Actividad 4: Identificación de los Métodos de las Clases

En esta actividad, se identificarán los **métodos** que corresponden a cada una de las clases previamente identificadas en el sistema de **sistematización de expedientes y asuntos** del **Centro Práctico de Asesoría Jurídica**. Los métodos definen las acciones que cada clase puede realizar, permitiendo que el sistema sea interactivo y funcional.

1. Usuario

Descripción: Clase que representa a los usuarios del sistema (estudiantes, asesores jurídicos, administradores).

- **Métodos:**
  - `iniciarSesion()`:
    - **Descripción:** Permite al usuario autenticarse en el sistema con sus credenciales (CIF y contraseña).
    - **Parámetros:** `cif` (String), `contraseña` (String).
    - **Retorno:** `boolean` (Indica si la autenticación fue exitosa o no).
  - `cerrarSesion()`:
    - **Descripción:** Permite cerrar la sesión del usuario en el sistema.
    - **Parámetros:** Ninguno.
    - **Retorno:** `void`.
  - `actualizarPerfil()`:
    - **Descripción:** Permite al usuario actualizar su información personal (nombre, correo, etc.).
    - **Parámetros:** `nombre` (String), `apellido` (String), `correo` (String).
    - **Retorno:** `boolean` (Indica si la actualización fue exitosa o no).

2. Expediente

Descripción: Representa los expedientes legales de los estudiantes gestionados por los asesores jurídicos.

- **Métodos:**
  - `asignarAsesor(asesor: Usuario)`:
    - **Descripción:** Asigna un asesor jurídico a un expediente.
    - **Parámetros:** `asesor` (Usuario) - El asesor jurídico asignado.
    - **Retorno:** `void`.
  - `actualizarEstado(estado: String)`:
    - **Descripción:** Actualiza el estado del expediente (Pendiente, En proceso, Resuelto).
    - **Parámetros:** `estado` (String) - El nuevo estado del expediente.

        - **Retorno:** `void`.
    - `generarReporte()`:
        - **Descripción:** Genera un reporte detallado sobre el expediente, incluyendo el progreso y el estado actual.
        - **Parámetros:** Ninguno.
        - **Retorno:** `String` (El contenido del reporte generado).

3. SolicitudInasistencia

Descripción: Representa las solicitudes de justificación de inasistencia enviadas por los estudiantes.

- **Métodos:**
    - `enviarSolicitud()`:
        - **Descripción:** Envía una solicitud de justificación de inasistencia al sistema.
        - **Parámetros:** Ninguno.
        - **Retorno:** `boolean` (Indica si la solicitud fue enviada correctamente o no).
    - `actualizarEstado(estado: String)`:
        - **Descripción:** Actualiza el estado de la solicitud (Pendiente, Aprobada, Rechazada).
        - **Parámetros:** `estado` (String) - El nuevo estado de la solicitud.
        - **Retorno:** `void`.
    - `verEstado()`:
        - **Descripción:** Permite al estudiante ver el estado actual de su solicitud de inasistencia.
        - **Parámetros:** Ninguno.
        - **Retorno:** `String` (Estado actual de la solicitud: Pendiente, Aprobada, Rechazada).

4. Reporte

Descripción: Representa los reportes generados sobre expedientes, solicitudes de justificación de inasistencia y otros datos relevantes.

- **Métodos:**
    - `generarReporte()`:
        - **Descripción:** Genera un reporte basado en los datos almacenados en el sistema (ej. expedientes, solicitudes).
        - **Parámetros:** `tipo` (String) - Tipo de reporte a generar (Ej. Solicitudes de inasistencia, Expedientes cerrados, etc.).

        - **Retorno:** `String` (El contenido del reporte generado).
    - `exportarPDF()`:
        - **Descripción:** Permite exportar el reporte generado en formato PDF.
        - **Parámetros:** Ninguno.
        - **Retorno:** `String` (Enlace al archivo PDF generado).
    - `mostrarReporte()`:
        - **Descripción:** Muestra el reporte en la interfaz de usuario.
        - **Parámetros:** Ninguno.
        - **Retorno:** `void`.

5. Administrador

Descripción: Representa a los administradores del sistema, quienes gestionan expedientes, solicitudes y reportes.

