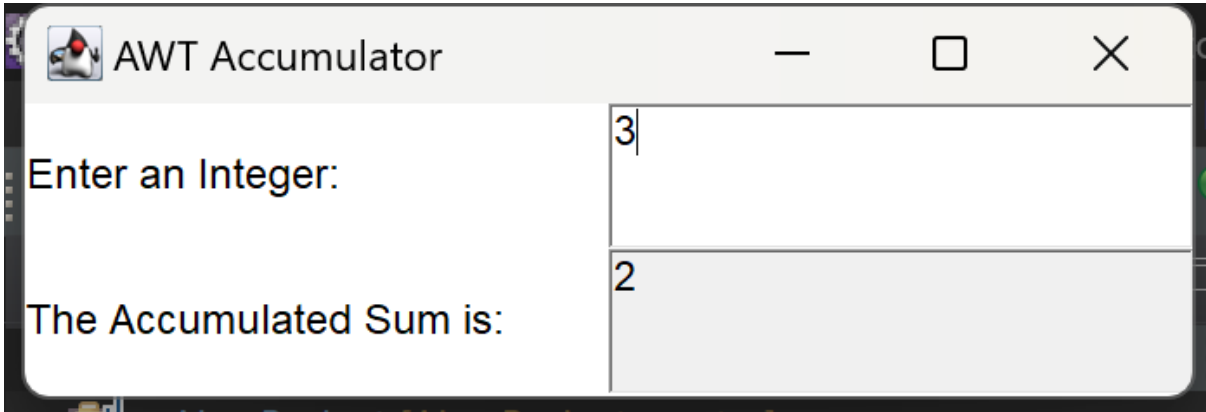
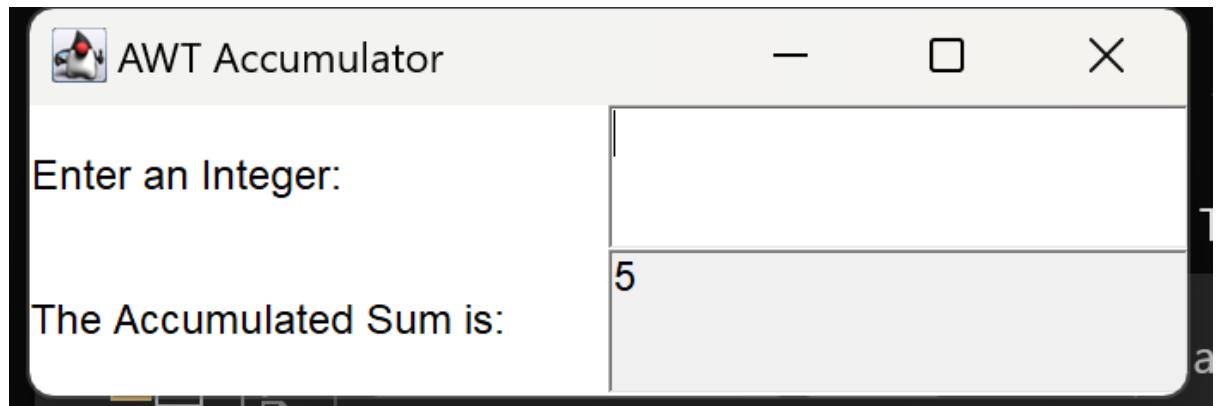


1. SWING

1.1 AWTAccumulator

```
AWTAccumulator.java X
1 package hust.soict.dsai.swing;
2
3 import java.awt.Frame;
4 import java.awt.GridLayout;
5 import java.awt.Label;
6 import java.awt.TextField;
7 import java.awt.event.ActionEvent;
8 import java.awt.event.ActionListener;
9
10 public class AWTAccumulator extends Frame {
11     private TextField tfInput;
12     private TextField tfOutput;
13     private int sum = 0;
14     // Accumulated sum, init to 0
15     // Constructor to setup the GUI components and event handlers
16     public AWTAccumulator() {
17         setLayout(new GridLayout(2, 2));
18         add(new Label("Enter an Integer: "));
19         tfInput = new TextField(10);
20         add(tfInput);
21         tfInput.addActionListener(new TFInputListener());
22         add(new Label("The Accumulated Sum is: "));
23         tfOutput = new TextField(10);
24         tfOutput.setEditable(false);
25         add(tfOutput);
26         setTitle("AWT Accumulator");
27         setSize(350, 120);
28         setVisible(true);
29     }
30     public static void main(String[] args) {
31         new AWTAccumulator();
32     }
33     private class TFInputListener implements ActionListener {
34         @Override
35         public void actionPerformed(ActionEvent evt) {
36             int numberIn = Integer.parseInt(tfInput.getText());
37             sum += numberIn;
38             tfInput.setText("");
39             tfOutput.setText(sum + "");
40         }
41     }
42 }
```





