S

Search and Replace

Project ID: 5645

Forked from an inaccessible project.

Solution Push
Mads Pedersen authored 1 month ago

Name	Last commit	Last update
Lesson02SearchReplaceSolution	Solution Push	1 month ago
assets	Moves files	1 month ago
searchandreplace	Moves files	1 month ago
.gitignore	Moves files	1 month ago
README.md	Moves files	1 month ago

README.md

Lesson 2: Exercise, Search And Replace

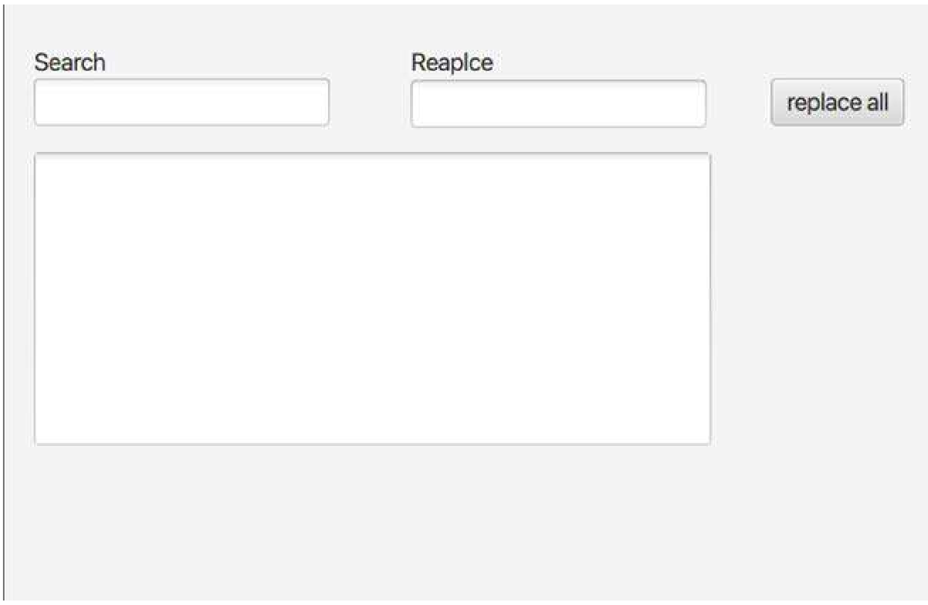
From ordinary exam, VOP F15.

Task 1 - Search and Replace

You will be working with the following file(s) for this exercise: **FXMLController.java, Primary.FXML**

Purpose: Using JavaFX to create a GUI application with search and replace functionality.

1. Add the following components to your scene:
   - A `Label` and a `TextField` for the text you will be searching for
   - A `Label` and a `TextField` for the text you will be replacing it with.
   - A `TextArea` for showcasing the text you will be search-and-replacing in
   - A `Button` for implementing the *Replace All* functionality.
2. Program an `actionHandler` for the button, so every occurrence of the text in the *"search"-field* will be replaced by the text in the *"replace"-field*



Your stage should look like this at the end of Task 1

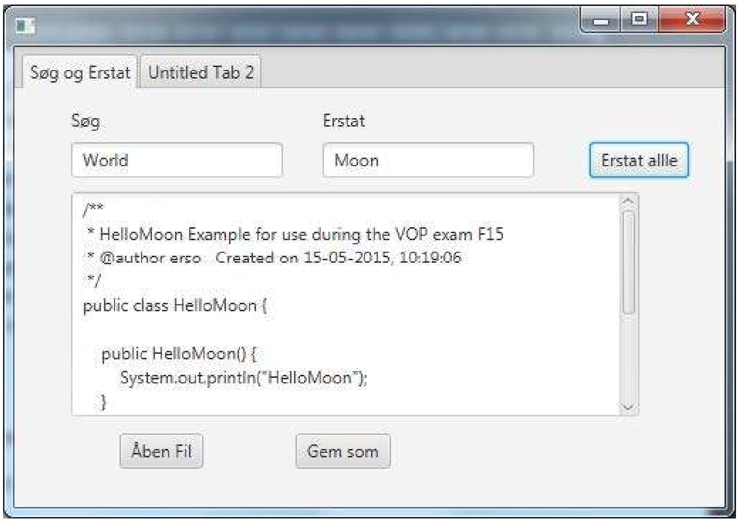
Task 2 - Simple File-I/O

Purpose: To be able to load the text you wish to use the search-and-replace functionality from a file and saving the altered text in another file.

