package SchedulingTest;

import java.io.BufferedReader;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.FileReader;

import java.io.FileWriter;

import java.io.IOException;

import java.io.StringWriter;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import java.text.SimpleDateFormat;

import java.util.Date;

import java.util.concurrent.Future;

import javax.ws.rs.core.MediaType;

import org.apache.commons.io.FileUtils;

import org.flowable.engine.delegate.DelegateExecution;

import org.flowable.engine.delegate.JavaDelegate;

import org.json.simple.JSONArray;

import org.json.simple.JSONObject;

import com.mysql.jdbc.PreparedStatement;

import com.mysql.jdbc.Statement;

import com.sun.jersey.api.client.AsyncWebResource;

import com.sun.jersey.api.client.Client;

import com.sun.jersey.api.client.WebResource;

public class Work implements JavaDelegate {

public void execute(DelegateExecution execution) {

String url = "jdbc:mysql://localhost:3306/flowable";

String user = "root";

String password = "";

Statement stmt = null;

File file1 = new File("D:/MY\_DIRECTORY"); //Source Directory

String s[]=file1.list();

File source = new File("D:/MY\_DIRECTORY"); //Source Directory

File dest = new File("D:/MY\_DIRECTORY1"); //Destination Directory

try {

FileUtils.copyDirectory(source, dest); //Copy files from Source Directory to Destination Directory

} catch (IOException e) {

e.printStackTrace();

}

JSONArray ja = new JSONArray(); //JSON Array

for(String s1 :s )

{

File f=new File("D:/MY\_DIRECTORY1/"+s1); //File of Destination Directory

JSONObject jo = new JSONObject(); //JSON Object

jo.put("FileName", s1); //Adding file name to JSON Object

jo.put("Path", f.getAbsolutePath()); //Adding file absolute path to JSON Object

ja.add(jo);

try {

Connection conn = DriverManager.getConnection(url, user, password);

String query = " insert into Json (file\_name,file\_path)"

+ " values (?,?)";

// create the mysql insert preparedstatement

PreparedStatement preparedStmt = (PreparedStatement) conn.prepareStatement(query);

preparedStmt.setString (1, s1);

preparedStmt.setString (2, f.getAbsolutePath());

// execute the preparedstatement

preparedStmt.execute();

conn.close();

} catch (SQLException ex) {

ex.printStackTrace();

}

}

JSONObject mainObj = new JSONObject();

// JSON Main Object

mainObj.put("Files", ja);

// Adding JSON Array to JSON Main Object

try {

String timeStamp = new SimpleDateFormat("yyyy.MM.dd.HH.mm.ss").format(new Date()); //Time Stamp

FileWriter file2=new FileWriter("D://MY\_DIRECTORY1//"+"Batch\_"+timeStamp+".txt"); //New file for JSON

file2.write(mainObj.toJSONString()); //Writing JSON Object to New File

file2.flush(); //flush file

file2.close(); //close file

} catch (IOException e) {

e.printStackTrace();

}

try {

String x = null;

BufferedReader br = null;

try {

br = new BufferedReader(new FileReader("D:/input2.txt"));

} catch (FileNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

try {

x=br .readLine();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

StringWriter out1 = new StringWriter();

mainObj.writeJSONString(out1);

String jsonText = out1.toString();

Client client=Client.create();

client.setReadTimeout(10000);

client.setConnectTimeout(10000);

AsyncWebResource resource=client.asyncResource(x);

Future<String> res=resource.type(MediaType.APPLICATION\_JSON).post(String.class,jsonText);

System.out.println(res);

} catch (Exception e) {

System.out.println("\nError while calling hello REST Service");

System.out.println(e);

}

for(String s1 :s)

{

File f=new File("D:/MY\_DIRECTORY/"+s1);

f.delete(); //Delete files from Source Directory

}

}

}

import java.io.BufferedReader;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.FileOutputStream;

import java.io.FileReader;

import java.io.IOException;

import java.util.ArrayList;

import org.apache.pdfbox.pdmodel.PDDocument;

import org.apache.pdfbox.text.PDFTextStripper;

import org.apache.poi.ss.usermodel.Cell;

import org.apache.poi.ss.usermodel.Row;

import org.apache.poi.ss.usermodel.Sheet;

import org.apache.poi.xssf.usermodel.XSSFWorkbook;

import org.flowable.engine.delegate.DelegateExecution;

import org.flowable.engine.delegate.JavaDelegate;

public class Work extends PDFTextStripper implements JavaDelegate {

public Work() throws IOException {

super();

// TODO Auto-generated constructor stub

}

public void execute(DelegateExecution execution)

{

File file = new File("D:/PDFWork/MY\_PDF\_DIRECTORY");

String s[]=file.list();

XSSFWorkbook workbook = new XSSFWorkbook();

Sheet sheet = workbook.createSheet("Work");

int value = 0;

BufferedReader br = null;

try {

br = new BufferedReader(new FileReader("D:/input1.txt"));

} catch (FileNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

try {

value=Integer.parseInt(br .readLine());

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

for(String s1 :s )

{

File f=new File("D:/PDFWork/MY\_PDF\_DIRECTORY/"+s1);

PDDocument document = null;

try {

document = PDDocument.load(f);

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

PDFTextStripper pdfStripper = null;

try {

pdfStripper = new PDFTextStripper();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

pdfStripper.setParagraphStart("/t");

ArrayList<String> l= new ArrayList<String>();

ArrayList<String> l1= new ArrayList<String>();

ArrayList<Integer> l2= new ArrayList<Integer>();

int n=-1;int n1=-1;String x="";

try {

for (String line: pdfStripper.getText(document).split(pdfStripper.getParagraphStart()))

{

int c=0;

line=line.trim();

for(int i=0;i<line.length();i++)

{

if( ((i>0)&&(line.charAt(i)!=' ')&&(line.charAt(i-1)==' ')) || ((line.charAt(i)!=' ')&&(i==0)) )

{

c++;

}

}

if(c<=value && c>0)

{

n++;n1++;x="";

l.add(n,line);

l1.add(n1,x);

l2.add(n,getCurrentPageNo());

System.out.println(line);

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

}

else

{

if(line.length()!=0)

{

x=x+line+"\n";

l1.add(n1, x);;

System.out.println(x);

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

}

}

}

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

int rowCount = -1;

for(int i=0;i<=n;i++)

{

Row row = sheet.createRow(++rowCount);

int columnCount = -1;

Cell cell = row.createCell(++columnCount);

cell.setCellValue(l.get(i));

Cell cell1 = row.createCell(++columnCount);

cell1.setCellValue(l1.get(i));

Cell cell2 = row.createCell(++columnCount);

cell2.setCellValue(l2.get(i));

}

try (FileOutputStream outputStream = new FileOutputStream("D://Work.xlsx")) {

workbook.write(outputStream);

} catch (FileNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

}