\*\*Assignment : Build an end to end AI system\*\*

=====================================

### Part 1: Clone Content Creator's Style (Instagram Scraper)

```python

def scrape\_instagram(username):

"""

Scrape the latest captions from an Instagram user's profile.

Note: This method uses Selenium, so ensure you have ChromeDriver installed.

"""

print(f"[INFO] Starting Instagram scrape for user: {username}...")

url = f"https://www.instagram.com/{username}/"

options = webdriver.ChromeOptions()

options.add\_argument("--headless")

driver = webdriver.Chrome(service=Service(ChromeDriverManager().install()), options=options)

# Capturing data

captions = []

try:

driver.get(url)

time.sleep(5) # Wait for the page to load fully

posts = driver.find\_elements(By.CLASS\_NAME, "\_aagv")

print("[INFO] Extracting captions...")

for post in posts:

text = post.text.strip()

if text:

captions.append(text)

driver.quit()

print(f"[SUCCESS] Captured {len(captions)} captions")

return captions

except Exception as e:

print(f"[ERROR] Failed to scrape Instagram: {e}")

driver.quit()

return []

```

### Part 2: Generate Personalized Product Reviews (Template-based)

```python

def generate\_review(product\_description):

"""

Generate a personalized product review using a template-based approach.

This simulates human-created reviews by filling in predefined templates.

"""

# Template for the review

templates = [

"This product is fantastic! The {product\_name} has an impressive {feature}, making it ideal for {use\_case}.",

"I've been using the {product\_name} for a while now, and I love how {feature} makes my {use\_case} much easier.",

"If you're looking for a product that excels in {feature}, the {product\_name} is a great choice for {use\_case}."

]

# Sample dynamic elements

product\_name = product\_description.get("name", "Product")

feature = product\_description.get("feature", "great performance")

use\_case = product\_description.get("use\_case", "daily use")

# Select a random template and generate the review

review = templates[0].format(product\_name=product\_name, feature=feature, use\_case=use\_case)

return review

```

### Part 3: Generate Video Scripts (Template-based)

```python

def generate\_video\_script(topic):

"""

Generate a video script using a template-based approach.

This simulates a human-created script by following a basic structure.

"""

# Template for the video script

templates = [

"Welcome to today's video where we discuss {topic}. In this video, we'll cover {main\_points}. Stay tuned until the end for {call\_to\_action}.",

"Hi everyone! In today's video, we're going to talk about {topic}. First, we'll explore {main\_points}, and then I'll give you some tips on {call\_to\_action}.",

"Welcome back! In this video, we're diving deep into {topic}. Make sure to stick around as we discuss {main\_points} and share {call\_to\_action}."

]

# Sample dynamic elements

main\_points = "the key benefits of productivity hacks"

call\_to\_action = "how you can boost your efficiency"

# Select a template and generate the video script

script = templates[0].format(topic=topic, main\_points=main\_points, call\_to\_action=call\_to\_action)

return script

```

### Part 4: Synthesizing Voice Clips

```python

def synthesize\_voice(text):

"""

Convert a given text to speech using pyttsx3 and save as an audio file.

"""

print(f"[INFO] Synthesizing voice for text length: {len(text)}")

engine = pyttsx3.init()

engine.setProperty('rate', 150) # Speed of speech

engine.setProperty('volume', 0.9) # Volume level

engine.save\_to\_file(text, 'voice\_output.mp3')

engine.runAndWait()

print("[SUCCESS] Voice clip saved as 'voice\_output.mp3'.")

```

### Main Execution Function

```python

def main():

# Sample inputs for testing

instagram\_username = "creator\_username"

# Step 1: Scrape Instagram Content (Unchanged)

captions = scrape\_instagram(instagram\_username)

if captions:

print("\n--- Instagram Captions (Sample) ---")

for i, caption in enumerate(captions[:3]):

print(f"{i+1}. {caption}")

# Step 2: Generate a Product Review

product\_description = {

"name": "SuperFast Smartphone",

"feature": "super-fast processing speed",

"use\_case": "gaming"

}

review = generate\_review(product\_description)

print("\n--- Generated Product Review ---\n", review)

# Step 3: Generate Video Script

video\_topic = "Top 5 Tips for Boosting Your Productivity"

video\_script = generate\_video\_script(video\_topic)

print("\n--- Generated Video Script ---\n", video\_script)

# Step 4: Synthesize Voice

synthesize\_voice(video\_script)

print("\n[INFO] Assignment completed successfully!")

# Only run the main function if this script is executed directly

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

main()

```