You will be working with the following file(s) for this assignment: **FXMLController.java, Primary.FXML, ”HelloWorld.txt”**

1. Create a FileChooser *instance*-variable.
2. *Initialize* the FileChooser, in the **FXMLController’s `initialize()`** -method and set the `InitialDirectory` to the project folder `(new File("."))` , as follows `fileChooser.setInitialDirectory(new File("."))`
3. Add 2 buttons to your scene, *“Open file”* and *“Save as*".
4. Create an `ActionHandler` for the *”Open file”*-button which uses the `FileChooser` , to select the desired file as follows: `File inFile = fileChooser.showOpenDialog(null);` Read the content of the inFile and load into the `TextArea.`
5. Create an `ActionHandler` for the *“Save as”*-button, which uses the `FileChooser` , to pick an existing file, or creating a new file, as follows: `File outFile = fileChooser.showSaveDialog(null);` The content of the `TextArea,` should be written in the outFile.

Example of the final stage visible below:



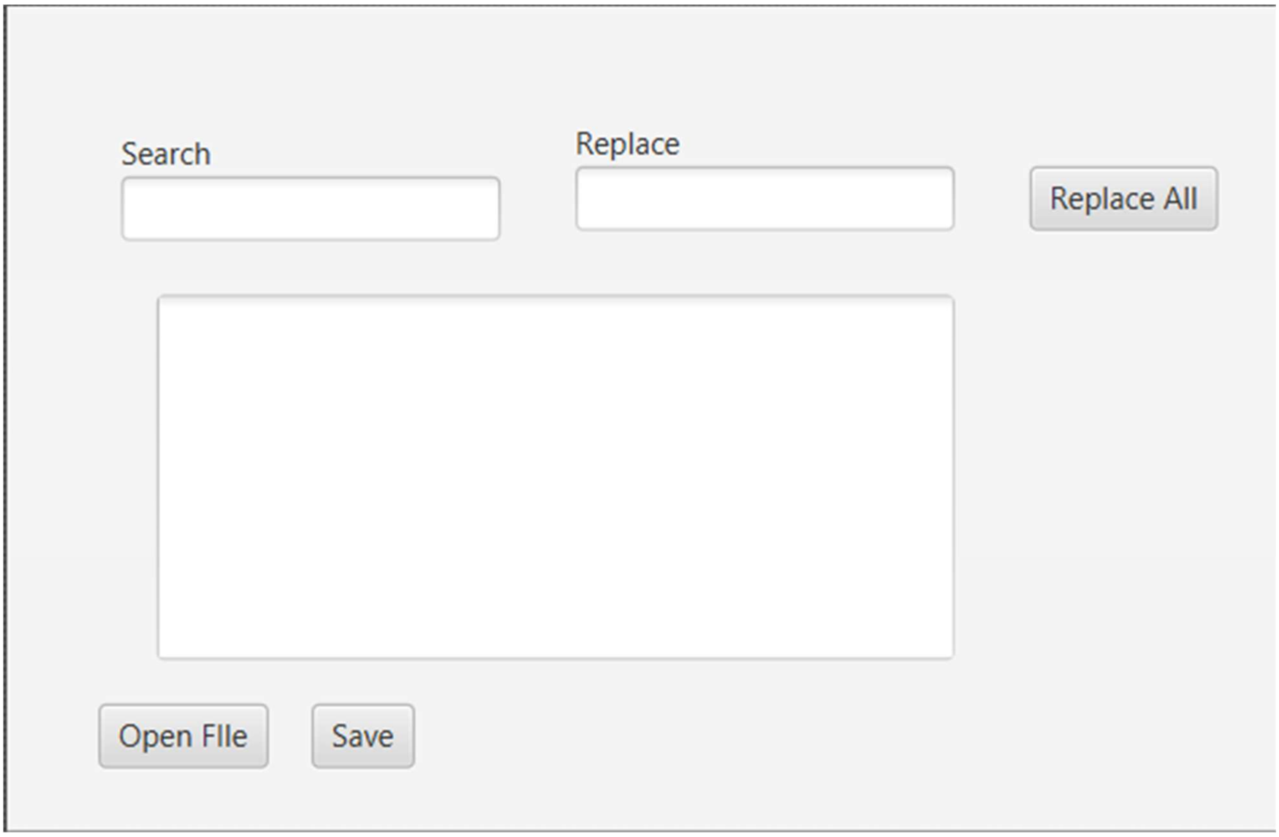
Søg: World

Erstat: Moon

```
/**
 * HelloMoon Example for use during the VOP exam F15
 * @author erso   Created on 15-05-2015, 10:19:06
 */
public class HelloMoon {

    public HelloMoon() {
        System.out.println("HelloMoon");
    }
```

In this example *”HelloWorld.txt”* has been loaded. (found in the root of the project folder), all occurrences of the word *”World”* has been changed to *”Moon”* and the result has been saved to *”HelloMoon.txt”*.



