

BÁO CÁO THỰC HÀNH LAB 5

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

1. Swing components	2
2. Organizing Swing components with Layout Managers	5
3. Create a graphical user interface for AIMS with Swing	7
4. JavaFX API	11
5. Setting up the View Cart Screen with ScreenBuilder	14
6. Integrating JavaFX into Swing application – The JFXPanel class	14
7. View the items in cart – JavaFX's data-driven UI	15

<https://github.com/VuThuongDat/IT3103.732871.2023.1.20215031.VuThuongDat>

1. Swing components

1.1. AWTAccumulator

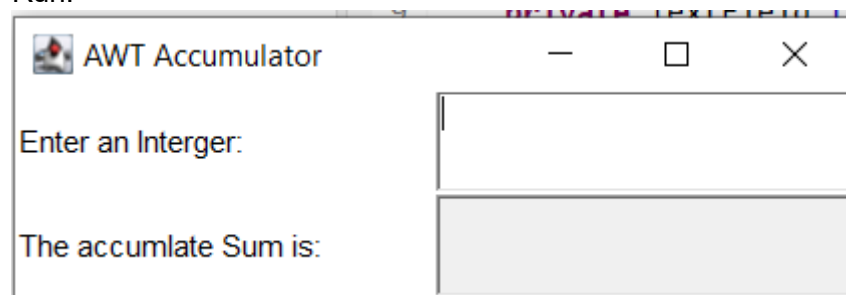
Code:

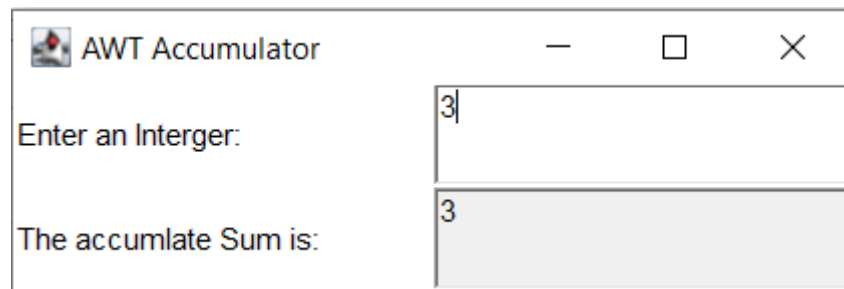
```

1 //Vu Thường Đạt 20215031
2 package h.GUIProject/src/hust/soict/hedspi/javafx/Painter.fxml
3
4 import java.awt.*;
5
6
7 public class AWTAccumulator extends Frame {
8     private TextField tfInput;
9     private TextField tfOutput;
10    private int sum = 0;
11    public AWTAccumulator() {
12        setLayout(new GridLayout(2,2));
13
14        add(new Label("Enter an Interger: "));
15
16        tfInput = new TextField(10);
17        add(tfInput);
18        tfInput.addActionListener(new TFInputListener());
19
20        add(new Label("The accumlate Sum is: "));
21
22        tfOutput = new TextField(10);
23        tfOutput.setEditable(false);
24        add(tfOutput);
25
26        setTitle("AWT Accumulator");
27        setSize(350,120);
28        setVisible(true);
29    }
30
31    public static void main(String[] args) {
32        new AWTAccumulator();
33    }
34
35    private class TFInputListener implements ActionListener {
36        @Override
37        public void actionPerformed (ActionEvent evt) {
38            int numberIn = Integer.parseInt(tfInput.getText());
39            sum += numberIn;
40            tfInput.setText("");
41            tfOutput.setText(sum + "");
42        }
43    }
44 }
45 }

```

Run:





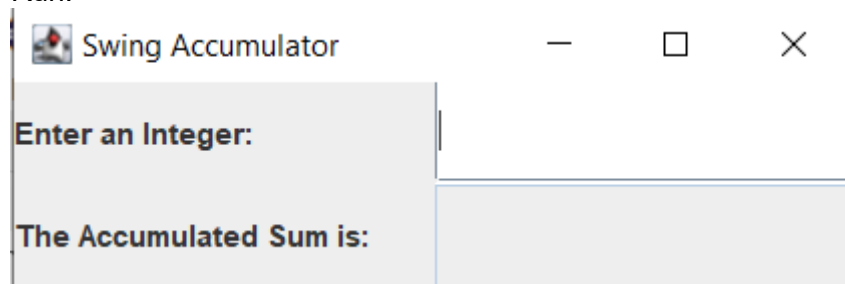
1.2. SwingAccumulator

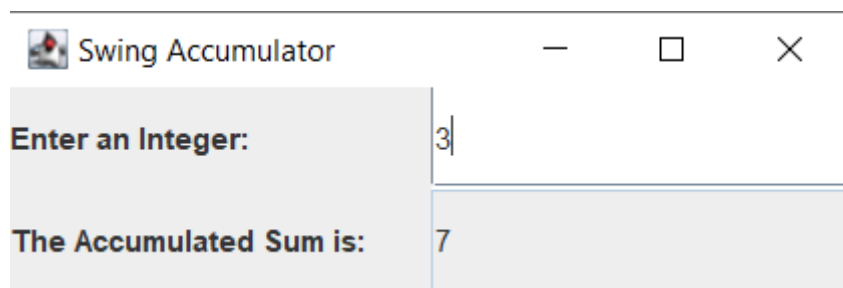
```

1 //Vu Thuong Dat 20215031
2 package hust.soict.hedspi.swing;
3
4 import java.awt.*;
5
6
7
8
9 public class SwingAccumulator extends JFrame {
10     private JTextField tfInput;
11     private JTextField tfOutput;
12     private int sum = 0;
13     public SwingAccumulator() {
14         Container cp = getContentPane();
15         cp.setLayout(new GridLayout(2, 2));
16
17         cp.add(new JLabel("Enter an Integer: "));
18
19         tfInput = new JTextField(10);
20         cp.add(tfInput);
21         tfInput.addActionListener(new TFInputListener());
22
23         cp.add (new JLabel("The Accumulated Sum is: "));
24
25         tfOutput = new JTextField(10);
26         tfOutput.setEditable(false);
27         cp.add(tfOutput);
28
29         setTitle("Swing Accumulator");
30         setSize(350, 120);
31         setVisible(true);
32     }
33
34     public static void main(String[] args) {
35         new SwingAccumulator();
36     }
37
38     private class TFInputListener implements ActionListener {
39         @Override
40         public void actionPerformed(ActionEvent evt) {
41             int numberIn = Integer.parseInt(tfInput.getText());
42             sum += numberIn;
43             tfOutput.setText(sum + "");
44         }
45     }
46 }

```

Run:





2. Organizing Swing components with Layout Managers

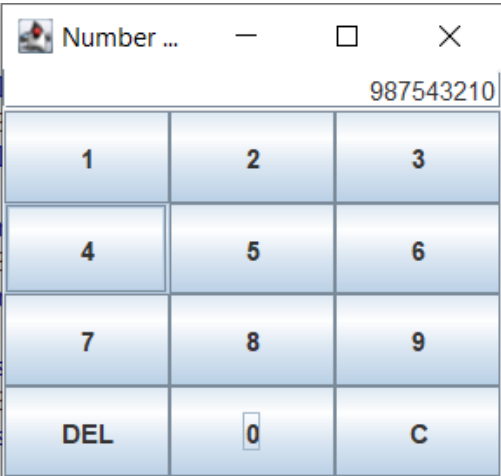
Code:

```

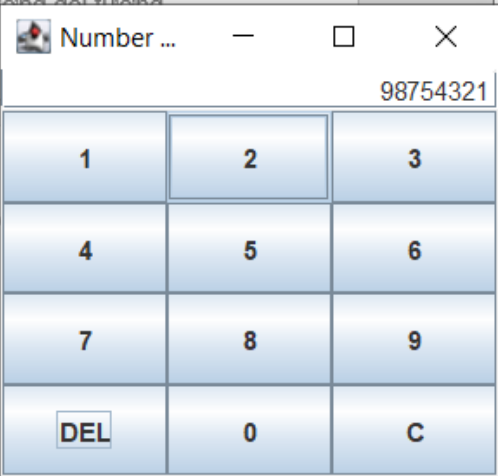
1  //Vu.Thuong.Dat 20215031
2  package hust.soict.hedspi.swing;
3
4  import java.awt.*;
5
6
7
8
9  public class NumberGrid extends JFrame{
10     private JButton [] btnNumbers = new JButton [10];
11     private JButton btnDelete, btnReset;
12     private JTextField tfDisplay;
13
14     public NumberGrid() {
15         tfDisplay = new JTextField();
16         tfDisplay.setComponentOrientation (ComponentOrientation. RIGHT_TO_LEFT);
17
18         JPanel panelButtons = new JPanel(new GridLayout (4, 3));
19         addButtons(panelButtons);
20
21         Container cp = getContentPane();
22         cp.setLayout(new BorderLayout());
23         cp.add(tfDisplay, BorderLayout.NORTH);
24         cp.add(panelButtons, BorderLayout.CENTER);
25
26         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
27         setTitle("Number Grid");
28         setSize(200, 200);
29         setVisible(true);
30     }
31
32     void addButtons(JPanel panelButtons) {
33         ButtonListener btnListener = new ButtonListener();
34         for (int i = 1; i <= 9; i++) {
35             btnNumbers[i] = new JButton("" + i);
36             panelButtons.add(btnNumbers[i]);
37             btnNumbers[i].addActionListener(btnListener);
38         }
39
40         btnDelete = new JButton("DEL");
41         panelButtons.add(btnDelete);
42         btnDelete.addActionListener(btnListener);
43
44         btnNumbers[0] = new JButton("0");
45         panelButtons.add(btnNumbers[0]);
46         btnNumbers[0].addActionListener(btnListener);
47
48         btnReset = new JButton("C");
49         panelButtons.add(btnReset);
50         btnReset.addActionListener(btnListener);
51     }
52
53     private class ButtonListener implements ActionListener {
54         public void actionPerformed(ActionEvent e) {
55             String button = e.getActionCommand();
56             if (button.charAt(0) >= '0' && button.charAt(0) <= '9') {
57                 tfDisplay.setText(tfDisplay.getText() + button);
58             } else if (button.equals("DEL")) {
59                 String text = tfDisplay.getText();
60                 if (!text.isEmpty()) {
61                     tfDisplay.setText(text.substring(0, text.length() - 1));
62                 }
63             } else {
64                 tfDisplay.setText("");
65             }
66         }
67     }
68     public static void main(String[] args) {
69         new NumberGrid();
70     }
71 }

```

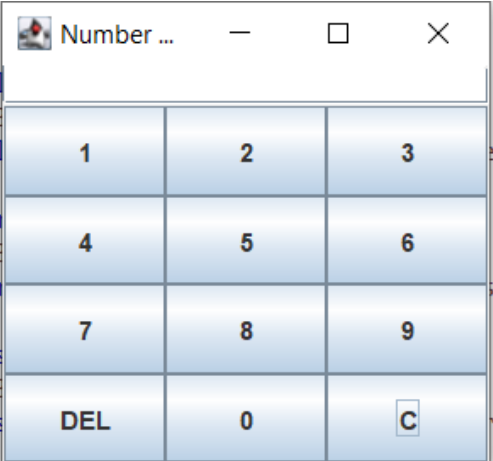
Run:
9->0:



Del:



C:



3. Create a graphical user interface for AIMS with Swing

Code:

StoreScreen

```

1 //Vu Thuong Dat 20215031
2 package hust.soict.hedspi.aims.screen;
3 import javax.swing.*;
17 public class StoreScreen extends JFrame {
18     private Store store;
19     private Cart cart;
20     private static List<Track> tracks;
21     private static List<String> authors;
22     public StoreScreen(Store store, Cart cart) {
23         this.store = store;
24         this.cart = cart;
25         Container cp = getContentPane();
26         cp.setLayout(new BorderLayout());
27
28         cp.add(createNorth(), BorderLayout.NORTH);
29         cp.add(createCenter(), BorderLayout.CENTER);
30
31         setVisible(true);
32         setTitle("Store");
33         setSize(1024, 768);
34     }
35
36     JPanel createNorth() {
37         JPanel north = new JPanel();
38         north.setLayout(new BoxLayout(north, BoxLayout.Y_AXIS));
39         north.add(createMenuBar());
40         north.add(createHeader());
41         return north;
42     }
43
44     JMenuBar createMenuBar() {
45         JMenu menu = new JMenu("Options");
46
47         JMenu smUpdateStore = new JMenu("Update Store");
48         smUpdateStore.add(new JMenuItem("Add Book"));
49         smUpdateStore.add(new JMenuItem("Add CD"));
50         smUpdateStore.add(new JMenuItem("Add DVD"));
51
52         menu.add(smUpdateStore);
53         menu.add(new JMenuItem("View store"));
54         menu.add(new JMenuItem("View cart"));

```

```

55
56     JMenuBar menuBar = new JMenuBar();
57     menuBar.setLayout(new FlowLayout(FlowLayout.LEFT));
58     menuBar.add(menu);
59
60     return menuBar;
61 }
62
63 JPanel createHeader() {
64     JPanel header = new JPanel();
65     header.setLayout(new BoxLayout(header, BoxLayout.X_AXIS));
66
67     JLabel title = new JLabel("AIMS");
68     title.setFont(new Font(title.getFont().getName(), Font.PLAIN, 50));
69     title.setForeground(Color.CYAN);
70
71     JButton viewcart = new JButton("View cart");
72     viewcart.setPreferredSize(new Dimension(100, 50));
73     viewcart.setMaximumSize(new Dimension(100, 50));
74     viewcart.addActionListener(new ButtonListener(cart));
75
76     header.add(Box.createRigidArea(new Dimension(10, 10)));
77     header.add(title);
78     header.add(Box.createHorizontalGlue());
79     header.add(viewcart);
80     header.add(Box.createRigidArea(new Dimension(10, 10)));
81
82     return header;
83 }
84
85 JPanel createCenter() {
86     JPanel center = new JPanel();
87     center.setLayout(new GridLayout(3, 3, 2, 2));
88     MediaStore.setCart(cart);
89     ArrayList<Media> mediaInStore = store.getItemsInStore();
90     int numItems = Math.min(9, mediaInStore.size());
91     for (int i = 0; i < numItems; i++) {
92         MediaStore cell = new MediaStore(mediaInStore.get(i));
93         center.add(cell);
94     }
95     return center;
96 }
97 public static void main(String[] args) throws Exception {
98     Store store = new Store();
99     Cart cart = new Cart();
100     Media media1 = new DigitalVideoDisc("The Lion King", "Animation", "John Director", 87, 19.95f);
101     store.addMedia(media1);
102     authors = new ArrayList<>();
103     authors.add("Antoine de Saint-Exupéry");
104     Media media2 = new Book("The little prince", "Animation", authors, 18.15f);
105     store.addMedia(media2);
106     tracks = new ArrayList<>();
107     tracks.add(new Track("I do", 5));
108     tracks.add(new Track("The night", 3));
109     Media media3 = new CompactDisc("Music", "US-UK", "Artist X", tracks, 20.81f);
110     store.addMedia(media3);
111     new StoreScreen(store, cart);
112 }
113 private class ButtonListener implements ActionListener {
114     private Cart cart;
115
116     public ButtonListener(Cart cart) {
117         super();
118         this.cart = cart;
119     }
120
121     @Override
122     public void actionPerformed(ActionEvent e) {
123         String button = e.getActionCommand();
124         if (button.equals("View cart")) {
125             cart.printCart();
126         }
127     }
128 }
129 }

