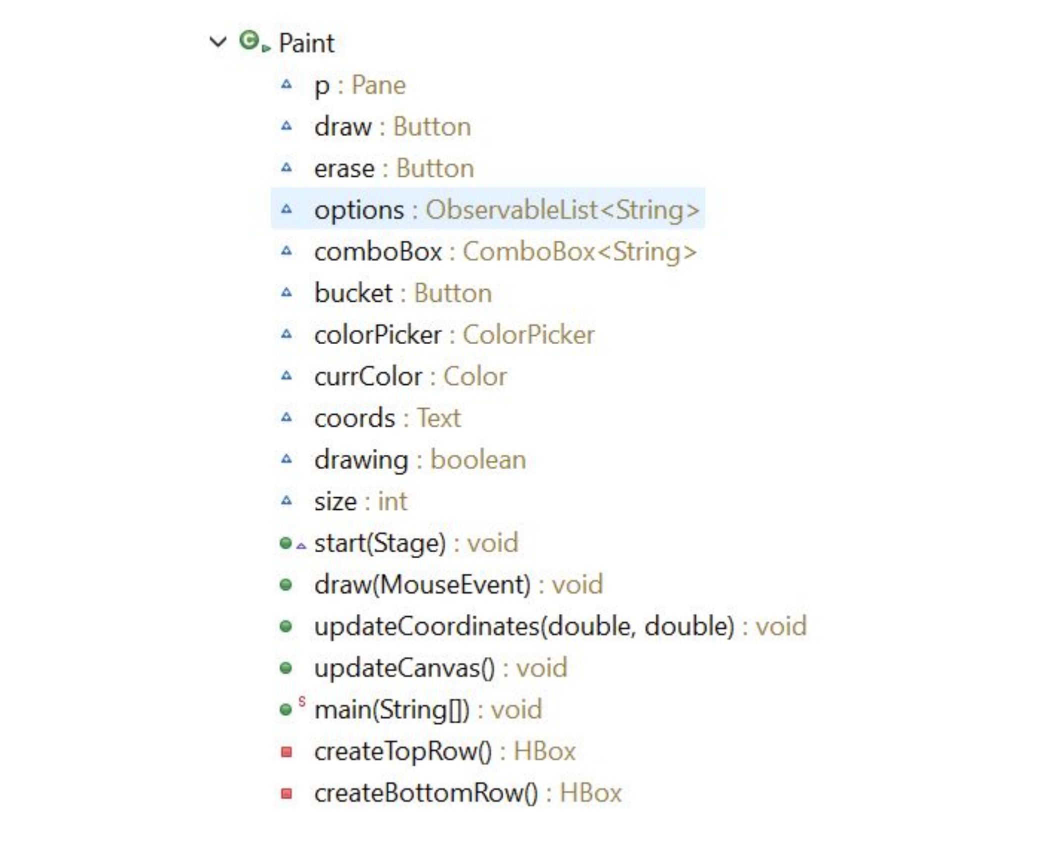
JavaFX Paint

Adam Spera - Brey Rivera

WHY PAINT?

Program Outline

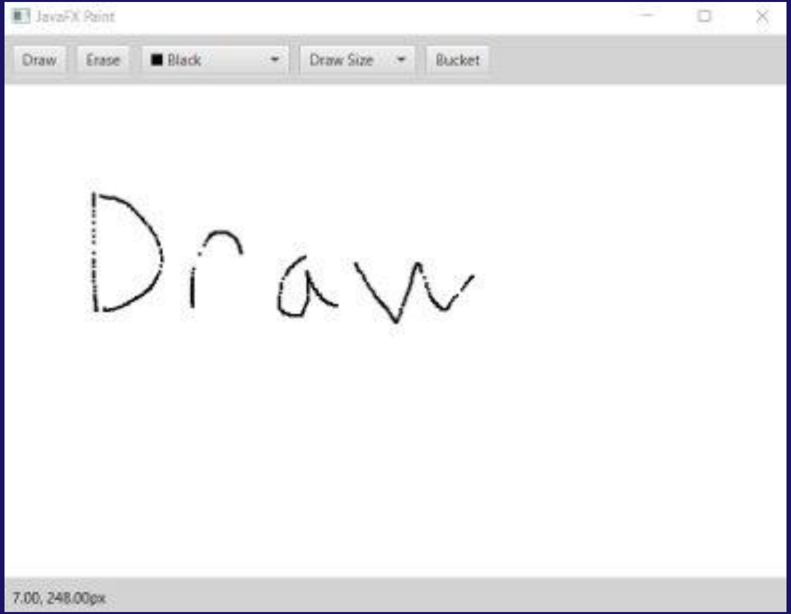


```
Paint
    p : Pane
    draw : Button
    erase : Button
    options : ObservableList<String>
    comboBox : ComboBox<String>
    bucket : Button
    colorPicker : ColorPicker
    currColor : Color
    coords : Text
    drawing : boolean
    size : int
    start(Stage) : void
    draw(MouseEvent) : void
    updateCoordinates(double, double) : void
    updateCanvas() : void
    main(String[]) : void
    createTopRow() : HBox
    createBottomRow() : HBox
```

