

welcome-to-colab

July 27, 2025

1 Welcome to Colab!

1.1 Access Popular LLMs via Google-Colab-AI Without an API Key

Users with Colab's paid plans have free access to most popular LLMs via google-colab-ai Python library. For more details, refer to the [getting started with google colab ai](#).

```
from google.colab import ai
response = ai.generate_text("What is the capital of France?")
print(response)
```

1.2 Explore 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? * Go to [Google AI Studio](#) and log in with your Google account. * [Create an API key](#). * Use a quickstart for [Python](#), or call the REST API using [curl](#).

Discover Gemini's advanced capabilities * Play with Gemini [multimodal outputs](#), mixing text and images in an iterative way. * Discover the [multimodal Live API](#) (demo [here](#)). * Learn how to [analyze images and detect items in your pictures](#) using Gemini (bonus, there's a [3D version](#) as well!). * Unlock the power of [Gemini thinking model](#), capable of solving complex task with its inner thoughts.

Explore complex use cases * Use [Gemini grounding capabilities](#) to create a report on a company based on what the model can find on internet. * Extract [invoices and form data from PDF](#) in a structured way. * Create [illustrations based on a whole book](#) using Gemini large context window and Imagen.

To learn more, check out the [Gemini cookbook](#) or visit the [Gemini API documentation](#).

Colab now has AI features powered by [Gemini](#). The video below provides information on how to use these features, whether you're new to Python, or a seasoned veteran.

What is Colab?

Colab, or “Colaboratory”, allows you to write and execute Python in your browser, with - Zero configuration required - Access to GPUs free of charge - Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch [Introduction to Colab](#) or [Colab Features You May Have Missed](#) to learn more, or

just get started below!

1.3 Getting started

The document you are reading is not a static web page, but an interactive environment called a **Colab notebook** that lets you write and execute code.

For example, here is a **code cell** with a short Python script that computes a value, stores it in a variable, and prints the result:

```
[ ]: seconds_in_a_day = 24 * 60 * 60
      seconds_in_a_day
```

```
[ ]: 86400
```

To execute the code in the above cell, select it with a click and then either press the play button to the left of the code, or use the keyboard shortcut “Command/Ctrl+Enter”. To edit the code, just click the cell and start editing.

Variables that you define in one cell can later be used in other cells:

```
[ ]: seconds_in_a_week = 7 * seconds_in_a_day
      seconds_in_a_week
```

```
[ ]: 604800
```

Colab notebooks allow you to combine **executable code** and **rich text** in a single document, along with **images**, **HTML**, **LaTeX** and more. When you create your own Colab notebooks, they are stored in your Google Drive account. You can easily share your Colab notebooks with co-workers or friends, allowing them to comment on your notebooks or even edit them. To learn more, see [Overview of Colab](#). To create a new Colab notebook you can use the File menu above, or use the following link: [create a new Colab notebook](#).

Colab notebooks are Jupyter notebooks that are hosted by Colab. To learn more about the Jupyter project, see [jupyter.org](#).

1.4 Data science

With Colab you can harness the full power of popular Python libraries to analyze and visualize data. The code cell below uses **numpy** to generate some random data, and uses **matplotlib** to visualize it. To edit the code, just click the cell and start editing.

You can import your own data into Colab notebooks from your Google Drive account, including from spreadsheets, as well as from Github and many other sources. To learn more about importing data, and how Colab can be used for data science, see the links below under Working with Data.

```
[ ]: import numpy as np
      import IPython.display as display
      from matplotlib import pyplot as plt
      import io
      import base64
```

```

ys = 200 + np.random.randn(100)
x = [x for x in range(len(ys))]

fig = plt.figure(figsize=(4, 3), facecolor='w')
plt.plot(x, ys, '-')
plt.fill_between(x, ys, 195, where=(ys > 195), facecolor='g', alpha=0.6)
plt.title("Sample Visualization", fontsize=10)

data = io.BytesIO()
plt.savefig(data)
image = F"data:image/png;base64,{base64.b64encode(data.getvalue()).decode()}"
alt = "Sample Visualization"
display.display(display.Markdown(F"!!! [{alt}] ({image}) """))
plt.close(fig)

```

Colab notebooks execute code on Google's cloud servers, meaning you can leverage the power of Google hardware, including GPUs and TPUs, regardless of the power of your machine. All you need is a browser.

