

## Actividad Práctica: Gestión de Lista de Tareas con Prioridades y ViewChild

**Objetivo:** Desarrollar un componente en Angular para gestionar una lista de tareas con diferentes niveles de prioridad, utilizando colores para indicar la importancia. Además, utilizar `ViewChild` y `ElementRef` para interactuar con elementos del DOM.

### Instrucciones:

1. Crea un nuevo componente llamado `TaskListComponent`.
2. Dentro de `TaskListComponent`, implementa una lista visual de tareas.
3. Cada tarea debe tener al menos los siguientes atributos: título, descripción y prioridad.
4. Utiliza colores para indicar la prioridad de cada tarea. Por ejemplo, asigna un fondo verde para las tareas de baja prioridad, naranja para las de prioridad media y rojo para las de alta prioridad.
5. Implementa un formulario en el que los usuarios puedan agregar nuevas tareas con sus respectivas prioridades.
6. Utiliza `ViewChild` y `ElementRef` para acceder y manipular elementos del DOM. Por ejemplo, al agregar una tarea, resalta visualmente el elemento correspondiente en la lista.
7. Ofrece la capacidad de marcar una tarea como completada y actualiza su estilo en consecuencia.
8. Agrega la opción de eliminar tareas de la lista.

### Consejos:

- Utiliza `ngFor` para iterar sobre la lista de tareas y mostrarlas dinámicamente en tu plantilla.
- Utiliza propiedades CSS condicionales para aplicar estilos basados en la prioridad de la tarea.
- Implementa lógica en el componente para gestionar las acciones del usuario (añadir, completar, eliminar).

- Experimenta con ``ViewChild`` y ``ElementRef`` para resaltar visualmente las tareas al agregarlas.

### **Extra:**

- Implementa una funcionalidad que permite cambiar la prioridad de una tarea después de haberla creada.
- Experimenta con animaciones o transiciones CSS para hacer la experiencia del usuario más fluida.