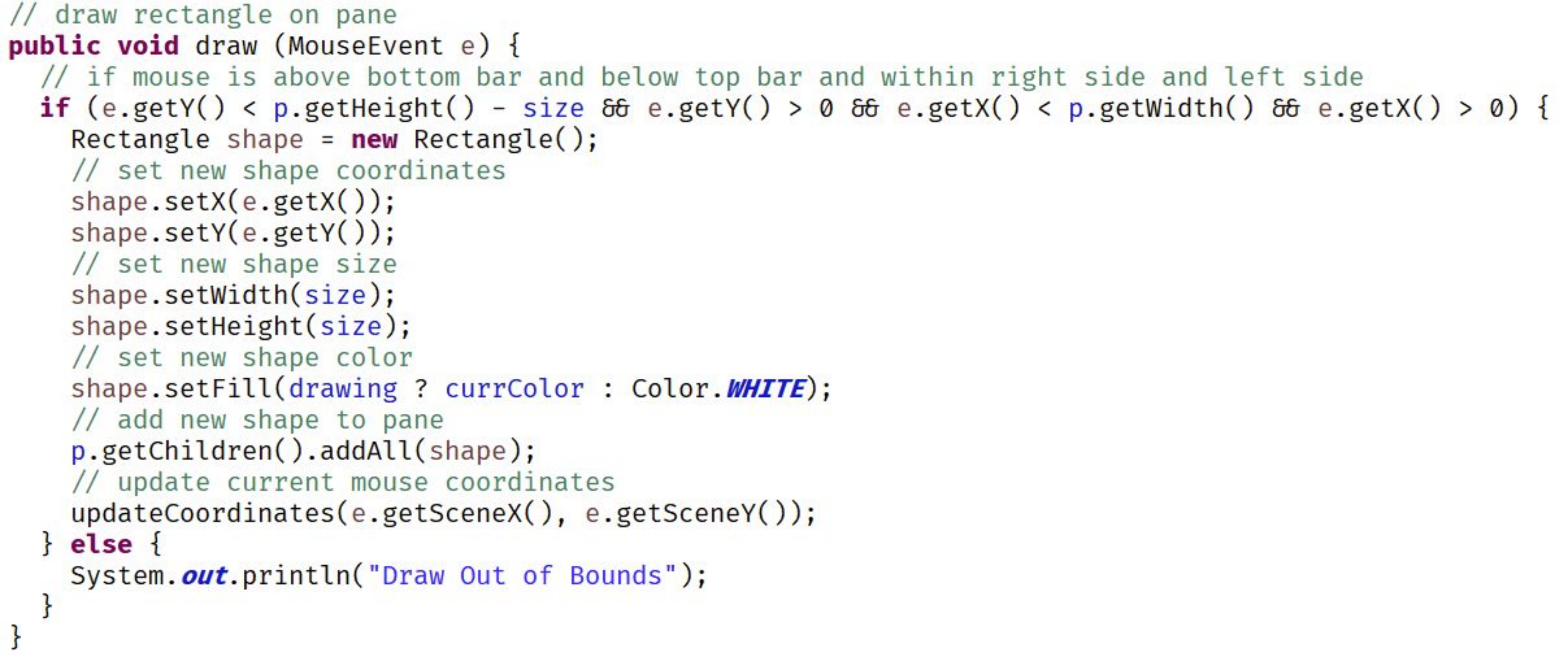
01

Functionality

Draw

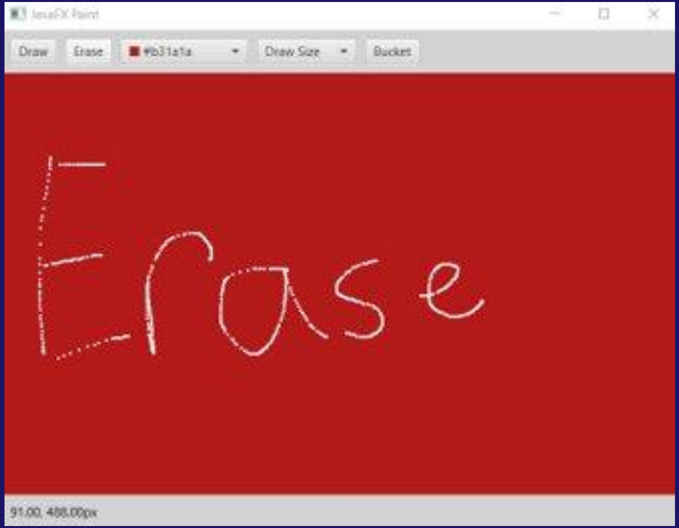

JavaFX Paint
Draw
Erase
■ Black
Draw Size
Bucket
Draw
7.00, 248.00px

Draw



```java
// draw rectangle on pane
public void draw (MouseEvent e) {
  // if mouse is above bottom bar and below top bar and within right side and left side
  if (e.getY() < p.getHeight() - size && e.getY() > 0 && e.getX() < p.getWidth() && e.getX() > 0) {
    Rectangle shape = new Rectangle();
    // set new shape coordinates
    shape.setX(e.getX());
    shape.setY(e.getY());
    // set new shape size
    shape.setWidth(size);
