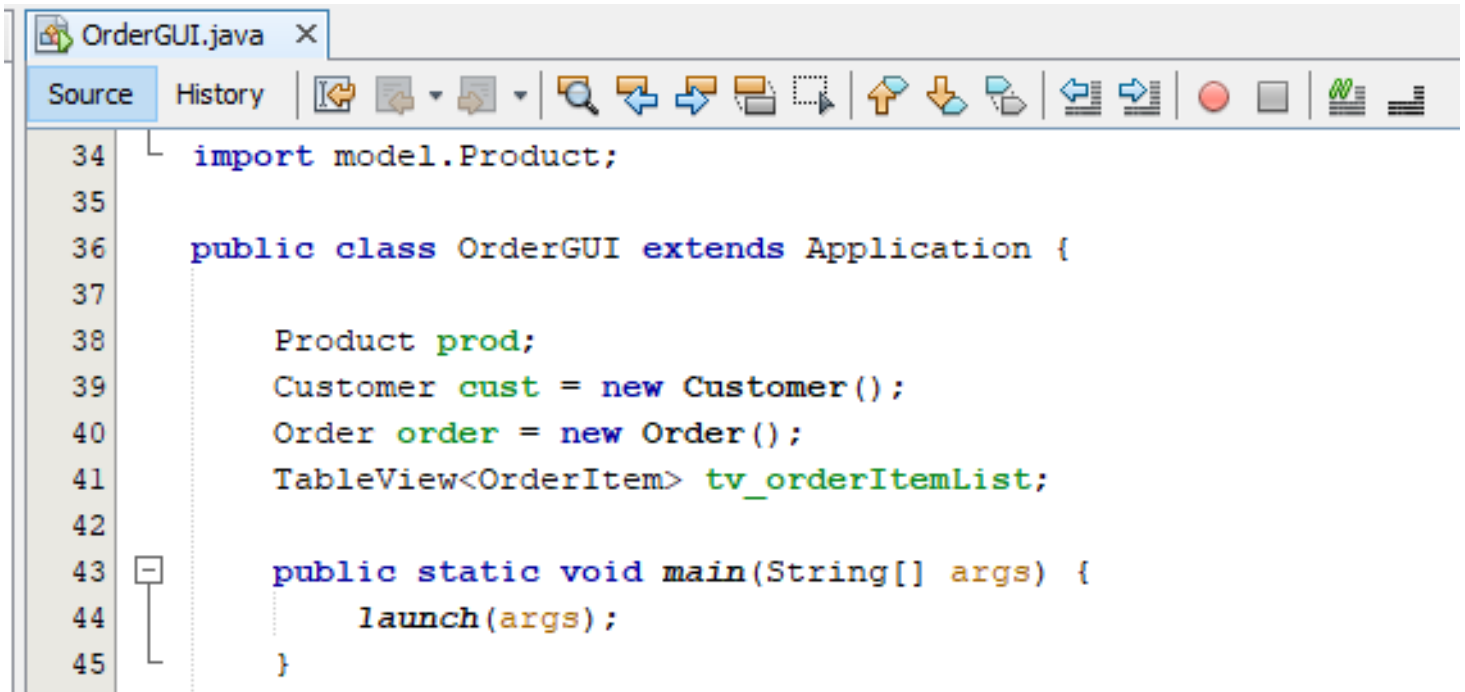




## GUI ADV



```
OrderGUI.java x
Source History
34 import model.Product;
35
36 public class OrderGUI extends Application {
37
38     Product prod;
39     Customer cust = new Customer();
40     Order order = new Order();
41     TableView<OrderItem> tv_orderItemList;
42
43     public static void main(String[] args) {
44         launch(args);
45     }
```

```
@Override
public void start(Stage primaryStage) throws Exception {

    Pane gridProduct = getProductPane();

    Pane gridCustomer = getCustomerPane();

    GridPane topPane = new GridPane();

    topPane.add(gridProduct, 0, 0);
    topPane.add(gridCustomer, 1, 0);

    tv_orderItemList = new TableView<>();
    drawTable(tv_orderItemList);

    Pane gridOrderDetail = getOrderDetailPane();

    VBox vbox = new VBox(topPane, tv_orderItemList, gridOrderDetail);
    vbox.setPadding(new Insets(10, 10, 10, 10));
    Scene scene = new Scene(vbox);

    primaryStage.setScene(scene);
    primaryStage.setTitle("Order detail");

    primaryStage.show();
}
```

