```
!pip install -q transformers accelerate torch fastapi uvicorn nest_asyncio huggingface_
!wget -q https://github.com/cloudflare/cloudflared/releases/latest/download/cloudflare
!chmod +x cloudflared
```

```
from huggingface_hub import login
login()                    # paste your HF access-token when prompted
```

```
import torch, gc
from transformers import AutoTokenizer, AutoModelForCausalLM

model_id = "google/gemma-3-12b-it"

print("⏳ Loading Gemma-3 12B…")
tokenizer = AutoTokenizer.from_pretrained(model_id)
model     = AutoModelForCausalLM.from_pretrained(
    model_id,
    torch_dtype=torch.float16,
    device_map="auto",
    low_cpu_mem_usage=True
)
model.eval(); gc.collect()
print("✅ Model ready")
```

```
⏳ Loading Gemma-3 12B…
tokenizer_config.json: 100%  1.16M/1.16M [00:00<00:00, 16.1MB/s]
tokenizer.model: 100%  4.69M/4.69M [00:00<00:00, 44.7MB/s]
tokenizer.json: 100%  33.4M/33.4M [00:00<00:00, 205MB/s]
added_tokens.json: 100%  35.0/35.0 [00:00<00:00, 4.62kB/s]
special_tokens_map.json: 100%  662/662 [00:00<00:00, 87.7kB/s]
config.json: 100%  916/916 [00:00<00:00, 123kB/s]
model.safetensors.index.json: 100%  109k/109k [00:00<00:00, 6.15MB/s]
Fetching 5 files: 100%  5/5 [01:02<00:00, 62.14s/it]
model-00001-of-00005.safetensors: 100%  4.98G/4.98G [01:02<00:00, 102MB/s]
model-00004-of-00005.safetensors: 100%  4.93G/4.93G [01:01<00:00, 98.9MB/s]
model-00005-of-00005.safetensors: 100%  4.60G/4.60G [01:00<00:00, 75.5MB/s]
model-00002-of-00005.safetensors: 100%  4.93G/4.93G [01:01<00:00, 80.7MB/s]
```

```
!ls
```

```
cloudflared  sample_data
```

Generated code may be subject to a license | openlangrid/mlgrid-services

```
def generate(prompt, **gen_kwargs):
    inputs = tokenizer(prompt, return_tensors="pt", truncation=True).to(model.device)

    with torch.no_grad():
        out_ids = model.generate(
            **inputs,
            max_new_tokens=150,
            do_sample=True,
            temperature=0.7,
            top_p=0.9,
            repetition_penalty=1.1,
            pad_token_id=tokenizer.eos_token_id,         # critical for out-of-vocab edg
            **gen_kwargs
```

dError RuntimeError RuntimeError •••

You are subscribed to Colab Pro. Learn more
Available: 36.89 compute units
Usage rate: approximately 7.62 per hour
You have 1 active session.

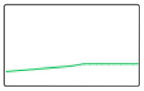
Manage sessions

Python 3 Google Compute Engine backend (GPU)
Showing resources from 9:50 AM to 9:53 AM

System RAM
4.1 / 83.5 GB

GPU RAM
25.1 / 40.0 GB

Disk
63.8 / 235.7 GB



```
        )

    decoded = tokenizer.decode(out_ids[0], skip_special_tokens=True)
    return decoded

# Test again
print(generate("Explain in one sentence why the sky appears blue."))
```

```
Asking to truncate to max_length but no maximum length is provided and the model h
---------------------------------------------------------------------------
RuntimeError                              Traceback (most recent call last)
<ipython-input-7-d05a8412a460> in <cell line: 0>()
     18
     19 # Test again
---> 20 print(generate("Explain in one sentence why the sky appears blue."))

                                 3 frames
/usr/local/lib/python3.11/dist-packages/transformers/generation/utils.py in
_sample(self, input_ids, logits_processor, stopping_criteria, generation_config,
synced_gpus, streamer, **model_kwargs)
   3474                 probs = nn.functional.softmax(next_token_scores, dim=-1)
   3475                 # TODO (joao): this OP throws "skipping cudagraphs due to
['incompatible ops']", find solution
-> 3476                 next_tokens = torch.multinomial(probs,
num_samples=1).squeeze(1)
   3477             else:
   3478                 next_tokens = torch.argmax(next_token_scores, dim=-1)

RuntimeError: CUDA error: device-side assert triggered
CUDA kernel errors might be asynchronously reported at some other API call, so
the stacktrace below might be incorrect.
For debugging consider passing CUDA_LAUNCH_BLOCKING=1
Compile with `TORCH_USE_CUDA_DSA` to enable device-side assertions.
```

Next steps: Explain error

```
!ls
```

```
cloudflared  sample_data
```

Change runtime type