    shape.setHeight(size);
    // set new shape color
    shape.setFill(drawing ? currColor : Color.WHITE);
    // add new shape to pane
    p.getChildren().addAll(shape);
    // update current mouse coordinates
    updateCoordinates(e.getSceneX(), e.getSceneY());
  } else {
    System.out.println("Draw Out of Bounds");
  }
}
```

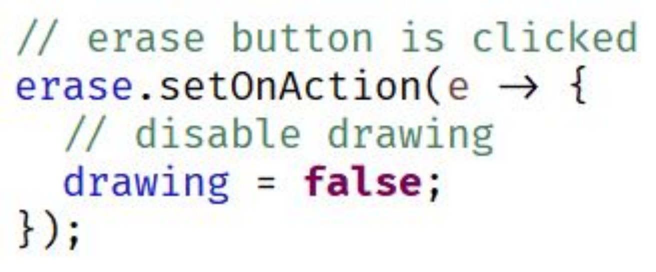
Erase



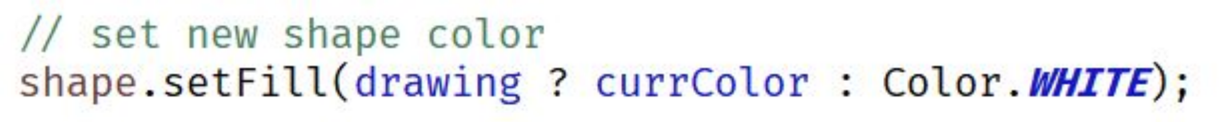
#b31a1a

91.00, 488.00px

Erase

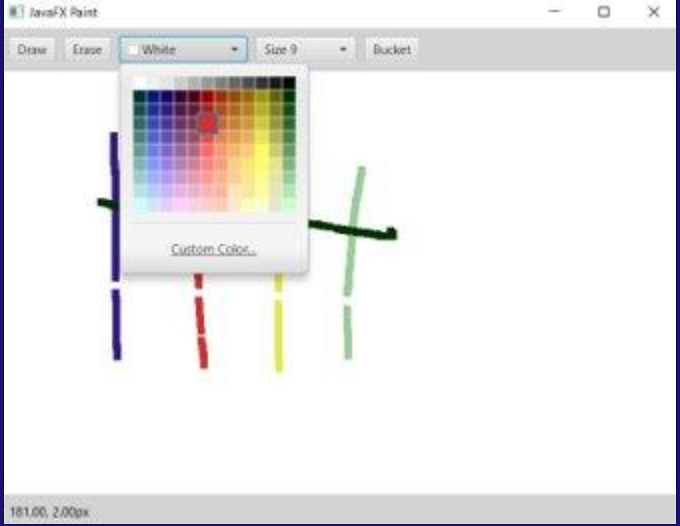


```java
// erase button is clicked
erase.setOnAction(e -> {
  // disable drawing
  drawing = false;
});
```

