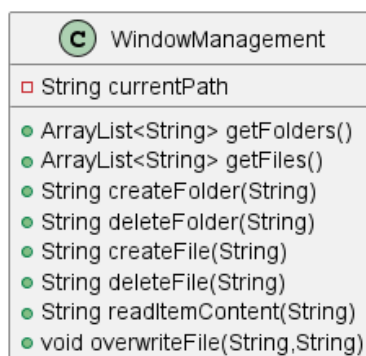
	CICLO FORMATIVO: Desarrollo de Aplicaciones Multiplataforma MÓDULO: Acceso a Datos CURSO: 2º B EXAMEN UT1	
---	--	--

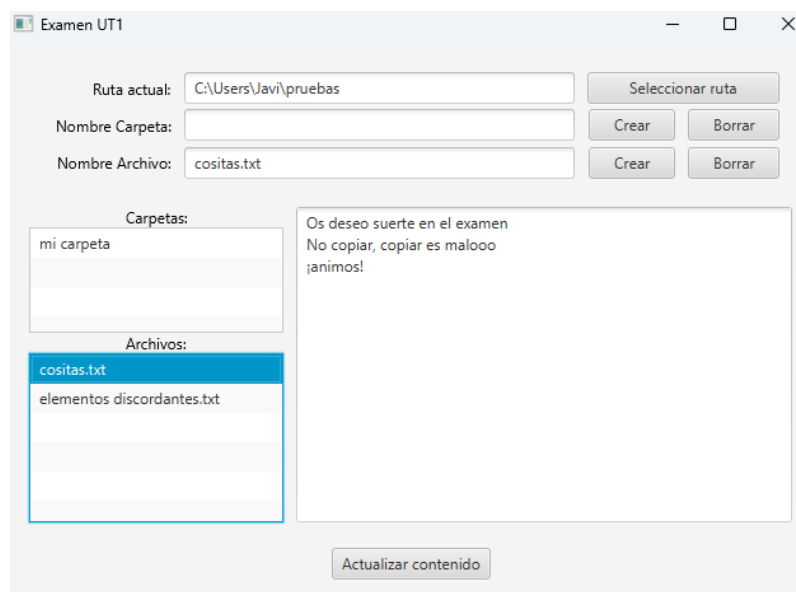
Nombre y apellidos: _____


El examen consiste en **completar los métodos de la clase WindowManagement** del proyecto ExamenUT1 facilitado en el moodle del módulo “2ºDAM-B Acceso a Datos (23-24)”, dentro de la actividad “Examen UT 1”, la cual **se cerrará automáticamente a las 16:55**.

Se deberá respetar el siguiente diagrama de clases sobre la clase WindowManagement, en caso contrario, podría ser causa de **suspenseo directo** (por favor, no incluir variables globales o métodos que no se indiquen):



El proyecto proporcionado se trata de una aplicación base desarrollada sobre JavaFX con la versión OpenJDK 21 de Oracle. Dicho proyecto abrirá una ventana de gestión de archivos que permitirá acceder al contenido de una carpeta concreta con capacidades de crear, editar y eliminar archivos y carpetas dentro de dicha ruta.



	CICLO FORMATIVO: Desarrollo de Aplicaciones Multiplataforma MÓDULO: Acceso a Datos CURSO: 2º B EXAMEN UT1	
---	--	--

Se deberán completar los siguientes métodos:

`public WindowManagement(String currentPath)` - Se deberá inicializar el constructor de la clase `WindowManagement`. Este constructor será invocado siempre que se seleccione el botón “seleccionar ruta” con la ruta indicada. Deberá seleccionarse una ruta de una carpeta concreta del sistema operativo. (0.75 puntos)


`public ArrayList<String> getFolders()` - Deberá devolver los nombres de todas aquellas carpetas del path seleccionado. En caso de error, deberá devolver una lista sin inicializar. Este método será invocado siempre que se seleccione el botón “seleccionar ruta” o “crear/borrar” un archivo o carpeta. (1 puntos)

