

El objetivo de este ejercicio es crear nuestra primera aplicación de JavaFX y familiarizarnos con el uso de SceneBuilder

ENUNCIADO

Estructura de la aplicación

Antes que nada, vamos a crear tres paquetes separados para seguir un Modelo-Vista-Controlador (MVC).

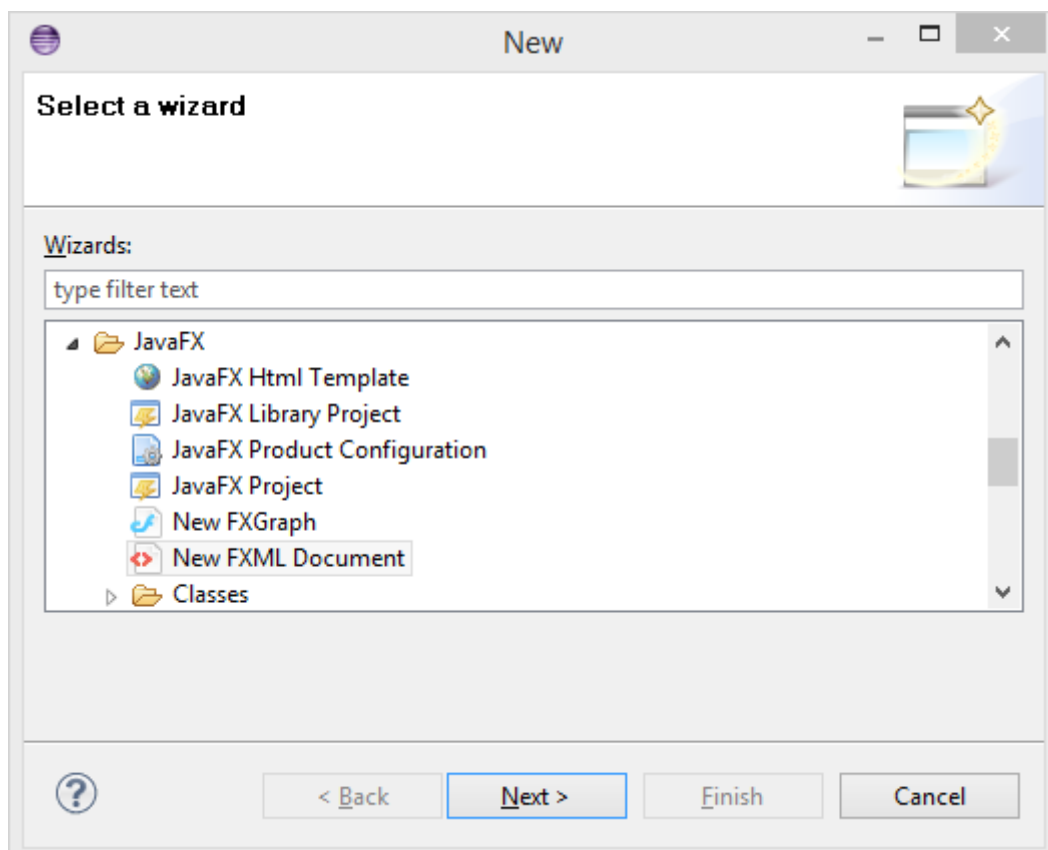
En el ratón hacemos clic derecho en la carpeta src, New | Package (podemos borrar el paquete por defecto application):

