

Building First Gradio Project

```
! pip install gradio

Collecting gradio
  Downloading gradio-5.23.3-py3-none-any.whl.metadata (16 kB)
Collecting aiofiles<24.0,>=22.0 (from gradio)
  Downloading aiofiles-23.2.1-py3-none-any.whl.metadata (9.7 kB)
Requirement already satisfied: anyio<5.0,>=3.0 in
/usr/local/lib/python3.11/dist-packages (from gradio) (4.9.0)
Collecting fastapi<1.0,>=0.115.2 (from gradio)
  Downloading fastapi-0.115.12-py3-none-any.whl.metadata (27 kB)
Collecting ffmpy (from gradio)
  Downloading ffmpy-0.5.0-py3-none-any.whl.metadata (3.0 kB)
Collecting gradio-client==1.8.0 (from gradio)
  Downloading gradio_client-1.8.0-py3-none-any.whl.metadata (7.1 kB)
Collecting groovy~=0.1 (from gradio)
  Downloading groovy-0.1.2-py3-none-any.whl.metadata (6.1 kB)
Requirement already satisfied: httpx>=0.24.1 in
/usr/local/lib/python3.11/dist-packages (from gradio) (0.28.1)
Requirement already satisfied: huggingface-hub>=0.28.1 in
/usr/local/lib/python3.11/dist-packages (from gradio) (0.30.1)
Requirement already satisfied: jinja2<4.0 in
/usr/local/lib/python3.11/dist-packages (from gradio) (3.1.6)
Requirement already satisfied: markupsafe<4.0,>=2.0 in
/usr/local/lib/python3.11/dist-packages (from gradio) (3.0.2)
Requirement already satisfied: numpy<3.0,>=1.0 in
/usr/local/lib/python3.11/dist-packages (from gradio) (2.0.2)
Requirement already satisfied: orjson~=3.0 in
/usr/local/lib/python3.11/dist-packages (from gradio) (3.10.16)
Requirement already satisfied: packaging in
/usr/local/lib/python3.11/dist-packages (from gradio) (24.2)
Requirement already satisfied: pandas<3.0,>=1.0 in
/usr/local/lib/python3.11/dist-packages (from gradio) (2.2.2)
Requirement already satisfied: pillow<12.0,>=8.0 in
/usr/local/lib/python3.11/dist-packages (from gradio) (11.1.0)
Requirement already satisfied: pydantic<2.12,>=2.0 in
/usr/local/lib/python3.11/dist-packages (from gradio) (2.11.1)
Collecting pydub (from gradio)
  Downloading pydub-0.25.1-py2.py3-none-any.whl.metadata (1.4 kB)
Collecting python-multipart>=0.0.18 (from gradio)
  Downloading python_multipart-0.0.20-py3-none-any.whl.metadata (1.8
kB)
Requirement already satisfied: pyyaml<7.0,>=5.0 in
/usr/local/lib/python3.11/dist-packages (from gradio) (6.0.2)
Collecting ruff>=0.9.3 (from gradio)
  Downloading ruff-0.11.4-py3-none-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (25 kB)
Collecting safehttpx<0.2.0,>=0.1.6 (from gradio)
```

