```python
# ===============================
# STEP 1: Install required libraries (run once)
# ===============================
!pip install diffusers transformers torch accelerate safetensors matplotlib --quiet

# ===============================
# STEP 2: Imports
# ===============================
from diffusers import StableDiffusionPipeline
import torch
import os
import matplotlib.pyplot as plt

# Enable inline plotting (important for notebooks)
%matplotlib inline

# ===============================
# STEP 3: Device & dtype handling
# ===============================
device = "cuda" if torch.cuda.is_available() else "cpu"
dtype = torch.float16 if device == "cuda" else torch.float32

# ===============================
# STEP 4: Load Stable Diffusion
# ===============================
model_id = "runwayml/stable-diffusion-v1-5"

pipe = StableDiffusionPipeline.from_pretrained(
    model_id,
    torch_dtype=dtype,
    safety_checker=None  # optional: disables NSFW blocking
).to(device)

# ===============================
# STEP 5: Output directory
# ===============================
output_dir = "synthetic_image_dataset"
os.makedirs(output_dir, exist_ok=True)

# ===============================
# STEP 6: Prompts
# ---------------------------------
```

```
" -------------------------------
prompts = [
    "A futuristic city at night with neon lights",
    "A cyberpunk warrior standing in rain",
    "A realistic astronaut walking on Mars",
    "A fantasy dragon flying over mountains",
    "An AI robot reading books in a library"
]

# ==============================
# STEP 7: Generate, save & display
# ==============================
for idx, prompt in enumerate(prompts):
    with torch.autocast(device):
        result = pipe(
            prompt,
            num_inference_steps=30,
            guidance_scale=7.5
        )

    image = result.images[0]

    # Save image
    image_path = os.path.join(output_dir, f"image_{idx+1}.png")
    image.save(image_path)

    # Display image
    plt.figure(figsize=(6, 6))
    plt.imshow(image)
    plt.axis("off")
    plt.title(f"{prompt}\nSaved as: image_{idx+1}.png")
    plt.show()

print("✅ Images successfully GENERATED, DISPLAYED, and SAVED!")
```

```
Flax classes are deprecated and will be removed in Diffusers v1.0.0. We recommend migrating to PyTorch classes or pinning your version of Diffusers.
Flax classes are deprecated and will be removed in Diffusers v1.0.0. We recommend migrating to PyTorch classes or pinning your version of Diffusers.
/usr/local/lib/python3.12/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 resta
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: 100%   541/541 [00:00<00:00, 58.2kB/s]

Fetching 13 files: 100%   13/13 [00:39<00:00,   3.07s/it]

special_tokens_map.json: 100%   472/472 [00:00<00:00, 4.86kB/s]

scheduler_config.json: 100%   308/308 [00:00<00:00, 3.31kB/s]

merges.txt: 525k/? [00:00<00:00, 1.44MB/s]

config.json: 100%   617/617 [00:00<00:00, 9.01kB/s]

vocab.json: 1.06M/? [00:00<00:00, 1.62MB/s]

preprocessor_config.json: 100%   342/342 [00:00<00:00, 3.72kB/s]

tokenizer_config.json: 100%   806/806 [00:00<00:00, 9.98kB/s]

text_encoder/model.safetensors: 100%   492M/492M [00:37<00:00, 8.74MB/s]

config.json: 100%   743/743 [00:00<00:00, 35.7kB/s]

config.json: 100%   547/547 [00:00<00:00, 25.9kB/s]

unet/diffusion_pytorch_model.safetensors: 100%   3.44G/3.44G [00:39<00:00, 242MB/s]

vae/diffusion_pytorch_model.safetensors: 100%   335M/335M [00:05<00:00, 74.0MB/s]

Loading pipeline components...: 100%   6/6 [00:11<00:00,   2.54s/it]

```
`torch_dtype` is deprecated! Use `dtype` instead!