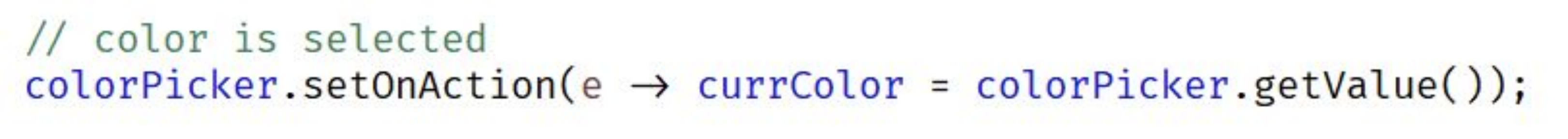


```java
// set new shape color
shape.setFill(drawing ? currColor : Color.WHITE);
```

Color Picker

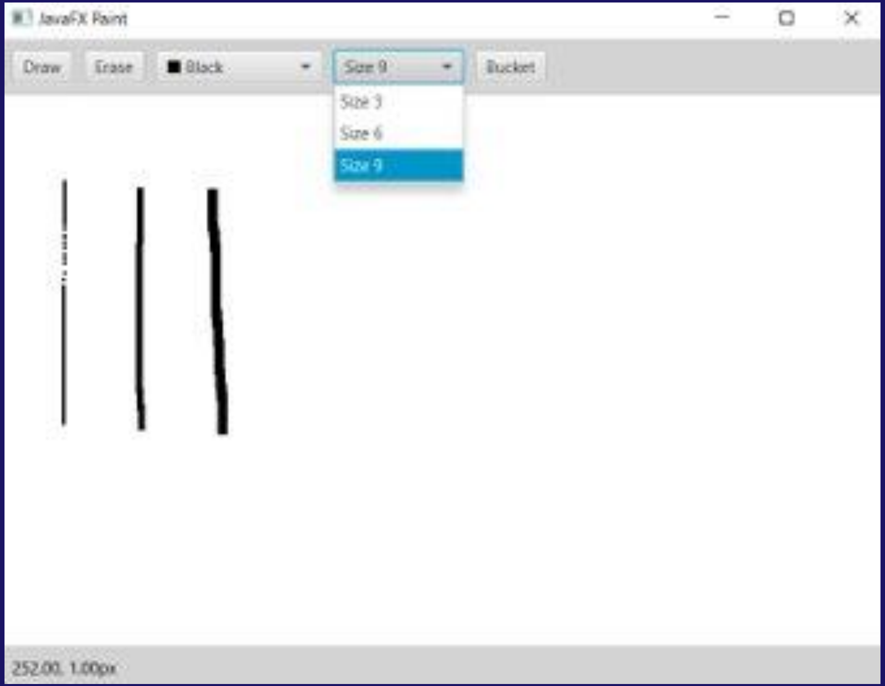


Color Picker

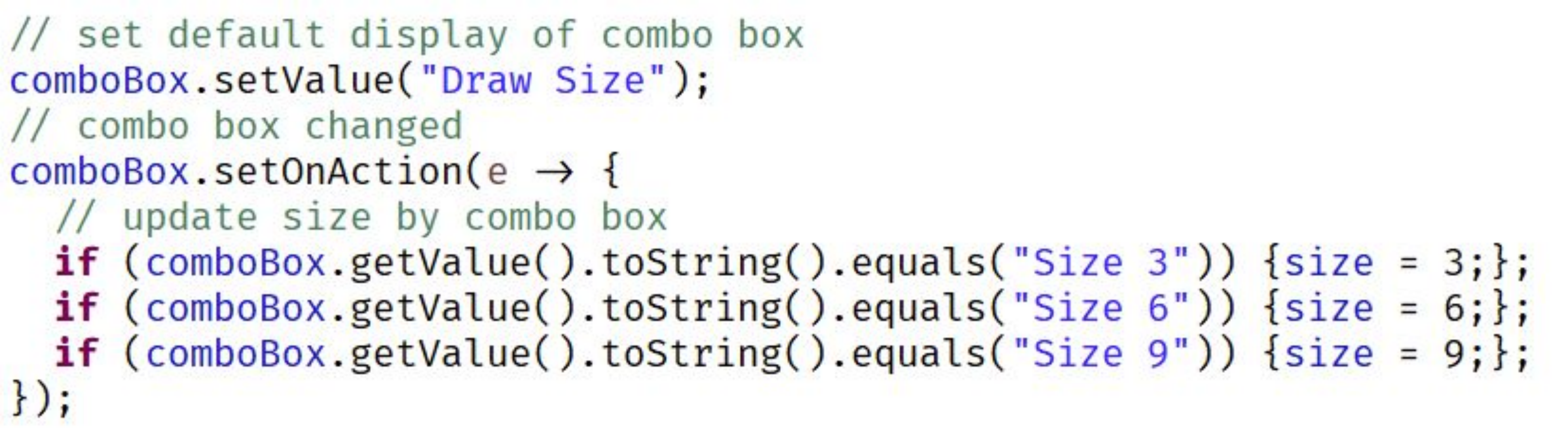


```
// color is selected
colorPicker.setOnAction(e → currColor = colorPicker.getValue());
```

Draw Size

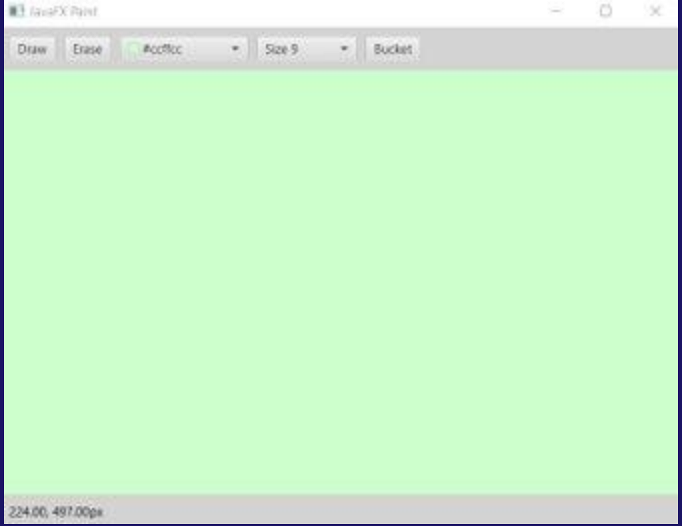

JavaFX Paint
Draw
Erase
■ Black
Size 9
Bucket
Size 3
