

N5 Sudoku

Consideraciones adicionales de diseño

Consejos prácticos para la construcción de la interfaz gráfica de la aplicación

1. El tamaño sugerido de la ventana principal es de 1000 píxeles de ancho y 550 píxeles de alto.
2. Para incluir una imagen en una etiqueta, se utiliza un ImageIcon de la siguiente manera:

```
JLabel lblImagen = new JLabel ( );  
ImageIcon icono = new ImageIcon( rutaImagen );  
lblImagen.setIcon( icono );
```

Donde rutaImagen es la ruta donde se encuentra ubicada la imagen. Por ejemplo, la imagen del banner se encuentra en `./data/imagenes/banner.png`.

3. Las imágenes de los iconos de navegación se encuentran en la siguiente ruta:
`./data/imagenes/`
 - abajo.png: Imagen del botón para navegar hacia abajo.
 - arriba.png: Imagen del botón para navegar hacia arriba.
 - derAb.png: Imagen del botón para navegar hacia abajo a la derecha.
 - derArr.png: Imagen del botón para navegar hacia arriba a la derecha.
 - derecha.png: Imagen del botón para navegar hacia la derecha.
 - izqAb.png: Imagen del botón para navegar hacia abajo a la izquierda.
 - izqArr.png: Imagen del botón para navegar hacia arriba a la izquierda.
 - izquierda.png: Imagen del botón para navegar hacia la izquierda.
4. Para centrar una imagen en una etiqueta, utilice `setHorizontalAlignment` de la siguiente manera:

```
lblImagen.setHorizontalAlignment( JLabel.CENTER );
```

5. Para definir las dimensiones que tiene una imagen usted puede utilizar la siguiente instrucción:

```
ImageIcon icono = new ImageIcon( new ImageIcon(  
ruta ).getImage().getScaledInstance( 200 , 200 , Image.SCALE_DEFAULT ) );
```

En esta instrucción se está definiendo el alto y el ancho de la imagen como 200 y 200.

6. Para que haya un espacio entre los elementos en un GridLayout, puede agregarle 2 parámetros adicionales al constructor del distribuidor gráfico, para indicar los espacios, en píxeles, entre cada una de sus zonas.

```
setLayout( new GridLayout( 2, 1, 2, 2 ) );
```

En este caso, se tiene una malla de 2 filas y una columna, y se deja un espacio de 2 píxeles entre las filas y un espacio de 2 filas entre las columnas.

Recomendaciones

- Antes de empezar a desarrollar la interfaz, es muy importante que entienda el ejercicio y el modelo del mundo. Es importante que conozca cuáles son los métodos existentes, para que después los pueda llamar desde la interfaz.
- En este nivel, debido a que las opciones de juego todavía no están habilitadas, todos los valores mostrados se están calculando de manera aleatoria. De esta manera, cada vez que cargue el juego, encontrará una información de estado de juego diferente. De igual manera, cada vez que presione alguno de los botones se calculará un valor diferente y aleatorio para algunos datos.
- Recuerde utilizar los mensajes de las excepciones para mostrar la información respecto al error ocurrido. No vuelva a escribir el mensaje.
- Tenga en cuenta que el ejercicio debe funcionar sin problemas si se cambia el archivo desde donde se cargan los sudokus.
- Recuerde que debe inicializar todos los elementos de la interfaz antes de utilizarlos.
- Se recomienda tener en la clase principal de la interfaz un método que actualice la información del juego, que en un futuro (cuando haga el ejercicio completo del nivel 6) será llamado cada vez que se realice un movimiento.
- Tenga en cuenta que este es el mismo ejercicio que utilizará como base para la entrega del ejercicio del siguiente nivel. Sea muy organizado con su código, nombre bien sus métodos y atributos, de manera que la modificación del mismo sea más fácil.
- En este nivel, en la clase principal del mundo encontrará un método mover sin parámetros que debe utilizar para todos los casos de navegación. En el siguiente nivel deberá implementar esta funcionalidad, pero está en libertad de implementarlo en este nivel si lo desea. Recuerde que para la implementación de este método debe recibir la dirección en la que se desea mover o puede crear varios métodos, uno para cada dirección en la que se puede navegar.}
- En este nivel, cuando se invoquen los métodos de cargar, reiniciar, validar y solucionar estos retornaran una cadena de caracteres con la ruta de una imagen que se deberá reemplazar por el tablero.