



# FLUX Installation and Usage Guide on Windows with GPU

Optimize your model generation experience with FLUX!



## System Requirements

Before you begin, ensure that your system meets these minimum requirements:

- Operating System: Windows 10 or higher
- CPU: 4-core processor or better
- RAM: Minimum 16 GB, 32 GB or more recommended
- GPU: NVIDIA with at least 8 GB of VRAM, CUDA-compatible
- Disk Space: At least 20 GB of free space
- Internet Connection: Required to download models and dependencies

**IMPORTANT:** FLUX models are quite large and require significant resources. A system with higher specs will provide better performance and faster generation times.



## Installation Steps

1. **Install Anaconda**
  - Download and install Anaconda: [Video](#)
  - Run the installer and follow the instructions
2. **Open Command Prompt or PowerShell as Administrator**
3. **Create and activate a Conda environment**

```
conda create --name flux python=3.10
```

```
conda activate flux
```

4. 📁 **Navigate to the directory where you want to install FLUX**

- Choose a location with enough free space
- For example: `cd C:\Projects`

5. 📄 **Clone the FLUX repository**

```
git clone https://github.com/black-forest-labs/flux
```

6. 📁 **Enter the project directory**

```
cd flux
```

7. ⚙️ **Install FLUX and its dependencies**

```
pip install -e .[all]
```

8. 🔥 **Install PyTorch with CUDA support**

```
conda install pytorch torchvision torchaudio pytorch-cuda=11.8 -c pytorch -c nvidia
```

This worked for me, but you can check the PyTorch page and try depending on your computer's specs [Pytorch.org](https://pytorch.org)

9. 📦 **Install additional dependencies**

```
pip install diffusers transformers accelerate jupyterlab
```

10. 🚀 **Launch Jupyter Lab**

```
jupyter lab
```

## Usage in Jupyter Lab

Once in Jupyter Lab, create a new notebook and use the following code:

```
import torch
from diffusers import FluxPipeline

# Choose the model you want to use
model_id = "black-forest-labs/FLUX.1-dev" # or "black-forest-labs/FLUX.1-schnell"

# Initialize the pipeline
pipe = FluxPipeline.from_pretrained(model_id, torch_dtype=torch.float16)
pipe = pipe.to("cuda")
pipe.enable_model_cpu_offload()

# Define the prompt and seed
prompt = "A cat holding a sign that says subscribe for more"
seed = 42

# Generate the image
image = pipe(
    prompt,
    output_type="pil",
    num_inference_steps=4, # Use a higher number if using [dev]
    generator=torch.Generator("cuda").manual_seed(seed)
).images[0]

# Save the image
image.save("flux-generated.png")






# Display the image in the notebook
from IPython.display import display
display(image)
```



## Important Notes

---

**⚠ NOTICE:** The first time you run the code, the selected model will be downloaded, which may take several minutes depending on your internet connection.

-  To switch between models, modify the `model_id` variable:
  - `"black-forest-labs/FLUX.1-schnell"` for the fast model
  - `"black-forest-labs/FLUX.1-dev"` for the development model (requires HuggingFace login)
-  If using FLUX.1-dev, consider increasing `num_inference_steps` for better quality.
-  Adjust `torch_dtype` to `torch.float32` if you encounter memory issues.
-  The image will be saved in the current working directory of Jupyter Lab.
-  If you experience memory or performance issues, close other resource-intensive applications.

## Additional Options

---

-  For interactive use from the command line:

```
python -m flux --name <name> --loop
```



-  To generate a single sample from the command line:

```
python -m flux --name <name> --height <height> --width <width> --prompt "<prompt>"
```

-  Streamlit and Gradio demos are also available. Check the GitHub repository for more details.

## Licensing Note

---

-  FLUX.1 [schnell] is under the Apache-2.0 license
-  FLUX.1 [dev] has a specific non-commercial license

**i INFO:** For more details on API usage or advanced information, refer to the official documentation in the FLUX GitHub repository.