Downloading safehttpx-0.1.6-py3-none-any.whl.metadata (4.2 kB)
Collecting semantic-version~=2.0 (from gradio)
Downloading semantic_version-2.10.0-py2.py3-none-any.whl.metadata (9.7 kB)
Collecting starlette<1.0,>=0.40.0 (from gradio)
Downloading starlette-0.46.1-py3-none-any.whl.metadata (6.2 kB)
Collecting tomlkit<0.14.0,>=0.12.0 (from gradio)
Downloading tomlkit-0.13.2-py3-none-any.whl.metadata (2.7 kB)
Requirement already satisfied: typer<1.0,>=0.12 in
/usr/local/lib/python3.11/dist-packages (from gradio) (0.15.2)
Requirement already satisfied: typing-extensions~=4.0 in
/usr/local/lib/python3.11/dist-packages (from gradio) (4.13.0)
Collecting uvicorn>=0.14.0 (from gradio)
Downloading uvicorn-0.34.0-py3-none-any.whl.metadata (6.5 kB)
Requirement already satisfied: fsspec in
/usr/local/lib/python3.11/dist-packages (from gradio-client==1.8.0->gradio) (2025.3.2)
Requirement already satisfied: websockets<16.0,>=10.0 in
/usr/local/lib/python3.11/dist-packages (from gradio-client==1.8.0->gradio) (15.0.1)
Requirement already satisfied: idna>=2.8 in
/usr/local/lib/python3.11/dist-packages (from anyio<5.0,>=3.0->gradio) (3.10)
Requirement already satisfied: sniffio>=1.1 in
/usr/local/lib/python3.11/dist-packages (from anyio<5.0,>=3.0->gradio) (1.3.1)
Requirement already satisfied: certifi in
/usr/local/lib/python3.11/dist-packages (from httpx>=0.24.1->gradio) (2025.1.31)
Requirement already satisfied: httpcore==1.* in
/usr/local/lib/python3.11/dist-packages (from httpx>=0.24.1->gradio) (1.0.7)
Requirement already satisfied: h11<0.15,>=0.13 in
/usr/local/lib/python3.11/dist-packages (from httpcore==1.*->httpx>=0.24.1->gradio) (0.14.0)
Requirement already satisfied: filelock in
/usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.28.1->gradio) (3.18.0)
Requirement already satisfied: requests in
/usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.28.1->gradio) (2.32.3)
Requirement already satisfied: tqdm>=4.42.1 in
/usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.28.1->gradio) (4.67.1)
Requirement already satisfied: python-dateutil>=2.8.2 in
/usr/local/lib/python3.11/dist-packages (from pandas<3.0,>=1.0->gradio) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in
/usr/local/lib/python3.11/dist-packages (from pandas<3.0,>=1.0-

```

>gradio) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in
/usr/local/lib/python3.11/dist-packages (from pandas<3.0,>=1.0-
>gradio) (2025.2)
Requirement already satisfied: annotated-types>=0.6.0 in
/usr/local/lib/python3.11/dist-packages (from pydantic<2.12,>=2.0-
>gradio) (0.7.0)
Requirement already satisfied: pydantic-core==2.33.0 in
/usr/local/lib/python3.11/dist-packages (from pydantic<2.12,>=2.0-
>gradio) (2.33.0)
Requirement already satisfied: typing-inspection>=0.4.0 in
/usr/local/lib/python3.11/dist-packages (from pydantic<2.12,>=2.0-
>gradio) (0.4.0)
Requirement already satisfied: click>=8.0.0 in
/usr/local/lib/python3.11/dist-packages (from typer<1.0,>=0.12-
>gradio) (8.1.8)
Requirement already satisfied: shellingham>=1.3.0 in
/usr/local/lib/python3.11/dist-packages (from typer<1.0,>=0.12-
>gradio) (1.5.4)
Requirement already satisfied: rich>=10.11.0 in
/usr/local/lib/python3.11/dist-packages (from typer<1.0,>=0.12-
>gradio) (13.9.4)
Requirement already satisfied: six>=1.5 in
/usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2-
>pandas<3.0,>=1.0->gradio) (1.17.0)
Requirement already satisfied: markdown-it-py>=2.2.0 in
/usr/local/lib/python3.11/dist-packages (from rich>=10.11.0-
>typer<1.0,>=0.12->gradio) (3.0.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in
/usr/local/lib/python3.11/dist-packages (from rich>=10.11.0-
>typer<1.0,>=0.12->gradio) (2.18.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.11/dist-packages (from requests->huggingface-
hub>=0.28.1->gradio) (3.4.1)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.11/dist-packages (from requests->huggingface-
hub>=0.28.1->gradio) (2.3.0)
Requirement already satisfied: mdurl~=0.1 in
/usr/local/lib/python3.11/dist-packages (from markdown-it-py>=2.2.0-
>rich>=10.11.0->typer<1.0,>=0.12->gradio) (0.1.2)
Downloading gradio-5.23.3-py3-none-any.whl (46.5 MB)
_____ 46.5/46.5 MB 13.0 MB/s eta
0:00:00
_____ 322.2/322.2 kB 10.3 MB/s eta
0:00:00
_____ 95.2/95.2 kB 6.6 MB/s eta
0:00:00
ultipart-0.0.20-py3-none-any.whl (24 kB)
Downloading ruff-0.11.4-py3-none-