```


MediaScreen

```

1 package hust.soict.hedspi.aims.screen;
2 import hust.soict.hedspi.aims.cart.Cart;
17 public class MediaStore extends JPanel {
18     private Media media;
19     private static Cart cart=new Cart();
20     public MediaStore(Media media) {
21         this.media = media;
22         this.setLayout(new BorderLayout(this, BorderLayout.Y_AXIS));
23
24         JLabel title = new JLabel(media.getTitle());
25         title.setFont(new Font(title.getFont().getName(), Font.PLAIN, 20));
26         title.setAlignmentX(CENTER_ALIGNMENT);
27
28         JLabel cost = new JLabel(""+ media.getCost()+" $");
29         cost.setAlignmentX(CENTER_ALIGNMENT);
30
31         JPanel container = new JPanel();
32         container.setLayout(new FlowLayout(FlowLayout.CENTER));
33
34         JButton addButton = new JButton("Add to cart");
35         container.add(addButton);
36         addButton.addActionListener(new ButtonListener(media));
37         if (media instanceof Playable) {
38             JButton playButton = new JButton("Play");
39             container.add(playButton);
40             playButton.addActionListener(new ButtonListener(media));
41         }
42
43         this.add(Box.createVerticalGlue());
44         this.add(title);
45         this.add(cost);
46         this.add(Box.createVerticalGlue());
47         this.add(container);
48
49         this.setBorder(BorderFactory.createLineBorder(Color.BLACK));
50     }
51     public static void setCart(Cart cart) {
52         MediaStore.cart = cart;
53     }
54     private class ButtonListener implements ActionListener {
55         private Media media;
56         public ButtonListener(Media media) {
57             super();
58             this.media = media;
59         }
60         @Override
61         public void actionPerformed(ActionEvent e) {
62             String button = e.getActionCommand();
63             if (button.equals("Add to cart")) {
64                 cart.addMedia(media);
65             } else if (button.equals("Play")) {
66                 media.play();
67             }
68         }
69     }
70 }

```

Run:

StoreScreen [Java Application] C:\Users\user\p2\pool\plugins\org.eclipse.justj.openj

Successfully added The Lion King (ID: 1) to store!
 Successfully added The little prince (ID: 2) to store!
 Successfully added Music (ID: 3) to store!
 Successfully added The Lion King (ID: 1) to cart!
 Successfully added The little prince (ID: 2) to cart!
 Can't add because The Lion King is already in the cart!
 Can't add because The little prince is already in the cart!
 Playing DVD: The Lion King
 DVD length: 87
 *****CART*****
 ID: 1 - The Lion King - Animation - 19,95 \$
 ID: 2 - The little prince - Animation - 18,15 \$
 Total cost: 38,10 \$

4. JavaFX API

Source:

Screen Build:



PaintControl:

```

1 package hust.soict.hedspi.javafx;
2 *import javafx.event.ActionEvent;
10 public class PainterController {
11     private Paint penColor;
12     @FXML
13     private ToggleGroup identical;
14
15     @FXML
16     private Pane drawingAreaPane;
17
18     @FXML
19     void drawingAreaMouseDragged(MouseEvent event) {
20         Circle newCircle = new Circle(event.getX(),
21             event.getY(), 4, penColor);
22         drawingAreaPane.getChildren().add(newCircle);
23     }
24
25     @FXML
26     void penButtonSelected(ActionEvent event) {
27         penColor = Color.BLACK;
28     }
29
30     @FXML
31     void eraserButtonSelected(ActionEvent event) {
32         penColor = Color.WHITE;
33     }
34
35     @FXML
36     void clearButtonPressed(ActionEvent event) {
37         drawingAreaPane.getChildren().clear();
38     }
39 }

```

Paint:

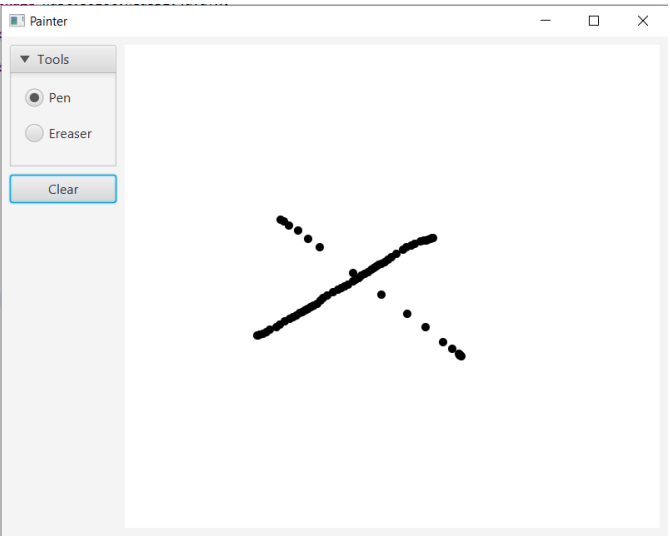
```

1 package hust.soict.hedspi.javafx;
2
3 *import javafx.application.Application;
8
9 public class Painter extends Application {
10     @Override
11     public void start(Stage stage) throws Exception {
12         Parent root = FXMLLoader.load(getClass()
13             .getResource("Painter.fxml"));
14         Scene scene = new Scene(root);
15         stage.setTitle("Painter");
16         stage.setScene(scene);
17         stage.show();
18     }
19
20     public static void main (String []args){
21         launch(args);
22     }
23 }

```

Run:

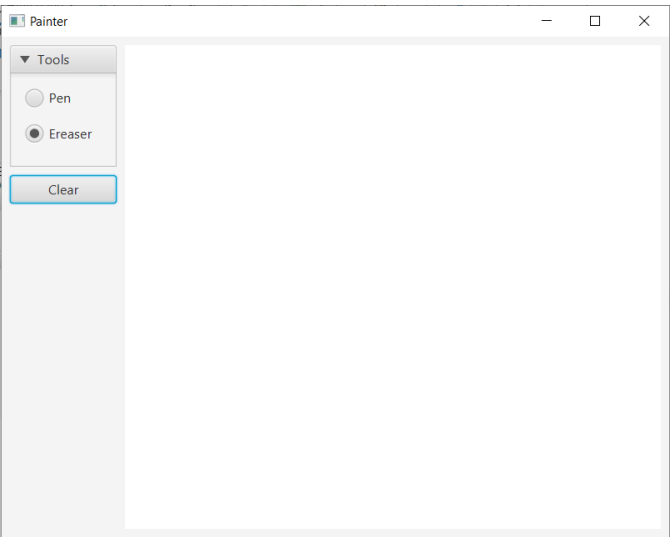
Pen



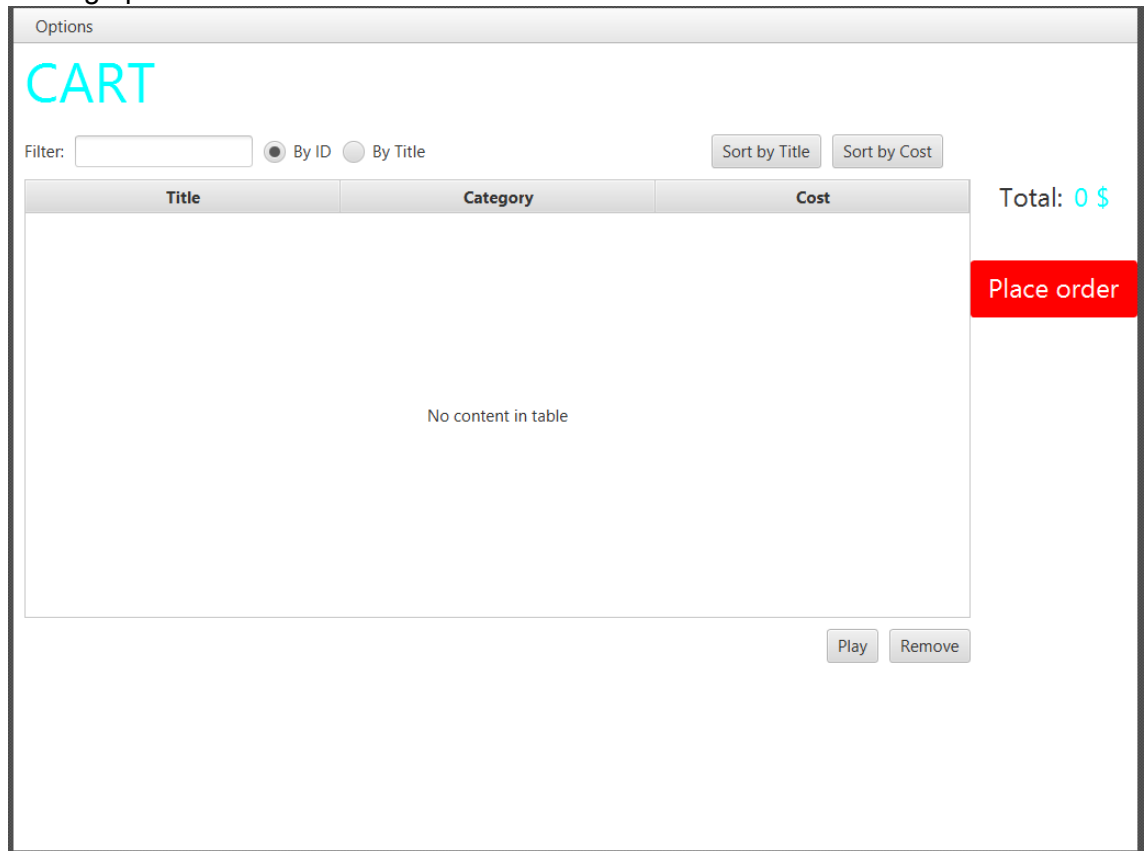
Eraser



Clear:



5. Setting up the View Cart Screen with ScreenBuilder



6. Integrating JavaFX into Swing application – The JFXPanel class

```

1 //Vu Thuong Dat 20215031
2 package hust.soict.hedspi.aims.screen;
3
4 import hust.soict.hedspi.aims.cart.Cart;
12 public class CartScreen extends JFrame{
13     private Cart cart;
14
15     public CartScreen(Cart cart) {
16         super();
17
18         this.cart = cart;
19
20         JFXPanel fxPanel = new JFXPanel();
21         this.add(fxPanel);
22
23         this.setTitle("Cart");
24         this.setVisible(true);
25         Platform.runLater(new Runnable() {
26             @Override
27             public void run() {
28                 try {
29                     FXMLLoader loader = new FXMLLoader(getClass().getResource("cart.fxml"));
30                     CartScreenController controller =
31                         new CartScreenController();
32                     controller.setCart(cart);
33                     loader.setController(controller);
34                     Parent root = loader.load();
35                     fxPanel.setScene(new Scene(root));
36                 } catch (IOException e) {
37                     e.printStackTrace();
38                 }
39             }
40         });
41     }
42 }

```

7. View the items in cart – JavaFX's data-driven UI

```

1 //Vu Thuong Dat 20215031
2 package hust.soict.hedspi.aims.screen;
3
4 import hust.soict.hedspi.aims.cart.Cart;
5
15 public class CartScreenController {
16     private Cart cart;
17
18     @FXML
19     private TableColumn<Media, String> colMediaCategory;
20
21     @FXML
22     private TableView<Media> tblMedia;
23
24     @FXML
25     private TableColumn<Media, Float> colMediaCost;
26
27     @FXML
28     private ToggleGroup filterCategory;
29
30     @FXML
31     private TableColumn<Media, String> colMediaTitle;
32
33     public void setCart(Cart cart) {
34         this.cart = cart;
35         initialize();
36     }
37
38     @FXML
39     private void initialize() {
40         colMediaTitle.setCellValueFactory(
41             new PropertyValueFactory<Media, String>("title"));
42         colMediaCategory.setCellValueFactory(
43             new PropertyValueFactory<Media, String>("category"));
44         colMediaCost.setCellValueFactory(
45             new PropertyValueFactory<Media, Float>("cost"));
46         tblMedia.setItems(this.cart.getItemsOrdered());
47     }
48 }
49

```