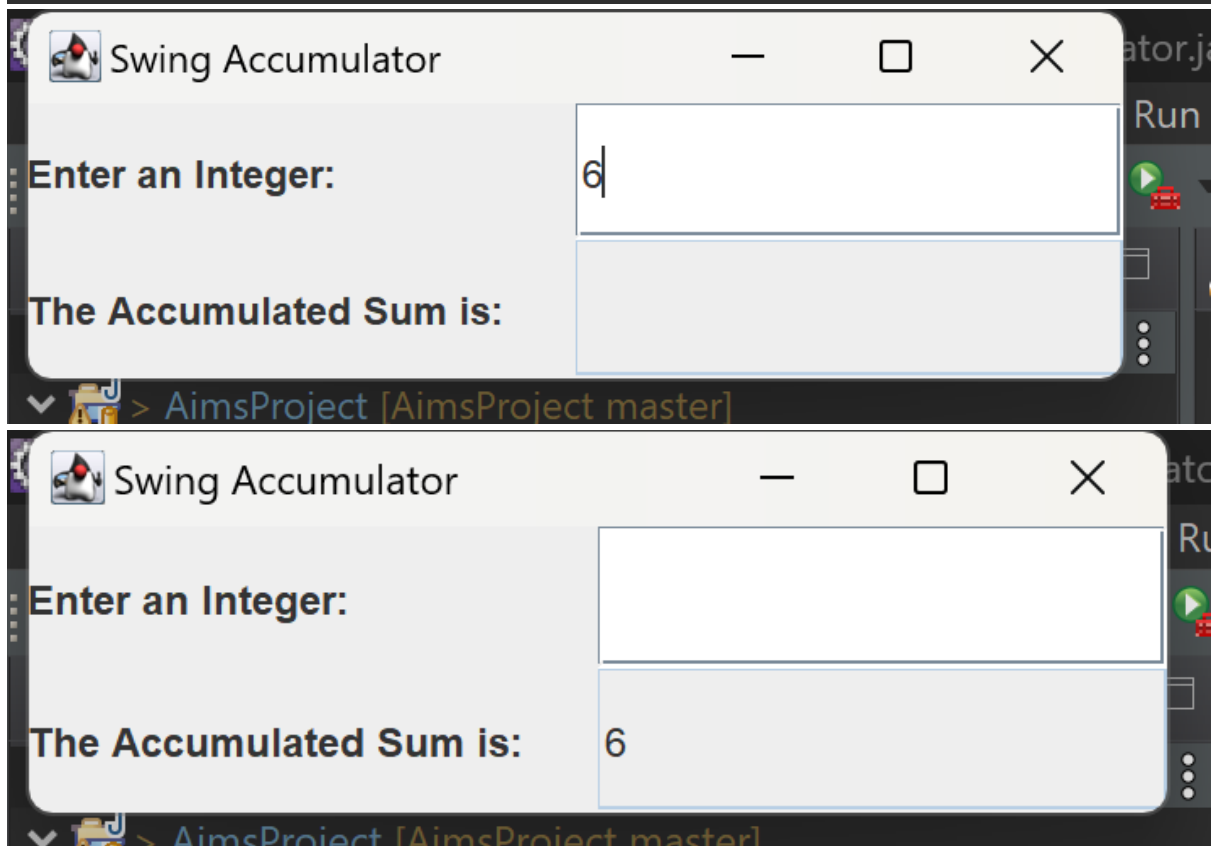
1.2 SwingAccumulator

```
AWTAccumulator.java SwingAccumulator.java X
1 package hust.soict.dsai.swing;
2
3 import java.awt.Container;
4 import java.awt.GridLayout;
5 import java.awt.event.ActionEvent;
6 import java.awt.event.ActionListener;
7
8 import javax.swing.JFrame;
9 import javax.swing.JLabel;
10 import javax.swing.JTextField;
11
12 public class SwingAccumulator extends JFrame {
13     private JTextField tfInput;
14     private JTextField tfOutput;
15     private int sum = 0;
16     // Accumulated sum, init to 0
17     // Constructor to setup the GUI components and event handlers
18     public SwingAccumulator() {
19         Container cp = getContentPane();
20         cp.setLayout(new GridLayout(2, 2));
21         cp.add(new JLabel("Enter an Integer: "));
22         tfInput = new JTextField(10);
23         cp.add(tfInput);
24         tfInput.addActionListener(new TFInputListener());
25         cp.add(new JLabel("The Accumulated Sum is: "));
26         tfOutput = new JTextField(10);
27         tfOutput.setEditable(false);
28         cp.add(tfOutput);
29         setTitle("Swing Accumulator");
30         setSize(350, 120);
31         setVisible(true);
    }
```

```

    }
    public static void main(String[] args) {
        new SwingAccumulator();
    }
    private class TFInputListener implements ActionListener {
        @Override
        public void actionPerformed(ActionEvent evt) {
            int numberIn = Integer.parseInt(tfInput.getText());
            sum += numberIn;
            tfInput.setText("");
            tfOutput.setText(sum + "");
        }
    }
}

```



2. Swing với Layout Managers
 - 2.2 NumberGrid

```

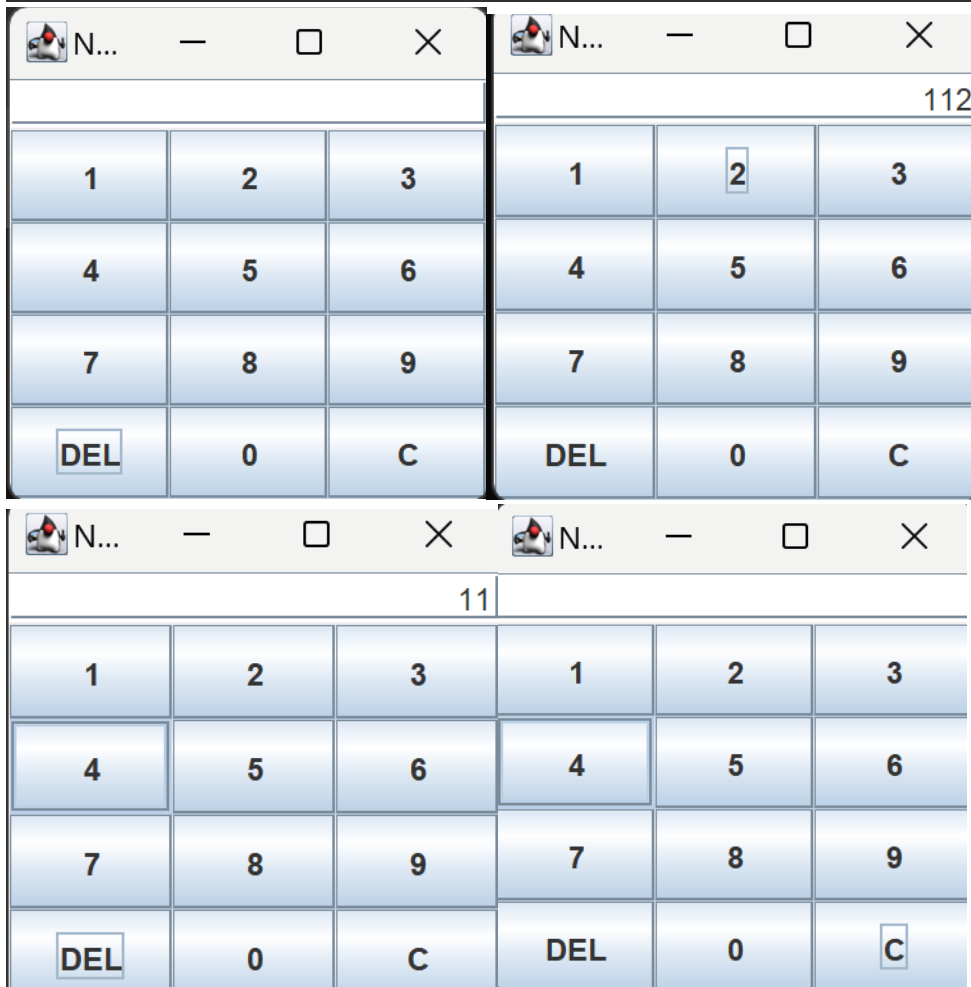
1 package hust.soict.dsai.swing;
2 import java.awt.BorderLayout;
3 import java.awt.ComponentOrientation;
4 import java.awt.Container;
5 import java.awt.GridLayout;
6 import java.awt.event.ActionEvent;
7 import java.awt.event.ActionListener;
8 import javax.swing.JButton;
9 import javax.swing.JFrame;
10 import javax.swing.JPanel;
11 import javax.swing.JTextField;
12
13 public class NumberGrid extends JFrame{
14     private JButton [] btnNumbers = new JButton [10];
15     private JButton btnDelete, btnReset;
16     private JTextField tfDisplay;
17     public NumberGrid() {
18         tfDisplay = new JTextField();
19         tfDisplay.setComponentOrientation( ComponentOrientation.RIGHT_TO_LEFT);
20         JPanel panelButtons = new JPanel(new GridLayout(4, 3));
21         addButtons (panelButtons);
22         Container cp = getContentPane();
23         cp.setLayout(new BorderLayout());
24         cp.add(tfDisplay, BorderLayout.NORTH);
25         cp.add(panelButtons, BorderLayout.CENTER);
26         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
27         setTitle("Number Grid");
28         setSize(200, 200);
29         setVisible(true);
30     }
31     void addButtons (JPanel panelButtons) {
32         ButtonListener btnListener = new ButtonListener();
33         for(int i = 1; i <= 9; i++) {
34             btnNumbers[i] = new JButton(""+i);
35             panelButtons.add(btnNumbers[i]);
36             btnNumbers[i].addActionListener(btnListener);
37         }

