

TALLER No. 9

PROGRAMACIÓN ORIENTADA A OBJETOS

Profesor: Jaime Alberto Guzmán Luna

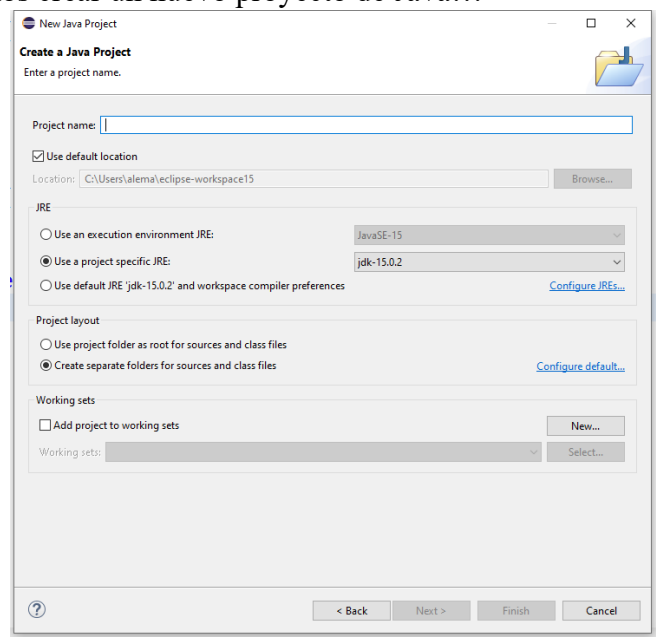
Contenido del taller:

1. Javafx
 2. Layouts
 3. Algunos componentes y eventos
-

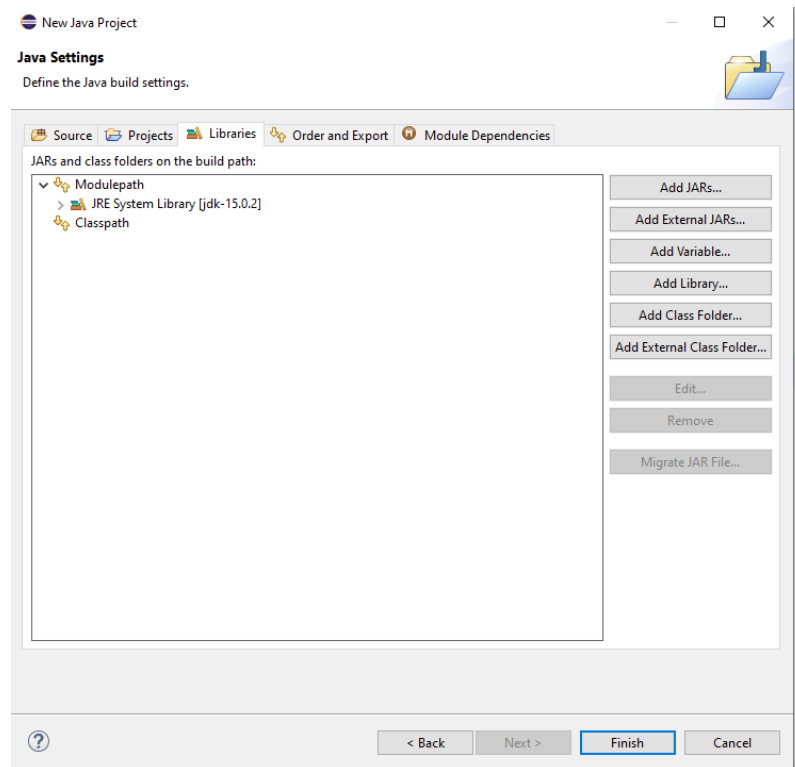
Ejercicio paso a paso

Vamos a realizar un pequeño ejercicio creando una ventana y añadiendo elementos.

Lo primero debemos crear un nuevo proyecto de Java...



Le damos un nombre y le damos siguiente...



Añadimos la librería JavaFX a nuestro proyecto, que previamente configuramos usando el tutorial dejado en el classroom. Y Le damos en finalizar.

