# .project

# org.eclipse.wst.common.project.facet.core.xml

<?xml version="1.0" encoding="UTF-8"?>
<faceted-project>
<installed facet="java" version="1.8"/>
</faceted-project>

# CmiConsoleClient.java

package cn.edu.scau.cmi.client;
import java.io.File;
import cn.edu.scau.cmi.utils.DocumentUtils;
public class CmiConsoleClient {
public static void main(String[] args) throws Exception {
String project="D:\\git\\copyProjectFiles2Word\\copyProjectFiles2Word";
File targePath=new File("D:\\git\\copyProjectFiles2Word\\copyProjectFiles2Word\\testResult.docx");
DocumentUtils.writeProject2Doc(project, targePath);
}
}

# CopyFiles2WordSwt.java

package cn.edu.scau.cmi.client;
import org.eclipse.swt.SWT;
import org.eclipse.swt.widgets.Display;
import org.eclipse.swt.widgets.Shell;
import org.eclipse.swt.widgets.Label;
import org.eclipse.swt.custom.CCombo;
import org.eclipse.core.databinding.DataBindingContext;
import org.eclipse.core.databinding.observable.value.IObservableValue;
import org.eclipse.jface.databinding.swt.WidgetProperties;
import org.eclipse.core.databinding.beans.PojoProperties;
import org.eclipse.core.databinding.observable.Realm;
import org.eclipse.jface.databinding.swt.SWTObservables;
import org.eclipse.swt.layout.GridLayout;
import org.eclipse.swt.layout.GridData;
public class CopyFiles2WordSwt {
private DataBindingContext m\_bindingContext;
protected Shell shell;
private Label label;
public static void main(String[] args) {
Display display = Display.getDefault();
Realm.runWithDefault(SWTObservables.getRealm(display), new Runnable() {
public void run() {
try {
CopyFiles2WordSwt window = new CopyFiles2WordSwt();
window.open();
} catch (Exception e) {
e.printStackTrace();
}
}
});
}
public void open() {
Display display = Display.getDefault();
createContents();
shell.open();
shell.layout();
while (!shell.isDisposed()) {
if (!display.readAndDispatch()) {
display.sleep();
}
}
}
protected void createContents() {
shell = new Shell();
shell.setSize(450, 300);
shell.setText("将项目的文本文件复制到一个Word文件中");
shell.setLayout(new GridLayout(2, false));
label = new Label(shell, SWT.NONE);
label.setText("请选择项目");
CCombo combo = new CCombo(shell, SWT.BORDER);
GridData gd\_combo = new GridData(SWT.LEFT, SWT.CENTER, false, false, 1, 1);
gd\_combo.widthHint = 179;
combo.setLayoutData(gd\_combo);
Label label\_1 = new Label(shell, SWT.NONE);
label\_1.setText("请选择文件类型");
new Label(shell, SWT.NONE);
m\_bindingContext = initDataBindings();
}
protected DataBindingContext initDataBindings() {
DataBindingContext bindingContext = new DataBindingContext();
//
IObservableValue observeSizeLabelObserveWidget = WidgetProperties.size().observe(label);
IObservableValue sizeShellObserveValue = PojoProperties.value("size").observe(shell);
bindingContext.bindValue(observeSizeLabelObserveWidget, sizeShellObserveValue, null, null);
//
return bindingContext;
}
}

# CmiFileUtils.java

package cn.edu.scau.cmi.utils;
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.InputStreamReader;
import java.io.OutputStreamWriter;
import org.apache.commons.io.FileUtils;
public class CmiFileUtils{
public static String readFile(File sourceProjectFile) {
String fileContent = "";
try {
// File sourceProjectFile = new File(filePathAndName);
if(sourceProjectFile.isFile()&&sourceProjectFile.exists()){
InputStreamReader read = new InputStreamReader(new FileInputStream(sourceProjectFile),"UTF-8");
BufferedReader reader=new BufferedReader(read);
String line;
while ((line = reader.readLine()) != null) {
fileContent = fileContent+line+"\r\n";
// 怎样添加字符，然后可以在word中变成段落呢
}
read.close();
}
} catch (Exception e) {
System.out.println("读取文件内容操作出错");
e.printStackTrace();
}
return fileContent;
}
public static void writeFile(String filePathAndName, String fileContent) {
try {
File sourceFilePath = new File(filePathAndName);
if (!sourceFilePath.exists()) {
sourceFilePath.createNewFile();
}
OutputStreamWriter write = new OutputStreamWriter(new FileOutputStream(sourceFilePath),"UTF-8");
BufferedWriter writer=new BufferedWriter(write);
writer.write(fileContent);
writer.close();
} catch (Exception e) {
System.out.println("写文件内容操作出错");
e.printStackTrace();
}
}
}