```

```

37     }
38     btnDelete = new JButton("DEL");
39     panelButtons.add(btnDelete);
40     btnDelete.addActionListener(btnListener);
41     btnNumbers [0] = new JButton("0");
42     panelButtons.add(btnNumbers[0]);
43     btnNumbers [0].addActionListener(btnListener);
44     btnReset = new JButton("C");
45     panelButtons.add(btnReset);
46     btnReset.addActionListener(btnListener);
47     }
48     private class ButtonListener implements ActionListener{
49         @Override
50         public void actionPerformed(ActionEvent e) {
51             String button = e.getActionCommand();
52             if(button.charAt(0) >= '0' && button.charAt(0) <= '9') { tfDisplay.setText(tfDisplay.getText() + button);
53             }
54             else if (button.equals("DEL")) {
55                 //handles the "DEL" case
56                 String currentText = tfDisplay.getText();
57                 if (!currentText.isEmpty()) {
58                     tfDisplay.setText(currentText.substring(0, currentText.length() - 1));
59                 }
60             }
61             else {
62                 //handles the "C" case
63                 tfDisplay.setText("");
64             }
65         }
66     }
67     public static void main(String[] args) {
68         new NumberGrid();
69     }
70 }
71
72

```



3. ViewStore

3.1 TÍNH

```
StoreScreen.java  MediaStore.java X
1 package hust.soict.dsai.aims.screen;
2
3 import java.awt.Color;
4 import java.awt.FlowLayout;
5 import java.awt.Font;
6
7 import javax.swing.BorderFactory;
8 import javax.swing.Box;
9 import javax.swing.BoxLayout;
10 import javax.swing.JButton;
11 import javax.swing.JLabel;
12 import javax.swing.JPanel;
13
14 import hust.soict.dsai.aims.media.Media;
15 import hust.soict.dsai.aims.media.Playable;
16
17 public class MediaStore extends JPanel{
18     private Media media;
19     public MediaStore (Media media) {
20         this.media = media;
21         this.setLayout(new BoxLayout(this, BoxLayout.Y_AXIS));
22         JLabel title = new JLabel(media.getTitle());
23         title.setFont(new Font(title.getFont().getName(), Font.PLAIN, 20));
24         title.setAlignmentX(CENTER_ALIGNMENT);
25         JLabel cost = new JLabel(""+media.getCost()+" $");
26         cost.setAlignmentX(CENTER_ALIGNMENT);
27         JPanel container = new JPanel();
28         container.setLayout(new FlowLayout (FlowLayout.CENTER));
29         container.add(new JButton("Add to cart"));
30         if(media instanceof Playable) {
31             container.add(new JButton("Play"));
32         }
33         this.add(Box.createVerticalGLue());
34         this.add(title);
35         this.add(cost);
36         this.add(Box.createVerticalGLue());
37         this.add(container);
38         this.setBorder(BorderFactory.createLineBorder(Color.BLACK));
39     }
40 }
41
42
1 package hust.soict.dsai.aims.screen;
2
3 import java.awt.BorderLayout;
4 import java.awt.Color;
5 import java.awt.Container;
6 import java.awt.Dimension;
7 import java.awt.FlowLayout;
8 import java.awt.Font;
9 import java.awt.GridLayout;
10 import java.util.ArrayList;
11
12 import javax.swing.Box;
13 import javax.swing.BoxLayout;
14 import javax.swing.JButton;
15 import javax.swing.JFrame;
16 import javax.swing.JLabel;
17 import javax.swing.JMenu;
18 import javax.swing.JMenuBar;
19 import javax.swing.JMenuItem;
20 import javax.swing.JPanel;
21
22 import hust.soict.dsai.aims.media.Book;
23 import hust.soict.dsai.aims.media.CompactDisc;
24 import hust.soict.dsai.aims.media.DigitalVideoDisc;
25 import hust.soict.dsai.aims.media.Media;
26 import hust.soict.dsai.aims.store.Store;
27
28 public class StoreScreen extends JFrame{
29     private Store store;
30     JPanel createNorth() {
```

```

31         JPanel north = new JPanel();
32         north.setLayout(new BorderLayout(north, BorderLayout.Y_AXIS));
33         north.add(createMenuBar());
34         north.add(createHeader());
35         return north;
36     }
37     JMenuBar createMenuBar() {
38         JMenu menu = new JMenu("Options");
39         JMenu smUpdateStore = new JMenu("Update Store");
40         smUpdateStore.add(new JMenuItem("Add Book"));
41         smUpdateStore.add(new JMenuItem("Add CD"));
42         smUpdateStore.add(new JMenuItem("Add DVD"));
43         menu.add(smUpdateStore);
44         menu.add(new JMenuItem("View store"));
45         menu.add(new JMenuItem("View cart"));
46         JMenuBar menuBar = new JMenuBar();
47         menuBar.setLayout(new FlowLayout(FlowLayout.LEFT));
48         menuBar.add(menu);
49         return menuBar;
50     }
51     JPanel createHeader() {
52         JPanel header = new JPanel();
53         header.setLayout(new BorderLayout(header, BorderLayout.X_AXIS));
54         JLabel title = new JLabel("AIMS");
55         title.setFont(new Font(title.getFont().getName(), Font.PLAIN, 50));
56         title.setForeground(Color.CYAN);
57         JButton cart = new JButton("View cart");
58         cart.setPreferredSize(new Dimension(100, 50));
59         cart.setMaximumSize(new Dimension(100, 50));
60         header.add(Box.createRigidArea(new Dimension(10, 10)));
61         header.add(title);
62         header.add(Box.createHorizontalGLue());
63         header.add(cart);
64         header.add(Box.createRigidArea(new Dimension(10, 10)));
65         return header;
66     }
67     JPanel createCenter() {
68         JPanel center = new JPanel();
69         center.setLayout(new GridLayout(3, 3, 2, 2));
70         ArrayList<Media> mediaInStore = store.getItemsInStore();
71         for (int i = 0; i < 9; i++) {
72             MediaStore cell = new MediaStore(mediaInStore.get(i));
73             center.add(cell);
74         }
75         return center;
76     }
77     public StoreScreen (Store store) {
78         this.store = store;
79         Container cp = getContentPane();
80         cp.setLayout(new BorderLayout());
81         cp.add(createNorth(), BorderLayout.NORTH);
82         cp.add(createCenter(), BorderLayout.CENTER);
83         setVisible(true);
84         setTitle("Store");
85         setSize(1024, 768);
86     }

