Task Description: Build a Simple "Task Manager" Application with WebSocket

Time limit: 30 minutes.

Objective: Create a task management application that includes backend and frontend components and utilizing WebSocket for real-time updates.

Backend (Java - Spring Boot):-

1. Setup: Create a Spring Boot application with the following specifications:
   - Use Spring Web for building RESTful APIs.
   - Use Spring WebSocket for real-time communication.
   - Use an in-memory database (like H2) for simplicity.
2. Requirements:
   - Implement a REST API with the following endpoints:
   - Create a new task :-The request body should include title ,description , status (default pending)
   - Add validation:- both string, mandatory ,title maximum length 10, description min length 10 and maximum 500
   - Retrieve all tasks.
   - Retrieve a specific task by its ID.
   - Update a specific task by its ID.
   - Update a task's status (status can be Pending, In Progress, Completed).
   - Delete a task by its ID.
3. WebSocket Endpoint: Create a WebSocket endpoint to broadcast updates when tasks are created, updated, or deleted.
   - Error Handling: Implement basic error handling (e.g., return 404 if a task is not found).

Frontend (Angular):-

1. Use Angular Forms for task creation and updates.
2. Use WebSocket to listen for task updates in real-time.
3. Requirements:
   - Create a component for adding tasks, including: A form with inputs for title and description.
   - Display a list of tasks retrieved from the backend
     - Display task details and current status.
     - Implement a dropdown to change the task's status.
   - Include a button to edit and delete each task.
   - Establish a WebSocket connection to listen for task updates and refresh the task list accordingly.
   - When a task is created, updated, or deleted, the task list should update in real-time without refreshing the page.