You have disabled the safety checker for <class 'diffusers.pipelines.stable_diffusion.pipeline_stable_diffusion.StableDiffusionPipeline'> by passing `safety_checker=Non
```

100%   30/30 [00:05<00:00,   6.84it/s]

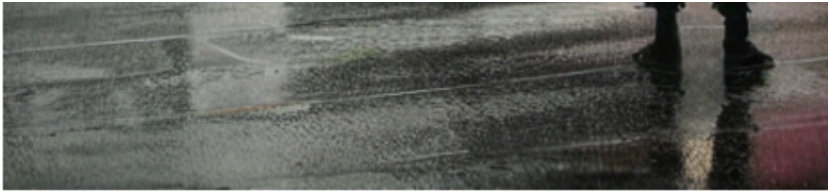A futuristic city at night with neon lights
Saved as: image_1.png

100%   30/30 [00:04<00:00,   6.82it/s]
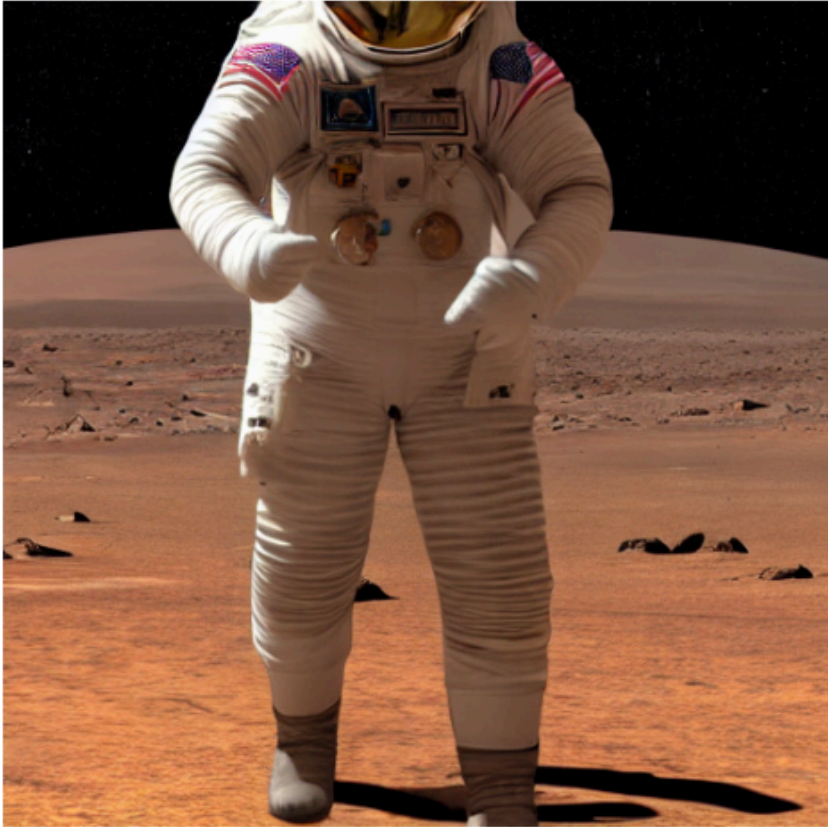
A cyberpunk warrior standing in rain
Saved as: image_2.png

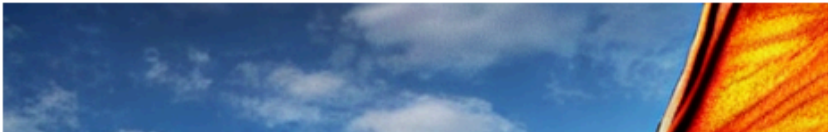100%   30/30 [00:04<00:00,   6.74it/s]

A realistic astronaut walking on Mars
Saved as: image_3.png

100%   30/30 [00:07<00:00,   3.73it/s]

A fantasy dragon flying over mountains
Saved as: image_4.png

100%   30/30 [00:07<00:00,   6.66it/s]