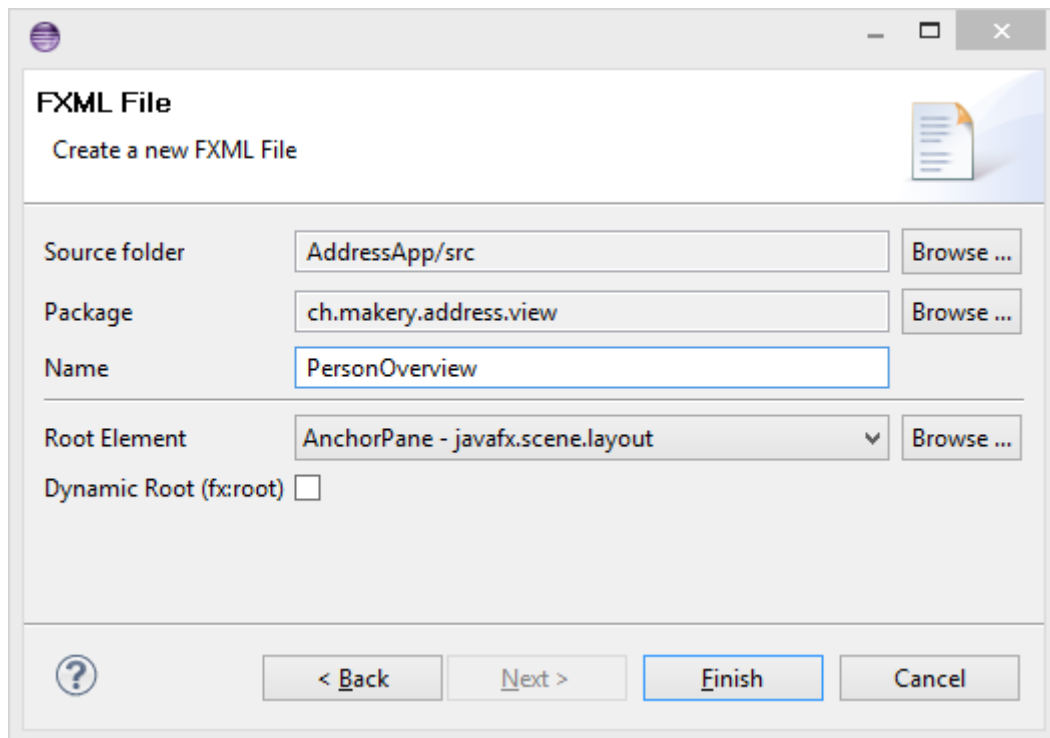
- **ch.makery.address:** Contendrá la mayoría de clases de control (C)
- **ch.makery.address.model:** Contendrá las clases del modelo (M)
- **ch.makery.address.view:** Contendrá las vistas (V)

Creando el archivo FXML de diseño

Hay dos formas de crear la interfaz de usuario. Programándolo en Java o mediante un archivo XML. En este caso usaremos XML (archivo con la extensión .fxml). Además, podemos usar la herramienta de edición visual Scene Builder, la cual nos evita tener que trabajar directamente con el XML.

Haz clic derecho el paquete *view* y crea un nuevo archivo FXML (*New | Other | FXML | New FXML Document*) llamado **PersonOverview**.

