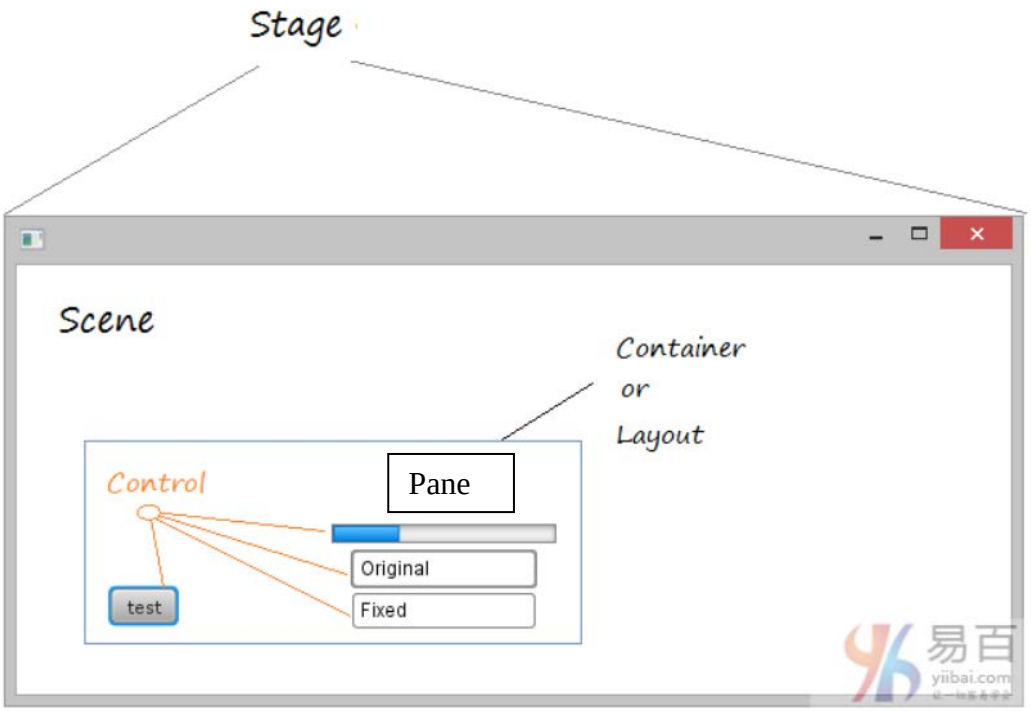
Java2 Lab 9
(JavaFx)

[Experimental Objective]

1. 学习理解JavaFx
2. 通过编写和调试程序，掌握JAVA图形界面设计的基本方法

[Javafx组成]

下图显示JavaFx中，舞台（Stage），场景（Scene），容器（Container），布局（Layout）和控件（Controls）之间的关系：



javafx 一些简单类名

1、Scene：创建场景
2、Stage：创建舞台
3、Pane：面板的基类，用getChildren（）可以返回面板中的节点列表
4、stackPane：面板，节点放置在面板中央
5、FlowPane：节点以一行一行，或一列一列放置
6、GridPane：节点放置在一个二维网络的单元格中
7、BorderPane：节点放置在四周及中央
8、HBox：节点放置在单行中
9、VBox：节点放置在单列中
10、Color：定义颜色
11、Font：定义字体大小，格式，粗细等
12、Image：获取图像
13、ImageView：按要求显示图像

14、Text：创建文本
15、Line：创建直线
16、Rectangle：创建矩形
17、Circle：创建圆
18、Ellipse：创建椭圆
19、Arc：创建弧
20、Polyogn： 创建封口的多边形
21、Polyline：创建未封口的多边形

[Javafx安装]

安装**e(fx)clipse:**

参见教程:

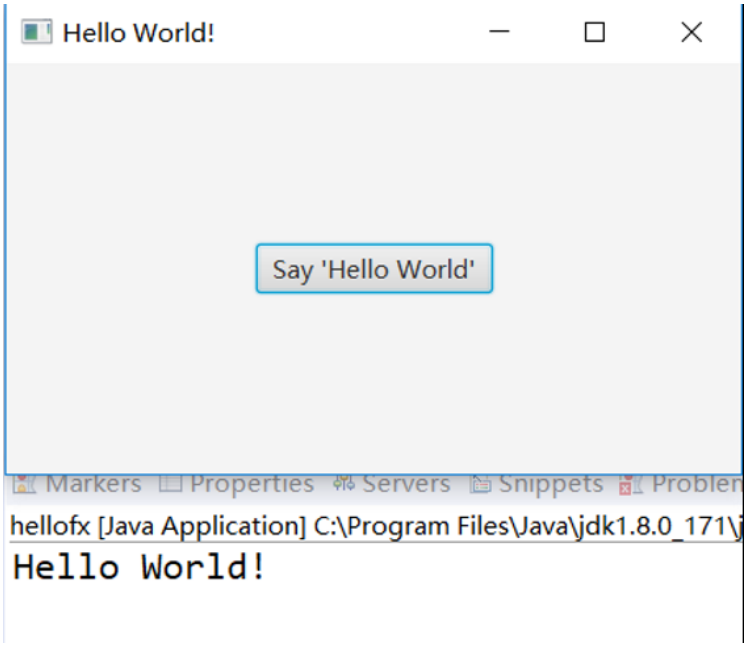
https://www.yiibai.com/javafx/install-efxclipse-into-eclipse.html

在install new software中输入:
Name: e(fx)clipse
Location: http://download.eclipse.org/efxclipse/updates-released/2.3.0/site

[Exercise]

练习1:

参照lab8的练习1，使用JavaFx编写hello world程序，并显示窗口和按钮内容为hello world，点击按钮可以在控制台输出hello world，效果如图。

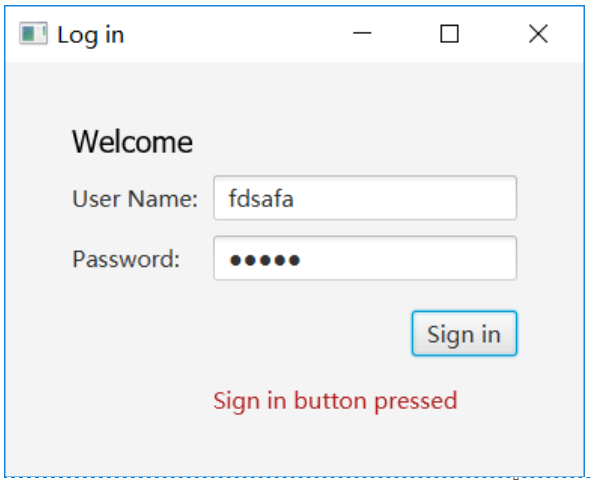


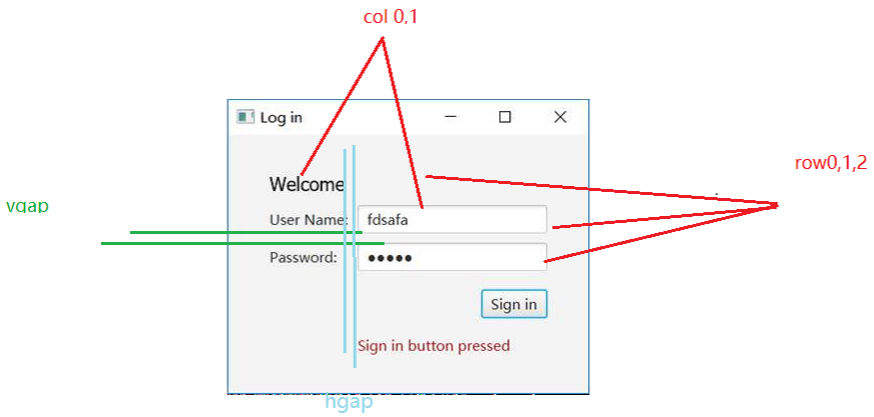
提示:
设置按钮事件，并将scene, button, stage等各个部分依次组装起来即可。

```
    public void start(Stage primaryStage) {
        try {
            Button button = new Button();
            button.setText("hello world");
            button.setOnAction(new EventHandler<ActionEvent>() {
                @Override
                public void handle(ActionEvent event) {
                    // TODO Auto-generated method stub
                    System.out.println("hello world");
                }
            });
            StackPane stackPane = new StackPane();
            stackPane.getChildren().add(button);
            Scene scene = new Scene(stackPane, 350, 200);

            primaryStage.setTitle("hello world");
            primaryStage.setScene(scene);
            primaryStage.show();
        } catch(Exception e) {
            e.printStackTrace();
        }
    }

    public static void main(String[] args) {
        launch(args);
    }
}
```

注：StackPane：它是类似于以前的CardLayout的一种卡片布局方式，就是后面的内容会显示在前面内容之上。

练习2：





思路提示:

在练习1的基础上，沿用上次lab8实验2的思路，将text, label,textfield,password field等各个组件放到GridPane中（因为GridPane可以创建二维的行和列网格），再将gridPane调整大小后添加到scene中，并设置scene为stage的场景即可。

```
@Override
public void start(Stage primaryStage) {
        primaryStage.setTitle("Log in");

        GridPane grid = new GridPane();
        grid.setAlignment(Pos.CENTER);
        grid.setHgap(10);
        grid.setVgap(10);
        grid.setPadding(new Insets(25, 25, 25, 25));

        placeComponents(grid);

        Scene scene = new Scene(grid, 400, 375);
        primaryStage.setScene(scene);
        scene.getStylesheets().add(getClass().getResource("application.css").toExternalForm());
        primaryStage.show();
}
```

将其他组件依次放置到GridPane中:

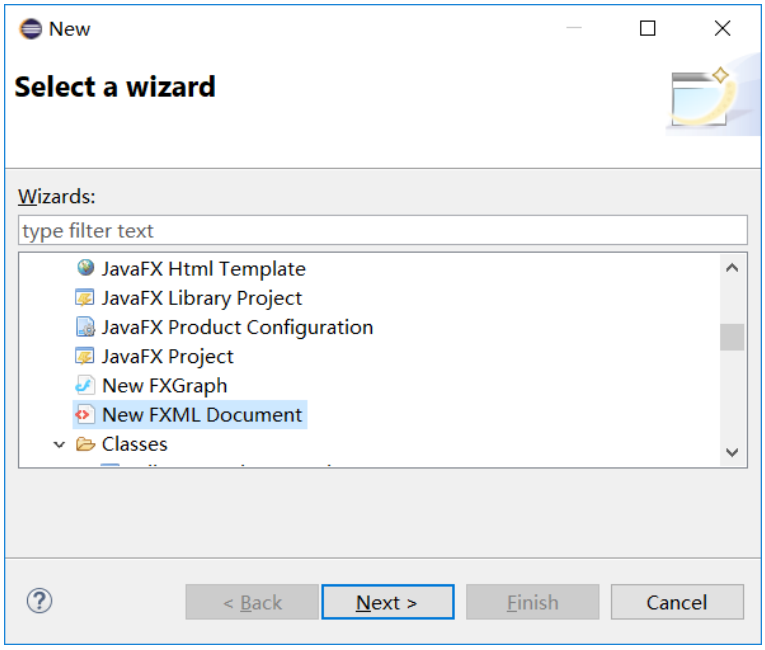
```
public static void placeComponents(GridPane grid) {
    Text text = new Text("Welcome");
    text.setFont(Font.font("Tahoma", FontWeight.NORMAL, 20));
    grid.add(text, 0, 0, 2, 1);

    Label userName = new Label("User Name:");
    grid.add(userName, 0, 1);
    TextField userTextField = new TextField();
    grid.add(userTextField, 1, 1);

    Label pw = new Label("Password:");
    grid.add(pw, 0, 2);
    PasswordField pwBox = new PasswordField();
    grid.add(pwBox, 1, 2);

    Button btn = new Button("Sign in");
    HBox hbBtn = new HBox(10);
    hbBtn.setAlignment(Pos.BOTTOM_RIGHT);
    hbBtn.getChildren().add(btn);
    grid.add(hbBtn, 1, 4);

    final Text actiontarget = new Text();
    grid.add(actiontarget, 1, 6);

        btn.setOnAction(new EventHandler<ActionEvent>() {
            @Override
            public void handle(ActionEvent e) {
                actiontarget.setFill(Color.FIREBRICK);
                actiontarget.setText("Sign in button pressed");
            }
        });


    }

    public static void main(String[] args) {
        launch(args);
    }
```