For example, if you find yourself waiting for **pandas** code to finish running and want to go faster, you can switch to a GPU Runtime and use libraries like [RAPIDS cuDF](#) that provide zero-code-change acceleration.

To learn more about accelerating pandas on Colab, see the [10 minute guide](#) or [US stock market data analysis demo](#).

1.5 Machine learning

With Colab you can import an image dataset, train an image classifier on it, and evaluate the model, all in just [a few lines of code](#).

Colab is used extensively in the machine learning community with applications including: - Getting started with TensorFlow - Developing and training neural networks - Experimenting with TPUs - Disseminating AI research - Creating tutorials

To see sample Colab notebooks that demonstrate machine learning applications, see the machine learning examples below.

1.6 More Resources

1.6.1 Working with Notebooks in Colab

- [Overview of Colab](#)
- [Guide to Markdown](#)
- [Importing libraries and installing dependencies](#)
- [Saving and loading notebooks in GitHub](#)
- [Interactive forms](#)

- [Interactive widgets](#)

Working with Data

- [Loading data: Drive, Sheets, and Google Cloud Storage](#)
- [Charts: visualizing data](#)
- [Getting started with BigQuery](#)

1.6.2 Machine Learning

These are a few of the notebooks related to Machine Learning, including Google's online Machine Learning course. See the [full course website](#) for more. - [Intro to Pandas DataFrame](#) - [Intro to RAPIDS cuDF to accelerate pandas](#) - [Getting Started with cuML's accelerator mode](#) - [Linear regression with tf.keras using synthetic data](#)

Using Accelerated Hardware

- [TensorFlow with GPUs](#)
- [TPUs in Colab](#)

1.6.3 Featured examples

- [Retraining an Image Classifier](#): Build a Keras model on top of a pre-trained image classifier to distinguish flowers.
- [Text Classification](#): Classify IMDB movie reviews as either *positive* or *negative*.
- [Style Transfer](#): Use deep learning to transfer style between images.
- [Multilingual Universal Sentence Encoder Q&A](#): Use a machine learning model to answer questions from the SQuAD dataset.
- [Video Interpolation](#): Predict what happened in a video between the first and the last frame.

```
[12]: from google.colab import drive
drive.mount('/content/drive')
```

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

```
[2]: pip install diffusers --upgrade
```

```
Requirement already satisfied: diffusers in /usr/local/lib/python3.11/dist-packages (0.34.0)
Requirement already satisfied: importlib_metadata in /usr/local/lib/python3.11/dist-packages (from diffusers) (8.7.0)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from diffusers) (3.18.0)
Requirement already satisfied: huggingface-hub>=0.27.0 in /usr/local/lib/python3.11/dist-packages (from diffusers) (0.33.4)
Requirement already satisfied: numpy in /usr/local/lib/python3.11/dist-packages (from diffusers) (2.0.2)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.11/dist-packages (from diffusers) (2024.11.6)
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages
```

```

packages (from diffusers) (2.32.3)
Requirement already satisfied: safetensors>=0.3.1 in
/usr/local/lib/python3.11/dist-packages (from diffusers) (0.5.3)
Requirement already satisfied: Pillow in /usr/local/lib/python3.11/dist-packages
(from diffusers) (11.3.0)
Requirement already satisfied: fsspec>=2023.5.0 in
/usr/local/lib/python3.11/dist-packages (from huggingface-
hub>=0.27.0->diffusers) (2025.3.0)
Requirement already satisfied: packaging>=20.9 in
/usr/local/lib/python3.11/dist-packages (from huggingface-
hub>=0.27.0->diffusers) (25.0)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.11/dist-
packages (from huggingface-hub>=0.27.0->diffusers) (6.0.2)
Requirement already satisfied: tqdm>=4.42.1 in /usr/local/lib/python3.11/dist-
packages (from huggingface-hub>=0.27.0->diffusers) (4.67.1)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/usr/local/lib/python3.11/dist-packages (from huggingface-
hub>=0.27.0->diffusers) (4.14.1)
Requirement already satisfied: hf-xet<2.0.0,>=1.1.2 in
/usr/local/lib/python3.11/dist-packages (from huggingface-
hub>=0.27.0->diffusers) (1.1.5)
Requirement already satisfied: zipp>=3.20 in /usr/local/lib/python3.11/dist-
packages (from importlib_metadata->diffusers) (3.23.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.11/dist-packages (from requests->diffusers) (3.4.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-
packages (from requests->diffusers) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.11/dist-packages (from requests->diffusers) (2.5.0)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.11/dist-packages (from requests->diffusers) (2025.7.14)

