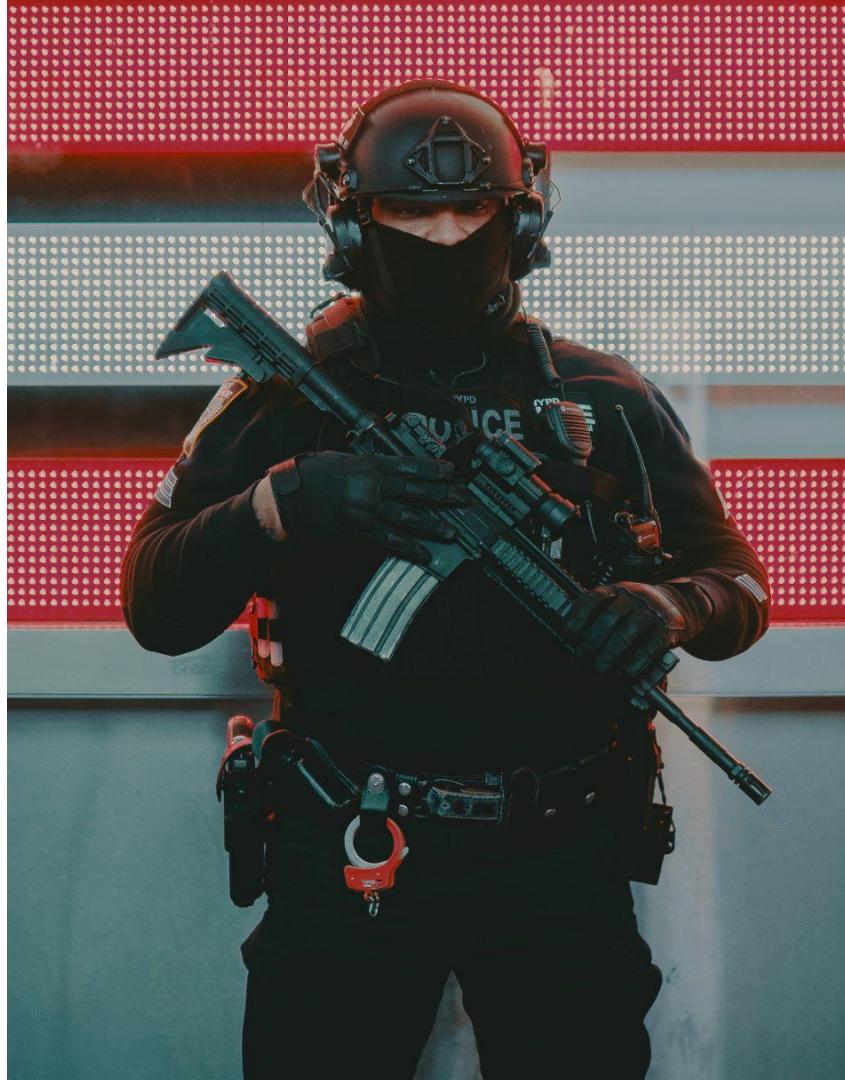


Settings

Safety Ratings

Harm Categories

- Harassment
- Hate Speech
- Sexually Explicit
- Dangerous Content



Settings

Safety Ratings

Harm Categories

- Harassment
- Hate Speech
- Sexually Explicit
- Dangerous Content

Harm Probabilities

- HIGH
- MEDIUM
- LOW
- NEGLIGIBLE

Build with AI

“The hottest new
programming language is
English.”

Andrey Karpathy
OpenAI

Prompt Engineering

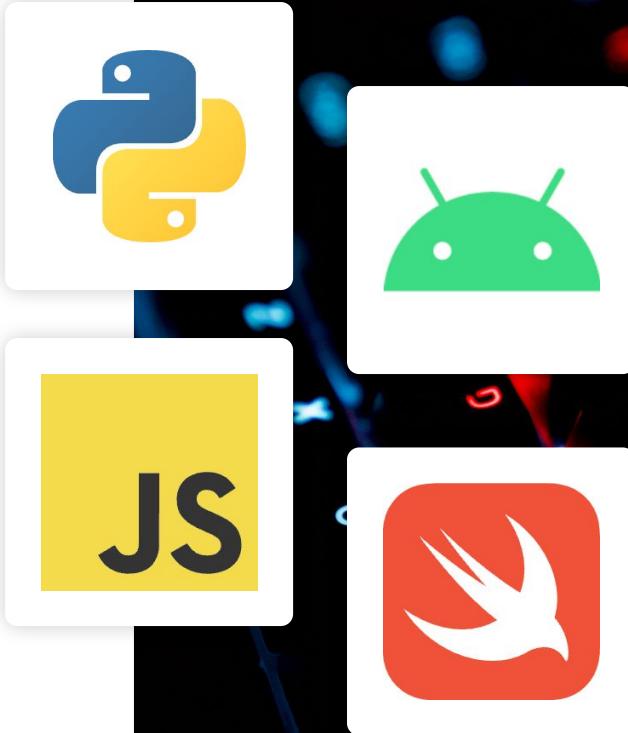
- Clear & Specific Instructions
- Give Examples
- Step by Step



