PROJECT

JAVA FX IN 4 DAYS

DAY 2 - HOW POWERFUL

DATE	5/31	CLIENT	ZAW MIN LWIN

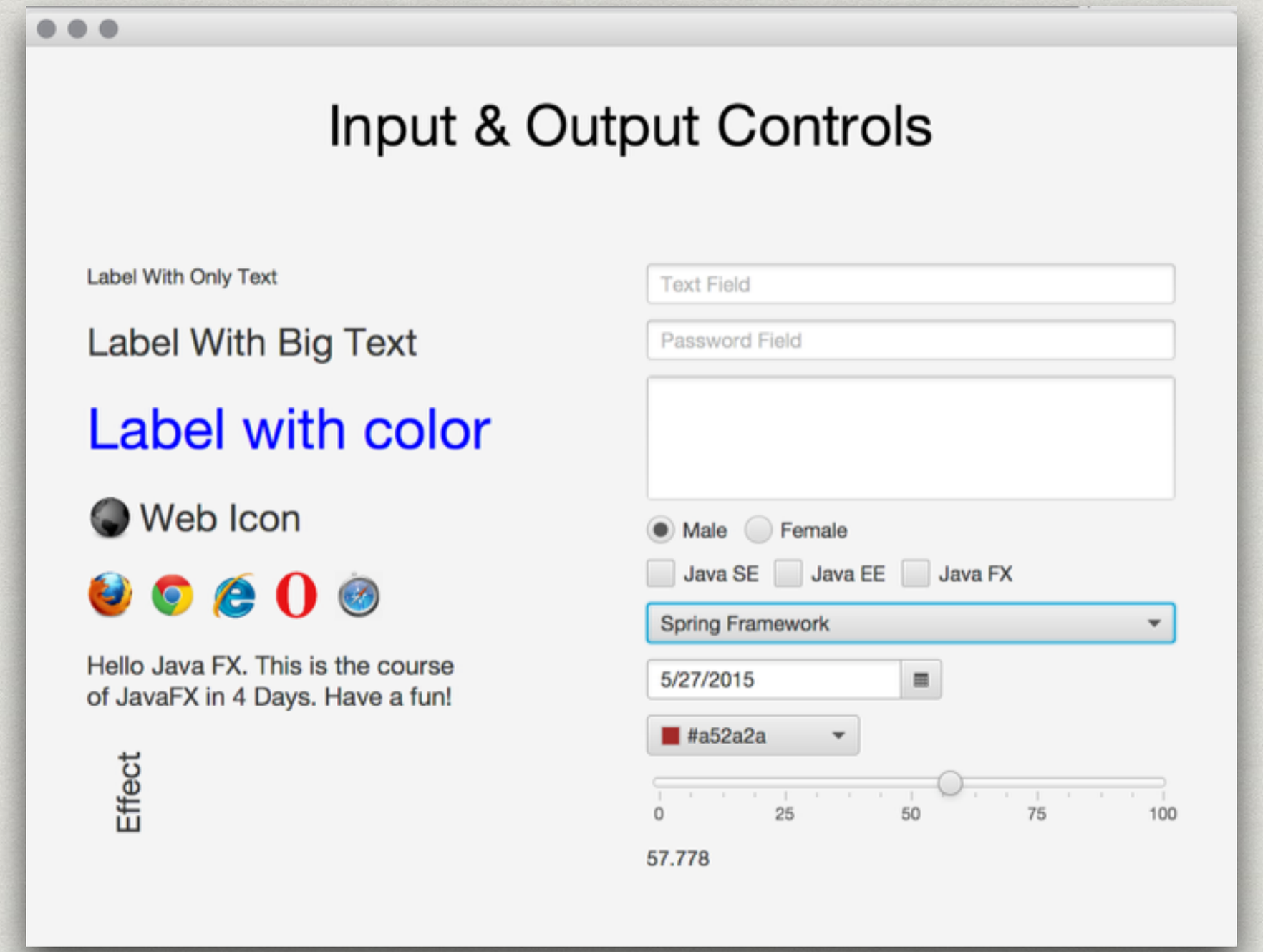
2.1
UI CONTROLS

JAVA FX IN 4 DAYS

Contents

- Inputs and Outputs Controls
- Controls using Collections

Inputs and Outputs



Label

```
VBox box = new VBox(20);
box.setPrefWidth(340);

// label
Label label1 = new Label("Label With Only Text");

Label label2 = new Label("Label With Big Text");
label2.setFont(new Font(24));

Label label3 = new Label("Label with color");
label3.setStyle("-fx-text-fill:blue;-fx-font-size:36;");
box.getChildren().addAll(label1, label2, label3);
```