Size 6
Size 9
252.00, 1.00px

Draw Size



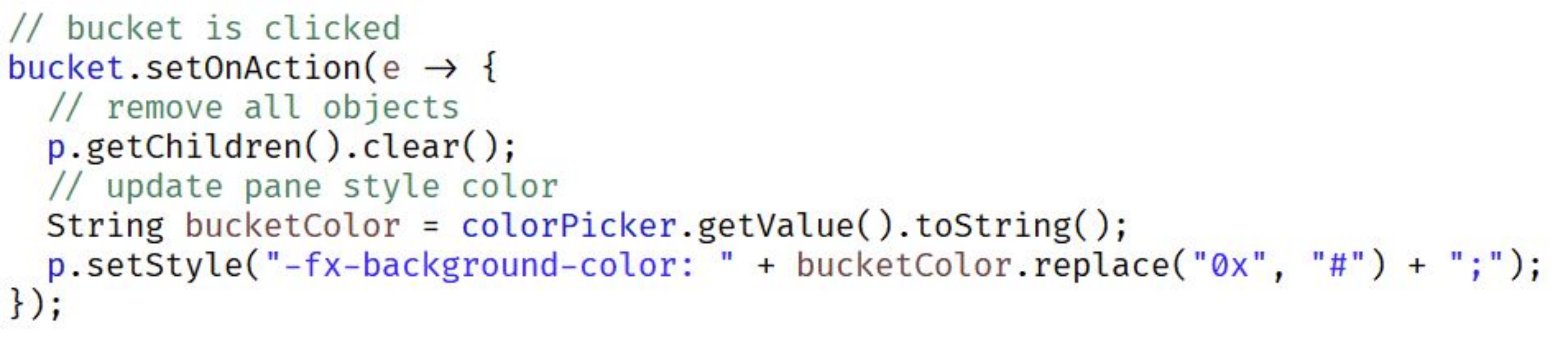
```
// set default display of combo box
comboBox.setValue("Draw Size");
// combo box changed
comboBox.setOnAction(e → {
  // update size by combo box
  if (comboBox.getValue().toString().equals("Size 3")) {size = 3;};
  if (comboBox.getValue().toString().equals("Size 6")) {size = 6;};
  if (comboBox.getValue().toString().equals("Size 9")) {size = 9;};
});
```

