**Project Introduction: Advanced Text Generation Application**

**Objective**

The goal of this project is to create an advanced text generation application using the Hugging Face transformers library and the GPT-2 model. This application will allow users to input a prompt, customize text generation parameters, generate multiple text outputs, and save these outputs to a file.

**Prerequisites**

Before you start, ensure you have the following:

1. **Python Version**: Python 3.7 or later installed on your machine.

<https://www.python.org/downloads/>

1. **Package Installation**: Necessary libraries required for this project (refer below).
2. **Operating System**: Compatibility with Windows, macOS, or Linux (recommended).

**Step-by-Step Instructions**

**Step 1: Install Required Packages**

1. **Open your terminal or command prompt**.
2. **Install Required Libraries**: Run the following command to install the necessary packages:

pip install transformers torch or

pip3 install transformers torch

**Step 2: Create the Python Script and implement the code**

1. **Create a new Python file**:
   * Name it ‘text\_generator.py’ or any name you prefer.
2. **Open the file in your text editor or IDE**.
3. **Implement the Code**

Copy and paste the following code into your text\_generator.py file. This code includes the necessary logic for text generation.

import time

from transformers import pipeline

# Load the text generation pipeline using GPT-2

generator = pipeline('text-generation', model='gpt2', device=-1)  # Use CPU

def generate\_text(prompt, max\_length=100, temperature=0.7, num\_return\_sequences=1):

    results = generator(prompt, max\_length=max\_length, num\_return\_sequences=num\_return\_sequences, temperature=temperature)

    return [result['generated\_text'] for result in results]

def save\_to\_file(text, filename='generated\_text.txt'):  # Change the filename to save the results to a different file

    with open(filename, 'w', encoding='utf-8') as file:

        file.write(text)

    print(f"Generated text saved to {filename}")

def main():

    print("Welcome to the Advanced Text Generator!")

    prompt = input("Enter your prompt text: ")

    max\_length = int(input("Enter the maximum length of generated text (recommended 30-100): "))

    temperature = float(input("Enter temperature (recommended range 0.5-1.0, higher means more random): "))

    num\_return\_sequences = int(input("Enter the number of text paragraphs to generate: "))

    # Start timing

    start\_time = time.time()

    # Generate text

    generated\_texts = generate\_text(prompt, max\_length=max\_length, temperature=temperature, num\_return\_sequences=num\_return\_sequences)

    # Stop timing

    end\_time = time.time()

    print(f"\nText generation took: {end\_time - start\_time:.2f} seconds\n")

    for i, generated\_text in enumerate(generated\_texts, 1):

        print(f"Generated Text Paragraph {i}:")

        print(generated\_text)

        print("-" \* 40)

    save\_choice = input("Would you like to save the generated text to a file? (yes/no): ")

    if save\_choice.lower() == 'yes':

        all\_texts = "\n\n".join(generated\_texts)

        save\_to\_file(all\_texts)

if \_\_name\_\_ == "\_\_main\_\_":

    main()

**Step 3: Run the Application**

1. **Open your terminal/command prompt**.
2. **Navigate to the directory** where your text\_generator.py file is located.
3. **Run the Python script**:

python text\_generator.py or

python3 text\_generator.py

1. **Follow the prompts** in the terminal to input your text prompt, set text generation parameters, and choose whether to save the generated text.

**Step 4: Test and Experiment**

1. **Try different prompts**: Explore how the model responds to various input prompts.
2. **Adjust the parameters**: Experiment with different values for maximum length, temperature, and number of outputs. Observe how it influences the generated text.

**Step 5: Make Observation and Conclusion**

1. Make your own observations and conclusions based on the results and save it to a Word Document.
2. Please write down your observations and conclusions in a Word document, and take a screenshot of your outputs and put it in the Word document. Once completed, upload the document to the quiz.