REST APIs

Client libraries for

- Python
- JavaScript
- Android (Kotlin)
- Swift
- cURL



Setup

Install & import libraries

```
$ pip install google-generativeai
```

```
import google.generativeai as genai  
genai.configure(api_key="")
```

Generate Text

Text only prompt

```
model = genai.GenerativeModel('gemini-pro')

response = model.generate_content("Write a story about a
boy and a backpack.")

print(response.text)
```

Generate Text

Text and image prompt

```
model = genai.GenerativeModel('gemini-pro-vision')
img = PIL.Image.open('image.jpg')
response = model.generate_content(["Write a blog based
on this photo.", img])
print(response.text)
```

Chat Conversations

For interactive applications

```
model = genai.GenerativeModel('gemini-pro')
chat = model.start_chat(history=[])

response = chat.send_message("Hello, how are you?")
print(response.text)
```

Demo App

- Text Prompt
- Image Caption
- Chatbot

[https://
cheng2-demo
.streamlit.app](https://cheng2-demo.streamlit.app)

Gemini 101 · Streamlit

cheng2-demo.streamlit.app/Image_Caption

Fork this app

Drag and drop file here
Limit 200MB per file • PNG, JPG, JPEG

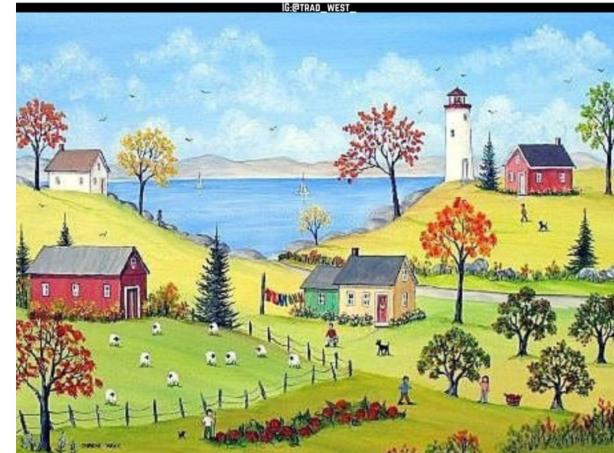
Browse files

30-30-rule.png 2.8MB

X

TO AVOID STRAINING YOUR EYES AT WORK, USE THE 30-30 RULE.
AFTER 30 MINUTES OF WORK, QUIT YOUR JOB AND BUY A 30 ACRE FARM IN THE MIDDLE OF NOWHERE.

© PTRA WEST



Prompt

Write a haiku about this

To avoid straining eyes At work, use 30-30 rule Quit job, buy farm instead

Demo App

- Text Prompt
- Image Caption
- Chatbot

[https://
cheng2-demo
.streamlit.app](https://cheng2-demo.streamlit.app)

Gemini 101 - Streamlit

cheng2-demo.streamlit.app/Chatbot

Hello

Minimal Setup

Text Prompt

Image Caption

Chatbot

Where should I go for summer vacation?

Domestic Destinations:

- Cape Cod, Massachusetts: Picturesque beaches, charming towns, and historical landmarks
- Maui, Hawaii: Tropical beaches, lush rainforests, and stunning sunsets
- Yosemite National Park, California: Stunning granite cliffs, giant sequoia trees, and cascading waterfalls
- Yellowstone National Park, Wyoming: Geysers, hot springs, wildlife sightings, and scenic hikes
- Acadia National Park, Maine: Rugged coastline, granite peaks, and picturesque lighthouses

International Destinations:

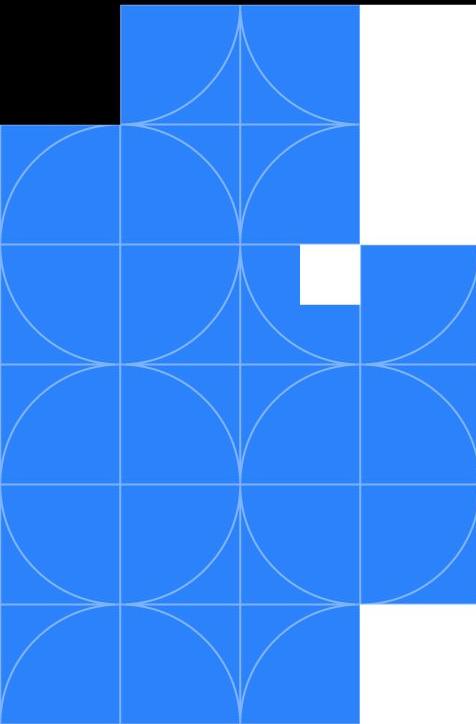
- Cancun, Mexico: White-sand beaches, turquoise waters, and vibrant nightlife
- Punta Cana, Dominican Republic: All-inclusive resorts, palm-fringed beaches, and crystal-clear waters
- Santorini, Greece: Stunning whitewashed villages, dramatic cliffs, and picturesque sunsets
- Barcelona, Spain: Vibrant city with stunning architecture, world-class cuisine, and lively beaches
- Rome, Italy: Historical ruins, iconic landmarks, delicious food, and lively piazzas

Other Factors to Consider:

- **Interests:** Determine what you enjoy doing on vacation, such as hiking, swimming, sightseeing, or relaxing.
- **Budget:** Set a realistic budget for accommodations, transportation, food, and activities.
- **Travel dates:** Peak season tends to be more expensive and crowded. Consider traveling during shoulder seasons or off-peak periods.
- **Proximity to airports:** Consider destinations that are easily accessible by plane from your home city, especially if you have limited time.

Message to Gemini...

Build with AI



Google Developer Groups
Capital Region

Now Your Turn!