Ya teniendo creado nuestro proyecto con la librería empezaremos a codificar nuestra interfaz gráfica llamada Basica.

```
package ventanas;

import javafx.application.Application;
import javafx.stage.Stage;

public class Basica extends Application {

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

    @Override
    public void start(Stage window) throws Exception {

    }

}
```

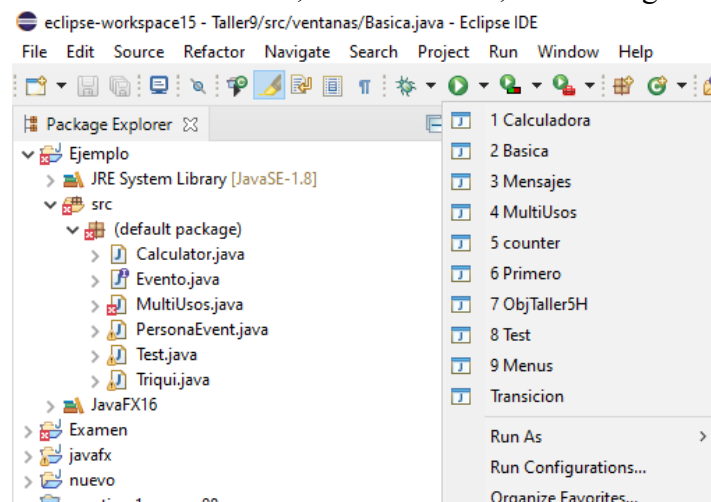
Notemos que para crear una interfaz grafica debemos heredar de la clase Application de JavaFX, y luego definir el método start, y en el main, usar el método launch(args), para iniciar la aplicación.

Ahora agreguemos un titulo a nuestra aplicacion, para eso usamos el método de la clase Stage, setTitle(), con el podemos darle un nombre a nuestra ventana y debemos usar el método show() para pintar o mostrar lo que hemos agregado.

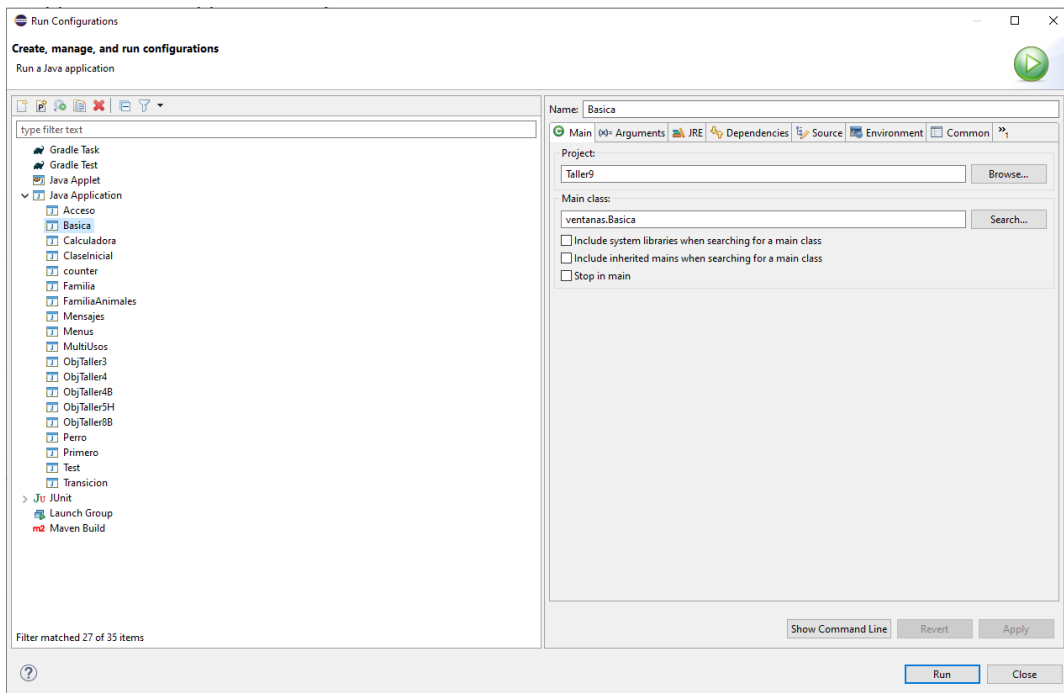
```
public class Basica extends Application {  
  
    public static void main(String args[]){  
        launch(args);  
    }  
  
    @Override  
    public void start(Stage window) throws Exception {  
        window.setTitle("Ventana");  
  
        window.show();  
    }  
}
```

Ahora vamos a ejecutarlo, recuerda que para ejecutar un programa de JavaFX debemos configurar la forma en que se ejecuta, para ello debemos seguir los siguientes pasos...