```

```

manylinux_2_17_x86_64.manylinux2014_x86_64.whl (11.3 MB)
----- 11.3/11.3 MB 20.4 MB/s eta
0:00:00
antic_version-2.10.0-py2.py3-none-any.whl (15 kB)
Downloading starlette-0.46.1-py3-none-any.whl (71 kB)
----- 72.0/72.0 kB 4.5 MB/s eta
0:00:00
lkit-0.13.2-py3-none-any.whl (37 kB)
Downloading uvicorn-0.34.0-py3-none-any.whl (62 kB)
----- 62.3/62.3 kB 3.6 MB/s eta
0:00:00
py-0.5.0-py3-none-any.whl (6.0 kB)
Downloading pydub-0.25.1-py2.py3-none-any.whl (32 kB)
Installing collected packages: pydub, uvicorn, tomlkit, semantic-
version, ruff, python-multipart, groovy, ffmpeg, aiofiles, starlette,
safehttpx, gradio-client, fastapi, gradio
Successfully installed aiofiles-23.2.1 fastapi-0.115.12 ffmpeg-0.5.0
gradio-5.23.3 gradio-client-1.8.0 groovy-0.1.2 pydub-0.25.1 python-
multipart-0.0.20 ruff-0.11.4 safehttpx-0.1.6 semantic-version-2.10.0
starlette-0.46.1 tomlkit-0.13.2 uvicorn-0.34.0

import gradio as gr

def hello_world(name):
    return "Hello... " + name + "!!"

hello_world("Aqib")

{"type": "string"}

interface=gr.Interface(fn=hello_world,inputs="text",outputs="text")
interface.launch()

Running Gradio in a Colab notebook requires sharing enabled.
Automatically setting `share=True` (you can turn this off by setting
`share=False` in `launch()` explicitly).

Colab notebook detected. To show errors in colab notebook, set
debug=True in launch()
* Running on public URL: https://dfd887db0af89a8442.gradio.live

This share link expires in 72 hours. 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.co/spaces)

<IPython.core.display.HTML object>

```

Customizing Gradio Components

Image Applications

```
interface=gr.Interface(fn=hello_world,inputs=gr.Textbox(lines=10,placeholder="Enter your input here..."),outputs="text")
```

```
interface.launch()
```

Running Gradio in a Colab notebook requires sharing enabled. Automatically setting `share=True` (you can turn this off by setting `share=False` in `launch()` explicitly).

Colab notebook detected. To show errors in colab notebook, set debug=True in launch()

* Running on public URL: <https://f5ad0b3372f0d17632.gradio.live>

This share link expires in 72 hours. 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.co/spaces>)

<IPython.core.display.HTML object>

```
import numpy as np
```

```
def sepia(input_img):  
    sepia_filter=np.array([  
        [.393, .769, .189],  
        [.349, .686, .168],  
        [.272, .534, .131]  
    ])  
    sepia_img=input_img.dot(sepia_filter.T)  
    sepia_img/=sepia_img.max()  
    return sepia_img
```

```
iface=gr.Interface(sepia, gr.Image(type="numpy", image_mode="RGB"),  
"image")
```

```
iface.launch()
```

Running Gradio in a Colab notebook requires sharing enabled. Automatically setting `share=True` (you can turn this off by setting `share=False` in `launch()` explicitly).