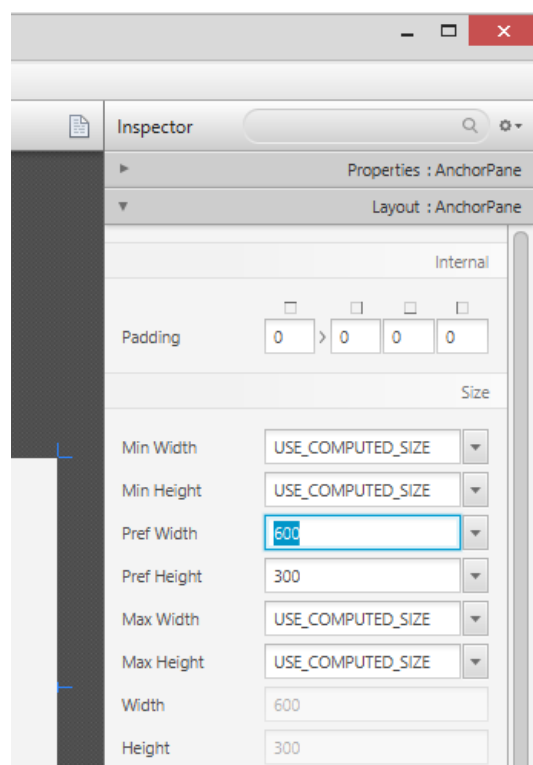




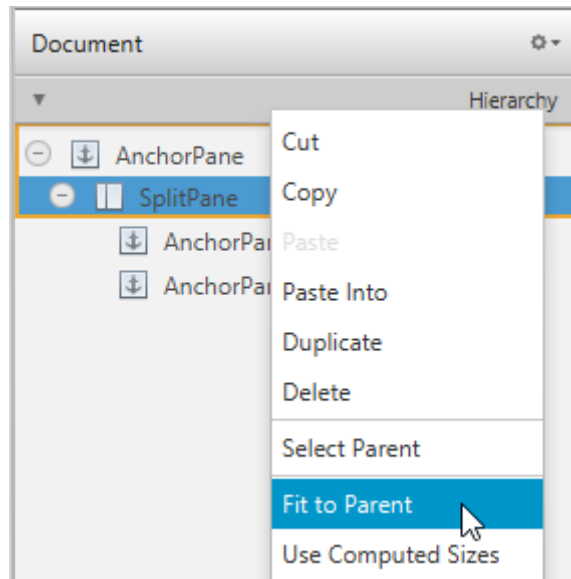
Diseño del FXML con Scene Builder

Haz clic derecho sobre **PersonOverview.fxml** y elige *Open with Scene Builder*. Ahora deberías ver el Scene Builder con un *AnchorPane* (visible en la jerarquía de componentes (herramienta Hierarchy) situada a la izquierda).

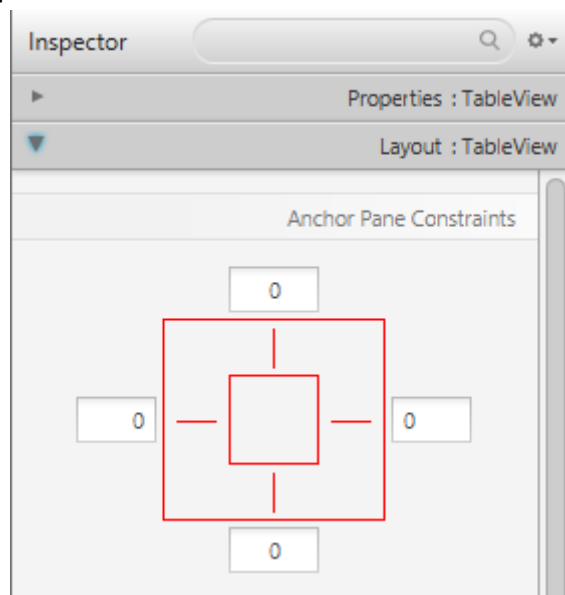
1. Selecciona el *AnchorPane* en tu jerarquía y ajusta el tamaño en el apartado Layout (a la derecha):



2. Añade un *SplitPane* (*Horizontal Flow*) arrastrándolo desde la librería (Library) al área principal de edición. Haz clic derecho sobre el *SplitPane* en la jerarquía y elige *Fit to Parent*.

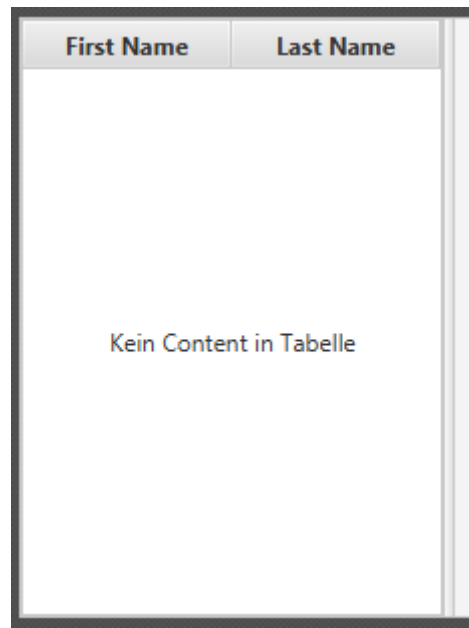


3. Arrastra un *TableView* (bajo *Controls*) al lado izquierdo del *SplitPane*. Selecciona la *TableView* (no una sola columna) y establece las siguientes restricciones de apariencia (Layout) para la *TableView*. Dentro de un *AnchorPane* siempre se pueden establecer anclajes (anchors) para los cuatro bordes.