```

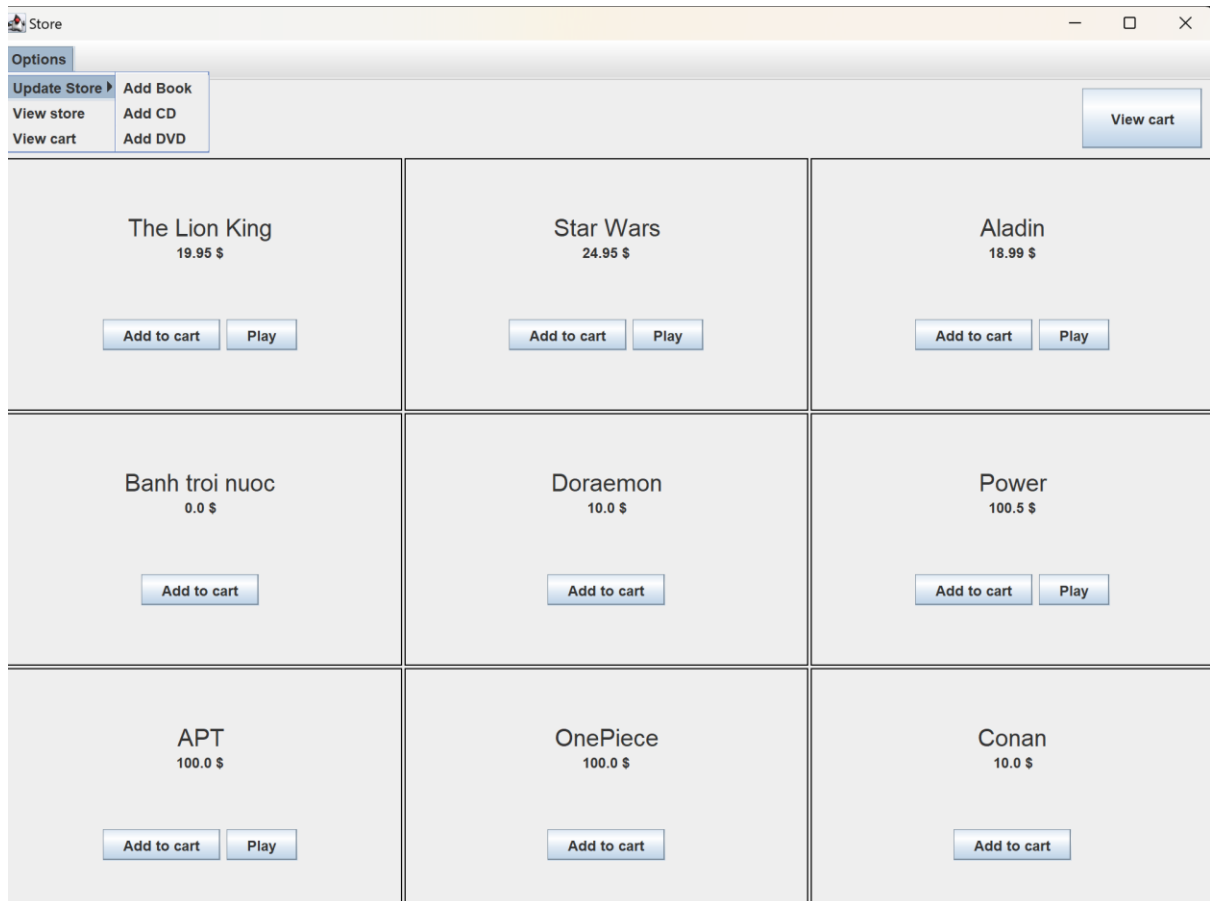
```

    public static void main(String[] args) {
        Store store=new Store();

        DigitalVideoDisc dvd1 = new DigitalVideoDisc("The Lion King", "Animation", "Roger Allers", 87, 19.95f);
        store.addMedia(dvd1);
        DigitalVideoDisc dvd2 = new DigitalVideoDisc("Star Wars", "Science Fiction", "George Lucas", 87, 24.95f);
        store.addMedia(dvd2);
        DigitalVideoDisc dvd3 = new DigitalVideoDisc("Aladin", "Animation", 18.99f);
        store.addMedia(dvd3);
        Book book1 = new Book("Banh troi nuoc", "ho xuan huong");
        store.addMedia(book1);
        Book book2 = new Book("Doraemon", "anime", 10.00f);
        store.addMedia(book2);
        CompactDisc track1 = new CompactDisc("Power", "kpop", 100.50f);
        store.addMedia(track1);
        CompactDisc track2 = new CompactDisc("APT", "kpop", 100f);
        store.addMedia(track2);
        Book book3 = new Book("OnePiece", "anime", 100f);
        store.addMedia(book3);
        Book book4 = new Book("Conan", "anime", 10.00f);
        store.addMedia(book4);

        new StoreScreen(store);
    }
}