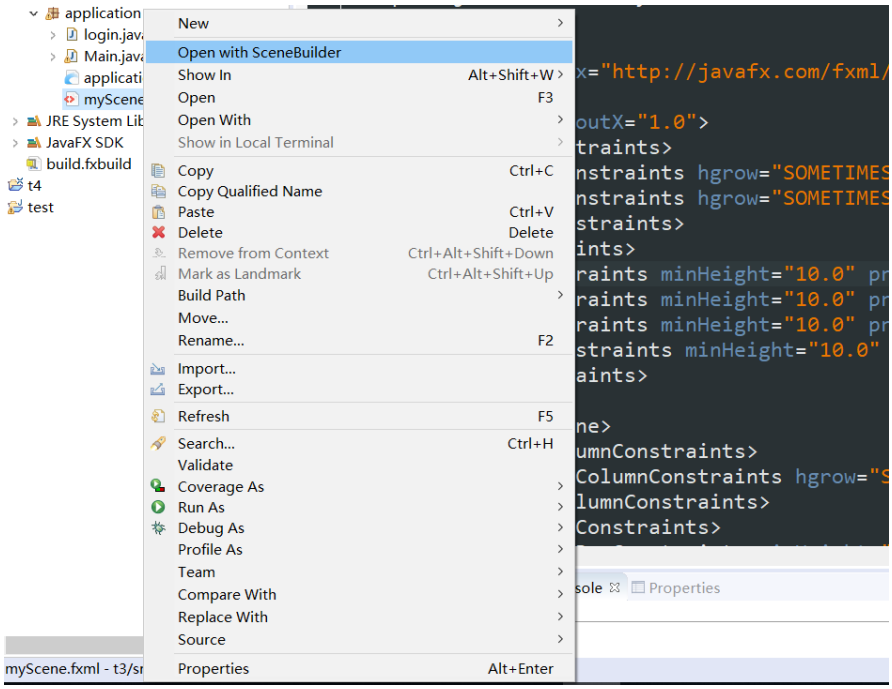
【可选】练习3：

在实验2的基础上，安装JavaFX Scene Builder，把实验2用图形化编码的方式实现一下，并比较图形化编码和用代码编写的优缺点。

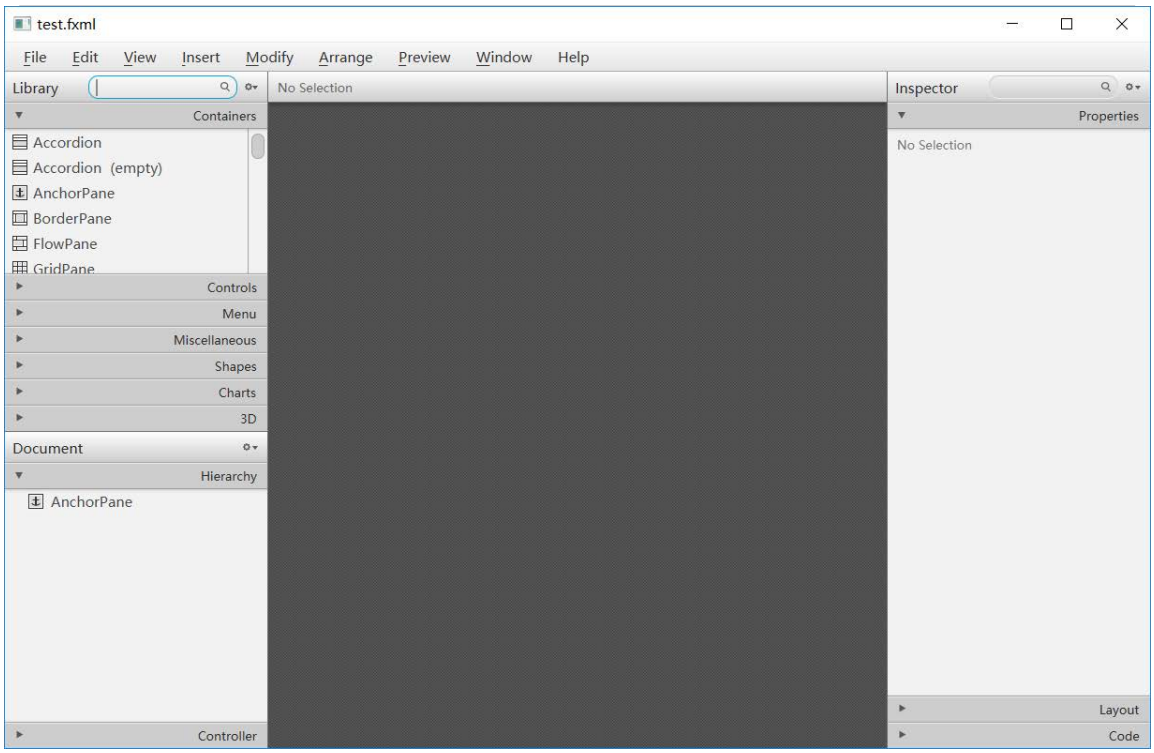
具体安装步骤参见教程：
https://www.yiibai.com/javafx/install-javafx-scene-builder-into-eclipse.html

在javaFx项目中新建new fxml document,并在fxml 文件右键选择 Open with Scene Builder，这样将会打开图形编辑界面





只需将各个元件依次添加组装即可



11849062@mail.sustc.edu.cn