Label with Image

```
// label with graphic and text
Label iconText = new Label();
ImageView imgView = new ImageView(
   new Image(getClass()
      .getResourceAsStream("Web-b-icon.png")));
imgView.setPreserveRatio(true);
imgView.setFitWidth(30);

iconText.setGraphic(imgView);
iconText.setText("Web Icon");
iconText.setFont(Font.font(24));
```

Wrapping Text in Label

```
// label get text wrapping
Label wrap = new Label();
wrap.setFont(Font.font(16));
wrap.setPrefWidth(240);
wrap.setWrapText(true);
wrap.setText("Hello Java FX. "
     +"This is the course of JavaFX "
     +"in 4 Days. Have a fun!");
box.getChildren().add(wrap);
```

Label with Effect

```
// label with effect
Label effect = new Label("Effect");
effect.setFont(Font.font(20));
effect.setRotate(270);
effect.setTranslateY(20);
box.getChildren().add(effect);
```

Text Inputs

```
// text field
TextField textField = new TextField();
textField.setPromptText("Text Field");
box.getChildren().add(textField);

// password field
PasswordField pass = new PasswordField();
pass.setPromptText("Password Field");
box.getChildren().add(pass);

// text area
TextArea textArea = new TextArea();
textArea.setPrefHeight(80);
box.getChildren().add(textArea);
```

Radio Buttons

```
// radio button
HBox radioGroup = new HBox(10);
ToggleGroup group = new ToggleGroup();

RadioButton male = new RadioButton("Male");
RadioButton female = new RadioButton("Female");
male.setSelected(true);
male.setToggleGroup(group);
female.setToggleGroup(group);

radioGroup.getChildren().addAll(male, female);

box.getChildren().add(radioGroup);
```

Check Box

```
// check box
HBox checkBoxes = new HBox(10);
checkBoxes.getChildren().addAll(FXCollections
   .observableArrayList(
      new CheckBox("Java SE"),
      new CheckBox("Java EE"),
      new CheckBox("Java FX")));
box.getChildren().add(checkBoxes);
```

Choice Box

```
// choice box
ChoiceBox<Object> choice = new ChoiceBox<>();
choice.setItems(FXCollections.observableArrayList(
        "Java SE",
        "JavaFX",
        new Separator(),
        "Java EE",
        "Spring Framework",
        new Separator(),
        "Android",
        new Separator(),
        "Java Developer Course for Japan"
        ));
choice.setPrefWidth(340);

box.getChildren().add(choice);
```