## drawTable(tableView localTableView)

```
private void drawTable(tableView localTableView) {

    TableColumn<OrderItem, String> column0 = new TableColumn<>("ID");
    column0.setCellValueFactory(param -> new ReadOnlyStringWrapper(Integer.toString(param.getValue().getProd().getProduct_id())));

    TableColumn<OrderItem, String> column1 = new TableColumn<>("Product Name");
    column1.setPrefWidth(350);
    column1.setCellValueFactory(param -> new ReadOnlyStringWrapper(param.getValue().getProd().getProduct_name()));

    TableColumn<OrderItem, String> column2 = new TableColumn<>("Category");
    column2.setPrefWidth(250);
    column2.setCellValueFactory(param -> new ReadOnlyStringWrapper(param.getValue().getProd().getCategory().getCategory_name()));

    TableColumn<OrderItem, String> column3 = new TableColumn<>("Department");
    column3.setPrefWidth(250);
    column3.setCellValueFactory(param -> new ReadOnlyStringWrapper(param.getValue().getProd().getCategory().getDepartment().getDepartment_name()));

    TableColumn<OrderItem, String> column4 = new TableColumn<>("Quantity");
    column4.setCellValueFactory(new PropertyValueFactory<>("quantity"));

    TableColumn<OrderItem, String> column5 = new TableColumn<>("Price");
    column5.setCellValueFactory(new PropertyValueFactory<>("price"));

    TableColumn<OrderItem, String> column6 = new TableColumn<>("Total");
    column6.setCellValueFactory(param -> new ReadOnlyStringWrapper(Double.toString(param.getValue().getPrice()*param.getValue().getQuantity())));

    localTableView.getColumns().add(column0);
    localTableView.getColumns().add(column1);
    localTableView.getColumns().add(column2);
    localTableView.getColumns().add(column3);
    localTableView.getColumns().add(column4);
    localTableView.getColumns().add(column5);
    localTableView.getColumns().add(column6);

    TableView<OrderItem> selectionModel = localTableView.getSelectionModel();

    selectionModel.setSelectionMode(SelectionMode.SINGLE);

    ObservableList<OrderItem> selectedItems = selectionModel.getSelectedItems();
}
```

## GridPane First - Fourth Row

```
private Pane getProductPane() {
    GridPane gridProduct = new GridPane();
    gridProduct.setVgap(10);
    gridProduct.setHgap(10);
    gridProduct.setPadding(new Insets(10, 10, 10, 10));

    //first row
    Text txt_productId = new Text("Product Id");
    TextField tf_productId = new TextField();
    tf_productId.setMinWidth(20);

    Button btn_search = new Button("Search");

    gridProduct.add(txt_productId, 0, 0);
    gridProduct.add(tf_productId, 1, 0);
    gridProduct.add(btn_search, 2, 0);

    //second row
    Text txt_productName = new Text("Product");
    Label lb_product = new Label();
    lb_product.setTextAlignment(TextAlignment.JUSTIFY);
    lb_product.setMinHeight(30);
    lb_product.setMaxWidth(600);
    lb_product.setMinWidth(600);

    gridProduct.add(txt_productName, 0, 1);
    gridProduct.add(lb_product, 1, 1, 3, 1);

    //third row
    Text txt_quantity = new Text("Quantity");
    TextField tf_quantity = new TextField("1");
    Button btn_add = new Button("Add");

    gridProduct.add(txt_quantity, 0, 2);
    gridProduct.add(tf_quantity, 1, 2);
    gridProduct.add(btn_add, 2, 2);

    //fourth row
    Text txt_total = new Text("Order Total");
    Text txt_baht = new Text("Thai baht (THB)");
    Label lb_total = new Label();
    gridProduct.add(txt_total, 1, 3);
    gridProduct.add(lb_total, 2, 3);
    gridProduct.add(txt_baht, 3, 3);
}
```

