```
from transformers import pipeline
import random


qa_pipeline = pipeline("question-answering")
summarizer = pipeline("summarization", model="sshleifer/distilbart-cnn-12-6")
```

```
No model was supplied, defaulted to distilbert/distilbert-base-cased-distilled-squad and revision 564e9b5 (https://huggingface.co/di
Using a pipeline without specifying a model name and revision in production is not recommended.
/usr/local/lib/python3.11/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 s
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(
config.json: 100%                                   473/473 [00:00<00:00, 37.9kB/s]
Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better p
WARNING:huggingface_hub.file_download:Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back
model.safetensors: 100%                                   261M/261M [00:01<00:00, 192MB/s]
tokenizer_config.json: 100%                                   49.0/49.0 [00:00<00:00, 3.57kB/s]
vocab.txt: 100%                                   213k/213k [00:00<00:00, 4.15MB/s]
tokenizer.json: 100%                                   436k/436k [00:00<00:00, 1.59MB/s]
Device set to use cpu
config.json: 100%                                   1.80k/1.80k [00:00<00:00, 117kB/s]
pytorch_model.bin: 100%                                   1.22G/1.22G [00:09<00:00, 183MB/s]
model.safetensors: 100%                                   1.22G/1.22G [00:12<00:00, 110MB/s]
tokenizer_config.json: 100%                                   26.0/26.0 [00:00<00:00, 782B/s]
vocab.json: 100%                                   899k/899k [00:00<00:00, 6.59MB/s]
merges.txt: 100%                                   456k/456k [00:00<00:00, 11.6MB/s]
Device set to use cpu
```

```
concepts = {
    "os": "An operating system (OS) manages computer hardware and software resources and provides common services for computer programs.
    "data structure": "A data structure is a storage format that enables efficient access and modification of data.",
    "recursion": "Recursion is a method of solving a problem where the solution depends on solving smaller instances of the same probler
}


sample_text = """
Python is an interpreted, high-level, general-purpose programming language. Its design philosophy emphasizes code readability.
Python supports multiple programming paradigms, including structured, object-oriented, and functional programming.
"""


def explain_concept():
    topic = input("Enter a topic (e.g., OS, Data Structure): ").lower()
    print("📘 Explanation:", concepts.get(topic, "Sorry, I don't have info on that."))


def summarize_topic():
    print("📚 Summary:")
    print(summarizer(sample_text, max_length=50, min_length=25, do_sample=False)[0]['summary_text'])


def ask_question():
    question = input("Ask a question: ")
    print("❓ Answer:")
    print(qa_pipeline({'question': question, 'context': sample_text})['answer'])


def generate_mcq():
    question = "What is Python primarily used for?"
    options = ['Low-level programming', 'Database management', 'Web and software development', 'Network configuration']
    correct = 2
    print("📝 MCQ:")
    print(question)
    for i, option in enumerate(options):
        print(f"{i+1}. {option}")
    print("✅ Answer:", options[correct])


while True:
    print("\n🤖 EduBot Options:\n1. Explain Concept\n2. Summarize Topic\n3. Ask a Question\n4. Generate MCQ\n5. Exit")
    choice = input("Choose: ")
    if choice == '1':
        explain_concept()
    elif choice == '2':
        summarize_topic()
    elif choice == '3':
        ask_question()
    elif choice == '4':
        generate_mcq()
    elif choice == '5':
        break
    else:
        print("Invalid choice.")
```

```
🤖 EduBot Options:
1. Explain Concept
2. Summarize Topic
3. Ask a Question
4. Generate MCQ
5. Exit
📘 Explanation: An operating system (OS) manages computer hardware and software resources and provides common services for computer

🤖 EduBot Options:
1. Explain Concept
2. Summarize Topic
3. Ask a Question
4. Generate MCQ
5. Exit
📘 Explanation: An operating system (OS) manages computer hardware and software resources and provides common services for computer

🤖 EduBot Options:
1. Explain Concept
2. Summarize Topic
3. Ask a Question
4. Generate MCQ
5. Exit
Your max_length is set to 50, but your input_length is only 49. Since this is a summarization task, where outputs shorter than the i
📚 Summary:
 Python is an interpreted, high-level, general-purpose programming language . Its design philosophy emphasizes code readability . It

🤖 EduBot Options:
1. Explain Concept
2. Summarize Topic
3. Ask a Question
4. Generate MCQ
5. Exit
```

Start coding or generate with AI.