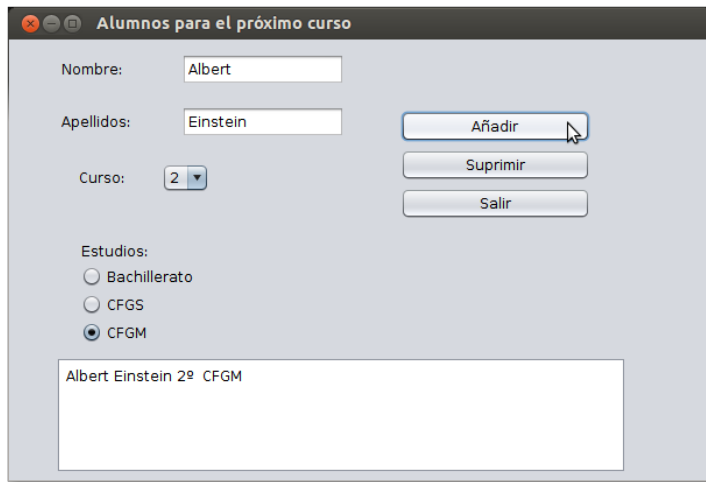


Realizar un programa Java que gestione un listado de alumnos de un centro: **40 puntos**



El programa debe tener la siguiente clase con sus correspondientes atributos y métodos:

Clase ListaAlumnos (Total 7 puntos)

- Tiene como atributo una **estructura** capaz de contener objetos de la clase String **(1 punto)**.
- **Constructor**: Sin argumentos, crea la estructura vacía. **(1 puntos)**.
- **Insertar(String)**: Inserta en la estructura una cadena de texto **(2 puntos)**.
- **Borrar(String)**: Elimina de la estructura el elemento de la estructura que se le pasa como argumento **(2 puntos)**.
- **toString()**: Devuelve una cadena que contendrá todos los textos de la lista, separados por un salto de línea. **(1 punto)**.

Ej:

Albert Einstein 2º CFGM

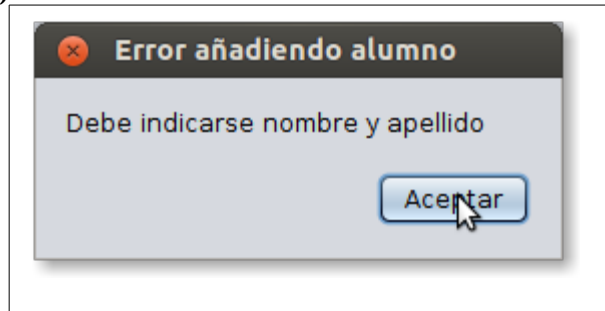
Pascal Blaise 1º CFGS

Consideraciones del interfaz (Total 13 puntos)

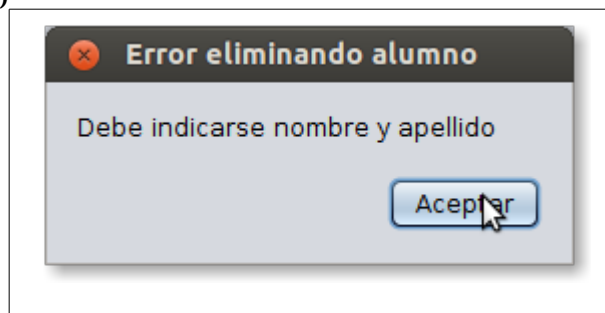
- Todos los elementos del interfaz deben tener un nombre descriptivo **(2 puntos)**.
- Cuando el programa comienza se debe crear, mediante su correspondiente constructor una estructura ListaAlumnos vacía **(1 punto)**.
- El marco principal tiene un título: Alumnos para el próximo curso **(1 punto)**.
- Diseño del conjunto de Estudios: **(2 puntos)**.
Los botones de radio deben tener un comportamiento conjunto.
- El cuadro inferior es un Área de Texto, que no puede editarse **(2 puntos)**.
- En la carga inicial del formulario los controles deben contener: **(5 puntos)**.
Nombre → Debe estar vacío
Apellido → Debe estar vacío
Curso → 1
Estudios → Bachillerato
Listado de alumnos → Vacío

Comportamiento (Total 15 puntos)