```



3.2 Tương tác

Cập nhật lớp MediaStore

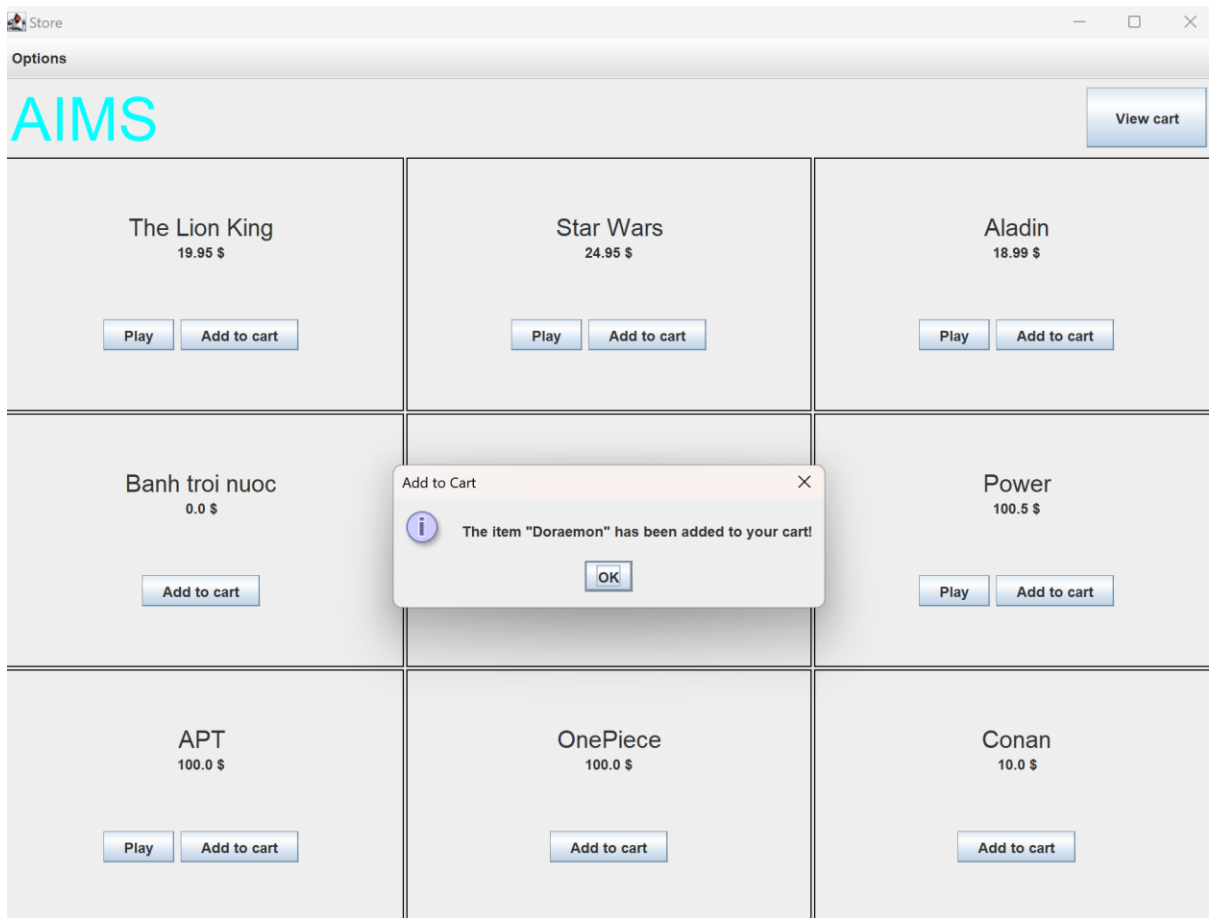
```
1 package hust.soict.dsai.aims.screen;
2
3 import java.awt.Color;
4 import java.awt.FlowLayout;
5 import java.awt.Font;
6
7 import javax.swing.BorderFactory;
8 import javax.swing.Box;
9 import javax.swing.BoxLayout;
10 import javax.swing.JButton;
11 import javax.swing.JLabel;
12 import javax.swing.JPanel;
13
14 import hust.soict.dsai.aims.media.Media;
15 import hust.soict.dsai.aims.media.Playable;
16
17 public class MediaStore extends JPanel {
18     private Media media;
19
20     public MediaStore(Media media) {
21         this.media = media;
22         this.setLayout(new BoxLayout(this, BoxLayout.Y_AXIS));
23         JLabel title = new JLabel(media.getTitle());
24         title.setFont(new Font(title.getFont().getName(), Font.PLAIN, 20));
25         title.setAlignmentX(CENTER_ALIGNMENT);
26         JLabel cost = new JLabel("" + media.getCost() + " $");
27         cost.setAlignmentX(CENTER_ALIGNMENT);
28         JPanel container = new JPanel();
29         container.setLayout(new FlowLayout(FlowLayout.CENTER));
30         if (media instanceof Playable) {
```

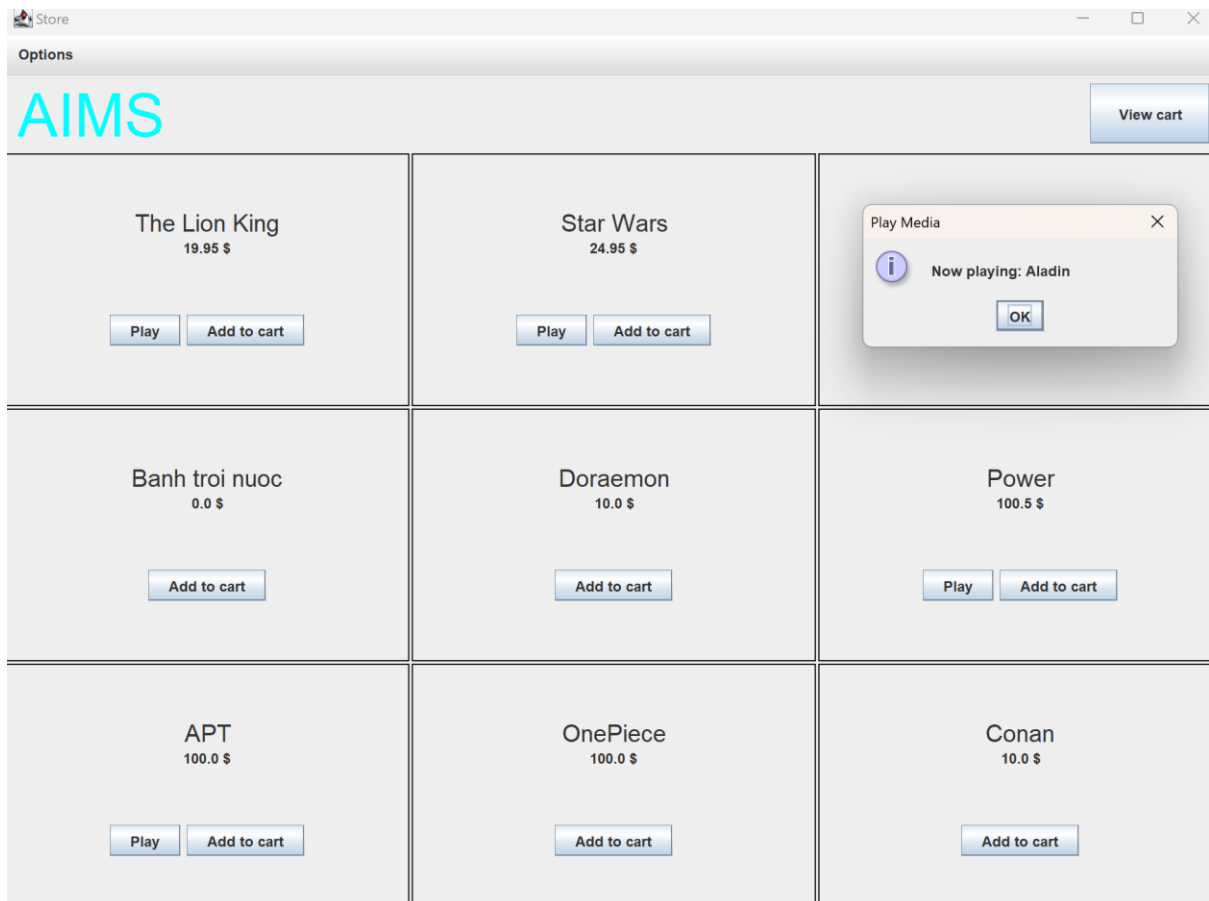
```
31             JButton btnPlay = new JButton("Play");
32             container.add(btnPlay);
33             btnPlay.addActionListener(e -> {
34                 // Xử lý phát Media
35                 System.out.println("Playing: " + media.getTitle());
36                 showPlayDialog(media.getTitle());
37             });
38             container.add(btnPlay);
39         }
40         // Add to cart button
41         JButton btnAddToCart = new JButton("Add to cart");
42         btnAddToCart.addActionListener(e -> {
43             // Xử lý thêm Media vào giỏ hàng
44             System.out.println("Added to cart: " + media.getTitle());
45             // Hiển thị hộp thoại thông báo
46             showAddToCartDialog(media.getTitle());
47         });
48         container.add(btnAddToCart);
49
50         this.add(Box.createVerticalGlue());
51         this.add(title);
52         this.add(cost);
53         this.add(Box.createVerticalGlue());
54         this.add(container);
55         this.setBorder(BorderFactory.createLineBorder(Color.BLACK));
56     }
```

```

57
58 // Hiển thị hộp thoại phát Media
59 private void showPlayDialog(String mediaTitle) {
60     javax.swing.JOptionPane.showMessageDialog(this, "Now playing: " + mediaTitle, "Play Media",
61         javax.swing.JOptionPane.INFORMATION_MESSAGE);
62 }
63 // Hiển thị hộp thoại thông báo đã thêm vào giỏ hàng
64 private void showAddToCartDialog(String mediaTitle) {
65     javax.swing.JOptionPane.showMessageDialog(
66         this,
67         "The item \"" + mediaTitle + "\" has been added to your cart!",
68         "Add to Cart",
69         javax.swing.JOptionPane.INFORMATION_MESSAGE
70     );
71 }
72 }
73