## Event

```
OrderGUI.java x
Source History

169
170 //event + Function
171 btn_search.addEventFilter(MouseEvent.MOUSE_RELEASED, (MouseEvent e) -> {
172     if (isEmpty(tf_productId)) {
173         return;
174     }
175     prod = new ProductDaoImp().findById(Integer.parseInt(tf_productId.getText()));
176     lb_product.setText(prod.toString());
177 });
178
179 //event
180 tf_productId.addEventFilter(KeyEvent.KEY_RELEASED, new EventHandler<KeyEvent>() {
181     @Override
182     public void handle(KeyEvent ke) {
183         if (ke.getCode().equals(KeyCode.ENTER)) {
184             if (isEmpty(tf_productId)) {
185                 return;
186             }
187             prod = new ProductDaoImp().findById(Integer.parseInt(tf_productId.getText()));
188             lb_product.setText(prod.toString());
189         }
190     }
191 });

DecimalFormat f = new DecimalFormat("##.00");

btn_add.addEventFilter(MouseEvent.MOUSE_RELEASED, new EventHandler<MouseEvent>() {
    @Override
    public void handle(MouseEvent e) {
        if (prod == null || isEmpty(tf_quantity)) {
            return;
        }
        int quantity = Integer.parseInt(tf_quantity.getText());
        OrderItem item = new OrderItem(
            0,
            prod,
            quantity,
            prod.getProduct_price()
        );
        tv_orderItemList.getItems().add(item);
        order.addItem(item);
        lb_total.setText(f.format(order.getTotlePrice()));

        prod = null;
        tf_productId.clear();
        lb_product.setText("");
        tf_quantity.setText(1 + "");
        tf_productId.requestFocus();
    }
});
return gridProduct;
}
```

## getOrderDetailPane()

```
private Pane getOrderDetailPane() {
    GridPane gridOrderDetail = new GridPane();
    gridOrderDetail.setVgap(10);
    gridOrderDetail.setHgap(10);
    gridOrderDetail.setPadding(new Insets(10, 10, 10, 10));

    ComboBox<Order.OrderStatus> cbx_Status = new ComboBox<>();
    cbx_Status.getItems().setAll(Order.OrderStatus.values());
    cbx_Status.getSelectionModel().select(2);
    Button btn_save = new Button("Save");

    gridOrderDetail.add(cbx_Status, 0, 0);
    gridOrderDetail.add(btn_save, 1, 0);

    //event
    btn_save.addEventFilter(MouseEvent.MOUSE_RELEASED, new EventHandler<MouseEvent>() {
        @Override
        public void handle(MouseEvent e) {
            order.setCustomer(cust);
            order.setOrder_status(cbx_Status.getValue());
            order.setOrder_date(new Date());
            new OrderDaoImp().insert(order);
            tv_orderItemList.getItems().clear();
        }
    });
    return gridOrderDetail;
}
```

## getCustomerPane()

```
private Pane getCustomerPane() {
    GridPane gridCustomer = new GridPane();
    gridCustomer.setVgap(10);
    gridCustomer.setHgap(10);
    gridCustomer.setPadding(new Insets(10, 10, 10, 10));

    //first row
    Text txt_custId = new Text("Customer Id");
    TextField tf_custId = new TextField();
    tf_custId.setMinWidth(20);
    Button btn_search = new Button("Search");
    Button btn_clear = new Button("Clear");

    gridCustomer.add(txt_custId, 0, 0);
    gridCustomer.add(tf_custId, 1, 0);
    gridCustomer.add(btn_search, 2, 0);
    gridCustomer.add(btn_clear, 3, 0);

    //second row
    Text txt_custName = new Text("Customer Name");
    Label lb_custName = new Label();
    lb_custName.setTextAlignment(TextAlignment.JUSTIFY);
    lb_custName.setMinHeight(30);
    lb_custName.setMaxWidth(600);
    lb_custName.setMinWidth(600);

    gridCustomer.add(txt_custName, 0, 1);
    gridCustomer.add(lb_custName, 1, 1, 3, 1);
}
```

## getCustomerPane() (Event)

```
//event
btn_search.addEventFilter(MouseEvent.MOUSE_RELEASED, new EventHandler<MouseEvent>() {
    @Override
    public void handle(MouseEvent e) {
        if (isEmpty(tf_custId)) {
            return;
        }

        cust = new CustomerDaoImp().findById(Integer.parseInt(tf_custId.getText()));
        lb_custName.setText(cust.toString());
    }
});

btn_clear.addEventFilter(MouseEvent.MOUSE_RELEASED, new EventHandler<MouseEvent>() {
    @Override
    public void handle(MouseEvent e) {
        cust = new Customer();
        lb_custName.setText("");
        tf_custId.clear();
    }
});
return gridCustomer;
}
```

## isEmpty(TextField tf)

```
private boolean isEmpty(TextField tf) {
    if (tf.getText() == null || tf.getText().trim().isEmpty()) {
        return true;
    }
    return false;
}
```