`public ArrayList<String> getFiles()` - Deberá devolver los nombres de todos aquellos archivos con extensión "txt" del path actual. En caso de error, deberá devolver una lista sin inicializar. Este método será invocado siempre que se seleccione el botón “seleccionar ruta” o “crear/borrar” un archivo o carpeta. (1.25 puntos)

`public String createFolder(String folderName)` - Deberá crear la carpeta solicitada sobre el path actual. En caso de error, deberá devolver un mensaje que indique dicho error. Además, deberá devolver un mensaje de aviso en caso de que dicha carpeta ya exista o no se pueda crear. Este método será invocado siempre que se seleccione el botón “crear” en la fila de “carpetas”. (0.75 puntos)

`public String deleteFolder(String folderName)` - Deberá borrar la carpeta solicitada sobre el path actual (Y SOLO LA CARPETA, ¿Qué pasa si tiene dentro cosas?). En caso de error, deberá devolver un mensaje que indique dicho error. Además, deberá devolver un mensaje de aviso en caso de que dicha carpeta no exista. Este método será invocado siempre que se seleccione el botón “borrar” en la fila de “carpetas”. (1.25 puntos)

`public String createFile(String fileName)` - Deberá crear el archivo solicitado sobre el path actual. ÚNICAMENTE podrá crear archivos con extension "txt". En caso de error, deberá devolver un mensaje que indique dicho error. Además, deberá devolver un mensaje de aviso en caso de que dicho archivo ya exista o no se pueda crear. Este método será invocado siempre que se seleccione el botón “crear” en la fila de “archivos”. (1 puntos)

	CICLO FORMATIVO: Desarrollo de Aplicaciones Multiplataforma MÓDULO: Acceso a Datos CURSO: 2º B EXAMEN UT1	
---	--	--


`public String deleteFile(String fileName)` - Deberá borrar el archivo solicitado sobre el path actual. En caso de error, deberá devolver un mensaje que indique dicho error. Además, deberá devolver un mensaje de aviso en caso de que dicho archivo no exista. Este método será invocado siempre que se seleccione el botón “borrar” en la fila de “archivos”. (1.25 puntos)

`public String readItemContent(String selectedItem)` - Deberá devolver el contenido del elemento seleccionado, únicamente si es un archivo (es decir, que si es una carpeta, no deberá devolver nada). Se deberá emplear la clase `BufferedReader` para ello, otro sistema de lectura no será válido. Este método será invocado siempre que se seleccione un archivo en el listado de archivos. (1.75 puntos)

`public void overwriteFile(String fileName, String fileContent)` - Deberá sobrescribir el contenido del archivo indicado. Se deberá emplear la clase `PrintWriter` para ello, otro sistema de escritura no será válido. Este método será invocado siempre que se seleccione el botón “Actualizar contenido” sobre el archivo seleccionado en el listado de archivos. (1 puntos)

Adicionalmente, se podrán recibir **puntuaciones negativas o penalizaciones** en los siguientes casos (además de los previamente mencionados):

- **Error en la compilación** del proyecto: 5 puntos.
- **Error de ejecución** en la aplicación: 2.5 puntos.
- **Errores de sintaxis**: 0.25 puntos cada una, hasta 2.5 puntos.

	CICLO FORMATIVO: Desarrollo de Aplicaciones Multiplataforma MÓDULO: Acceso a Datos CURSO: 2º B EXAMEN UT1	
---	--	--

Desarrollo de métodos	
WindowManagement (0.75 puntos)	
getFolders (1 puntos)	
getFiles (1.25 puntos)	
createFolder (0.75 puntos)	
deleteFolder (1.25 puntos)	
createFile (1 puntos)	
deleteFile (1.25 puntos)	
readItemContent (1.75 puntos)	
overwriteFile (1 puntos)	
Penalizaciones	
Error de compilación (-5 puntos)	
Error de ejecución (-2.5 puntos)	
Error de sintaxis (-2.5 puntos)	
Puntuación total	
TOTAL	