Task 3 : Web Scraper for News Headlines.

- **Objective: Scrape top headlines from a news website.**
- **Tools : Python, requests, BeautifulSoup**
- **Deliverables: Python script + .txt file of headlines**

Hints/Mini Guide:

1. Use requests to fetch HTML
2. Use BeautifulSoup to parse <h2> or title tags
3. Save the titles in a .txt file

- **Outcome: :** Automate data collection from a public website

Interview Questions:

1. What is a GET request?
2. How do you install external packages in Python?
3. What is a User-Agent in HTTP?
4. What is soup.find_all() used for?
5. What are the risks of web scraping?
6. What's the difference between id and class in HTML?
7. What is an HTML tag?
8. What does .text return in BeautifulSoup?
9. What is a try-except block?
10. What are HTTP status codes?

Key Concepts: HTTP requests, Web scraping, HTML parsing

📤 **Submit Here:**

After completing the task, paste your GitHub repo link and submit it using the link below:

- 👉 [Submission Link]

📌 Task Submission Guidelines

- ⏰ **Time Window:**

You can complete the task anytime between 10:00 AM to 10:00 PM on the given day. Submission link closes at 10 :00 PM

- 🔍 **Self-Research Allowed:**

You are free to explore, Google, or refer to tutorials to understand concepts and complete the task effectively.

- 🛠️ **Debug Yourself:**

Try to resolve all errors by yourself. This helps you learn problem-solving and ensures you don't face the same issues in future tasks.

- 💸 **No Paid Tools:**

If the task involves any paid software/tools, do not purchase anything. Just learn the process or find free alternatives.

- 📁 **GitHub Submission:**

Create a new GitHub repository for each task.

Add everything you used for the task – code, datasets, screenshots (if any), and a **short README.md** explaining what you did.

📤 **Submit Here:**

After completing the task, paste your GitHub repo link and submit it using the link below:

- 👉 [Submission Link]