Progress Tracker

1. Prototyping in Google AI Studio
2. Coding with Python Quickstart
3. Build & Deploy Streamlit App

bit.ly/cheng2-workshop

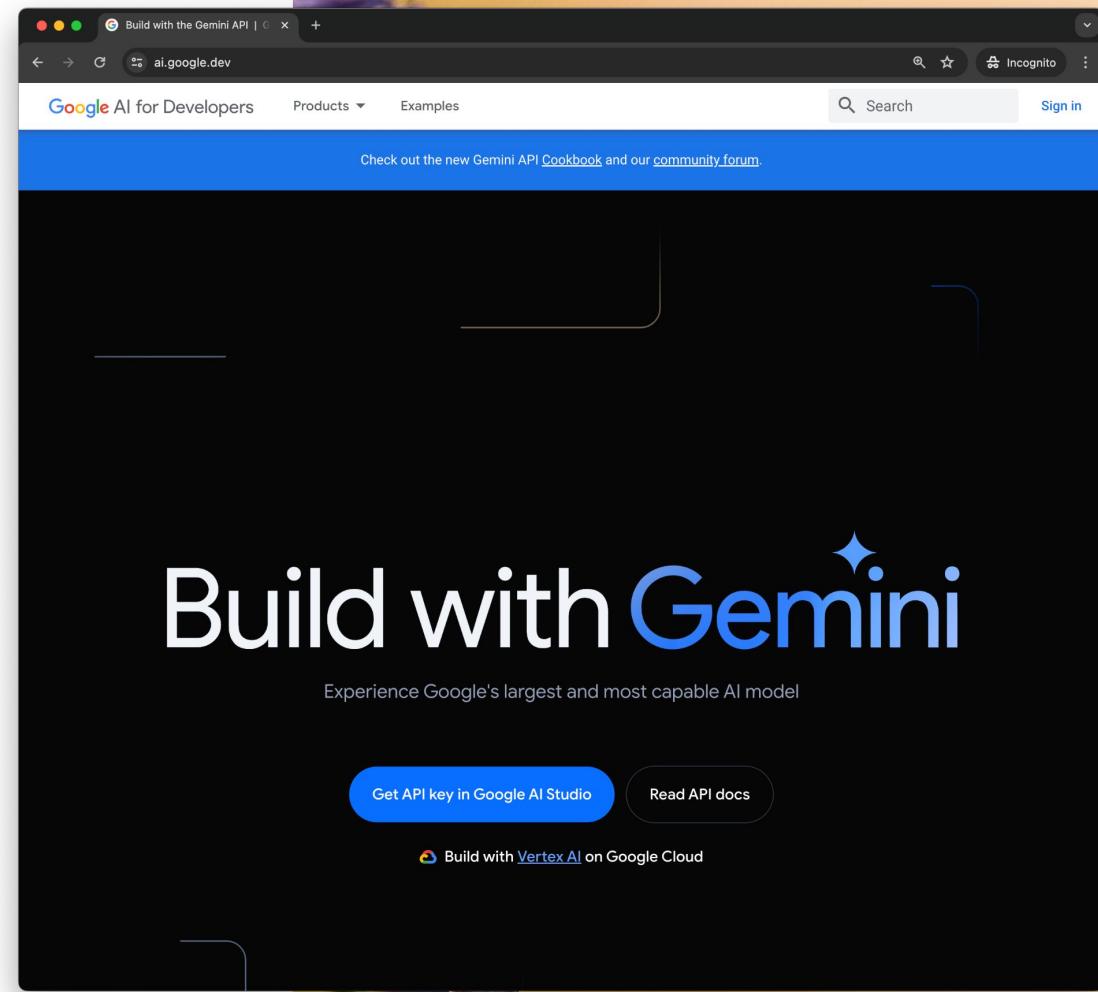
The screenshot shows a Google Sheets document titled "Cheng2 Workshop - Progress Tracker". The spreadsheet has a header row with columns A through Y. Row 1 contains the column headers: Name, Cheng, Cheng, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y. Rows 2 and 3 show the user's information: Gmail (cheng2_ccstan) and GitHub ID (cheng2_ccstan). Rows 5 through 39 list various workshop tasks, each with a checkbox in column C. The tasks are categorized into sections: Prototyping (Google AI Studio, Freeform prompt, Prompt gallery, Structured prompt, Get code, API key -- SAVE for later), Coding (Python Quickstart, Open colab, Setup, Add secrets, List models, Generate text (2), Chat conversations), Streamlit (Run Demo, Fork this app, Sign in with GitHub, Advance settings + secrets, Codespace + Simple Brows, Check deployed, Edit & test, Commit + Sync + Re-deploy, Share, App URL), and other items (34, 35, 36, 37, 38, 39). The date "2024-04-26" is visible at the bottom left of the sheet.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
1	Name	Cheng	Cheng																						
2	Gmail	cheng2_ccstan																							
3	Github ID	cheng2_ccstan																							
4																									
5	Prototyping																								
6	Google AI Studio																								
7	Freeform prompt																								
8	Prompt gallery																								
9	Structured prompt																								
10	Get code																								
11	API key -- SAVE for later																								
12																									
13	Coding																								
14	Python Quickstart																								
15	Open colab																								
16	Setup																								
17	Add secrets																								
18	List models																								
19	Generate text (2)																								
20	Chat conversations																								
21																									
22	Streamlit																								
23	Run Demo																								
24	Fork this app																								
25	Sign in with GitHub																								
26	Advance settings + secrets																								
27	Codespace + Simple Brows																								
28	Check deployed																								
29	Edit & test																								
30	Commit + Sync + Re-deploy																								
31	Share																								
32	App URL																								
33																									
34																									
35																									
36																									
37																									
38																									
39																									

Google AI Studio

- Freeform prompt
- Prompt gallery
- Structured prompt
- Get code
- API key

ai.google.dev



Colab Notebook

- Setup
- Add secrets
- List models
- Generate text
- Chat conversations

[ai.google.dev/tutorials/
python_quickstart](https://ai.google.dev/tutorials/python_quickstart)

The screenshot shows a Google Colab notebook titled "python.ipynb". The left sidebar contains a "Table of contents" with sections like "Copyright 2024 Google LLC.", "Get started with the Gemini API: Python", "Prerequisites", "Setup", "List models", "Generate text from text inputs", "Generate text from image and text inputs", "Chat conversations", "Count tokens", "Use embeddings", "Advanced use cases", "Safety settings", "Encode messages", "Multi-turn conversations", "Generation configuration", and "What's next". The main content area displays the "Get started with the Gemini API: Python" section, which includes a "View on Google AI" button, a "Run in Google Colab" button, and a "View source on GitHub" button. It also contains a brief description of the quickstart, a numbered list of steps, and a "Prerequisites" section.

python.ipynb

File Edit View Insert Runtime Tools Help

Table of contents

Copyright 2024 Google LLC.