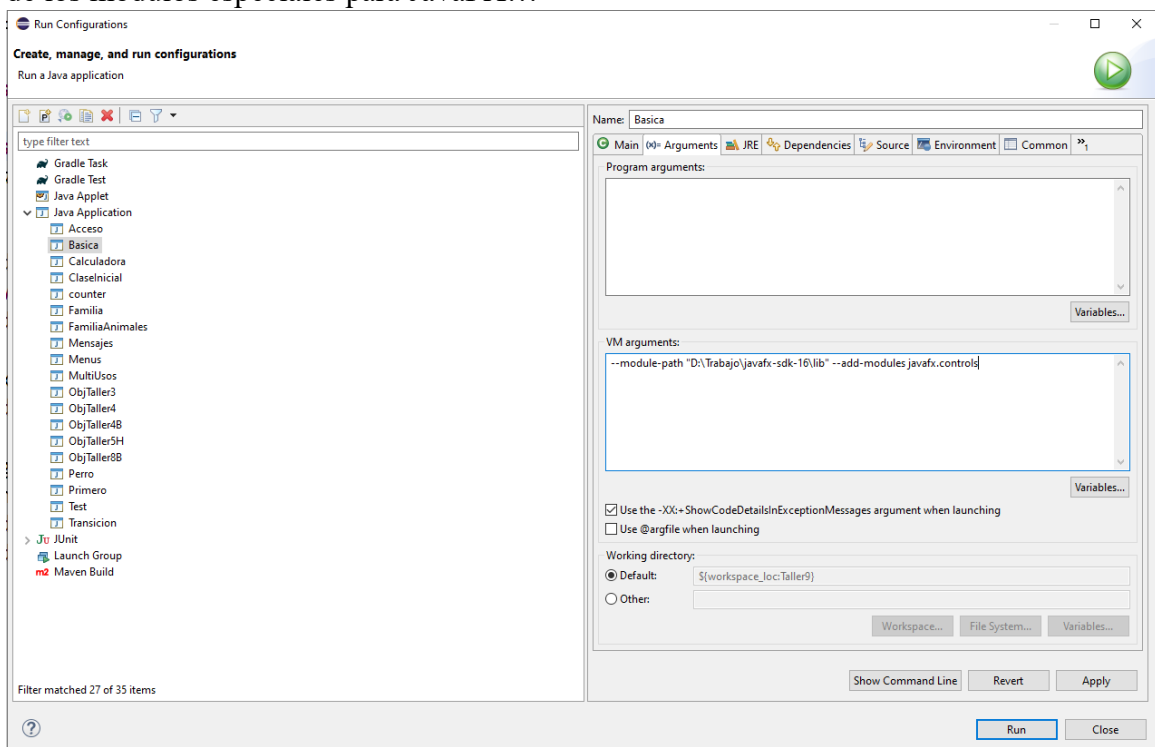
Al momento de darle correr, seleccionamos, Run configurations...



Ahora revisamos que estemos parados en la configuración de la clase Básica...



Seleccionamos Arguments y en el apartado de VM arguments debemos poner la direccion de los módulos especiales para JavaFX...

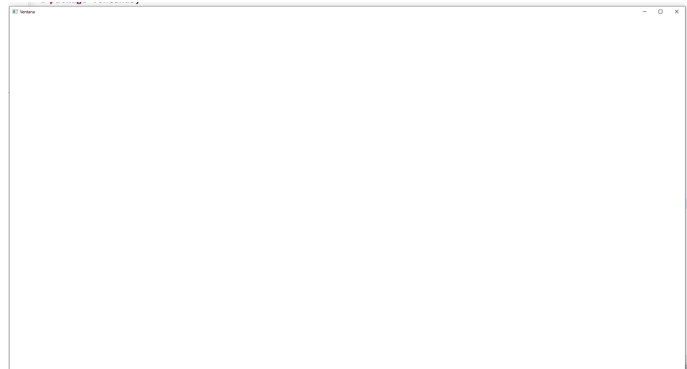


Esto es lo que se debe pegar en dicho espacio

--module-path "su directorio\javafx-sdk-16\lib" --add-modules javafx.controls

Nota: Si aparece dentro de los Arguments el siguiente checkbox: "Use the -XstartOnFirstThread argument when launching with SWT" **verifique que NO esté chequeado.**

Debe indicar el lugar de instalación de JavaFX y ya quedaría configurado para correr sin problemas un programa con JavaFX.



Este seria el resultado, aun no hemos agregado ningún elemento.