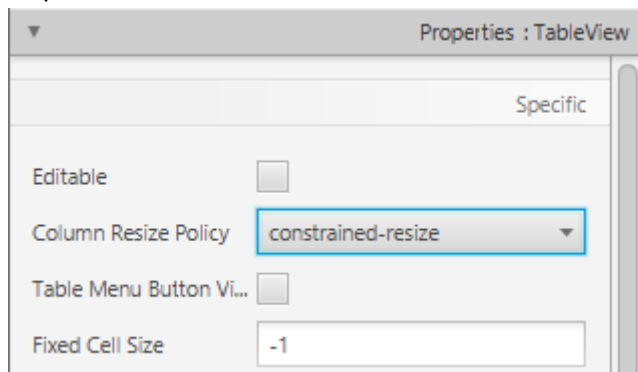


4. Ve al menú *Preview / Show Preview in Window* para comprobar si se visualiza correctamente. Intenta cambiar el tamaño de la ventana. La *TableView* debería ajustar su tamaño al tamaño de la ventana, pues está “anclada” a sus bordes.

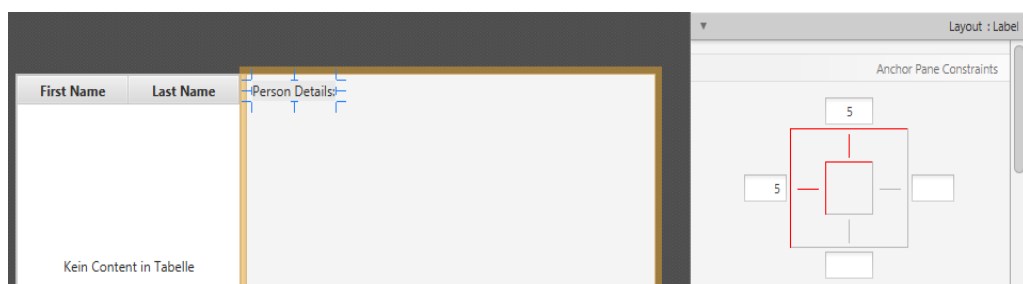
5. Cambia el texto de las columnas (bajo Properties) a “First Name” y “Last Name”.



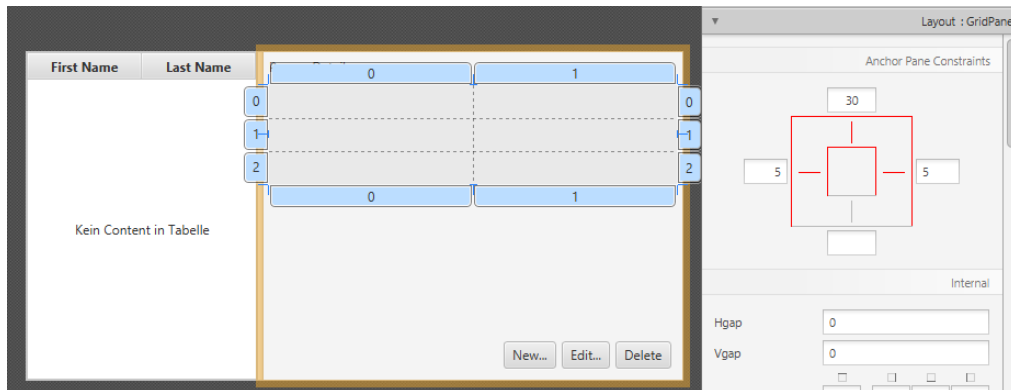
6. Selecciona la *TableView* y elige *constrained-resize* para la *Column Resize Policy* (en Properties). Esto asegura que las columnas usarán siempre todo el espacio que tengan disponible.