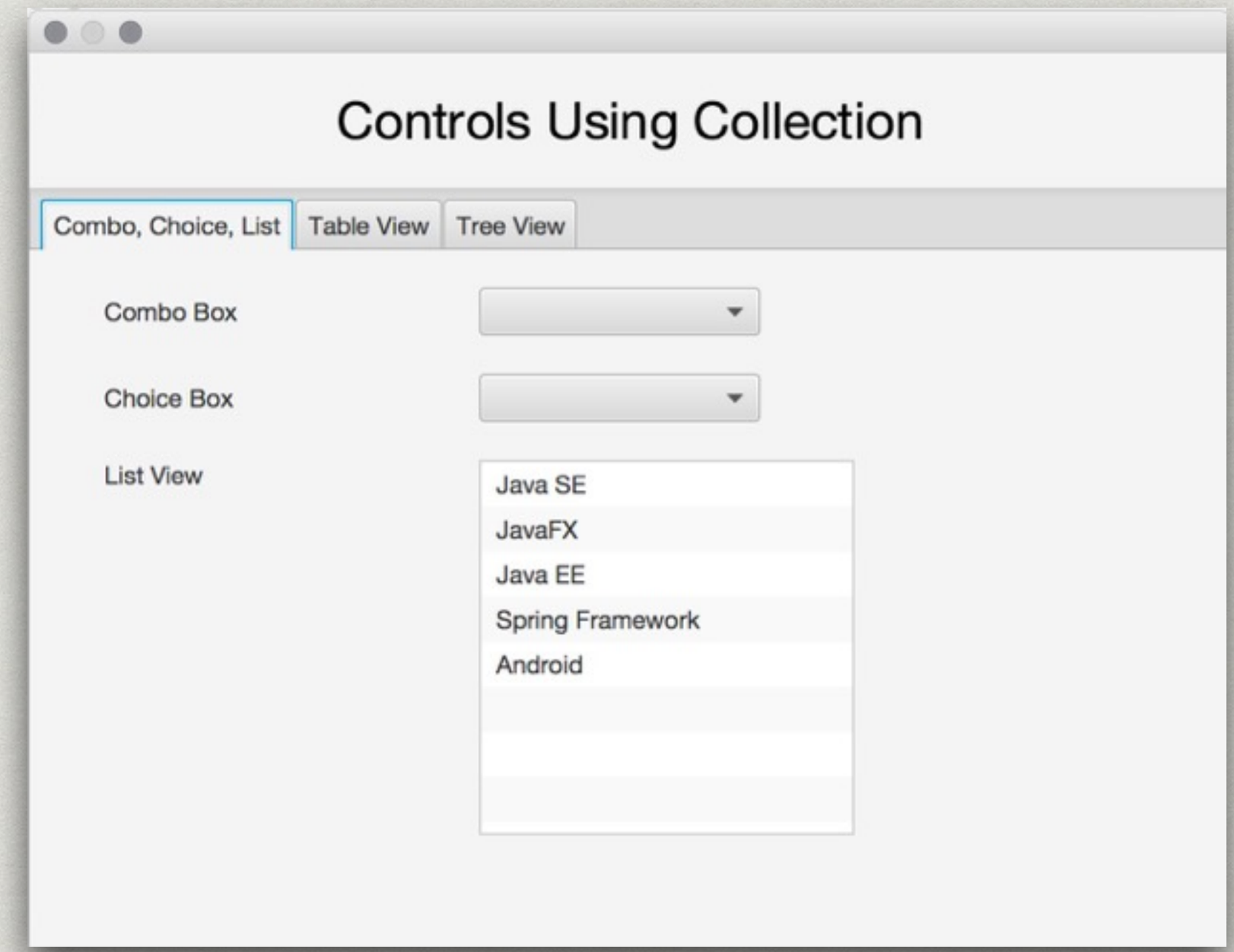
Slider

```
// slider
Slider slider = new Slider(0, 100, 30);
slider.setMajorTickUnit(25);
slider.setShowTickLabels(true);
slider.setShowTickMarks(true);
slider.setBlockIncrement(5);

box.getChildren().add(slider);

Label result = new Label();
Bindings.bindBidirectional(
      result.textProperty(),
      slider.valueProperty(),
      new NumberStringConverter());
box.getChildren().add(result);
```

Controls using Collections

- ChoiceBox
- ComboBox
- ListView
- TableView
- TreeView

Choice, Combo, List



Combo, Choice & List

```
@FXML
private ComboBox<String> combo;
@FXML
private ChoiceBox<Object> choice;
@FXML
private ListView<Course> listView;
```

Adding Data to Control

```
// combo, choice, list view
combo.getItems().addAll(getListString());
choice.getItems().addAll(getListString());
listView.getItems().addAll(Sample.getCourses());
listView.setCellFactory(CourseCell::new);
```

Custom List Cell

```
public class CourseCell extends ListCell<Course> {

    public CourseCell(ListView<Course> param) {
    }

    @Override
    protected void updateItem(Course item, boolean empty) {
        super.updateItem(item, empty);

        if (!empty) {
            setText(item.getName());
        }
    }

}
```

