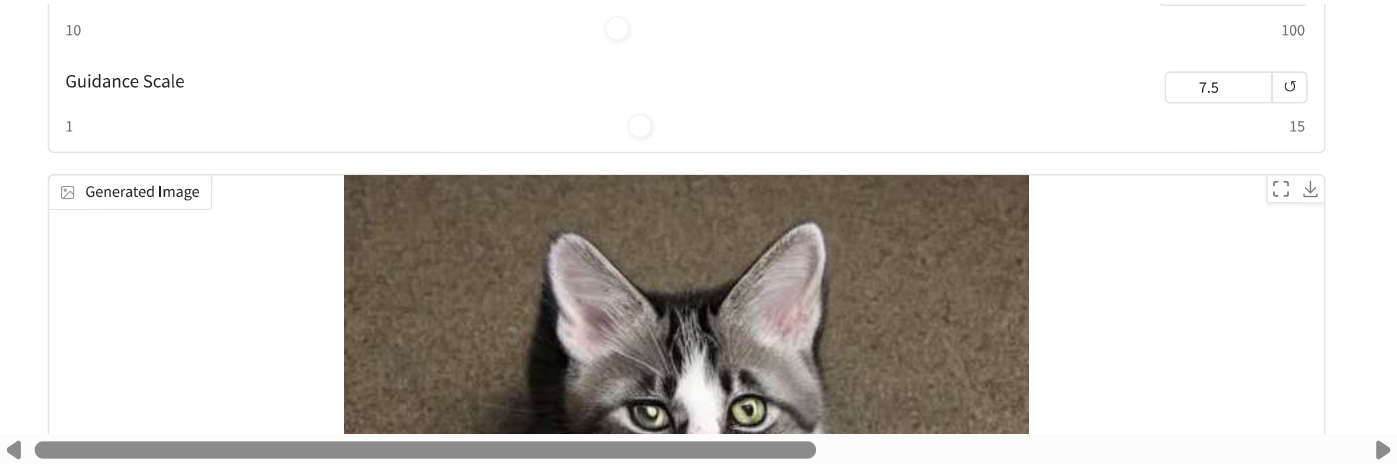
```
# Step 1: Install necessary packages
!pip install --quiet diffusers transformers accelerate safetensors

# Step 2: Import dependencies
from diffusers import StableDiffusionPipeline
import torch

# Step 3: Load the model from Hugging Face
model_id = "runwayml/stable-diffusion-v1-5"
pipe = StableDiffusionPipeline.from_pretrained(
    model_id,
    torch_dtype=torch.float16, # Specify the data type
    use_safetensors=True,
    # Remove the revision="fp16" parameter as it's not a valid revision for this model ID
)
pipe = pipe.to("cuda")

# Step 4: Define inference function
def generate_image(prompt, num_inference_steps=50, guidance_scale=7.5):
    image = pipe(prompt=prompt, num_inference_steps=num_inference_steps, guidance_scale=guidance_scale).images[0]
    return image

# Example usage (can be commented out in production)
# image = generate_image("a futuristic city at sunset")
# image.show()
```

```
model_index.json: 100%  541/541 [00:00<00:00, 39.4kB/s]
Fetching 15 files: 100%  15/15 [00:37<00:00,  2.66s/it]
model.safetensors: 100%  1.22G/1.22G [00:22<00:00, 111MB/s]
model.safetensors: 100%  492M/492M [00:14<00:00, 71.8MB/s]
config.json: 100%  4.72k/4.72k [00:00<00:00, 176kB/s]
preprocessor_config.json: 100%  342/342 [00:00<00:00, 19.5kB/s]
merges.txt: 100%  525k/525k [00:00<00:00, 3.73MB/s]
config.json: 100%  617/617 [00:00<00:00, 11.0kB/s]
special_tokens_map.json: 100%  472/472 [00:00<00:00, 16.2kB/s]
scheduler_config.json: 100%  308/308 [00:00<00:00, 3.86kB/s]
tokenizer_config.json: 100%  806/806 [00:00<00:00, 28.9kB/s]
diffusion_pytorch_model.safetensors: 100%  3.44G/3.44G [00:37<00:00, 159MB/s]
config.json: 100%  743/743 [00:00<00:00, 17.7kB/s]
config.json: 100%  547/547 [00:00<00:00, 18.3kB/s]
diffusion_pytorch_model.safetensors: 100%  335M/335M [00:10<00:00, 59.9MB/s]
vocab.json: 100%  1.06M/1.06M [00:00<00:00, 7.22MB/s]
Loading pipeline components...: 100%  7/7 [00:23<00:00,  5.22s/it]
```

```
# Step 1: Install packages
!pip install --quiet gradio diffusers transformers accelerate safetensors

# Step 2: Import libraries
import gradio as gr
from diffusers import StableDiffusionPipeline
import torch

# Step 3: Load model
model_id = "runwayml/stable-diffusion-v1-5"
pipe = StableDiffusionPipeline.from_pretrained(
    model_id,
    torch_dtype=torch.float16,
    use_safetensors=True,
    # Remove the revision="fp16" parameter as it's not a valid revision
).to("cuda")

# Step 4: Define generation function
def generate(prompt, num_inference_steps=50, guidance_scale=7.5):
```

```
    image = pipe(prompt, num_inference_steps=num_inference_steps, guidance_scale=guidance_scale).images[0]
    return image

# Step 5: Create Gradio UI
with gr.Blocks() as demo:
    gr.Markdown("# Text-to-Image Generator using Stable Diffusion")
    prompt = gr.Textbox(label="Prompt", placeholder="Enter your imagination...")
    steps = gr.Slider(10, 100, value=50, label="Inference Steps")
    scale = gr.Slider(1.0, 15.0, value=7.5, label="Guidance Scale")
    output = gr.Image(label="Generated Image")

    btn = gr.Button("Generate")
    btn.click(fn=generate, inputs=[prompt, steps, scale], outputs=output)

demo.launch()
```

```
Loading pipeline components...: 100%  7/7 [00:16<00:00,  3.17s/it]
It looks like you are running Gradio on a hosted a Jupyter notebook. For the Gradio app to work, sharing must be enabled. Automatically

Colab notebook detected. To show errors in colab notebook, set debug=True in launch()
* Running on public URL: https://1649f9940063edb5eb.gradio.live

This share link expires in 1 week. For free permanent hosting and GPU upgrades, run `gradio deploy` from the terminal in the working dir
```



Start coding or generate with AI.