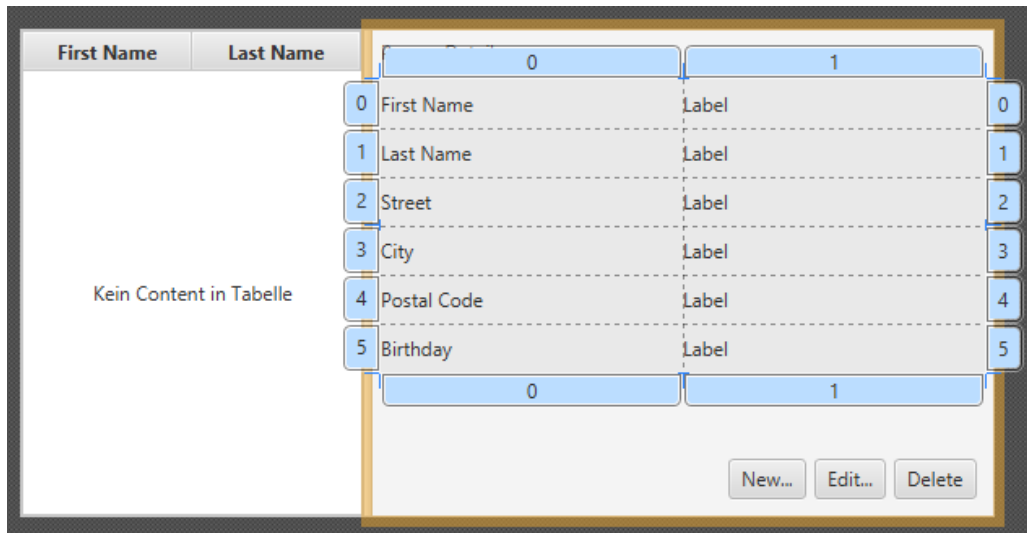
7. Añade una *Label* al lado derecho del *SplitPane* con el texto “Person Details” (truco: usa la búsqueda en la librería para encontrar el componente *Label*). Ajusta su apariencia usando anclajes.



8. Añade un `GridPane`* al lado derecho, selecciónalo y ajusta su apariencia usando anclajes (superior, derecho e izquierdo).

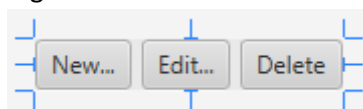


9. Añade las siguientes etiquetas (`Label`) a las celdas del `GridPane`.
Nota: Para añadir una fila al `GridPane` selecciona un número de fila existente (se volverá de color amarillo), haz clic derecho sobre el número de fila y elige "Add Row".

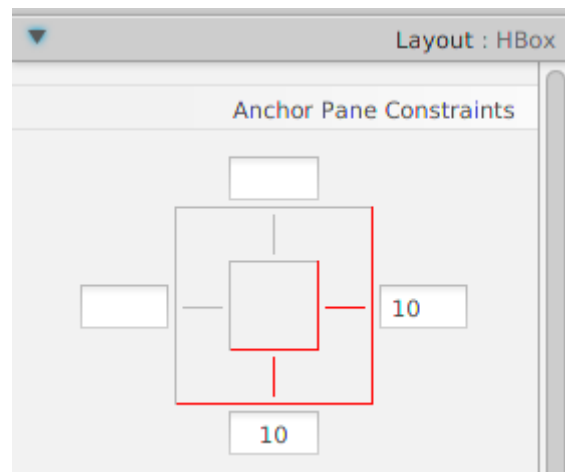


10. Añade 3 botones a la parte inferior. Truco: Selecciónalos todos, haz clic derecho e invoca `Wrap In / HBox`. Esto los pondrá a los 3 juntos en un `HBox`. Podrías necesitar establecer un espaciado `spacing` dentro del `HBox` (en el menú de propiedades situado a la derecha).

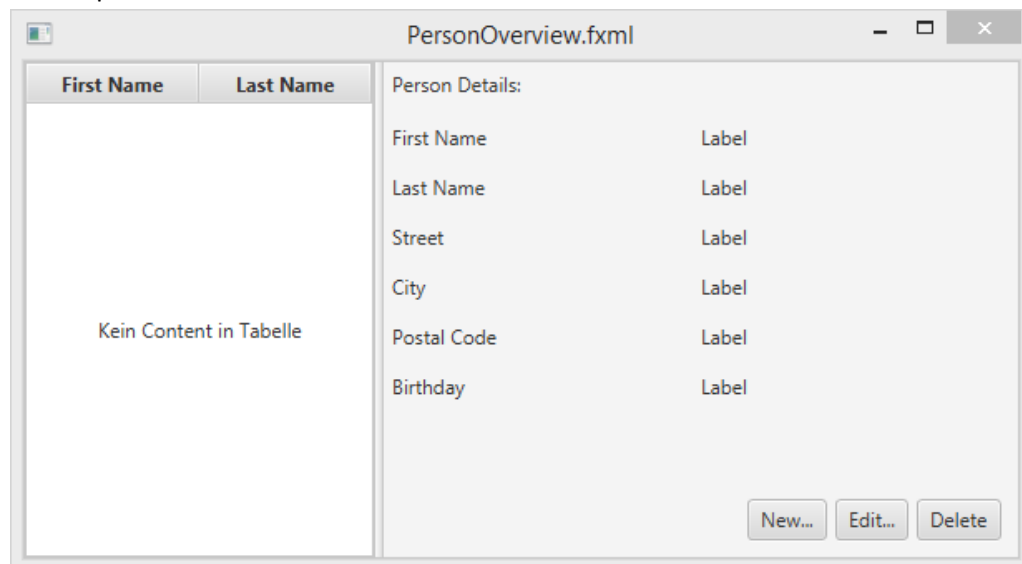
Después, establece también anclajes (derecho e inferior) para que se mantengan en el lugar correcto.