Table View

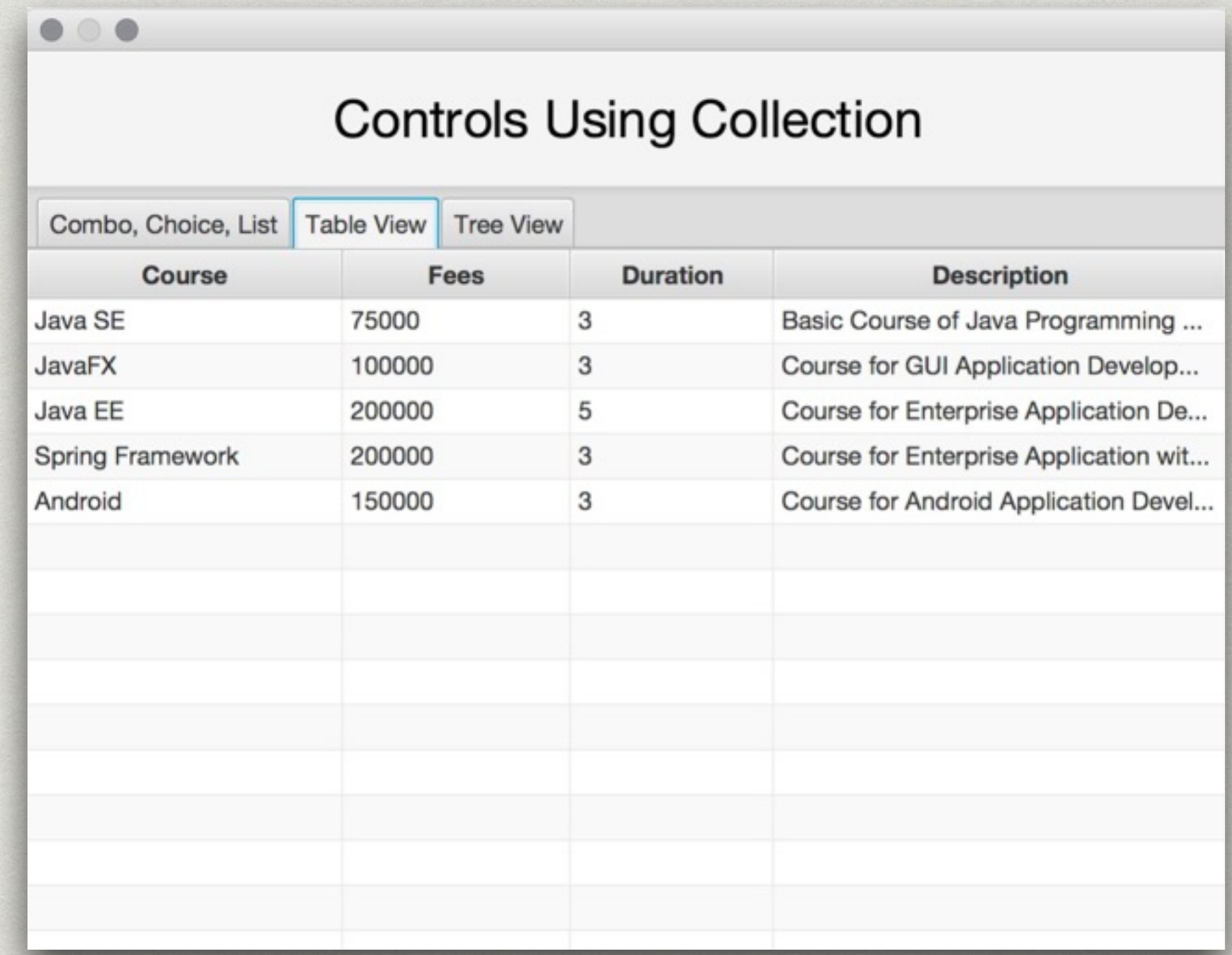

Controls Using Collection
Combo, Choice, List
Table View
Tree View
Course
Fees
Duration
Description
Java SE
75000
3
Basic Course of Java Programming ...
JavaFX
100000
3
Course for GUI Application Develop...
Java EE
200000
5
Course for Enterprise Application De...
Spring Framework
200000
3
Course for Enterprise Application wit...
Android
150000
3
Course for Android Application Devel...

Table View

```
@FXML
private TableView<Course> table;
@FXML
private TableColumn<Course, String> colName;
@FXML
private TableColumn<Course, String> colPrice;
@FXML
private TableColumn<Course, String> colDuration;
@FXML
private TableColumn<Course, String> colDescription;
```

