

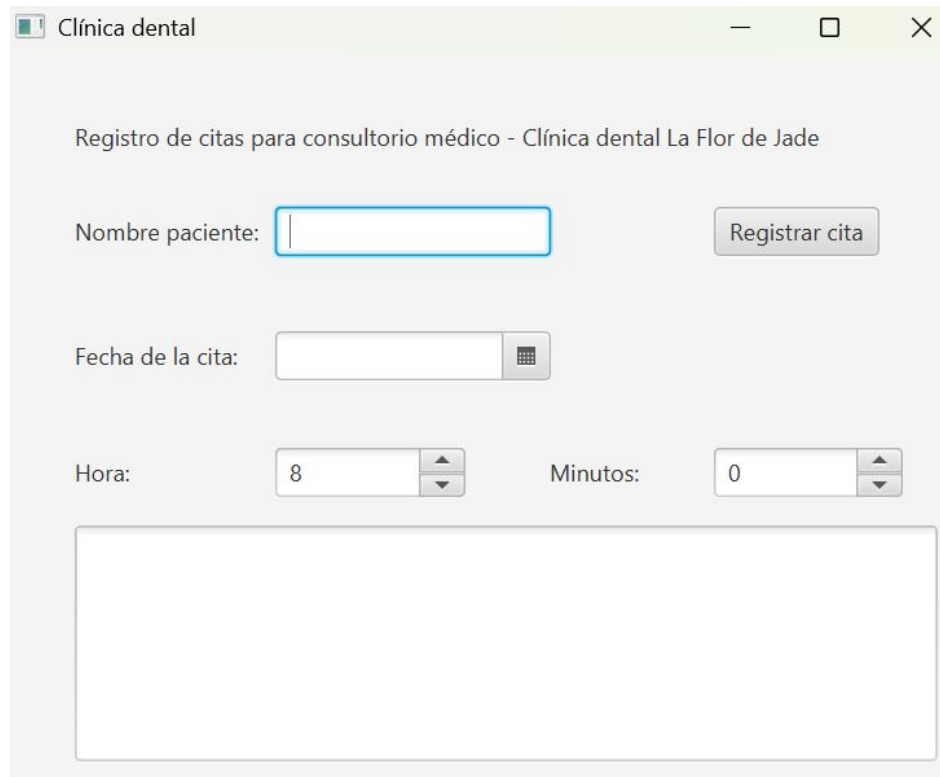
UD 1.- Ejercicios de repaso

DI DAM2

Ejercicio 1

Utilizar un GridPane para la interfaz

¿Cuántas filas y
cuántas columnas se
van a necesitar?








Clínica dental

Registro de citas para consultorio médico - Clínica dental La Flor de Jade

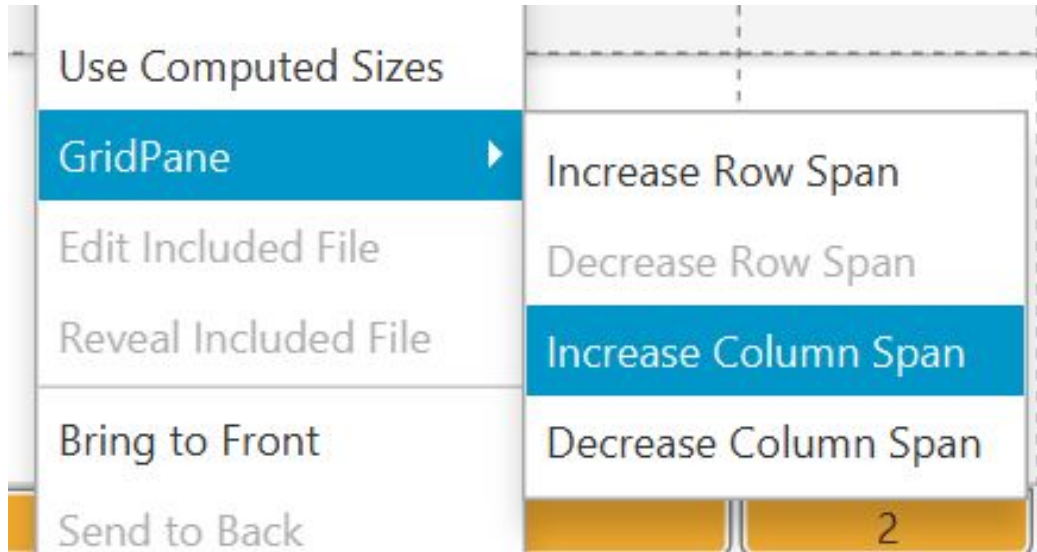
Nombre paciente:

Fecha de la cita:

Hora: Minutos:

| 0 | 1 | 2 | 3 | | |
|---|--|---|----------------|---|---|
| 0 | Registro de citas para consultorio médico - Clínica dental La Flor de Jade | | | 0 | |
| 1 | Nombre paciente: | <input type="text"/> | Registrar cita | 1 | |
| 2 | Fecha de la cita: | <input type="text"/>  | | 2 | |
| 3 | Hora: | <input type="text"/>   | Minutos: | <input type="text"/>   | 3 |
| 4 | | | | 4 | |
| 0 | 1 | 2 | 3 | | |

Utilizar *Increase Column Span* para que un control ocupe varias columnas





Registro de citas para consultorio médico - Clínica dental La Flor de Jade

Nombre paciente:

Fecha de la cita:



Hora:



Minutos:



Nombre de paciente: Ana García

Fecha de la cita: 20/09/2025

Hora de la cita es: 08:05

Añadir patrón de número

```
DecimalFormat formatonumero = new DecimalFormat("00");  
formatonumero.format(hora.getValue());
```

Ejercicio 2

Dar color a una etiqueta

Utilizando el sistema de color RGB, colorear una etiqueta en función de los valores establecidos en los ComboBox

Color.rgb(int r, int g, int b)

Rojo:

7



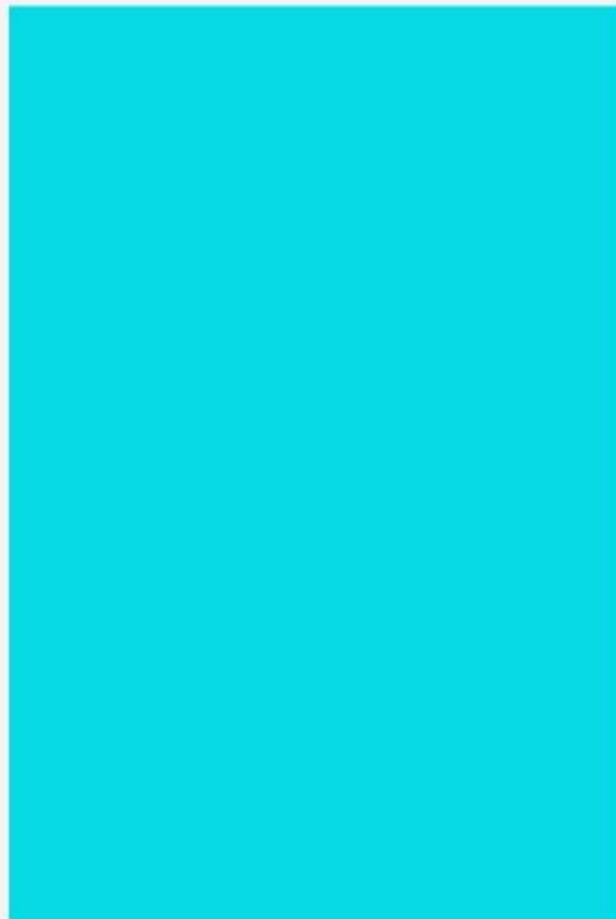
Verde:

218



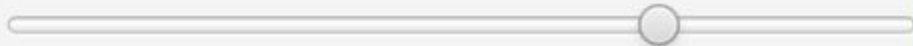
Azul:

228

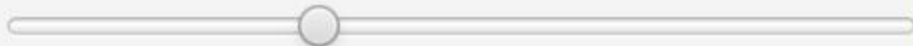


Ejercicio 3

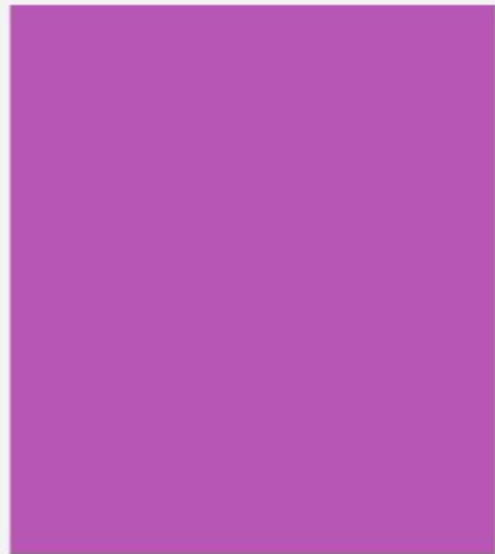
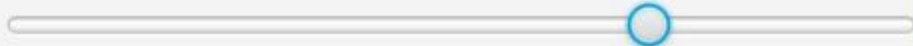
Rojo