Bucket



224.00, 497.00px

Bucket

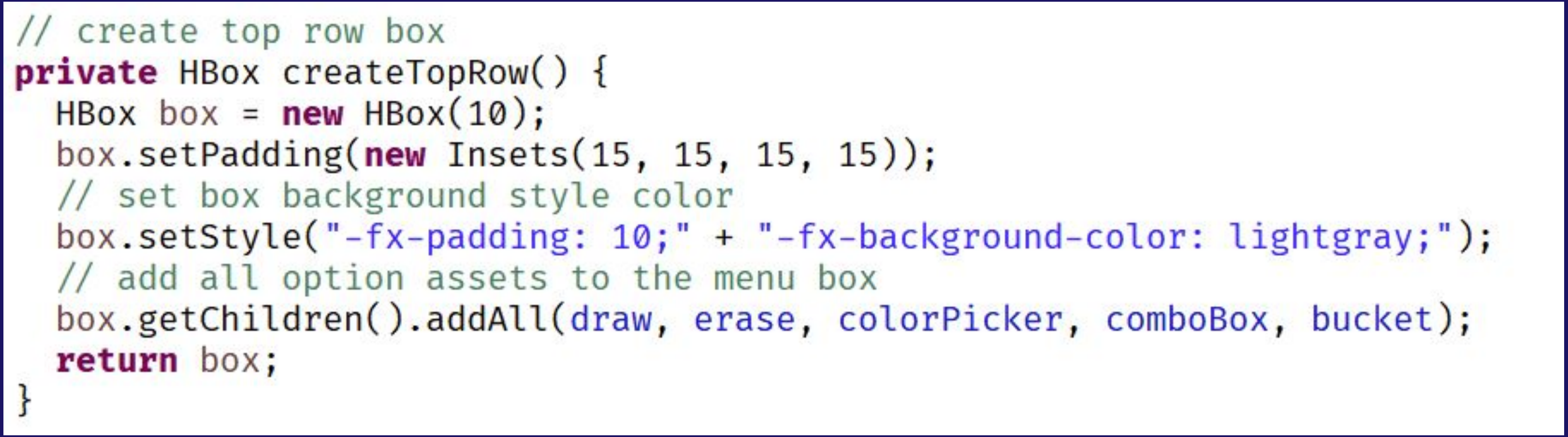


```
// bucket is clicked
bucket.setOnAction(e -> {
  // remove all objects
  p.getChildren().clear();
  // update pane style color
  String bucketColor = colorPicker.getValue().toString();
  p.setStyle("-fx-background-color: " + bucketColor.replace("0x", "#") + ";");
});
```