Ovenpå, kan det ses at jeg har lavet de forskellige komponenter for JavaFX i SceneBuilder.

I kan ignorere størrelsen på scenen, det vigtigste er bare at indsætte de rigtige knapper, figurer og labeler med tekst og id.

```java
package com.example;

import java.io.File;
import java.io.FileNotFoundException;
import java.io.PrintWriter;
import java.net.URL;
import java.util.ResourceBundle;
import java.util.Scanner;

import com.sun.javafx.charts.Legend;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.Initializable;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.control.TextArea;
import javafx.scene.control.TextField;
import javafx.stage.FileChooser;

//Scanner
//Replace all tager to argumenter. Den første hvad der skal erstattes og den anden hvad der skal erstattes med.

public class FXMLController implements Initializable {

    //Vi starter med, at bruge skeletten fra SceneBuilder og derved indsætter det herinde.
    FileChooser fc;
//Vi erklærer en FileChooser med variablen fc.

    //Nu tager vi alle ting fra JavaFX, som er defineret fra SceneBuilder.
    //Vi gør det for alle de variabler, der har @FXML.
    @FXML
    private Button openfilebutton;

    @FXML
    private TextField replace_field;

    @FXML
    private Button savebutton;

    @FXML
    private TextField search_field;

    @FXML
    private TextArea text_field;


//Herunder har vi ActionHandler for Replace-Knappen.
    //Vi tager teksten, som søges igennem.
    //Derefter tager vi teksten, som skal søges efter.
    //Derefter tager vi det som vi søger efter, som skal erstattes med.
    //Nu udskifter vi teksten i search og replace, hvor der er "match" på søgestrengen og derefter udskifter det med udskiftningsstrengen.
    //Nu gemmer vi det i variablen replacedText.
    //Til sidst sætter vi teksten til at være i Text-Field.

    //Vi kan se, at vi har lavet den sidste del af Opgave 1 her.
    @FXML
    void ActionHandler(ActionEvent event) {
        //String textToSearchAndReplace = text_field.getText();
        String searchString = search_field.getText();
        String replaceString = replace_field.getText();
        String replacedText = text_field.getText().replaceAll(searchString,replaceString);
        text_field.setText(replacedText);
    }

//Vi tager og åbner en fil.
    //Derefter læser vi igennem i filen.
    //Vi gemmer det i en streng midlertidigt.
    //Så læser vi igennem Filen og gemmer linjerne i strengen.
    @FXML
    void openfile(ActionEvent event) {
        try {
            //Opgave 2.2 er løst her.
            File openenedFile = fc.showOpenDialog(null);
            Scanner scan = new Scanner(openenedFile);
            String fileScanned = null;

            while (scan.hasNextLine()) {
                fileScanned += scan.nextLine();
                fileScanned += "\n";
            }
            scan.close();
            text_field.setText(fileScanned);
        } catch (FileNotFoundException e) {
            System.out.println("File is not found");

        }
    }

    @FXML
    void save(ActionEvent event) {
        //Opgave 2.5 er løst her.
        File outfile = fc.showSaveDialog(null);
        try{
            PrintWriter skriver = new PrintWriter(outfile);
            skriver.print(text_field.getText());
            skriver.close();
        }
        catch(FileNotFoundException e){
            System.out.println("Problem With Writing To The File");
        }
    }

//Tag mappen som koden er i - betyder
```

```
setInitialDirectory
        @Override
    public void initialize(URL url, ResourceBundle rb) {
        //Opgave 2.a er lavet her.
        fc = new FileChooser();
        //Opgave 2.b er lavet her.
        fc.setInitialDirectory(new File(".."));

        // TODO
    }
}

```

```java
package com.example;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.stage.Stage;

import java.io.IOException;


public class MainApp extends Application {
    private static Stage stage;

    @Override
    public void start(@SuppressWarnings("exports") Stage s) throws IOException {
        stage=s;
        setRoot("primary","");
    }

    static void setRoot(String fxml) throws IOException {
        setRoot(fxml,stage.getTitle());
    }

    static void setRoot(String fxml, String title) throws IOException {
        Scene scene = new Scene(loadFXML(fxml));
        stage.setTitle(title);
        stage.setScene(scene);
        stage.show();
    }

    private static Parent loadFXML(String fxml) throws IOException {
        FXMLLoader fxmlLoader = new FXMLLoader(MainApp.class.getResource("/fxml/"+fxml + ".fxml"));
        return fxmlLoader.load();
    }


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

}

```