Verde



Azul



Ejercicio 4

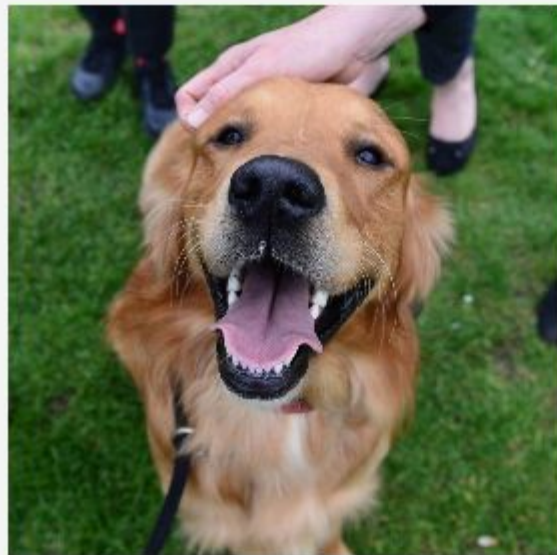
Altura y anchura de la imagen

```
img.getFitHeight();
```

```
img.getFitWidth();
```

```
img.setFitHeight();
```

```
img.setFitWidth();
```



Ejercicio 5



Pestañas



Datos conductor



Coche



Monovolumen

Nombre

Apellidos


☐


Coche

☐


Monovolumen

Al iniciar la aplicación, las pestañas Coche y Monovolumen estarán deshabilitadas.


 Pestañas



Datos conductor



Coche



Monovolumen

Nombre

Apellidos

☐ Coche

☐ Monovolumen

Ejercicio 6

Añadir tabla al ejercicio

Nombre:

Álvaro

Apellidos:

Pérez

Localidad:

Caudé

Salario:

1.700,00 €



Nuevo registro

Guardar

Añadir a la tabla

[illegible]

Crear la clase Empleado

- Declarar las variables
- Crear el constructor con todos los campos
- Crear los Getter y los Setter

Declaración de la tabla y las columnas

nombre de la clase

@FXML

```
private TableView<Empleado> tabla;
```

@FXML

```
private TableColumn<Empleado,String> Cnombre;
```

repetir para cada
columna

Añadir registros a la tabla

Nombre de la variable
en la Clase empleado

```
Cnombre.setCellValueFactory(new PropertyValueFactory<>("nombre"));
```

Repetir por cada columna
de la tabla

```
Empleado p = new Empleado(Tnombre.getText(), ...);
```

```
tabla.getItems().add(p);
```

Añadir los textos de los
TextField de la interfaz

Ejercicio 7

Interfaz con tabla

- Crear una tabla en la interfaz
- Al mostrar la interfaz que se carguen todos los registros en la tabla

| Nombre | Apellidos | Localidad | Salario |
|-----------|-----------|-----------|---------|
| Marta | Soler | Teruel | 1300 |
| Luis | Martínez | Caudé | 1500 |
| Esther | García | Cella | 1200 |
| Ismael | Pérez | Sarrión | 1000 |
| Ana | García | Teruel | 1200 |
| Carlos | Marzo | Cella | 1000 |
| Álvaro | Pérez | Caudé | 1700 |
| Francisco | Gutierrez | Sarrión | 1500 |
| Arturo | Sebastián | Teruel | 1600 |
| Pedro | García | Cella | 1200 |
| Alba | Estrada | Sarrión | 1500 |
| | | | |
| | | | |

Búsquedas en una tabla

- Añadir un *TextField*, donde el usuario escriba la localidad a buscar
- Al pulsar el botón *Mostrar*, se mostrarán sólo los registros de esa localidad

Cargar valores de la tabla en el ComboBox

Modificar el ejercicio anterior, sustituyendo el *TextField* por un *ComboBox* donde aparezcan un listado de localidades extraídas de la tabla y sin repetir valores