02

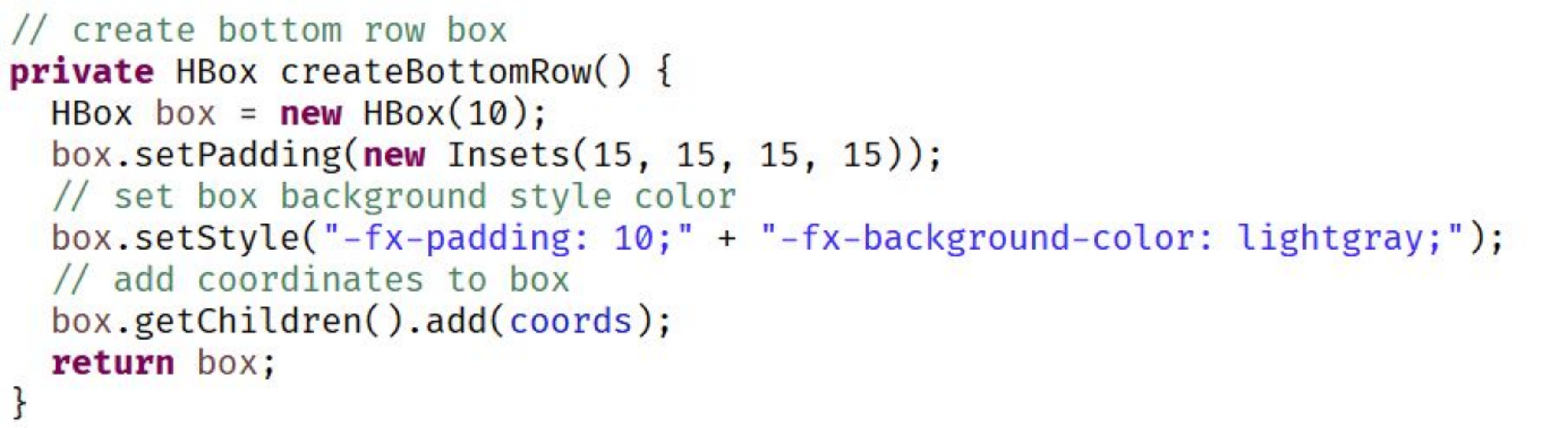
Methods

createTopRow()



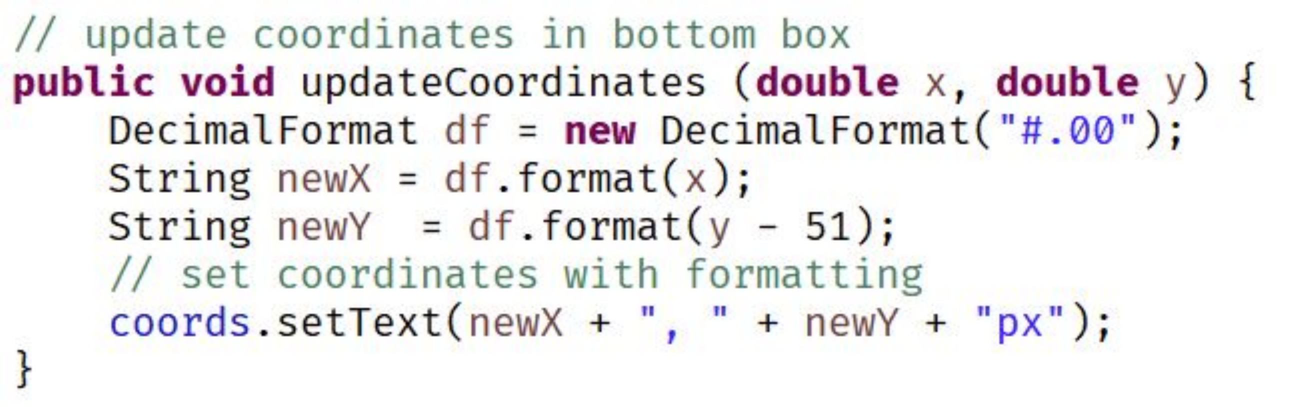
```java
// create top row box
private HBox createTopRow() {
  HBox box = new HBox(10);
  box.setPadding(new Insets(15, 15, 15, 15));
  // set box background style color
  box.setStyle("-fx-padding: 10;" + "-fx-background-color: lightgray;");
  // add all option assets to the menu box
  box.getChildren().addAll(draw, erase, colorPicker, comboBox, bucket);
  return box;
}
```

createBottomRow()



