程式設計畢業專題簡報

前言：我們所使用的是java外掛的輔助程式JavaFX，則在coding方面我們都是從網路上與朋友摸索出來的，所以在我們的畢業專案中沒有用到任何跟swing有關的程式碼，我們這樣做的起由是因為我們現在學到的swing技巧離我們想做出來的成品有很大的距離。

程式碼展示：

1. 轉換ＦＸＭＬ檔案的程式碼（轉換page之間的工具），其中紅字的程式碼是我們為了要使用css所增加的並使用css語法去寫我們的背景圖

public void switchToScene1(ActionEvent event)throws IOException

{

Exploder loader = new Exploder(get Class().get Resource("homePage.fxml"));

Parent root = loader.load();

homePageController scene2Controller = loader.getController();

scene2Controller.getname(userName.getText());

stage = (Stage)((Node)event.getSource()).getScene().getWindow();

scene = new Scene(root);

scene.getStylesheets().add(getClass().getResource("application.css").toExternalForm());

stage.setScene(scene);

stage.show();

}

1. 除了使用第一點的轉換FXML檔案的技巧外，我們也使用了getController() methed去使用另一個ＦＸＭＬ的工具，而在這裡我們使用的是搜尋，也就是必須要透過這行程式碼先把我們下一個要變換的Page的搜尋列先列出來再作轉換的部分，而在這也用了我們資料庫連結的class Database conn 去取得與資料庫的連結

public void goTosearchPage(Event event)throws IOException, SQLException

{

conn=datebase.getconn();

FXMLLoader loader = new FXMLLoader(getClass().getResource("searchPage.fxml"));

Parent root = loader.load();

searchPageController scene2Controller = loader.getController();

if(TextField.getText()!=""&&TextField.getText()!=" "&&TextField.getText()!=" ")

{scene2Controller.setTextField(TextField.getText());

scene2Controller.setListView();

}

scene2Controller.getname(userName.getText());

stage = (Stage)((Node)event.getSource()).getScene().getWindow();

scene = new Scene(root);

scene.getStylesheets().add(getClass().getResource("application.css").toExternalForm());

stage.setScene(scene);

stage.show();

}

1. SQL語法的運用（搜尋系統）活用了%與ResultSetMetaData去取得getColumnCount()這個功能去做後續的處理

public void setListView(Event event) throws IOException, SQLException

{

reset();

image.setVisible(true);

String a=Textfeild.getText().toString();

if(a.length()>1)

{conn=datebase.getconn();

String re="SELECT \* FROM products WHERE name LIKE '%"+a+"%'";

PreparedStatement pst=conn.prepareStatement(re);

ResultSet result=pst.executeQuery();

ResultSetMetaData metaData = result.getMetaData();

int columnCount = metaData.getColumnCount();

while(result.next()) {

for(int i=1;i<=columnCount;i++) {

if(i>=1) {

myListView.getItems().add(result.getString(i));

}

}

4.使用getSelectionModel().getSelectedItem()這兩個工具去取得使用者所選擇的商品去做相對應的SQL功能並善用餘數去對InfoArea跟CommentArea做排版

public void getinfo(ActionEvent event) throws IOException, SQLException

{.

………………….

String a=myListView.getSelectionModel().getSelectedItem();

conn=datebase.getconn();

stat=conn.createStatement();

String re="SELECT \* FROM IDname WHERE name LIKE '%"+a+"%'";

…………………….

while(result.next()) {

String av=null;

if(averge()>0)

{av=String.format("%.1f", averge())+"星";}

else

{av="尚未有評價";}

comArea.appendText("綜合評價: "+av+"\n-----------------------------------------\n");

for(int i=1;i<=columnCount;i++) {

if(i%5==1) {

textArea.appendText("ID: ");

product=result.getString(i);

textArea.appendText(product+"\n");

}

else if(i%5==2) {

textArea.appendText("name: ");

product=result.getString(i);

textArea.appendText(product+"\n");

……………………………………

result.close();

Textfield2.setText("");

String re1="SELECT \* FROM commentTable WHERE name LIKE '%"+myListView.getSelectionModel().getSelectedItem()+"%'";

PreparedStatement pst1=conn.prepareStatement(re1);

…………….

5.KeyAction的編排，使用到getCode()這個工具配合switch去確認是否為enter鍵且用觀察版面的擺置去給予適當的功能，像在這例子裡我們使用了確認signUp跟LogIn button的visible去判斷現在版面的樣貌並給予相對的功能

Ex:

public void KeyBoardEnter(KeyEvent event) throws IOException, SQLException {

switch(event.getCode())

{case ENTER:

if(loginbut.isVisible()) {LoginButton(event);}else{signupButton(event);}

default: break;}}

6.其餘程式碼概要我們在LaunchPage所使用的choiceBox也使用getSelectionModel().getSelectedItem()這兩個工具去做資料庫的刪除加入等等

另外我們也多次使用到button與資料庫使用與排版的連結，

Ex:

TranslateTransition slide=new TranslateTransition();

slide.setDuration(Duration.seconds(0.7));

slide.setNode(layer2);

slide.setToX(-365);

slide.play();

用TranslateTransition工具去做signPage（第一頁）的版面轉換

我們也在登錄系統設定了不能有中文只能有中英，而我們使用的對應程式碼就是如下圖，使用match()功能去給他回傳一個boolean值

Ex:

public boolean isEnglishDit(String str)

{

boolean result2 = str.matches("[a-zA-Z0-9]+");

return result2;

}

而在使用者輸入錯誤時我們有跳出視窗的提醒，而在這裡我們使用的是Alert

Ex:

Alert alt1=new Alert(Alert.AlertType.ERROR);

alt1.setTitle("Log in Fail");

alt1.setContentText("Please input the correct username and password");

alt1.show();

而在css語法我們主要是使用圖片的獲取與排版

Ex:

.bg11img

{

-fx-background-image:url("p11.jpg");

-fx-background-size:cover ;

}

而在超連結的部分,使用的是Desktop這個工具

Ex:

Desktop browser=Desktop.getDesktop();

browser.browse(new URI(url));

剩下的程式碼就大多以sceneController的語法有比較大關聯，像是setListView()

和我們熟悉的setText()等等，並且善用SQL語法去使用就能達到很好的排版效果，給予每個button搭配的SQL語法（insert,delete,like…）並在用結果去做Page的編排