```





4. Painter

```

Painter.java
1 package hust.soict.dsai.javafx;
2
3 import javafx.application.Application;
4
5 public class Painter extends Application{
6     @Override
7     public void start (Stage stage) throws Exception {
8         Parent root = FXMLLoader.load(getClass().getResource("/hust/soict/dsai/javafx/Painter.fxml"));
9         Scene scene = new Scene(root);
10        stage.setTitle("Painter");
11        stage.setScene(scene);
12        stage.show();
13    }
14    public static void main(String[] args) {
15        launch(args);
16    }
17 }

```

```

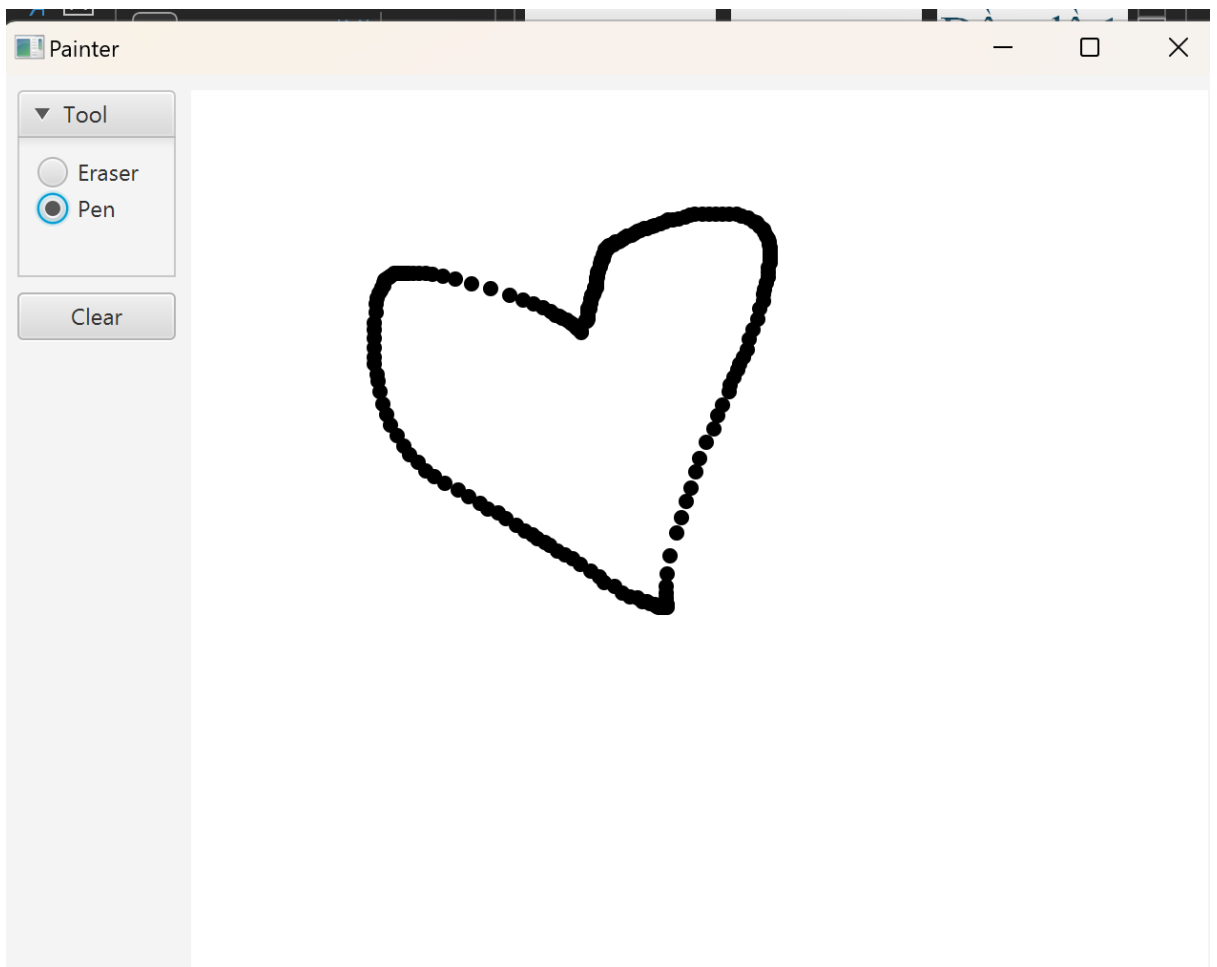
Painter.java *PainterController.java
1 package hust.soict.dsai.javafx;
2 import javafx.event.ActionEvent;
3
4
5
6
7
8
9 public class PainterController {
10     private boolean isEraser = false;
11
12     @FXML
13     private Pane drawingAreaPane;
14
15     @FXML
16     void clearButtonPressed(ActionEvent event) {
17         drawingAreaPane.getChildren().clear();
18     }
19     @FXML
20     void penSelected(ActionEvent event) {
21         isEraser = false; // Switch to Pen mode
22     }
23
24     @FXML
25     void eraserSelected(ActionEvent event) {
26         isEraser = true; // Switch to Eraser mode
27     }
28
29     @FXML
30     void drawingAreaMouseDragged(MouseEvent event) {
31         Color color = isEraser ? Color.WHITE : Color.BLACK;
32
33         Circle newCircle = new Circle(event.getX(), event.getY(), 4, color);
34         drawingAreaPane.getChildren().add(newCircle);
35     }
36 }

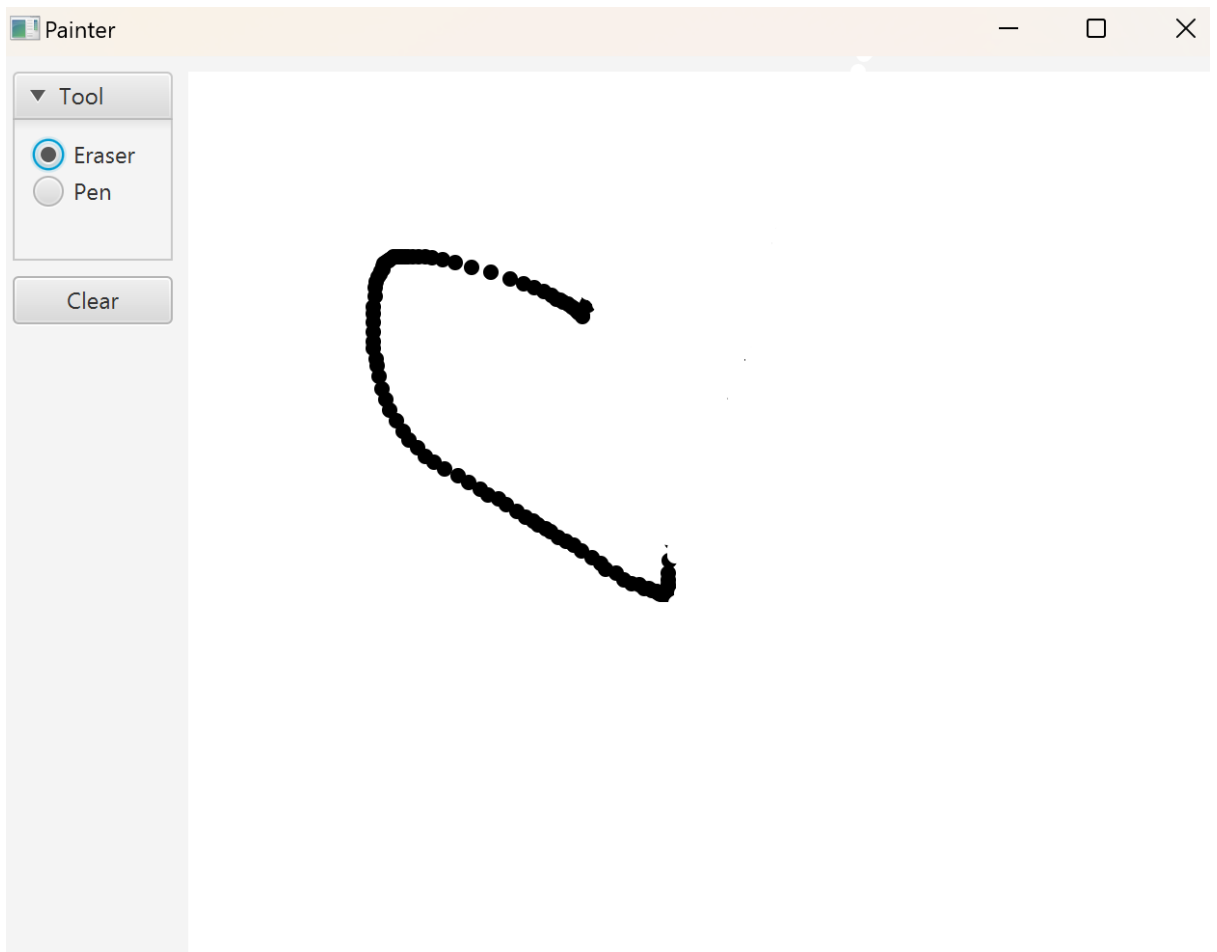
```