```

```
[3]: pip install invisible_watermark transformers accelerate safetensors
```

```

Collecting invisible_watermark
  Downloading invisible_watermark-0.2.0-py3-none-any.whl.metadata (8.2 kB)
Requirement already satisfied: transformers in /usr/local/lib/python3.11/dist-
packages (4.53.3)
Requirement already satisfied: accelerate in /usr/local/lib/python3.11/dist-
packages (1.9.0)
Requirement already satisfied: safetensors in /usr/local/lib/python3.11/dist-
packages (0.5.3)
Requirement already satisfied: Pillow>=6.0.0 in /usr/local/lib/python3.11/dist-
packages (from invisible_watermark) (11.3.0)
Requirement already satisfied: PyWavelets>=1.1.1 in
/usr/local/lib/python3.11/dist-packages (from invisible_watermark) (1.8.0)
Requirement already satisfied: numpy>=1.17.0 in /usr/local/lib/python3.11/dist-
packages (from invisible_watermark) (2.0.2)

```

Requirement already satisfied: opencv-python>=4.1.0.25 in /usr/local/lib/python3.11/dist-packages (from invisible_watermark) (4.12.0.88)

Requirement already satisfied: torch in /usr/local/lib/python3.11/dist-packages (from invisible_watermark) (2.6.0+cu124)

Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from transformers) (3.18.0)

Requirement already satisfied: huggingface-hub<1.0,>=0.30.0 in /usr/local/lib/python3.11/dist-packages (from transformers) (0.33.4)

Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from transformers) (25.0)

Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.11/dist-packages (from transformers) (6.0.2)

Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.11/dist-packages (from transformers) (2024.11.6)

Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from transformers) (2.32.3)

Requirement already satisfied: tokenizers<0.22,>=0.21 in /usr/local/lib/python3.11/dist-packages (from transformers) (0.21.2)

Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.11/dist-packages (from transformers) (4.67.1)

Requirement already satisfied: psutil in /usr/local/lib/python3.11/dist-packages (from accelerate) (5.9.5)

Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub<1.0,>=0.30.0->transformers) (2025.3.0)

Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub<1.0,>=0.30.0->transformers) (4.14.1)

Requirement already satisfied: hf-xet<2.0.0,>=1.1.2 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub<1.0,>=0.30.0->transformers) (1.1.5)

Requirement already satisfied: networkx in /usr/local/lib/python3.11/dist-packages (from torch->invisible_watermark) (3.5)

Requirement already satisfied: jinja2 in /usr/local/lib/python3.11/dist-packages (from torch->invisible_watermark) (3.1.6)

Collecting nvidia-cuda-nvrtc-cu12==12.4.127 (from torch->invisible_watermark)
 Downloading nvidia_cuda_nvrtc_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)

Collecting nvidia-cuda-runtime-cu12==12.4.127 (from torch->invisible_watermark)
 Downloading nvidia_cuda_runtime_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)

Collecting nvidia-cuda-cupti-cu12==12.4.127 (from torch->invisible_watermark)
 Downloading nvidia_cuda_cupti_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)

Collecting nvidia-cudnn-cu12==9.1.0.70 (from torch->invisible_watermark)
 Downloading nvidia_cudnn_cu12-9.1.0.70-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)

Collecting nvidia-cublas-cu12==12.4.5.8 (from torch->invisible_watermark)