Colab notebook detected. To show errors in colab notebook, set debug=True in launch()

* Running on public URL: <https://c28a15ab2b4d8d9a44.gradio.live>

This share link expires in 72 hours. 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.co/spaces>)

<IPython.core.display.HTML object>

Working with tabular data

```
import pandas as pd
import matplotlib.pyplot as plt
import gradio as gr
import numpy as np

def sales_projections(employee_data):
    sales_data = employee_data.iloc[:, 1:4].astype("int").to_numpy()
    regression_values = np.apply_along_axis(lambda row:
np.array(np.polyld(np.polyfit([0, 1, 2], row, 2))), 0, sales_data)
    projected_months = np.repeat(np.expand_dims(np.arange(3, 12), 0),
len(sales_data), axis=0) # Use np.arange instead of np.arrange
    projected_values = np.array([
        month * month * regression[0] + month * regression[1] +
        regression[2]
        for month, regression in zip(projected_months,
regression_values)])
    plt.plot(projected_values.T)
    plt.legend(employee_data["Name"])
    return employee_data, plt.gcf(), regression_values

# Create the DataFrame outside of gr.Interface and pass it as the
value argument
df = pd.DataFrame(data=[["Jon", 12, 14, 18], ["Alice", 14, 17, 2],
["Sana", 8, 9.5, 12]],
                  columns=["Name", "Jan Sales", "Feb Sales", "Mar
Sales"])

iface = gr.Interface(sales_projections,
                    gr.Dataframe(value=df, # Pass the DataFrame using
the 'value' argument
                                headers=["Name", "Jan Sales", "Feb
Sales", "Mar Sales"])),
                    ["dataframe", "plot", "numpy"],
                    description="Enter sales figures for employees to
predict sales trajectory over year.")
iface.launch()
```

Running Gradio in a Colab notebook requires sharing enabled.
Automatically setting `share=True` (you can turn this off by setting
`share=False` in `launch()` explicitly).

Colab notebook detected. To show errors in colab notebook, set
debug=True in launch()
* Running on public URL: <https://45aaf6d65e5e5e3b14.gradio.live>

This share link expires in 72 hours. 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.co/spaces>)

<IPython.core.display.HTML object>

Gradio Multiple Inputs

```
def bmi(name, height, weight):  
    bmi_val=weight/(height**2)  
    result_emoticon="😊" if bmi_val<30 else "😞"  
    return f"{name} has a BMI of {round(bmi_val,2)} and the result will  
make you: {result_emoticon}"
```

```
bmi("Aqib", 1.2, 80)
```

```
{"type": "string"}
```

```
interface = gr.Interface(fn=bmi, inputs=["text",  
gr.Slider(0,200,label="Height in Meters"),  
gr.Slider(0,100,label="Height in Kg")], outputs="text")  
interface.launch()
```

Running Gradio in a Colab notebook requires sharing enabled.
Automatically setting `share=True` (you can turn this off by setting
`share=False` in `launch()` explicitly).

Colab notebook detected. To show errors in colab notebook, set
debug=True in launch()
* Running on public URL: <https://111cb43e677c9c008f.gradio.live>

This share link expires in 72 hours. 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.co/spaces>)

<IPython.core.display.HTML object>

Debuggin and Flagging

```
def bmi(name, height, weight, feeling):  
    bmi_val=weight/(height**2)
```

```

result_emoticon="😊" if bmi_val<30 else "😞"
output_str = f"Hello {name}, your BMI is: {round(bmi_val,2)}"
txt = "Happy" if feeling else "Sad"
return (output_str, result_emoticon, txt)

interface = gr.Interface(fn=bmi,
                        inputs=[gr.Textbox(lines=1, label="Name"),
                                gr.Slider(1,200,label="Height in Meters"),
                                gr.Slider(1,100,label="Weight in Kg"), gr.Checkbox(label="You are
feeling all good")],
                        outputs=[gr.Textbox(lines=1, label="BMI
Result"),gr.Textbox(lines=1, label="Result
Emotion"),gr.Textbox(lines=1, label="How you are feeling now?")])
interface.launch(debug=True)

```

Running Gradio in a Colab notebook requires sharing enabled.
Automatically setting `share=True` (you can turn this off by setting
`share=False` in `launch()` explicitly).