Es decir, en la pestaña `Layout` debes indicar los siguientes valores (y el espaciado mencionado en el párrafo anterior).



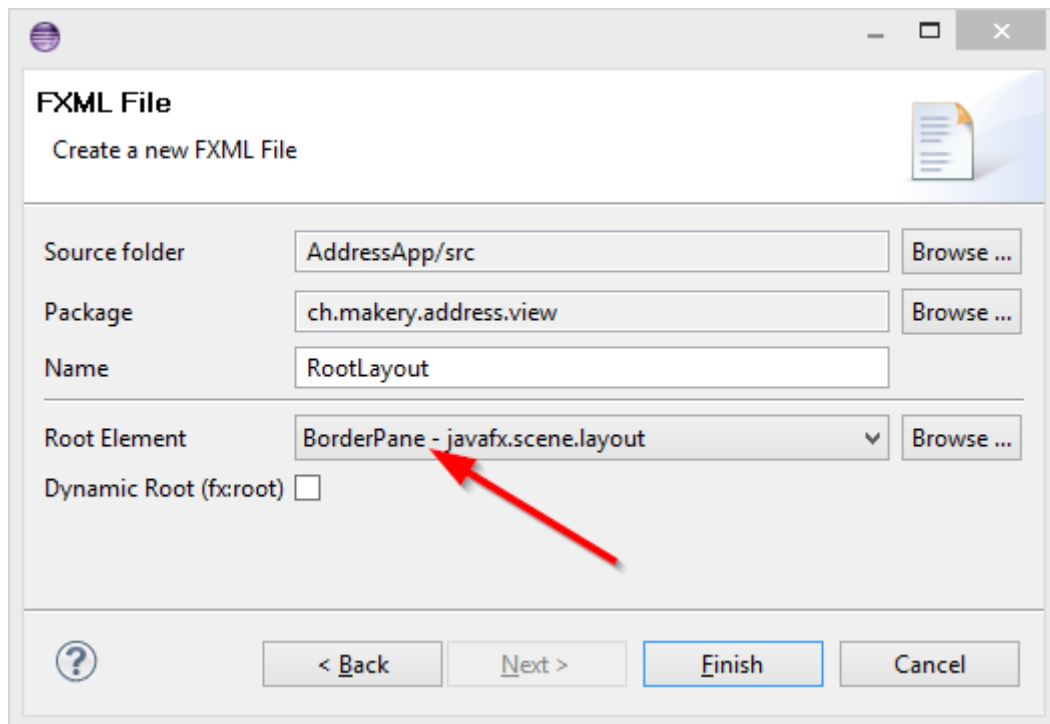
11. Ahora deberías ver algo parecido a lo siguiente. Usa el menú *Preview* para comprobar su comportamiento al cambiar el tamaño de la ventana.



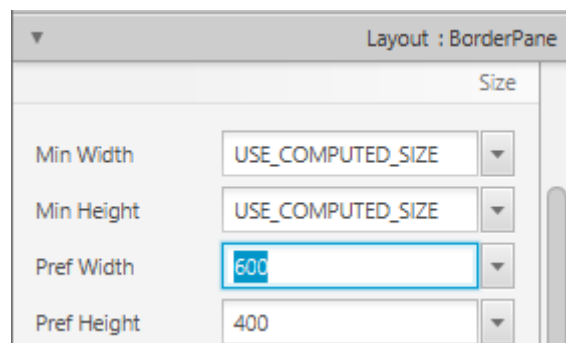
Creando la aplicación principal

Necesitamos otro archivo FXML para nuestra vista raíz, la cual contendrá una barra de menús y encapsulará la vista recién creada **PersonOverview.fxml**.