Downloading nvidia_cublas_cu12-12.4.5.8-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
 Collecting nvidia-cufft-cu12==11.2.1.3 (from torch->invisible_watermark)
 Downloading nvidia_cufft_cu12-11.2.1.3-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
 Collecting nvidia-curand-cu12==10.3.5.147 (from torch->invisible_watermark)
 Downloading nvidia_curand_cu12-10.3.5.147-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
 Collecting nvidia-cusolver-cu12==11.6.1.9 (from torch->invisible_watermark)
 Downloading nvidia_cusolver_cu12-11.6.1.9-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)
 Collecting nvidia-cusparselt-cu12==0.6.2 (from torch->invisible_watermark)
 Downloading nvidia_cusparselt_cu12-0.6.2-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)
 Requirement already satisfied: nvidia-cusparselt-cu12==0.6.2 in /usr/local/lib/python3.11/dist-packages (from torch->invisible_watermark) (0.6.2)
 Requirement already satisfied: nvidia-nccl-cu12==2.21.5 in /usr/local/lib/python3.11/dist-packages (from torch->invisible_watermark) (2.21.5)
 Requirement already satisfied: nvidia-nvtx-cu12==12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch->invisible_watermark) (12.4.127)
 Collecting nvidia-nvjitlink-cu12==12.4.127 (from torch->invisible_watermark)
 Downloading nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
 Requirement already satisfied: triton==3.2.0 in /usr/local/lib/python3.11/dist-packages (from torch->invisible_watermark) (3.2.0)
 Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.11/dist-packages (from torch->invisible_watermark) (1.13.1)
 Requirement already satisfied: mpmath<1.4, >=1.1.0 in /usr/local/lib/python3.11/dist-packages (from sympy==1.13.1->torch->invisible_watermark) (1.3.0)
 Requirement already satisfied: charset-normalizer<4, >=2 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (3.4.2)
 Requirement already satisfied: idna<4, >=2.5 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (3.10)
 Requirement already satisfied: urllib3<3, >=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (2.5.0)
 Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (2025.7.14)
 Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.11/dist-packages (from jinja2->torch->invisible_watermark) (3.0.2)
 Downloading invisible_watermark-0.2.0-py3-none-any.whl (1.6 MB)

1.6/1.6 MB

27.3 MB/s eta 0:00:00

```

Downloading nvidia_cublas_cu12-12.4.5.8-py3-none-manylinux2014_x86_64.whl
(363.4 MB)
363.4/363.4 MB
4.0 MB/s eta 0:00:00
Downloading nvidia_cuda_cupti_cu12-12.4.127-py3-none-
manylinux2014_x86_64.whl (13.8 MB)
13.8/13.8 MB
121.0 MB/s eta 0:00:00
Downloading nvidia_cuda_nvrtc_cu12-12.4.127-py3-none-
manylinux2014_x86_64.whl (24.6 MB)
24.6/24.6 MB
84.1 MB/s eta 0:00:00
Downloading nvidia_cuda_runtime_cu12-12.4.127-py3-none-
manylinux2014_x86_64.whl (883 kB)
883.7/883.7 kB
61.3 MB/s eta 0:00:00
Downloading nvidia_cudnn_cu12-9.1.0.70-py3-none-manylinux2014_x86_64.whl
(664.8 MB)
664.8/664.8 MB
2.2 MB/s eta 0:00:00
Downloading nvidia_cufft_cu12-11.2.1.3-py3-none-manylinux2014_x86_64.whl
(211.5 MB)
211.5/211.5 MB
6.2 MB/s eta 0:00:00
Downloading nvidia_curand_cu12-10.3.5.147-py3-none-
manylinux2014_x86_64.whl (56.3 MB)
56.3/56.3 MB
12.4 MB/s eta 0:00:00
Downloading nvidia_cusolver_cu12-11.6.1.9-py3-none-
manylinux2014_x86_64.whl (127.9 MB)
127.9/127.9 MB
8.3 MB/s eta 0:00:00
Downloading nvidia_cusparses_cu12-12.3.1.170-py3-none-
manylinux2014_x86_64.whl (207.5 MB)
207.5/207.5 MB
6.8 MB/s eta 0:00:00
Downloading nvidia_nvjitlink_cu12-12.4.127-py3-none-
manylinux2014_x86_64.whl (21.1 MB)
21.1/21.1 MB
35.8 MB/s eta 0:00:00
Installing collected packages: nvidia-nvjitlink-cu12, nvidia-curand-cu12,
nvidia-cufft-cu12, nvidia-cuda-runtime-cu12, nvidia-cuda-nvrtc-cu12, nvidia-
cuda-cupti-cu12, nvidia-cublas-cu12, nvidia-cusparses-cu12, nvidia-cudnn-cu12,
nvidia-cusolver-cu12, invisible_watermark
Attempting uninstall: nvidia-nvjitlink-cu12
Found existing installation: nvidia-nvjitlink-cu12 12.5.82
Uninstalling nvidia-nvjitlink-cu12-12.5.82:
Successfully uninstalled nvidia-nvjitlink-cu12-12.5.82

