DAY 5

Image Generation & Transformation Using AI

Today was a super exciting day. Everything was visual and interactive! I got to explore two really fun AI tools:

1. Text-to-Image Generation using Hugging Face
2. Image-to-Image Transformation using Stability AI (via Gradio).

Both projects were built using Python, and each of them showed how AI can turn simple text or image prompts into something really creative and visually impressive.

Prompt to Image (Hugging Face API)

This tool was all about taking a text prompt and generating a completely new image using an AI model hosted on Hugging Face.

What I Did:

- Loaded my Hugging Face API key securely using a .env file.
- Wrote a script that sends a text prompt to the model black-forest-labs/FLUX.1-dev.
- The model returns a newly generated image — which I saved as output_image.png (or any name I choose).
- Used Python's Pillow to open the image and get details like:
  - Format (e.g. PNG)
  - Dimensions (e.g. 512x512)

- ➢ Color mode (e.g. RGB)
- ➢ File size

- ❖ Then, I created a small HTML report that shows:
  - ➢ The prompt I used
  - ➢ The image it generated
  - ➢ Image properties in a table
- ❖ The HTML opened automatically in my browser — very clean and professional!

Example Prompt:"A robot drinking coffee on a snowy mountain during sunrise"

The output was a gorgeous AI art piece — surprisingly accurate and vibrant!

Image-to-Image with Prompt (Stability AI + Gradio)

The second tool was even more fun — I built a Gradio web app where I upload an image and provide a prompt. The AI then modifies or reimagines that image based on what I describe.

How It Worked:

- ❖ I used Stability AI's stable-diffusion-v1-6 engine via their API.
- ❖ The user uploads any image (e.g., a dog photo), and enters a creative prompt (e.g., "Turn this dog into a superhero").
- ❖ The script:
  - ➢ Resizes the image to 512x512 (required format)
  - ➢ Sends the image and prompt to the API
  - ➢ Gets back a newly generated version of the image

- I then displayed the result in Gradio, along with some cool image info:
  - Saved file path
  - Size, format, mode, etc.
- The final output was displayed right there on the Gradio interface with HTML styling.

Real Example:

- Upload: A selfie-style photo of a cat
- Prompt: “Make this cat look like an astronaut floating in space”
- Result: An amazing illustrated version of the cat, wearing a helmet and floating among stars!

What I Learned

- How to securely use and manage API keys using .env files
- How to make HTTP requests (both with and without files)
- Used Pillow (PIL) to read, resize, and inspect images
- Explored Gradio to make an interactive app with a clean web UI
- Learned how prompt tuning and parameters like cfg_scale and image_strength affect AI outputs

Tech Stack

Tool/Library	Purpose

requests	Making API calls
dotenv	Loading API keys securely
Pillow (PIL)	Handling image processing in Python
Gradio	Creating a UI for the image transformer
Hugging Face API	For prompt-to-image generation
Stability AI API	For image-to-image transformation

Why This Is Useful

- You can generate art, illustrations, and mockups in seconds.
- Useful for creative projects, content creators, game designers, and developers.
- Can be used in education, for example by turning concepts into visuals.
- Good for marketing and design work — generate drafts instantly.
- Combines text + visual prompts, making this a real example of multimodal AI.

Final Thoughts

Today's work was honestly one of the most fun experiences in the training so far. I got to:

- Work with real-world APIs
- Build an actual interactive image app
- Exploring how text and images can come together with the help of AI.