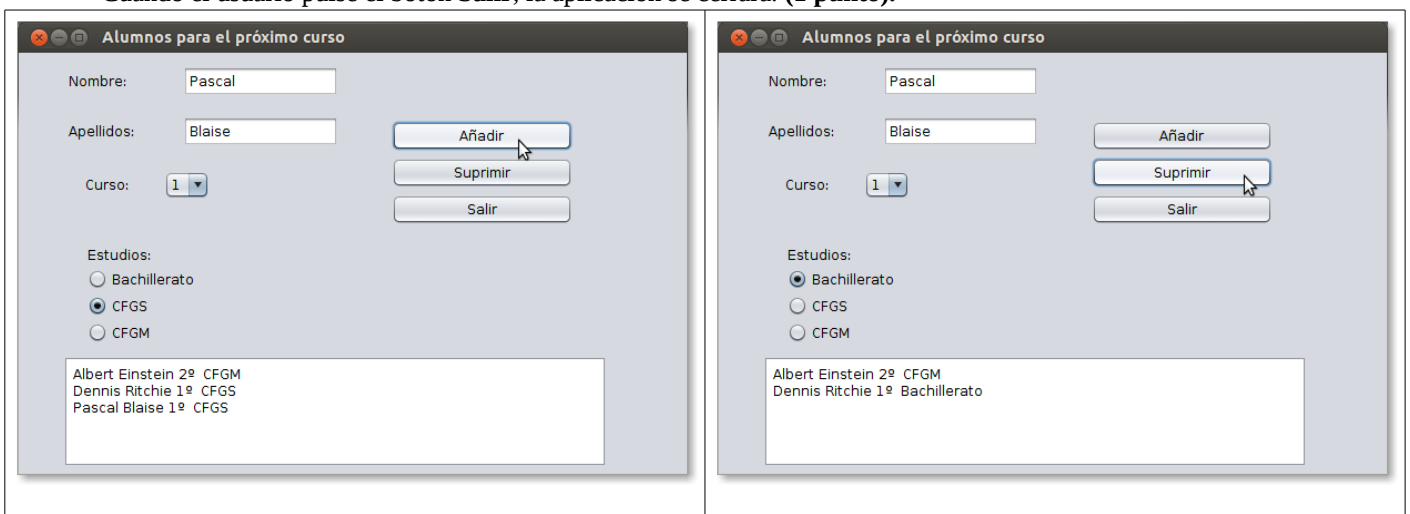
- Cuando el usuario pulsa el botón **Añadir** se realizarán las siguientes acciones:
 - Comprobar que los cuadros de texto Nombre y Apellidos contienen algo y si es así: **(1 punto)**
 - Concatenar el Nombre, un espacio en blanco, el Apellido, un espacio en blanco, el curso que este seleccionado, el carácter °, un espacio en blanco y el texto del grupo Estudios que esté seleccionado. **(2 puntos)**
Por ejemplo para la imagen de arriba, se generará el texto: “Albert Einstein 2º CFGS”
 - Introducir el String resultante en la estructura ListaAlumnos, usando el método Insertar de la misma **(1 punto)**
 - Mostrar en el TextArea la lista completa almacenada en la estructura ListaAlumnos, usando el método toString de la misma. **(1 punto)**
 - Si alguno de los cuadros de texto Nombre o Apellidos no contienen nada debe mostrarse un cuadro de diálogo como el siguiente: **(2 puntos)**



- Cuando el usuario pulsa el botón **Suprimir** se realizarán las siguientes acciones:
 - Comprobar que los cuadros de texto Nombre y Apellidos contienen algo y si es así: **(1 punto)**
 - Concatenar el Nombre, un espacio en blanco y el Apellido. **(2 puntos)**
Por ejemplo para la imagen de arriba, se generará el texto: “Albert Einstein”
 - Llamar a la función Borrar de la estructura ListaAlumnos con este texto generado anteriormente. **(1 punto)**
 - Mostrar en el TextArea la lista completa almacenada en la estructura ListaAlumnos, usando el método toString de la misma. **(1 punto)**
 - Si alguno de los cuadros de texto Nombre o Apellidos no contienen nada debe mostrarse un cuadro de diálogo como el siguiente: **(2 puntos)**



- Cuando el usuario pulse el botón **Salir**, la aplicación se cerrará. **(1 punto).**



Claridad del código y eficiencia: (5 puntos).

Sólo se aplicará si el programa funciona PERFECTAMENTE