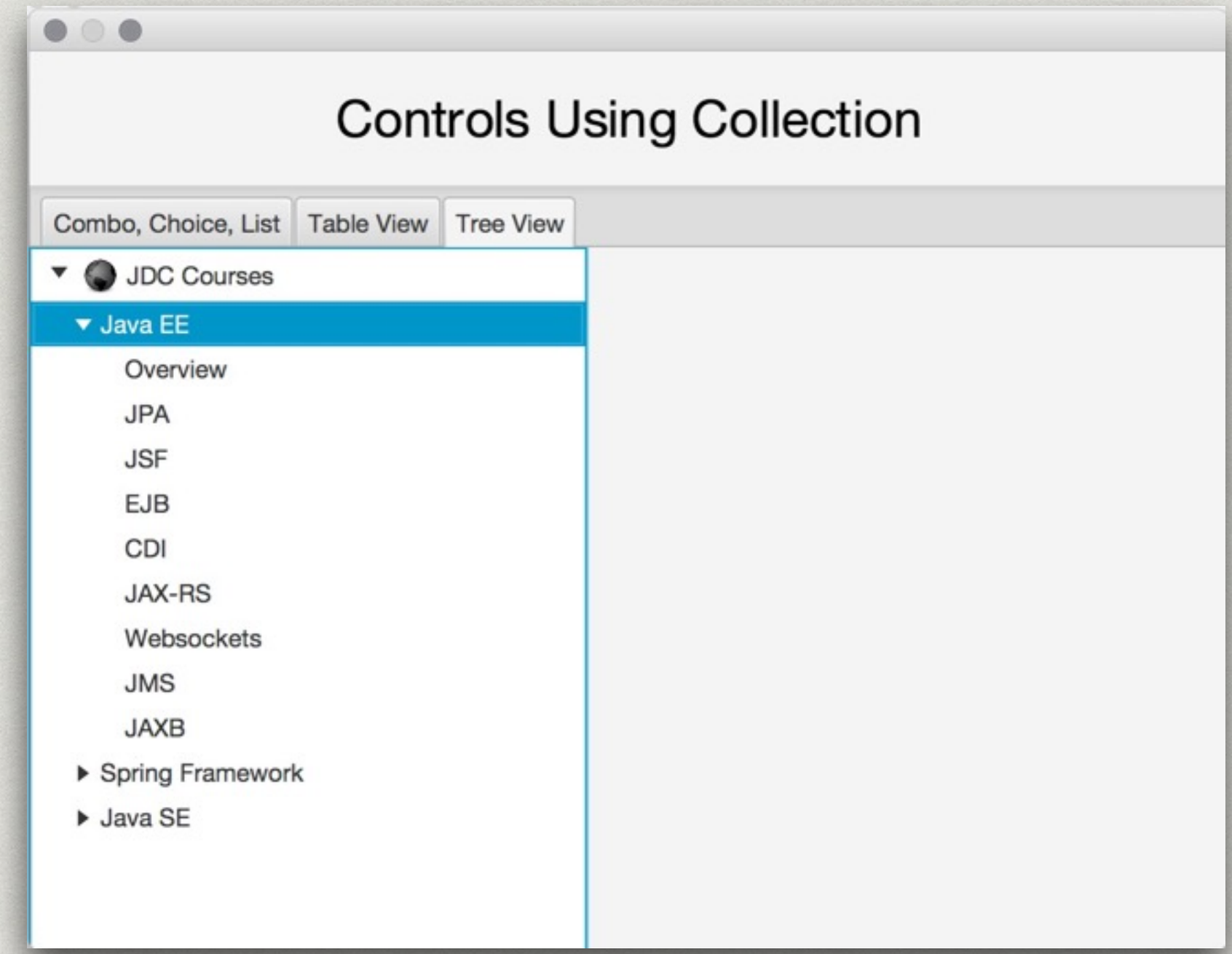
Setting Factory for Column Cell

```
// table view
colName.setCellValueFactory(
     new PropertyValueFactory<>("name"));
colPrice.setCellValueFactory(
     new PropertyValueFactory<>("price"));
colDuration.setCellValueFactory(
     new PropertyValueFactory<>("duration"));
colDescription.setCellValueFactory(
     new PropertyValueFactory<>("description"));
```

Adding Data to Table

```
table.getItems().clear();
table.getItems().addAll(Sample.getCourses());
```

TreeView


Controls Using Collection
Combo, Choice, List
Table View
Tree View
JDC Courses
Java EE
Overview
JPA
JSF
EJB
CDI
JAX-RS
Websockets
JMS
JAXB
Spring Framework
Java SE