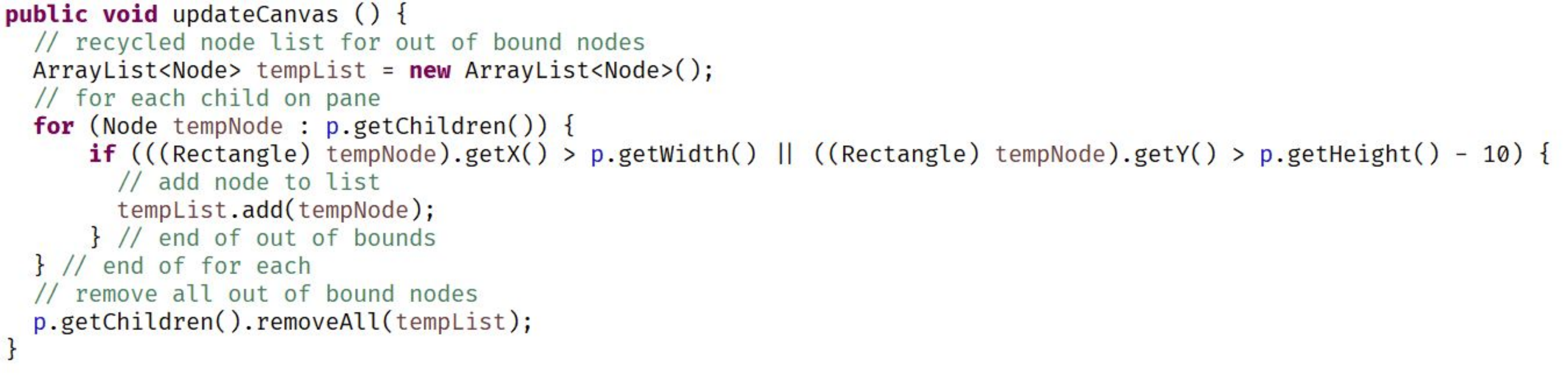
```
// create bottom row box
private HBox createBottomRow() {
  HBox box = new HBox(10);
  box.setPadding(new Insets(15, 15, 15, 15));
  // set box background style color
  box.setStyle("-fx-padding: 10;" + "-fx-background-color: lightgray;");
  // add coordinates to box
  box.getChildren().add(coords);
  return box;
}
```

updateCoordinates()



```java
// update coordinates in bottom box
public void updateCoordinates (double x, double y) {
    DecimalFormat df = new DecimalFormat("#.00");
    String newX = df.format(x);
    String newY  = df.format(y - 51);
    // set coordinates with formatting
    coords.setText(newX + ", " + newY + "px");
}
```

updateCanvas()

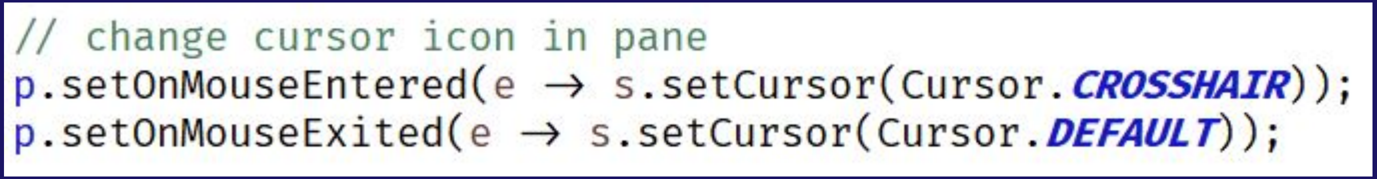


```java
public void updateCanvas () {
  // recycled node list for out of bound nodes
  ArrayList<Node> tempList = new ArrayList<Node>();
  // for each child on pane
  for (Node tempNode : p.getChildren()) {
      if (((Rectangle) tempNode).getX() > p.getWidth() || ((Rectangle) tempNode).getY() > p.getHeight() - 10) {
        // add node to list
        tempList.add(tempNode);
      } // end of out of bounds
  } // end of for each
  // remove all out of bound nodes
  p.getChildren().removeAll(tempList);
}
```

03

Cool Feature

Cursor



```
// change cursor icon in pane
p.setOnMouseEntered(e -> s.setCursor(Cursor.CROSSHAIR));
p.setOnMouseExited(e -> s.setCursor(Cursor.DEFAULT));
```

04

Bugs

Bugs

01

Out of Bounds Drawing

02

Coordinate Accuracy

03

Window Resizing

Future Improvements

Demo / Questions

Thank You