Colab notebook detected. This cell will run indefinitely so that you
can see errors and logs. To turn off, set debug=False in launch().
* Running on public URL: <https://827882fc701f7b346d.gradio.live>

This share link expires in 72 hours. 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.co/spaces>)

<IPython.core.display.HTML object>

Using existing dataset file at: .gradio/flagged/dataset1.csv
Keyboard interruption in main thread... closing server.
Killing tunnel 127.0.0.1:7868 <>
<https://827882fc701f7b346d.gradio.live>

```
pd.read_csv(".gradio/flagged/dataset1.csv")
```

```

{"summary":{"\n  \"name\": \"pd\",\n  \"rows\": 2,\n  \"fields\": [\n    {\n      \"column\": \"Name\",\n      \"properties\": {\n        \"dtype\": \"string\",\n        \"num_unique_values\": 2,\n        \"samples\": [\n          \"Ashiq\",\n          \"Aqib\"\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      },\n      \"column\": \"Height in Meters\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 0,\n        \"min\": 1,\n        \"max\": 1,\n        \"num_unique_values\": 1,\n        \"samples\": [\n          1\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      },\n      \"column\": \"Weight in Kg\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\":

```



```

19.16259377015544,\n          \"min\": 22.9,\n          \"max\": 50.0,\n          \"num_unique_values\": 2,\n          \"samples\": [\n              22.9\n          ],\n          \"semantic_type\": \"\",\n          \"description\": \"\"\n      },\n      {\n          \"column\": \"You are feeling all good\",\n          \"properties\": {\n              \"dtype\": \"boolean\",\n              \"num_unique_values\": 2,\n              \"samples\": [\n                  true\n              ],\n              \"semantic_type\": \"\",\n              \"description\": \"\"\n          },\n          {\n              \"column\": \"BMI Result\",\n              \"properties\": {\n                  \"dtype\": \"string\",\n                  \"num_unique_values\": 2,\n                  \"samples\": [\n                      \"Hello Ashiq, your BMI is: 22.9\"\n                  ],\n                  \"semantic_type\": \"\",\n                  \"description\": \"\"\n              },\n              {\n                  \"column\": \"Result Emotion\",\n                  \"properties\": {\n                      \"dtype\": \"string\",\n                      \"num_unique_values\": 2,\n                      \"samples\": [\n                          \"\\ud83d\\ude0a\"\n                      ],\n                      \"semantic_type\": \"\",\n                      \"description\": \"\"\n                  },\n                  {\n                      \"column\": \"How you are feeling now?\",\n                      \"properties\": {\n                          \"dtype\": \"string\",\n                          \"num_unique_values\": 2,\n                          \"samples\": [\n                              \"Happy\"\n                          ],\n                          \"semantic_type\": \"\",\n                          \"description\": \"\"\n                      },\n                      {\n                          \"column\": \"timestamp\",\n                          \"properties\": {\n                              \"dtype\": \"object\",\n                              \"num_unique_values\": 2,\n                              \"samples\": [\n                                  \"2025-04-05 16:25:22.666752\"\n                              ],\n                              \"semantic_type\": \"\",\n                              \"description\": \"\"\n                          }\n                      }\n                  }\n              }\n          ],\n      },\n      {\n          \"column\": \"How you are feeling now?\",\n          \"properties\": {\n              \"dtype\": \"string\",\n              \"num_unique_values\": 2,\n              \"samples\": [\n                  \"Happy\"\n              ],\n              \"semantic_type\": \"\",\n              \"description\": \"\"\n          },\n          {\n              \"column\": \"timestamp\",\n              \"properties\": {\n                  \"dtype\": \"object\",\n                  \"num_unique_values\": 2,\n                  \"samples\": [\n                      \"2025-04-05 16:25:22.666752\"\n                  ],\n                  \"semantic_type\": \"\",\n                  \"description\": \"\"\n              }\n          }\n      }\n  ],\n  \"type\": \"dataframe\"}

