Aquí tienes una versión actualizada de la consigna para tu proyecto de lista de tareas utilizando Java Swing:

**Consigna: Aplicación de Lista de Tareas con Java Swing**

Desarrolla una aplicación de lista de tareas que permita a los usuarios crear, actualizar, eliminar y marcar como completadas las tareas. La aplicación debe incluir una interfaz gráfica utilizando Java Swing para que los usuarios puedan interactuar fácilmente con la lista de tareas.

Requisitos Funcionales:

1. **Interfaz de Usuario Swing:**
   * La interfaz debe mostrar la lista de tareas pendientes.
   * Los usuarios deben poder agregar nuevas tareas utilizando un campo de texto y un botón "Agregar".
   * Los usuarios deben poder marcar una tarea como completada haciendo clic en un botón asociado a cada tarea.
   * Los usuarios deben poder editar y eliminar tareas existentes utilizando botones de acción en la interfaz.
2. **Base de Datos JPA:**
   * Utiliza JPA (Java Persistence API) para gestionar las tareas en una base de datos relacional.
   * Cada tarea debe tener al menos los siguientes atributos: ID, descripción y estado (completada o pendiente).
3. **Operador Ternario:**
   * Utiliza el operador ternario para manejar casos simples de lógica condicional en tu aplicación.

Requisitos Técnicos:

1. Utiliza Java para el desarrollo del backend de la aplicación.
2. Utiliza Java Swing para crear la interfaz gráfica de usuario.
3. Utiliza JPA para mapear las entidades de la base de datos y realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) en las tareas.
4. Utiliza una base de datos relacional compatible con JPA, como MySQL, PostgreSQL o H2 Database.

Extras (Opcionales):

* Implementa la autenticación de usuarios para que solo los usuarios registrados puedan acceder a la lista de tareas.
* Permite que los usuarios filtren las tareas por estado (completadas o pendientes).
* Implementa la capacidad de ordenar las tareas por fecha de vencimiento o por orden alfabético.

Recuerda seguir buenas prácticas de codificación, como dividir tu código en clases y métodos bien estructurados, manejar excepciones de manera adecuada y documentar tu código de manera clara. ¡Buena suerte con tu proyecto!