Licensed under the Apache License, Version 2.0 (the "License");

Get started with the Gemini API: Python

Prerequisites

Setup

- Install the Python SDK
- Import packages
- Setup your API key

List models

Generate text from text inputs

Generate text from image and text inputs

Chat conversations

Count tokens

Use embeddings

Advanced use cases

- Safety settings
- Encode messages
- Multi-turn conversations
- Generation configuration

What's next

+ Section

Copyright 2024 Google LLC.

Licensed under the Apache License, Version 2.0 (the "License");

Show code

Get started with the Gemini API: Python

View on Google AI Run in Google Colab View source on GitHub

This quickstart demonstrates how to use the Python SDK for the Gemini API, which gives you access to Google's Gemini large language models. In this quickstart, you will learn how to:

1. Set up your development environment and API access to use Gemini.
2. Generate text responses from text inputs.
3. Generate text responses from multimodal inputs (text and images).
4. Use Gemini for multi-turn conversations (chat).
5. Use embeddings for large language models.

Prerequisites

You can run this quickstart in [Google Colab](#), which runs this notebook directly in the browser and does not require additional environment configuration.

Alternatively, to complete this quickstart locally, ensure that your development environment meets the following requirements:

- Python 3.9+
- An installation of `jupyter` to run the notebook.

Setup

Colab Notebook

- Setup
- Add secrets
- List models
- Generate text
- Chat conversations

[ai.google.dev/tutorials/
python_quickstart](https://ai.google.dev/tutorials/python_quickstart)

The screenshot shows a Google Colab notebook titled "python.ipynb". The left sidebar displays a "Secrets" panel where a secret named "GOOGLE_API_KEY" has been added with a value represented by a series of dots. Below this, code snippets demonstrate how to access the secret in Python:

```
from google.colab import userdata
userdata.get('secretName')
```

and

```
# Or use `os.getenv('GOOGLE_API_KEY')` to fetch an environment variable.
GOOGLE_API_KEY=userdata.get('GOOGLE_API_KEY')

genai.configure(api_key=GOOGLE_API_KEY)
```

The main content area contains two sections: "Setup your API key" and "List models". The "Setup your API key" section provides instructions for obtaining an API key from Google AI Studio and adding it to the secrets manager. It also shows how to pass the key to the Gemini API using environment variables or direct assignment. The "List models" section shows a code snippet for listing available Gemini models:

```
[ ] for m in genai.list_models():
    if 'generateContent' in m.supported_generation_methods:
        print(m.name)
```

Note: For detailed information about the available models, including their capabilities and rate limits, see [Gemini models](#). There are options for requesting [rate limit increases](#). The rate limit for Gemini-Pro models is 60 requests per minute (RPM).

The genai package also supports the PaLM family of models, but only the Gemini models support the generic, multimodal capabilities of the generateContent method.

Streamlit App

- Fork the app
- Sign in with GitHub
- Setup secrets
- Edit & test
- Deploy & share!

[https://
cheng2-demo
.streamlit.app](https://cheng2-demo.streamlit.app)

The screenshot shows a Streamlit application window titled "Gemini 101 - Streamlit". The sidebar on the left lists several demo options: "Hello", "Minimal Setup", "Text Prompt", "Image Caption", and "Chatbot". Below the sidebar is a green button with the text "Select a demo above.". The main content area features a large heading "Welcome to the Gemini Demo! 🌟" and a sub-section titled "Resources to continue learning:" which lists links to Gemini + AI Studio, Gemini Quickstart Tutorials, Streamlit Tutorials, Introduction to Generative AI, and DeepLearning.AI. At the bottom of the main content area is a section titled "Feedback & stay in touch:" with a link to a feedback survey.

Gemini 101 - Streamlit

cheng2-demo.streamlit.app

Hello

Minimal Setup

Text Prompt

Image Caption

Chatbot

Select a demo above.

Welcome to the Gemini Demo! 🌟

This is the companion code for the Intro to AI + Gemini 101 workshops to quickly build and deploy Gemini-powered apps. ➔ Select a demo from the sidebar for ideas and adapt for your own projects!

Resources to continue learning:

- [Gemini + AI Studio](#) -- Prototyping environment with Gemini
- [Gemini Quickstart Tutorials](#) -- Examples to build in different programming languages
- [Streamlit Tutorials](#) -- Build web apps calling LLMs
- [Introduction to Generative AI](#) -- Google learning paths with videos & exercises
- [DeepLearning.AI](#) -- Full specializations + many short 1-hour courses

Feedback & stay in touch:

- [Feedback survey](#)

Free Credits!