```

Improving UI+UX theme

```

interface = gr.Interface(fn=bmi,
                        inputs=[gr.Textbox(lines=1, label="Name"),
                                gr.Slider(1,200,label="Height in Meters"),
                                gr.Slider(1,100,label="Weight in Kg"), gr.Checkbox(label="Are you
                                feeling all good?")],
                        outputs=[gr.Textbox(lines=1, label="BMI
                                Result"),gr.Textbox(lines=1, label="Result
                                Emotion"),gr.Textbox(lines=1, label="How you are feeling now?")],
                        examples=[['Aqib', 1.2, 50, True],
                                ['Sakib',2,80,False],
                                ['Rakib',1.5,70,True]],
                        # live=True,
                        flagging_options=["Yes","No","Maybe"],
                        theme="huggingface"
                        )
interface.launch(debug=True, share=True)

/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(
/usr/local/lib/python3.11/dist-packages/gradio/blocks.py:1115:
UserWarning: Cannot load huggingface. Caught Exception: 404 Client
Error: Not Found for url:
https://huggingface.co/api/spaces/huggingface (Request ID: Root=1-
67f15816-1b6b677a77ba760b41218429;b0c719f6-316b-4f14-980f-
92ce57696241)
```

Sorry, we can't find the page you are looking for.

```
warnings.warn(f"Cannot load {theme}. Caught Exception: {str(e)}")
```

Colab notebook detected. This cell will run indefinitely so that you can see errors and logs. To turn off, set debug=False in launch().
* Running on public URL: <https://dcb5d068f056da6d97.gradio.live>

This share link expires in 72 hours. 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.co/spaces>)

<IPython.core.display.HTML object>

Keyboard interruption in main thread... closing server.
Killing tunnel 127.0.0.1:7865 <>
<https://dcb5d068f056da6d97.gradio.live>

Chatbot Creation

```
import random
import gradio as gr

def chat(message, history):
    history = history or [] # Initialize history as an empty list if
None
    if message.startswith("How many"):
        response = random.choice(["1", "2", "3", "4", "5"])
    elif message.startswith("How"):
        response = random.choice(["Great", "Good", "Okay", "Bad"])
    elif message.startswith("Where"):
        response = random.choice(["Here", "There", "Somewhere"])
    else:
        response = "I don't know"
```

```

        history_result = history + [[message, response]] # Update history
        return history, history_result # Return history for both outputs

iface = gr.Interface(
    fn=chat,
    inputs=[gr.Textbox(lines=1, label="Ask your question here ☐",
placeholder="Ask any of this 3 question: How many are you replying
now? or How are you? or Where are you?"), "state"], # "state" is used
for the chat history
    outputs=["state", "chatbot"], # "chatbot" to display the
conversation
    allow_flagging="never"
)

```

```
iface.launch(debug=True)
```

```

/usr/local/lib/python3.11/dist-packages/gradio/components/base.py:423:
UserWarning: You have not specified a value for the `type` parameter.
Defaulting to the 'tuples' format for chatbot messages, but this is
deprecated and will be removed in a future version of Gradio. Please
set type='messages' instead, which uses openai-style dictionaries with
'role' and 'content' keys.

```

```

    obj = utils.component_or_layout_class(cls_name)(render=render)
/usr/local/lib/python3.11/dist-packages/gradio/interface.py:415:
UserWarning: The `allow_flagging` parameter in `Interface` is
deprecated.Use `flagging_mode` instead.
    warnings.warn(

```

Running Gradio in a Colab notebook requires sharing enabled.
Automatically setting `share=True` (you can turn this off by setting
`share=False` in `launch()`) explicitly).

Colab notebook detected. This cell will run indefinitely so that you
can see errors and logs. To turn off, set debug=False in launch().
* Running on public URL: <https://360b6201541caf905a.gradio.live>

This share link expires in 72 hours. 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.co/spaces>)

<IPython.core.display.HTML object>

Keyboard interruption in main thread... closing server.
Killing tunnel 127.0.0.1:7868 <>
<https://360b6201541caf905a.gradio.live>