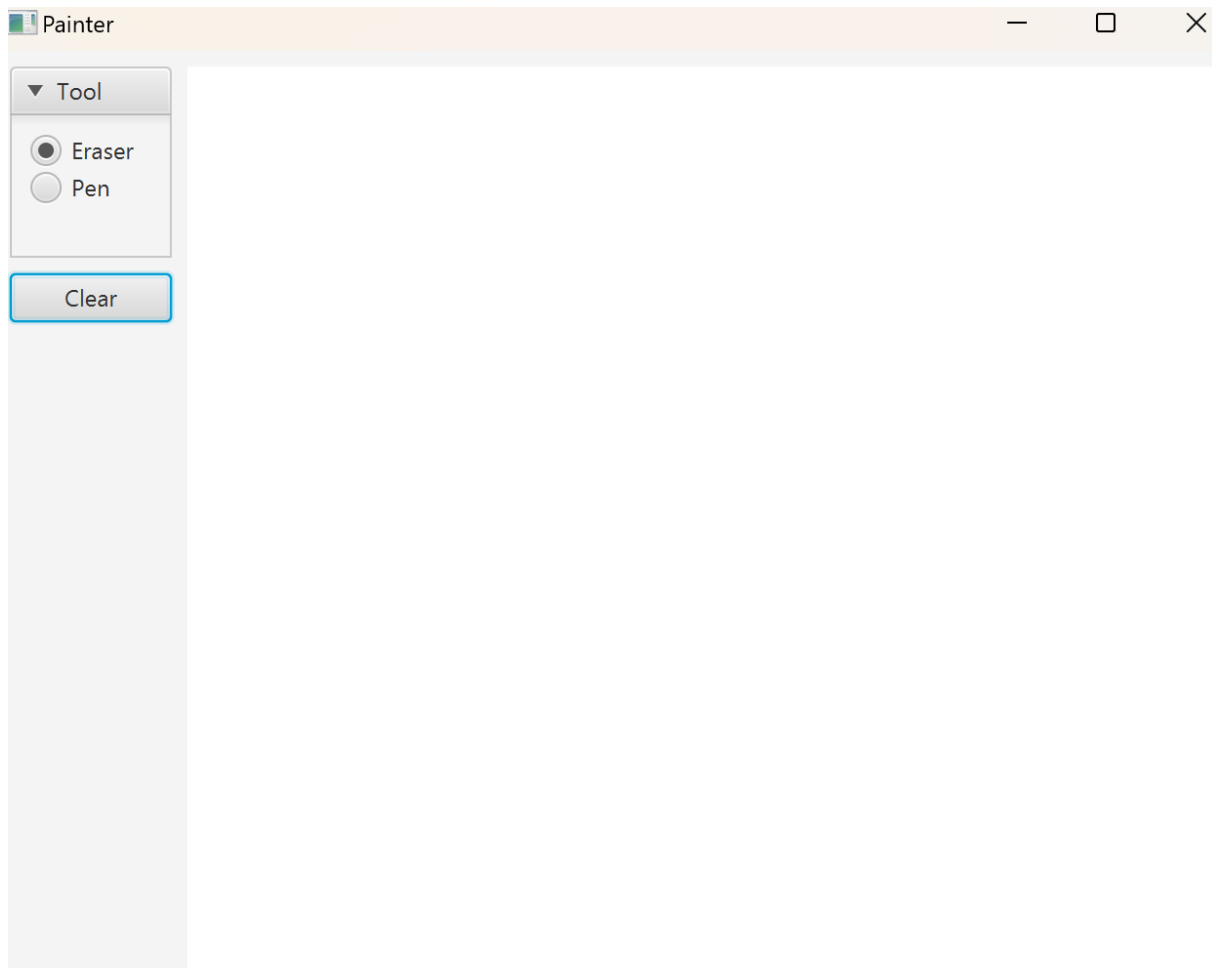
```

Painter.java *PainterController.java Painter.fxml
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <import javafx.geometry.Insets>
4 <import javafx.scene.control.Button>
5 <import javafx.scene.control.RadioButton>
6 <import javafx.scene.control.TitledPane>
7 <import javafx.scene.control.ToggleGroup>
8 <import javafx.scene.layout.BorderPane>
9 <import javafx.scene.layout.Pane>
10 <import javafx.scene.layout.VBox>
11
12 Bind to grammar/schema...
13 <BorderPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" minWidth="-Infinity" prefHeight="480.0" prefWidth="640.0">
14     <padding>
15         <Insets bottom="8.0" left="8.0" right="8.0" top="8.0" />
16     </padding>
17     <left>
18         <VBox maxHeight="1.7976931348623157E308" prefHeight="464.0" prefWidth="83.0" spacing="8.0" BorderPane.alignment="CENTER">
19             <BorderPane.margin>
20                 <Insets right="8.0" />
21             </BorderPane.margin>
22             <children>
23                 <TitledPane prefHeight="98.0" prefWidth="57.0" text="Tool">
24                     <content>
25                         <VBox prefHeight="200.0" prefWidth="100.0">
26                             <children>
27                                 <RadioButton mnemonicParsing="false" onAction="#eraserSelected" text="Eraser">
28                                     <toggleGroup>
29                                         <ToggleGroup fx:id="Tool" />
30                                     </toggleGroup>
31                                 </RadioButton>
32                                 <RadioButton mnemonicParsing="false" onAction="#penSelected" prefHeight="21.0" prefWidth="41.0" text="Pen">
33                                     <toggleGroup>
34                                         <ToggleGroup fx:id="Tool" />
35                                     </toggleGroup>
36                                 </RadioButton>
37                             </children>
38                         </VBox>
39                     </content>
40                 </TitledPane>
41                 <Button maxHeight="1.7976931348623157E308" mnemonicParsing="false" onAction="#clearButtonPressed" text="Clear" />
42             </children>
43         </VBox>
44     </left>
45     <right>
46         <Pane prefHeight="480.0" prefWidth="557.0">
47             <children>
48                 <DrawingArea />
49             </children>
50         </Pane>
51     </right>
52 </BorderPane>