```



```

Attempting uninstall: nvidia-curand-cu12
  Found existing installation: nvidia-curand-cu12 10.3.6.82
  Uninstalling nvidia-curand-cu12-10.3.6.82:
    Successfully uninstalled nvidia-curand-cu12-10.3.6.82
Attempting uninstall: nvidia-cufft-cu12
  Found existing installation: nvidia-cufft-cu12 11.2.3.61
  Uninstalling nvidia-cufft-cu12-11.2.3.61:
    Successfully uninstalled nvidia-cufft-cu12-11.2.3.61
Attempting uninstall: nvidia-cuda-runtime-cu12
  Found existing installation: nvidia-cuda-runtime-cu12 12.5.82
  Uninstalling nvidia-cuda-runtime-cu12-12.5.82:
    Successfully uninstalled nvidia-cuda-runtime-cu12-12.5.82
Attempting uninstall: nvidia-cuda-nvrtc-cu12
  Found existing installation: nvidia-cuda-nvrtc-cu12 12.5.82
  Uninstalling nvidia-cuda-nvrtc-cu12-12.5.82:
    Successfully uninstalled nvidia-cuda-nvrtc-cu12-12.5.82
Attempting uninstall: nvidia-cuda-cupti-cu12
  Found existing installation: nvidia-cuda-cupti-cu12 12.5.82
  Uninstalling nvidia-cuda-cupti-cu12-12.5.82:
    Successfully uninstalled nvidia-cuda-cupti-cu12-12.5.82
Attempting uninstall: nvidia-cublas-cu12
  Found existing installation: nvidia-cublas-cu12 12.5.3.2
  Uninstalling nvidia-cublas-cu12-12.5.3.2:
    Successfully uninstalled nvidia-cublas-cu12-12.5.3.2
Attempting uninstall: nvidia-cusparse-cu12
  Found existing installation: nvidia-cusparse-cu12 12.5.1.3
  Uninstalling nvidia-cusparse-cu12-12.5.1.3:
    Successfully uninstalled nvidia-cusparse-cu12-12.5.1.3
Attempting uninstall: nvidia-cudnn-cu12
  Found existing installation: nvidia-cudnn-cu12 9.3.0.75
  Uninstalling nvidia-cudnn-cu12-9.3.0.75:
    Successfully uninstalled nvidia-cudnn-cu12-9.3.0.75
Attempting uninstall: nvidia-cusolver-cu12
  Found existing installation: nvidia-cusolver-cu12 11.6.3.83
  Uninstalling nvidia-cusolver-cu12-11.6.3.83:
    Successfully uninstalled nvidia-cusolver-cu12-11.6.3.83
Successfully installed invisible_watermark-0.2.0 nvidia-cublas-cu12-12.4.5.8
nvidia-cuda-cupti-cu12-12.4.127 nvidia-cuda-nvrtc-cu12-12.4.127 nvidia-cuda-
runtime-cu12-12.4.127 nvidia-cudnn-cu12-9.1.0.70 nvidia-cufft-cu12-11.2.1.3
nvidia-curand-cu12-10.3.5.147 nvidia-cusolver-cu12-11.6.1.9 nvidia-cusparse-
cu12-12.3.1.170 nvidia-nvjitlink-cu12-12.4.127

```

```

[4]: from diffusers import DiffusionPipeline
import torch

```

```

pipe = DiffusionPipeline.from_pretrained("stabilityai/
↳stable-diffusion-xl-base-1.0", torch_dtype=torch.float16,
↳use_safetensors=True, variant="fp16")
pipe.to("cuda")

# if using torch < 2.0
# pipe.enable_xformers_memory_efficient_attention()

prompt = "An astronaut riding a green horse"

images = pipe(prompt=prompt).images[0]

```

/usr/local/lib/python3.11/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(

model_index.json: 0%| | 0.00/609 [00:00<?, ?B/s]