Gradio Sharing+Auth

```
!pip install gradio_client
```

```
Requirement already satisfied: gradio_client in
/usr/local/lib/python3.11/dist-packages (1.8.0)
Requirement already satisfied: fsspec in
/usr/local/lib/python3.11/dist-packages (from gradio_client)
(2025.3.2)
Requirement already satisfied: httpx>=0.24.1 in
/usr/local/lib/python3.11/dist-packages (from gradio_client) (0.28.1)
Requirement already satisfied: huggingface-hub>=0.19.3 in
/usr/local/lib/python3.11/dist-packages (from gradio_client) (0.30.1)
Requirement already satisfied: packaging in
/usr/local/lib/python3.11/dist-packages (from gradio_client) (24.2)
Requirement already satisfied: typing-extensions~=4.0 in
/usr/local/lib/python3.11/dist-packages (from gradio_client) (4.13.0)
Requirement already satisfied: websockets<16.0,>=10.0 in
/usr/local/lib/python3.11/dist-packages (from gradio_client) (15.0.1)
Requirement already satisfied: anyio in
/usr/local/lib/python3.11/dist-packages (from httpx>=0.24.1-
>gradio_client) (4.9.0)
Requirement already satisfied: certifi in
/usr/local/lib/python3.11/dist-packages (from httpx>=0.24.1-
>gradio_client) (2025.1.31)
Requirement already satisfied: httpcore==1.* in
/usr/local/lib/python3.11/dist-packages (from httpx>=0.24.1-
>gradio_client) (1.0.7)
Requirement already satisfied: idna in /usr/local/lib/python3.11/dist-
packages (from httpx>=0.24.1->gradio_client) (3.10)
Requirement already satisfied: h11<0.15,>=0.13 in
/usr/local/lib/python3.11/dist-packages (from httpcore==1.*-
>httpx>=0.24.1->gradio_client) (0.14.0)
Requirement already satisfied: filelock in
/usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.19.3-
>gradio_client) (3.18.0)
Requirement already satisfied: pyyaml>=5.1 in
/usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.19.3-
>gradio_client) (6.0.2)
Requirement already satisfied: requests in
/usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.19.3-
>gradio_client) (2.32.3)
Requirement already satisfied: tqdm>=4.42.1 in
/usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.19.3-
>gradio_client) (4.67.1)
Requirement already satisfied: sniffio>=1.1 in
/usr/local/lib/python3.11/dist-packages (from anyio->httpx>=0.24.1-
>gradio_client) (1.3.1)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.11/dist-packages (from requests->huggingface-
```

```
hub>=0.19.3->gradio_client) (3.4.1)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.11/dist-packages (from requests->huggingface-
hub>=0.19.3->gradio_client) (2.3.0)
```

```
interface = gr.Interface(fn=bmi,
                        inputs=[gr.Textbox(lines=1, label="Name"),
                                gr.Slider(1,200,label="Height in Meters"),
                                gr.Slider(1,100,label="Weight in Kg"), gr.Checkbox(label="Are you
feeling all good?")],
                        outputs=[gr.Textbox(lines=1, label="BMI
Result"),gr.Textbox(lines=1, label="Result
Emotion"),gr.Textbox(lines=1, label="How you are feeling now?")],
                        examples=[[ 'Aqib', 1.2, 50, True],
                                [ 'Sakib',2,80,False],
                                [ 'Rakib',1.5,70,True]],
                                # live=True,
                                flagging_options=["Yes","No","Maybe"],
                                theme="huggingface"
                                )
#Launch interface with share=True parameter
interface.launch(share=True)
```

```
/usr/local/lib/python3.11/dist-packages/gradio/blocks.py:1115:
UserWarning: Cannot load huggingface. Caught Exception: 404 Client
Error: Not Found for url:
https://huggingface.co/api/spaces/huggingface (Request ID: Root=1-
67f15840-3435b5fe52f943b5388b713c;78f9ad70-80fa-4cc1-913d-
a12e2c9d9176)
```

```
Sorry, we can't find the page you are looking for.
warnings.warn(f"Cannot load {theme}. Caught Exception: {str(e)}")
```

```
Colab notebook detected. To show errors in colab notebook, set
debug=True in launch()
```

```
* Running on public URL: https://a84218222122a7bb26.gradio.live
```

```
This share link expires in 72 hours. 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.co/spaces)
```

```
<IPython.core.display.HTML object>
```

```
from gradio_client import Client
```

```
client = Client("https://a84218222122a7bb26.gradio.live/")
result = client.predict(
    name="Hello!!",
```

```

        height=1,
        weight=1,
        feeling=False,
        api_name="/predict"
    )
    print(result)

```

Loaded as API: <https://a84218222122a7bb26.gradio.live/> ✓
('Hello Hello!!', your BMI is: 1.0', '😊', 'Sad')

```

interface.launch(auth=("kutub", "kutub6789"))

```

Running Gradio in a Colab notebook requires sharing enabled.
Automatically setting `share=True` (you can turn this off by setting
`share=False` in `launch()` explicitly).

