Task 4: Make a Website Mobile-Friendly Using CSS Media Queries

- **Objective: Convert an existing desktop-only page to a mobile-friendly layout using media queries.**
- **Tools: Any existing HTML file, VS Code, Chrome DevTools.**
- **Deliverables: Updated CSS with responsive design working on mobile viewport.**

Hints/Mini Guide:

1. Open the HTML page in VS Code.
2. Identify fixed width elements or large images.
3. Write media queries targeting max-width 768px.
4. Adjust layout: stack columns vertically, reduce font sizes.
5. Make nav menu collapse or stack vertically.
6. Test with Chrome DevTools device toolbar.
7. Fix overflow and scrolling issues.
8. Ensure images scale within containers.

Outcome: : Grasp media queries, mobile-first design, flexible layouts.

Interview Questions:

1. What are media queries?
2. Explain mobile-first vs desktop-first CSS design.
3. How do you test responsiveness?
4. What units are best for responsive layouts?
5. What is viewport meta tag?
6. How does flexbox help in responsive design?
7. Difference between absolute and relative units?
8. How to handle images in responsive design?
9. What is adaptive vs responsive design?
10. Explain CSS grid responsiveness.

Key Concepts: Media Queries, Responsive Web Design, Viewport, CSS Units (%, rem, vw).

📤 **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]