Fetching 19 files: 0%| | 0/19 [00:00<?, ?it/s]

config.json: 0%| | 0.00/565 [00:00<?, ?B/s]

special_tokens_map.json: 0%| | 0.00/472 [00:00<?, ?B/s]

scheduler_config.json: 0%| | 0.00/479 [00:00<?, ?B/s]

tokenizer_config.json: 0%| | 0.00/737 [00:00<?, ?B/s]

text_encoder/model.fp16.safetensors: 0%| | 0.00/246M [00:00<?, ?B/s]

merges.txt: 0.00B [00:00, ?B/s]

text_encoder_2/model.fp16.safetensors: 0%| | 0.00/1.39G [00:00<?, ?B/
↳s]

special_tokens_map.json: 0%| | 0.00/460 [00:00<?, ?B/s]

vocab.json: 0.00B [00:00, ?B/s]

tokenizer_config.json: 0%| | 0.00/725 [00:00<?, ?B/s]

config.json: 0%| | 0.00/575 [00:00<?, ?B/s]

config.json: 0.00B [00:00, ?B/s]

config.json: 0%| | 0.00/642 [00:00<?, ?B/s]

```

unet/diffusion_pytorch_model.fp16.safete(...): 0%|          | 0.00/5.14G [00:00<?
↪, ?B/s]
vae_1_0/diffusion_pytorch_model.fp16.saf(...): 0%|          | 0.00/167M [00:00<?
↪, ?B/s]
vae/diffusion_pytorch_model.fp16.safeten(...): 0%|          | 0.00/167M [00:00<?
↪, ?B/s]
Loading pipeline components...: 0%|          | 0/7 [00:00<?, ?it/s]
0%|          | 0/50 [00:00<?, ?it/s]

```

```

[1]: from diffusers import StableDiffusionPipeline
import torch

model_id = "sd-legacy/stable-diffusion-v1-5"
pipe = StableDiffusionPipeline.from_pretrained(model_id, torch_dtype=torch.
↪float16)
pipe = pipe.to("cuda")

prompt = "a photo of an astronaut riding a horse on mars"
image = pipe(prompt).images[0]

image.save("astronaut_rides_horse.png")

```

```

/usr/local/lib/python3.11/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(
model_index.json: 0%|          | 0.00/541 [00:00<?, ?B/s]
Fetching 15 files: 0%|          | 0/15 [00:00<?, ?it/s]
preprocessor_config.json: 0%|          | 0.00/342 [00:00<?, ?B/s]
scheduler_config.json: 0%|          | 0.00/308 [00:00<?, ?B/s]
merges.txt: 0.00B [00:00, ?B/s]
config.json: 0.00B [00:00, ?B/s]
special_tokens_map.json: 0%|          | 0.00/472 [00:00<?, ?B/s]
config.json: 0%|          | 0.00/617 [00:00<?, ?B/s]
model.safetensors: 0%|          | 0.00/492M [00:00<?, ?B/s]

```

```

model.safetensors:  0%|          | 0.00/1.22G [00:00<?, ?B/s]
tokenizer_config.json:  0%|          | 0.00/806 [00:00<?, ?B/s]
vocab.json: 0.00B [00:00, ?B/s]
diffusion_pytorch_model.safetensors:  0%|          | 0.00/335M [00:00<?, ?B/s]
config.json:  0%|          | 0.00/743 [00:00<?, ?B/s]
diffusion_pytorch_model.safetensors:  0%|          | 0.00/3.44G [00:00<?, ?B/s]
config.json:  0%|          | 0.00/547 [00:00<?, ?B/s]
Loading pipeline components...:  0%|          | 0/7 [00:00<?, ?it/s]
  0%|          | 0/50 [00:00<?, ?it/s]

```

[14]: `!nvidia-smi`

```

Sun Jul 27 06:48:53 2025
+-----+
+-----+
| NVIDIA-SMI 550.54.15                Driver Version: 550.54.15          CUDA Version:
12.4          |
|-----+-----+-----+
+-----+
| GPU  Name            Persistence-M | Bus-Id        Disp.A | Volatile
Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |      Memory-Usage | GPU-Util
Compute M. |
|              |              |              |
MIG M. |
|=====+=====+=====+
=====|
|   0   Tesla T4               Off  | 00000000:00:04.0 Off |
0 |
| N/A   62C    P0              30W / 70W |  4242MiB /  15360MiB |      0%
Default |
|              |              |              |
N/A |
+-----+-----+-----+
+-----+
+-----+
| Processes:
|
| GPU   GI    CI          PID    Type    Process name
GPU Memory |
|       ID    ID
Usage      |

```

```
|=====
=====|
+-----+
-----+
```

```
[11]: from google.colab import drive
drive.mount('/content/drive')
```

Mounted at /content/drive

```
[ ]: import gradio as gr
from diffusers import StableDiffusionPipeline
model_id="sd-legacy/stable-diffusion-v1-5"
pipe=StableDiffusionPipeline.from_pretrained()

def generated_Image
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```