Ahora vamos a crear un layout donde poner nuestro elementos, añadiremos a este algunos elementos

```
public class Basica extends Application {

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

    @Override
    public void start(Stage window) throws Exception {
        window.setTitle("Ventana");

        // Configurar Layout
        FlowPane root = new FlowPane();
        root.setVgap(4);
        root.setHgap(4);

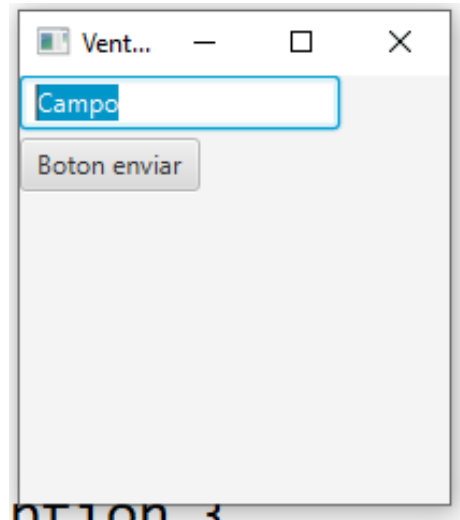
        // Crear un text field
        TextField t = new TextField("Campo");
        root.getChildren().add(t);

        // Generar Boton
        Button b1=new Button("Boton enviar");
        root.getChildren().add(b1);

        //Agregar el layout y darle una dimension a nuestra ventana
        Scene scene = new Scene(root, 200, 200);
        window.setScene(scene);
        window.show();
    }
}
```

```
}
```

El resultado sería el siguiente...



Experimente añadiendo otros elementos, otros layouts y las propiedades que tienen los elementos.

Ejercicio 1 - Saludo de bienvenida

Usted está encargada de desarrollar una interfaz gráfica que reciba el nombre de una persona y luego de dar clic sobre un botón imprimir el siguiente mensaje: 'Hola + nombre de la persona':

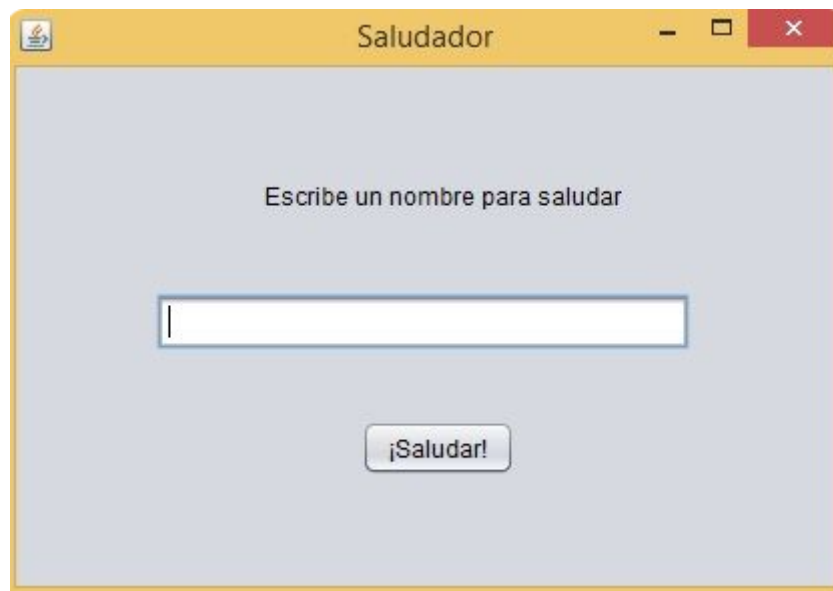


Imagen producto esperado

Ejercicio 2 - Posicionamiento de botones

Ahora trabajaremos un poco sobre el posicionamiento de elementos en la ventana principal, para ello debemos instanciar dos elementos (Botones) y ubicarlos en la parte central de la ventana como lo verá a continuación:

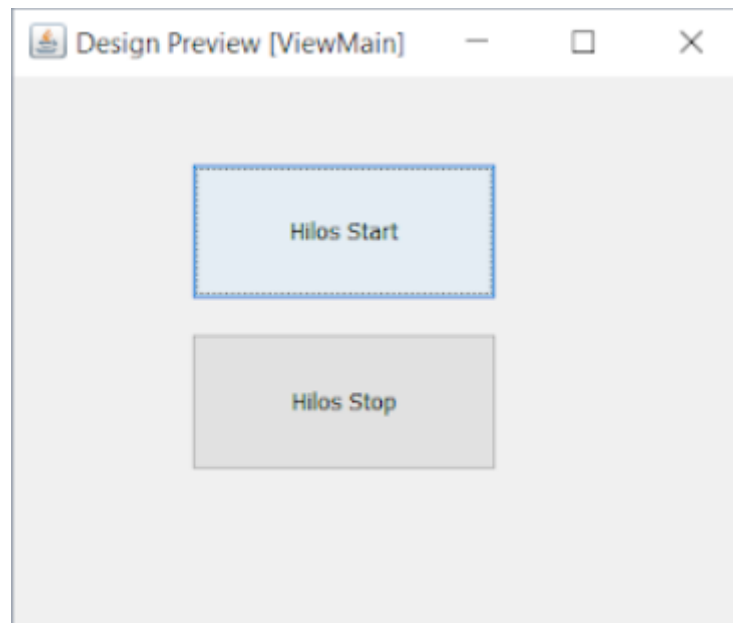
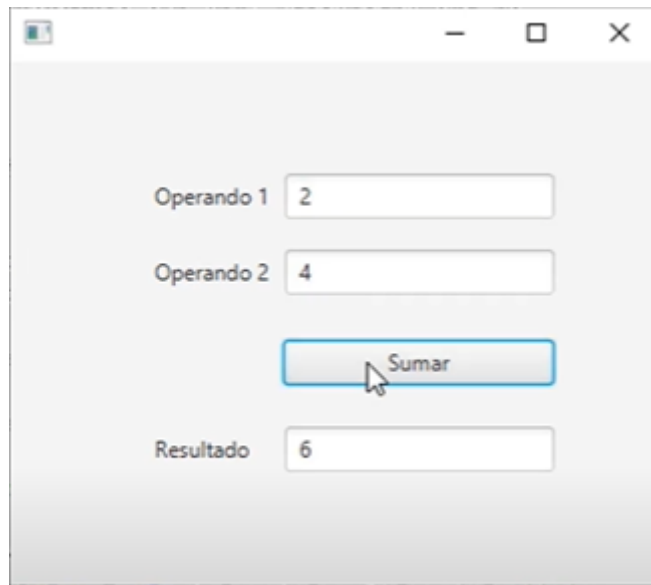


Imagen producto esperado

Ejercicio 3 - Suma aritmética básica

Vamos a empezar a trabajar algunos eventos asociados a instancias (Botones), en este caso se deben recibir dos números y posteriormente a ellos al oprimir sobre el botón sumar, debe mostrar el resultado de tal operación:



A screenshot of a graphical user interface for a basic arithmetic addition application. The window has a standard title bar with minimize, maximize, and close buttons. The interface contains four labeled text input fields and one button. The first field, labeled 'Operando 1', contains the number '2'. The second field, labeled 'Operando 2', contains the number '4'. Below these fields is a button labeled 'Sumar' with a mouse cursor hovering over it. At the bottom, a field labeled 'Resultado' contains the number '6'.

Label	Value
Operando 1	2
Operando 2	4
Botón	Sumar
Resultado	6

Imagen producto esperado

Ejercicio – GitHub

Ahora vamos a implementar la interfaz gráfica de una calculadora, su misión es dado un código, hacer las respectivas modificaciones en los espacios denotados con (***) de modo que el producto esperado sea el siguiente:

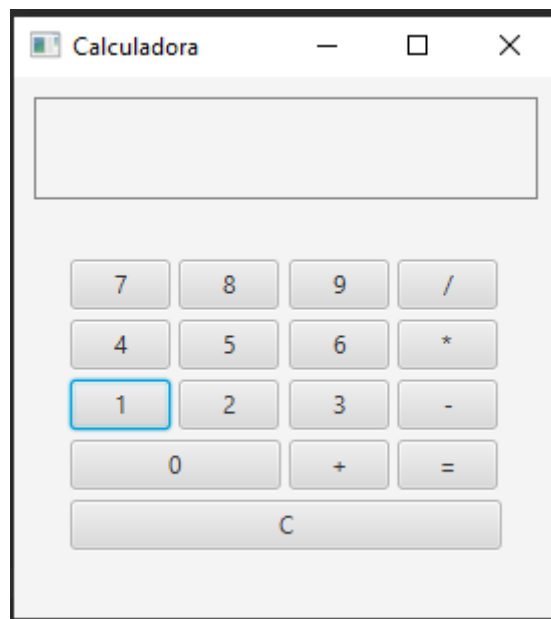


Imagen producto esperado

Adicionalmente, implemente las respectivas instrucciones para que la calculadora pueda ejecutar la operación suma, resta, multiplicación y división únicamente entre 2 números ingresados.

Enlace GitHub: <https://classroom.github.com/a/KiRZibEm>