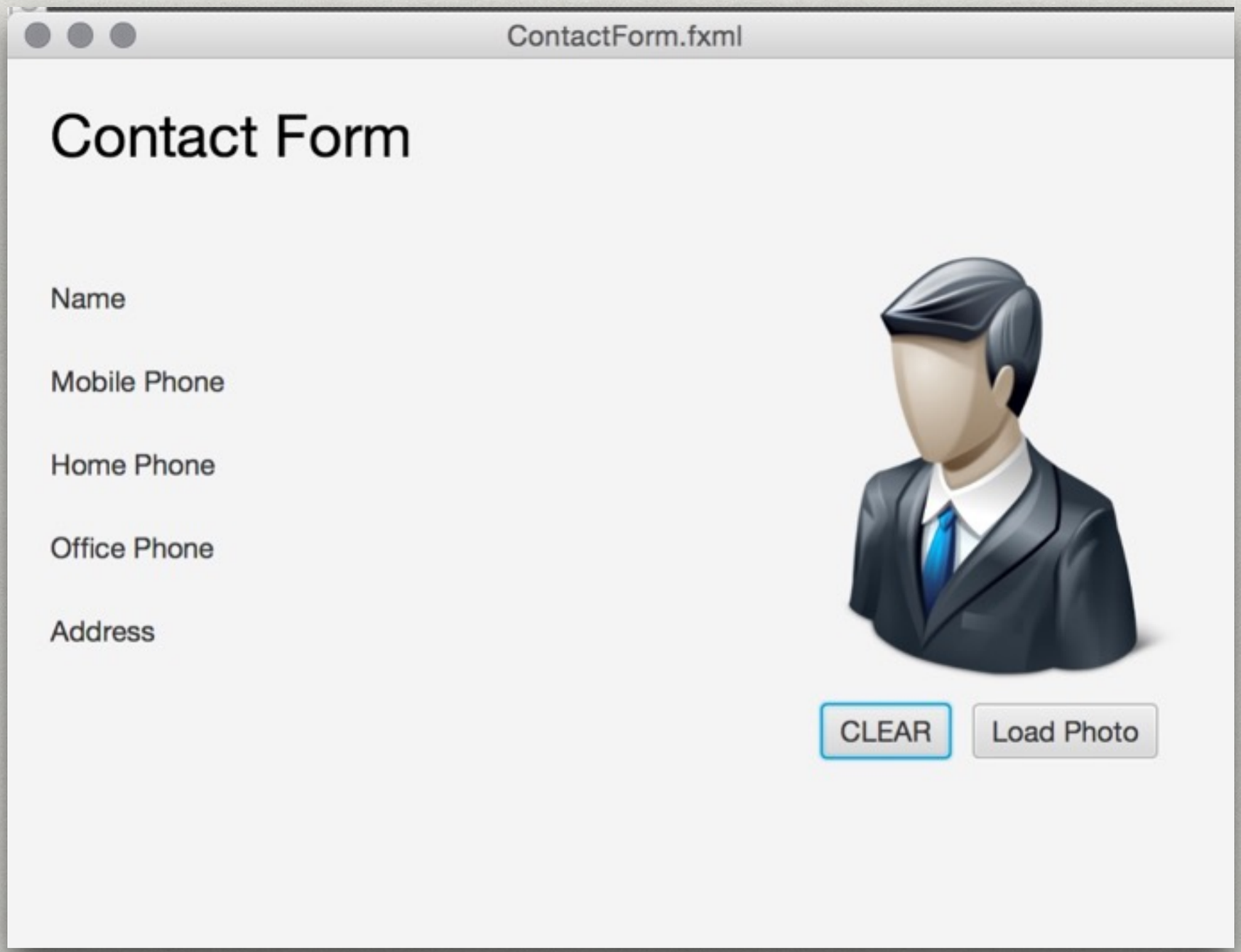
Declaring TreeView

```
@FXML
private TreeView<String> tree;

    // tree view
    ImageView imgView = new ImageView(
            new Image(getClass()
                .getResourceAsStream("Web-b-icon.png")));
    imgView.setPreserveRatio(true);
    imgView.setFitWidth(20);

    TreeItem<String> rootNode = new TreeItem<>("JDC Courses", imgView);
    rootNode.setExpanded(true);
```

Adding Data To Tree

```
getMap().forEach((key, list) -> {
    TreeItem<String> leaf = new TreeItem<>(key);
    leaf.getChildren().addAll(list
            .stream()
            .map(a -> new TreeItem<>(a))
            .collect(Collectors.toList()));
    rootNode.getChildren().add(leaf);
});

tree.setRoot(rootNode);
```

Let's Complete this View



Notes!

- How to write controls with SceneBuilder
- How to control button action
- How to get input data from view
- How to navigate other stage
- How to load data to table