1. Crea otro archivo FXML dentro del paquete view llamado **RootLayout.fxml**. Esta vez, elige *BorderPane* como elemento raíz.



2. Abre `RootLayout.fxml` en el Scene Builder.
3. Cambia el tamaño del `BorderPane` con la propiedad `Pref Width` establecida en 600 y `Pref Height` en 400.



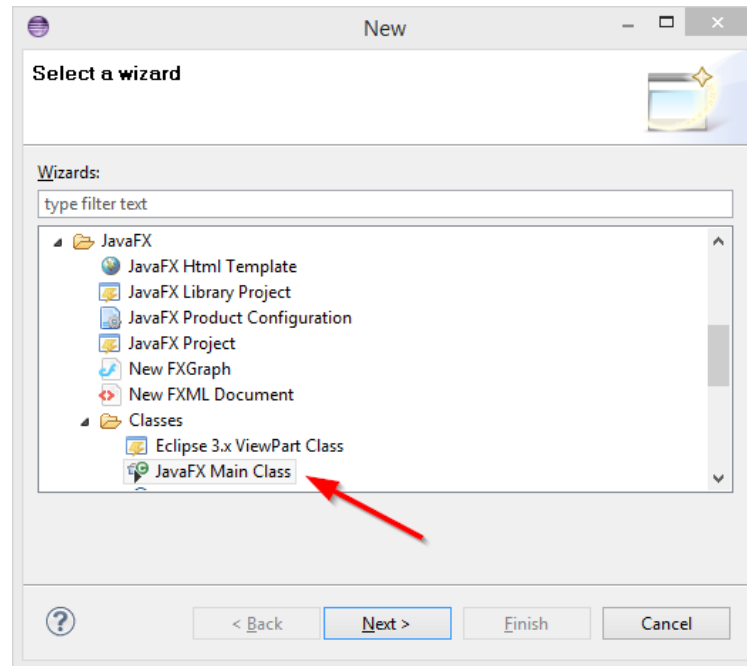
4. Añade una `MenuBar` en la ranura superior del `BorderPane`. De momento no vamos a implementar la funcionalidad del menú.