- **Métodos:**
    - `crearExpediente(estudiante: Usuario, asunto: String)`:
        - **Descripción:** Crea un nuevo expediente para un estudiante.
        - **Parámetros:** `estudiante` (Usuario), `asunto` (String).
        - **Retorno:** `Expediente` (Nuevo expediente creado).
    - `asignarExpediente(expediente: Expediente, asesor: Usuario)`:
        - **Descripción:** Asigna un expediente a un asesor jurídico.
        - **Parámetros:** `expediente` (Expediente), `asesor` (Usuario).
        - **Retorno:** `void`.
    - `generarReporte(tipo: String)`:
        - **Descripción:** Genera un reporte del sistema sobre expedientes, solicitudes, etc.
        - **Parámetros:** `tipo` (String) - El tipo de reporte (Ej. Resumen de casos, Estadísticas de inasistencias).
        - **Retorno:** `String` (El contenido del reporte generado).
    - `verSolicitudesInasistencia()`:
        - **Descripción:** Permite al administrador revisar todas las solicitudes de justificación de inasistencia.
        - **Parámetros:** Ninguno.
        - **Retorno:** `List<SolicitudInasistencia>` (Lista de todas las solicitudes de inasistencia).

6. AsesorJurídico

Descripción: Representa a los asesores jurídicos, responsables de gestionar y resolver los expedientes asignados.

- **Métodos:**
  - `revisarExpediente(expediente: Expediente)`:
    - **Descripción:** Permite al asesor revisar el expediente asignado.
    - **Parámetros:** `expediente` (Expediente).
    - **Retorno:** `void`.
  - `actualizarExpediente(expediente: Expediente, nuevoEstado: String)`:
    - **Descripción:** Permite actualizar el estado del expediente.
    - **Parámetros:** `expediente` (Expediente), `nuevoEstado` (String).
    - **Retorno:** `void`.
  - `resolverExpediente(expediente: Expediente)`:
    - **Descripción:** Permite cerrar un expediente y asignarle un estado final (Resuelto).
    - **Parámetros:** `expediente` (Expediente).
    - **Retorno:** `void`.

7. BaseDeDatos

Descripción: Representa la base de datos donde se almacenan todos los registros del sistema, como usuarios, expedientes y solicitudes.

- **Métodos:**
  - `guardarDatos(objeto: Object)`:
    - **Descripción:** Almacena un objeto (usuario, expediente, solicitud, etc.) en la base de datos.
    - **Parámetros:** `objeto` (Object) - Objeto a almacenar.
    - **Retorno:** `boolean` (Indica si el almacenamiento fue exitoso).
  - `obtenerDatos(clase: Class)`:
    - **Descripción:** Obtiene los datos de una clase específica de la base de datos (Ej. usuarios, expedientes).
    - **Parámetros:** `clase` (Class) - La clase que se desea obtener (Ej. `Usuario.class`, `Expediente.class`).
    - **Retorno:** `List<Object>` (Lista de objetos de la clase especificada).
  - `actualizarDatos(objeto: Object)`:
    - **Descripción:** Actualiza los datos de un objeto ya almacenado en la base de datos.
    - **Parámetros:** `objeto` (Object) - El objeto que se actualizará.
    - **Retorno:** `boolean` (Indica si la actualización fue exitosa).
  - `eliminarDatos(objeto: Object)`:
    - **Descripción:** Elimina un objeto de la base de datos.
    - **Parámetros:** `objeto` (Object) - El objeto que se eliminará.
    - **Retorno:** `boolean` (Indica si la eliminación fue exitosa).

Resumen de Métodos:

- **Usuario:** Métodos relacionados con el inicio y cierre de sesión, así como la actualización del perfil del usuario.
- **Expediente:** Métodos para asignar asesores, actualizar el estado del expediente y generar reportes sobre expedientes.
- **SolicitudInasistencia:** Métodos para enviar, actualizar y ver el estado de las solicitudes de inasistencia.
- **Reporte:** Métodos para generar y exportar reportes sobre el sistema.
- **Administrador:** Métodos relacionados con la creación, asignación de expedientes y la gestión de reportes y solicitudes de inasistencia.
- **AsesorJurídico:** Métodos para revisar, actualizar y resolver expedientes.
- **BaseDeDatos:** Métodos para interactuar con la base de datos y gestionar el almacenamiento, recuperación y eliminación de datos.

Estos métodos definen las acciones que cada clase puede realizar dentro del sistema, permitiendo la gestión eficiente de los expedientes y solicitudes de justificación de inasistencia.