```
import os
os.environ['GEMINI_API_KEY'] = 'AIzaSyCKN7Wuh467KfUhar14DJkw8wrZNrF2p4M'
```

```
!pip install -q -U google-generativeai
```

```
import google.generativeai as genai
genai.configure(api_key=os.environ['GEMINI_API_KEY'])
```

```
model = genai.GenerativeModel('gemini-2.5-flash-preview-04-17')
```

```
response = model.generate_content("create a table and explain about ai, generative ai and agentic ai")
print(response.text)
```

```
Okay, here is a table comparing AI, Generative AI, and Agentic AI, followed by explanations for each co

Think of it like this: **AI** is the vast field. **Generative AI** is a *specific type* of AI focused o

## Comparison Table

| Feature             | AI (Artificial Intelligence)                    | Generative AI
| :------------------ | :---------------------------------------------- | :-------------------------
| **Scope**           | Broadest field; encompasses all systems mimicking human intelligence. | A speci
| **Core Focus**      | Mimic human cognitive functions (learning, problem-solving, perception, decisio
| **Primary Function**| Process information, identify patterns, make predictions/decisions, automate ta
| **How it Works**    | Various techniques: Machine Learning (ML), Deep Learning, rules-based systems,
| **Key Capabilities**| Classification, regression, clustering, pattern recognition, prediction, optimi
| **Relationship**    | The overarching field. Generative AI and Agentic AI are developments *within* o
| **Examples**        | Traditional ML models (spam filters, recommendation systems), Expert Systems, R

## Explanations

1.  **AI (Artificial Intelligence):**
    Artificial Intelligence is the broadest field dedicated to creating systems or machines that can pe

2.  **Generative AI:**
    Generative AI is a specific and rapidly evolving *subset* of Artificial Intelligence focused on cre

3.  **Agentic AI:**
    Agentic AI refers to an *architectural approach* or *paradigm* where an AI system is designed to ac
```

```
import PIL.Image
img = PIL.Image.open(r"/content/sample_data/Balarama and SriKrishna.jpg")
img
```

```
model = genai.GenerativeModel('gemini-2.5-flash-preview-04-17')


response = model.generate_content(img)
print(response)
```

```
response:
GenerateContentResponse(
    done=True,
    iterator=None,
    result=protos.GenerateContentResponse({
      "candidates": [
        {
          "content": {
            "parts": [
              {
                "text": "This is a vibrant painting depicting a scene from Hindu mythology, likely feat
              }
            ],
            "role": "model"
          },
          "finish_reason": "STOP",
          "index": 0
        }
      ],
      "usage_metadata": {
        "prompt_token_count": 259,
        "candidates_token_count": 309,
```