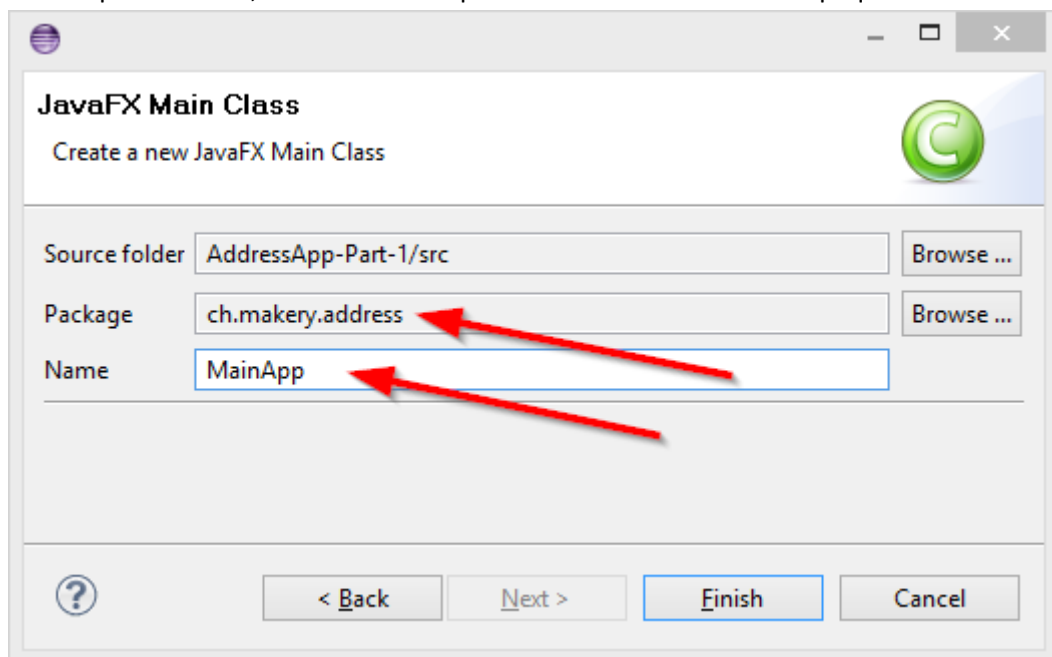
La clase Main en JavaFX

Ahora necesitamos crear la **clase java principal**, la cual iniciará nuestra aplicación mediante `RootLayout.fxml` y añadirá la vista `PersonOverview.fxml` en el centro.

1. Haz clic derecho en el proyecto y elige *New / Other / JavaFX / classes / JavaFX Main Class*. Es posible que esta clase ya esté en el proyecto y te puedas ahorrar este paso.

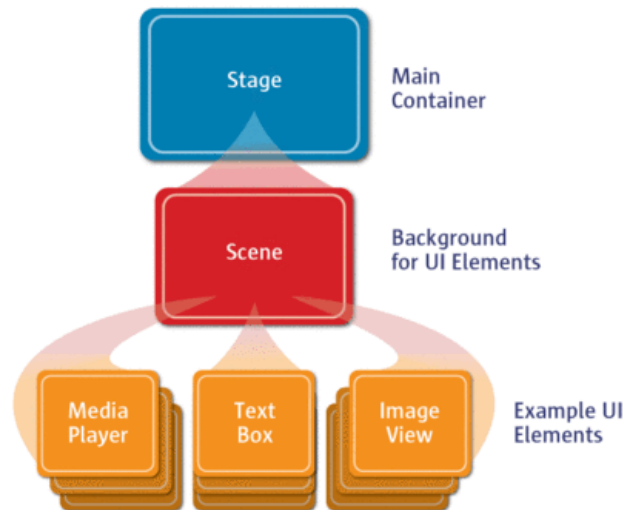


2. Llama a esta clase **Main** y ponla en el paquete de controladores **ch.makery.address** (nota: este es el paquete padre de los paquetes **view** y **model**). Si la clase Main existe previamente, bastará con que muevas el archivo al paquete indicado.



La clase generada (**Main.java**) extiende a la clase **Application** y contiene el método **start(Stage primaryStage)**. Este método es invocado automáticamente cuando la aplicación es lanzada desde el método **main**.

Recuerda el proceso para iniciar una aplicación con JavaFX.



Aún no hemos estudiado en detalle cómo crear controles, por lo que de momento copia el código de la página siguiente, que tiene como objetivo cargar los diseños realizados anteriormente con FXML y empleando el objeto Scene.

```
public class Main extends Application {

    private Stage primaryStage;
    private BorderPane rootLayout;

    @Override
    public void start(Stage primaryStage) {
        this.primaryStage = primaryStage;
        this.primaryStage.setTitle("AddressApp");

        initRootLayout();

        showPersonOverview();
    }

    /** Inicializa el diseño de la pantalla principal. */
    public void initRootLayout() {
        try {
            // Carga el XML con el diseño principal
            FXMLLoader loader = new FXMLLoader();

            loader.setLocation(Main.class.getResource("view/
                RootLayout.fxml"));
            rootLayout = (BorderPane) loader.load();

            // Se añade el diseño principal a la escena
            Scene scene = new Scene(rootLayout);
            primaryStage.setScene(scene);
            primaryStage.show();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    /** Muestra el diseño de PersonOverview dentro de la pantalla principal */
    public void showPersonOverview() {
        try {
            // Cargamos el archivo PersonOverview
            FXMLLoader loader = new FXMLLoader();
            loader.setLocation(Main.class.getResource("view/
                PersonOverview.fxml"));
            AnchorPane personOverview = (AnchorPane) loader.load();

            // Se sitúa en el centro del diseño principal
            rootLayout.setCenter(personOverview);
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    /** Returns the main stage. */
    public Stage getPrimaryStage() {
        return primaryStage;
    }

    public static void main(String[] args) {
        launch(args);
    }
}
```