- Google Cloud Platform
- Free tier + \$5 credit
- Cloud Innovator for Cloud Skill Boost training

[trygcp.dev/e/
build-ai-DC02](http://trygcp.dev/e/build-ai-DC02)

The screenshot shows a Google Cloud web interface. At the top, it displays "trygcp.dev/e/build-ai-DC02". The header includes the Google Cloud logo and a "Sign out" link. A greeting message says "Hi, welcome Cheng2 Workshops (cheng2.workshops@gmail.com)". Below this, the title "GDG Capital Region Build w/AI" is displayed. A message states: "Your credit will allow you to use Google Cloud [Free Tier products](#). It has an amount of **\$5**. Once redeemed, it will be valid for **180 days** or until the balance is depleted if you use non-free services." A button labeled "CLICK HERE TO ACCESS YOUR CREDITS" is present. At the bottom right, there is a small icon and the text "GCP Credits #5549820".

Resources

bit.ly/cheng2-slides

1. Gemini + AI Studio

Prototyping environment with Gemini

2. Gemini Quickstart Tutorials

Examples to build in different programming languages

3. Streamlit Tutorials

Build web apps calling LLMs

4. Introduction to Generative AI

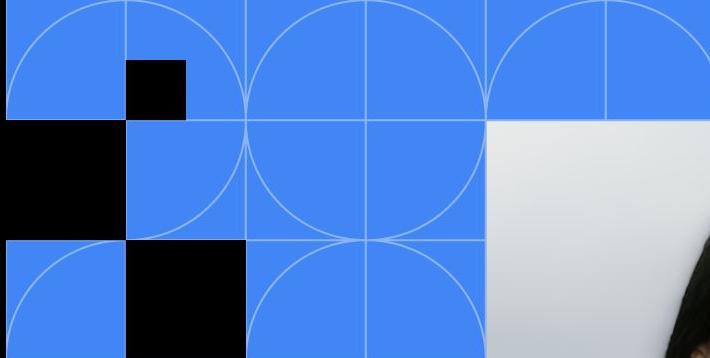
Google learning paths with videos & exercises

5. DeepLearning.AI

Full specializations + many short 1-hour courses



Build with AI



ChengCheng Tan

ccstan99@gmail.com

cheng2-tan

@cheng2_tan

Google Developer Groups
Capital Region

Build with AI

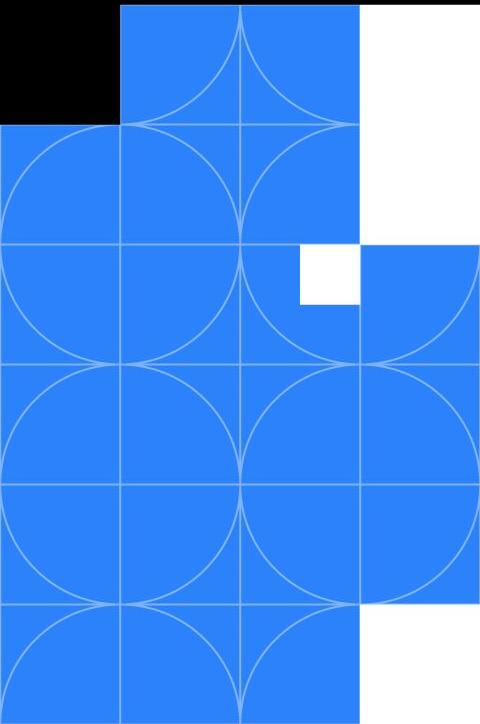


Image Credits

- Alpaca by Samantha Amidon on Unsplash
- Calligraphy Pen by Digital Content Writers India on Unsplash
- Calculator by Towfiq barbhuiya on Unsplash
- Fruits & Veggies by Brooke Lark on Unsplash
- Neuron by Hal Gatewood on Unsplash
- Scrabble Tiles by Merve Sehirli Nasir on Unsplash
- Runners Training by Fitsum Admasu on Unsplash
- Thumbs Up by Johan Godinez on Unsplash
- Butterfly & Flowers by Birger Strahl on Unsplash
- Tools by Haupes on Unsplash
- Security Guard by Alec Favale on Unsplash
- Reaching Hands by Matheus Viana on Unsplash
- Keyboard by Mohammad Rahmani on Unsplash
- Laptop by Andras Vas on Unsplash
- Stars by Aldebaran S on Unsplash
- Everything Everywhere All at Once (2022)
- Boy & Robot, Winter Plant, Granny images generated on stability.ai