Bài 1:

```
from transformers import pipeline

# 1. Tải pipeline "fill-mask"
mask_filler = pipeline("fill-mask")

# 2. Câu đầu vào có token MASK
input_sentence = "Hanoi is the <mask> of Vietnam."

# 3. Dự đoán 5 từ khả dĩ nhất
predictions = mask_filler(input_sentence, top_k=5)

# 4. In kết quả
print("Câu gốc:", input_sentence)
for pred in predictions:
    print(f"- {pred['token_str']:10s} | score = {pred['score']:.4f}")
    #print("  =>", pred['sequence'])
```

```
No model was supplied, defaulted to distilbert/distilroberta-base and revision fb53ab8 (https://huggingface.co/distilbert/distil
Using a pipeline without specifying a model name and revision in production is not recommended.
Some weights of the model checkpoint at distilbert/distilroberta-base were not used when initializing RobertaForMaskedLM: ['robe
- This IS expected if you are initializing RobertaForMaskedLM from the checkpoint of a model trained on another task or with ano
- This IS NOT expected if you are initializing RobertaForMaskedLM from the checkpoint of a model that you expect to be exactly i
Device set to use cpu
Câu gốc: Hanoi is the <mask> of Vietnam.
-  capital    | score = 0.9341
-  Republic   | score = 0.0300
-  Capital    | score = 0.0105
-  birthplace | score = 0.0054
-  heart      | score = 0.0014
```

Bài 2:

```
from transformers import pipeline

# Tải mô hình GPT-2
generator = pipeline("text-generation")

prompt = "The best thing about learning NLP is"

generated = generator(prompt, max_length=50, num_return_sequences=1)

print("Kết quả sinh:")
print(generated[0]["generated_text"])
```

```
No model was supplied, defaulted to openai-community/gpt2 and revision 607a30d (https://huggingface.co/openai-community/gpt2).
Using a pipeline without specifying a model name and revision in production is not recommended.
config.json:  100%                                          665/665  [00:00<00:00,  16.7kB/s]

model.safetensors:  100%                                        548M/548M  [00:11<00:00,  98.9MB/s]

generation_config.json:  100%                                      124/124  [00:00<00:00,  3.55kB/s]

tokenizer_config.json:  100%                                      26.0/26.0  [00:00<00:00,  643B/s]

vocab.json:       1.04M/?  [00:00<00:00,  18.0MB/s]

merges.txt:       456k/?  [00:00<00:00,  5.17MB/s]

tokenizer.json:       1.36M/?  [00:00<00:00,  24.1MB/s]

Device set to use cpu
Truncation was not explicitly activated but `max_length` is provided a specific value, please use `truncation=True` to explicitl
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
Both `max_new_tokens` (=256) and `max_length`(=50) seem to have been set. `max_new_tokens` will take precedence. Please refer to
Kết quả sinh:
The best thing about learning NLP is that sometimes you don't have to do everything. I have learned a lot of stuff from NLP, and

Q: Do you think that your best friend you know is at MIT?

A: Yes, I think he's very good. He's a really good person, and he's been doing the best job of writing the best books I have eve

Q: You've been making some interesting connections.

A: Oh, I'm sure. I think I've been in a lot of good places, but I've been in a lot of bad places. I think my best friend is at M
```

Bài 3:

```
import torch
from transformers import AutoTokenizer, AutoModel

model_name = "bert-base-uncased"

tokenizer = AutoTokenizer.from_pretrained(model_name)
model = AutoModel.from_pretrained(model_name)

sentence = ["This is a sample sentence."]

inputs = tokenizer(sentence, padding=True, truncation=True, return_tensors='pt')

with torch.no_grad():
    outputs = model(**inputs)

last_hidden_state = outputs.last_hidden_state

# Mean Pooling có mask
attention_mask = inputs['attention_mask']
mask_expanded = attention_mask.unsqueeze(-1).expand(last_hidden_state.size()).float()

sum_embeddings = torch.sum(last_hidden_state * mask_expanded, dim=1)
sum_mask = torch.clamp(mask_expanded.sum(1), min=1e-9)

sentence_embedding = sum_embeddings / sum_mask

print(sentence_embedding)
print("Shape:", sentence_embedding.shape)
```