```







5. Cart FX

```
1 package hust.soict.dsai.aims.screen;
2
3 import hust.soict.dsai.aims.cart.Cart;
4 import hust.soict.dsai.aims.media.Book;
5 import hust.soict.dsai.aims.media.CompactDisc;
6 import hust.soict.dsai.aims.media.DigitalVideoDisc;
7 import hust.soict.dsai.aims.media.Media;
8 import javafx.fxml.FXML;
9 import javafx.scene.control.TableColumn;
10 import javafx.scene.control.TableView;
11 import javafx.scene.control.cell.PropertyValueFactory;
12
13 public class CartScreenController {
14     private Cart cart;
15     @FXML
16     private TableView<Media> tblMedia;
17     @FXML
18     private TableColumn<Media, String> colMediaTitle;
19     @FXML
20     private TableColumn<Media, String> colMediacategory;
21     @FXML
22     private TableColumn<Media, Float> colMediaCost;
23     public CartScreenController (Cart cart) {
24         super();
25         this.cart = cart;
26     }
27     @FXML
28     private void initialize() {
29         colMediaTitle.setCellValueFactory(
30             new PropertyValueFactory<Media, String>("title"));
31         colMediacategory.setCellValueFactory(
32             new PropertyValueFactory<Media, String>("category"));
33         colMediaCost.setCellValueFactory(
34             new PropertyValueFactory<Media, Float>("cost"));
35         tblMedia.setItems(this.cart.getItemsOrdered());
36     }
37 }
```



```

19  @Override
20  public void start(Stage primaryStage) throws Exception {
21      try {
22          // Đường dẫn đến file FXML
23          FXMLLoader loader = new FXMLLoader(getClass().getResource("/hust/soict/dsai/aims/screen/Cart.fxml"));
24          // Gán controller
25          CartScreenController controller = new CartScreenController(cart);
26          loader.setController(controller);
27
28          // Load giao diện từ FXML
29          Parent root = loader.load();
30
31          // Thiết lập và hiển thị giao diện
32          primaryStage.setTitle("Cart");
33          primaryStage.setScene(new Scene(root));
34          primaryStage.show();
35      } catch (IOException e) {
36          e.printStackTrace();
37      }
38  }
39
40  public static void main(String[] args) {
41      // Khởi tạo dữ liệu giỏ hàng
42      cart = new Cart();
43      cart.addMedia(new DigitalVideoDisc("The Lion King", "Animation", "Roger Allers", 87, 19.95f));
44      cart.addMedia(new DigitalVideoDisc("Star Wars", "Science Fiction", "George Lucas", 87, 24.95f));
45      cart.addMedia(new DigitalVideoDisc("Aladin", "Animation", 18.99f));
46      cart.addMedia(new Book("Banh troi nuoc", "Ho Xuan Huong"));
47      cart.addMedia(new Book("Doraemon", "Anime", 10.00f));
48      cart.addMedia(new CompactDisc("Power"));
49
50      // Chạy ứng dụng JavaFX
51      Launch.launch(args);
52  }
53 }
54

```

```

1  <?xml version="1.0" encoding="UTF-8"?>
2
3  <?import javafx.geometry.Insets?>
4  <?import javafx.scene.control.Button?>
5  <?import javafx.scene.control.ButtonBar?>
6  <?import javafx.scene.control.Label?>
7  <?import javafx.scene.control.Menu?>
8  <?import javafx.scene.control.MenuBar?>
9  <?import javafx.scene.control.MenuItem?>
10 <?import javafx.scene.control.RadioButton?>
11 <?import javafx.scene.control.TableColumn?>
12 <?import javafx.scene.control.TableView?>
13 <?import javafx.scene.control.TextField?>
14 <?import javafx.scene.control.ToggleGroup?>
15 <?import javafx.scene.layout.BorderPane?>
16 <?import javafx.scene.layout.HBox?>
17 <?import javafx.scene.layout.VBox?>
18 <?import javafx.scene.text.Font?>
19
20  Bind to grammar/schema...
21 <BorderPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" minWidth="-Infinity" prefHeight="768.0" prefWidth="1024.0">
22     <top>
23         <VBox prefWidth="100.0" BorderPane.alignment="CENTER">
24             <children>
25                 <MenuBar prefWidth="1024.0">
26                     <menus>
27                         <Menu mnemonicParsing="false" text="Options">
28                             <items>
29                                 <Menu mnemonicParsing="false" text="Update Store">
30                                     <items>
31                                         <MenuItem mnemonicParsing="false" text="Add Book" />
32                                         <MenuItem mnemonicParsing="false" text="Add CD" />
33                                         <MenuItem mnemonicParsing="false" text="Add DVD" />
34                                     </items>
35                                 </Menu>
36                                 <MenuItem mnemonicParsing="false" text="View Store" />
37                             </items>
38                         </Menu>
39                     </menus>
40                 </MenuBar>
41             </children>
42         </VBox>
43     </top>
44 </BorderPane>

```

Cart

Options

CART

Filter:

☒ By ID ☐ By Title

Title	Category	Cost
The Lion King	Animation	19.95
Star Wars	Science Fiction	24.95
Aladin	Animation	18.99
Banh troi nuoc		0.0
Doraemon	Anime	10.0
Power		0.0

Total0\$

Play

Remove