Colab notebook detected. To show errors in colab notebook, set
debug=True in launch()
* Running on public URL: <https://a84218222122a7bb26.gradio.live>

This share link expires in 72 hours. 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.co/spaces>)

<IPython.core.display.HTML object>

Deploying a DL Gradio App with 😊 Spaces

```

!pip install transformers

```

Requirement already satisfied: transformers in
/usr/local/lib/python3.11/dist-packages (4.50.3)
Requirement already satisfied: filelock in
/usr/local/lib/python3.11/dist-packages (from transformers) (3.18.0)
Requirement already satisfied: huggingface-hub<1.0,>=0.26.0 in
/usr/local/lib/python3.11/dist-packages (from transformers) (0.30.1)
Requirement already satisfied: numpy>=1.17 in
/usr/local/lib/python3.11/dist-packages (from transformers) (2.0.2)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.11/dist-packages (from transformers) (24.2)
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) (0.21.1)
Requirement already satisfied: safetensors>=0.4.3 in
/usr/local/lib/python3.11/dist-packages (from transformers) (0.5.3)
Requirement already satisfied: tqdm>=4.27 in
/usr/local/lib/python3.11/dist-packages (from transformers) (4.67.1)
Requirement already satisfied: fsspec>=2023.5.0 in
/usr/local/lib/python3.11/dist-packages (from huggingface-
hub<1.0,>=0.26.0->transformers) (2025.3.2)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/usr/local/lib/python3.11/dist-packages (from huggingface-
hub<1.0,>=0.26.0->transformers) (4.13.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.11/dist-packages (from requests->transformers)
(3.4.1)
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)
(2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.11/dist-packages (from requests->transformers)
(2025.1.31)
```

```
from transformers import pipeline
```

```
sentiment = pipeline("sentiment-analysis")
```

No model was supplied, defaulted to distilbert/distilbert-base-uncased-finetuned-sst-2-english and revision 714eb0f (<https://huggingface.co/distilbert/distilbert-base-uncased-finetuned-sst-2-english>).

Using a pipeline without specifying a model name and revision in production is not recommended.

```
{"model_id": "2d1e7b165ef94724b3644d25abf81b0a", "version_major": 2, "version_minor": 0}
```

Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better performance, install the package with: `pip install huggingface_hub[hf_xet]` or `pip install hf_xet`

WARNING:huggingface_hub.file_download:Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better performance, install the package with: `pip install huggingface_hub[hf_xet]` or `pip install hf_xet`

```
{"model_id": "f1c008521e0c4438b4c0e30fbe92f70d", "version_major": 2, "version_minor": 0}
```



```
{"model_id": "6ecec83ec4944df1ba26798db5dbda62", "version_major": 2, "version_minor": 0}
```

```
{"model_id": "efca90b8c8ef41f692a16e25960f5ede", "version_major": 2, "version_minor": 0}
```

Device set to use cpu

```
def get_sentiment(input_text):  
    return sentiment(input_text)
```

```
result=get_sentiment("The boy is very good")
```

```
result
```

```
[{'label': 'POSITIVE', 'score': 0.9998672008514404}]
```

```
iface = gr.Interface(fn=get_sentiment,  
                    inputs=[gr.Textbox(lines=1, label="Write the  
sentence)],  
                    outputs=[gr.Textbox(lines=1, label="Understand  
the Sentiment Result)],  
                    title="Sentiment Analysis")  
iface.launch()
```

Running Gradio in a Colab notebook requires sharing enabled.
Automatically setting `share=True` (you can turn this off by setting
`share=False` in `launch()` explicitly).

Colab notebook detected. To show errors in colab notebook, set
debug=True in launch()
* Running on public URL: <https://aade4cc32219f25282.gradio.live>

This share link expires in 72 hours. 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.co/spaces>)

<IPython.core.display.HTML object>