# DocumentUtils.java

package cn.edu.scau.cmi.utils;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStream;
import java.util.ArrayList;
import org.apache.poi.xwpf.usermodel.XWPFDocument;
import org.apache.poi.xwpf.usermodel.XWPFParagraph;
import org.apache.poi.xwpf.usermodel.XWPFRun;
import org.apache.poi.xwpf.usermodel.XWPFStyles;
import org.apache.xmlbeans.XmlException;
import org.openxmlformats.schemas.wordprocessingml.x2006.main.CTStyles;
import cn.edu.scau.cmi.nouse.FileUtils;
public class DocumentUtils {
/\*\*
\* word整体样式
\*/
private static CTStyles style = null;
/\*\*
\* Word整体样式
\*/
static {
XWPFDocument template;
try {
// 读取模板文档
template = new XWPFDocument(new FileInputStream(System.getProperty("user.dir")+"/format.docx"));
// 获得模板文档的整体样式
style = template.getStyle();
} catch (FileNotFoundException e) {
e.printStackTrace();
} catch (IOException e) {
e.printStackTrace();
} catch (XmlException e) {
e.printStackTrace();
}
}
private static void WriteDoc(StringBuilder allSourceFilecontent, String targePath) throws IOException {
String allFileContent = allSourceFilecontent.toString();
XWPFDocument doc = new XWPFDocument();
// 创建一个段落
XWPFParagraph para = doc.createParagraph();
// 一个XWPFRun代表具有相同属性的一个区域。
XWPFRun run = para.createRun();
run.setText(allFileContent);
OutputStream os = new FileOutputStream(targePath);
// 把doc输出到输出流
doc.write(os);
doc.close();
}
public static void writeProject2Doc(String project, File targePath) throws IOException {
OutputStream os = new FileOutputStream(targePath);
XWPFDocument doc = new XWPFDocument();
XWPFStyles newStyles = doc.createStyles();
newStyles.setStyles(style);
// XWPFParagraph paraFileName = doc.createParagraph();
// XWPFParagraph paraFileContent = doc.createParagraph();
// StringBuilder sb=new StringBuilder("");
// 获取所有的项目文件
ArrayList<File> projectFileList = ProjectUtils.getProjectFiles(project);
for (File file : projectFileList) {
XWPFParagraph paraFileName = doc.createParagraph();
XWPFParagraph paraFileContent = doc.createParagraph();
System.out.println(file.getName());
System.out.println(FileUtils.readFile(file));
//设置文件名标题
paraFileName.setStyle("2");
XWPFRun fileNameRun = paraFileName.createRun();
XWPFRun fileContentRun = paraFileContent.createRun();
//写文件名
// fileNameRun.setBold(true);
fileNameRun.setText(file.getName());
//换行
// fileNameRun.addCarriageReturn();
//写文件内容
fileContentRun.setBold(false);
fileContentRun.setText(FileUtils.readFile(file));
}
doc.write(os);
doc.close();
System.out.println("所有的项目文件已经完整的写入到了文档");
}
}

# DocumentUtilsOlder.java

package cn.edu.scau.cmi.utils;
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStream;
import java.util.ArrayList;
import org.apache.poi.xwpf.usermodel.XWPFDocument;
import org.apache.poi.xwpf.usermodel.XWPFParagraph;
import org.apache.poi.xwpf.usermodel.XWPFRun;
import cn.edu.scau.cmi.utils.ProjectUtils;
public class DocumentUtilsOlder {
public static void WriteDoc(StringBuilder allSourceFilecontent,String targePath) throws IOException{
String allFIleContent=allSourceFilecontent.toString();
XWPFDocument doc = new XWPFDocument();
//创建一个段落
XWPFParagraph para = doc.createParagraph();
//一个XWPFRun代表具有相同属性的一个区域。
XWPFRun run = para.createRun();
run.setText(allFIleContent);
// run.setBold(true); //加粗
// run.setText("加入的内容");
OutputStream os = new FileOutputStream(targePath);
//把doc输出到输出流
doc.write(os);
doc.close();
}
public static void writeProject2Doc(String project, File targePath) throws IOException{
OutputStream os = new FileOutputStream(targePath);
XWPFDocument doc = new XWPFDocument();
XWPFParagraph para = doc.createParagraph();
// StringBuilder sb=new StringBuilder("");
ArrayList<File> projectFileList = ProjectUtils.getProjectFiles(project);
for(File file:projectFileList){
XWPFRun fileNameRun = para.createRun();
XWPFRun fileContentRun = para.createRun();
fileNameRun.setBold(true);
fileNameRun.setText(file.getName());
fileNameRun.setText("\r\n");
fileContentRun.setBold(false);
fileContentRun.setText(CmiFileUtils.readFile(file));
}
doc.write(os);
doc.close();
System.out.println("OK");
}
}

# ProjectUtils.java

# Client.java

package cn.edu.scau.cmi.nouse;
//package cn.edu.scau.cmi.implcode;
//
//import java.io.File;
//import java.util.ArrayList;
//
//public class Client {
//
// public static void main(String[] args) throws Exception {
// StringBuilder sbAllDocument4Word=new StringBuilder("");//所有准备写入到word文档中的内容
// String simpleDocumentString="";//每一个文档的内容
// DocumentUtils writeWordDocument=new DocumentUtils();
// ProjectUtils getProjectFile=new ProjectUtils();
// ArrayList<File> projectFile=getProjectFile.getProjectFiles("D:/Workspace/MyEclipse4Spring/t");
// for(int i=0;i<projectFile.size();i++){
// simpleDocumentString=FileUtils.readFile(projectFile.get(i).toString());
// String[] splitString=projectFile.get(i).toString().split("\\\\");
// sbAllDocument4Word.append(splitString[splitString.length-1]+"\r\n");
// sbAllDocument4Word.append(simpleDocumentString);
// System.out.println(projectFile.get(i));
// System.out.print(simpleDocumentString);
// }
// writeWordDocument.WriteDoc(sbAllDocument4Word.toString(),"D:\\simpleWrite.docx");
// System.out.println("OK");
// }
//
//}

# DocumentUtilsOlder.java

package cn.edu.scau.cmi.nouse;
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStream;
import java.util.ArrayList;
import org.apache.poi.xwpf.usermodel.XWPFDocument;
import org.apache.poi.xwpf.usermodel.XWPFParagraph;
import org.apache.poi.xwpf.usermodel.XWPFRun;
import cn.edu.scau.cmi.utils.ProjectUtils;
public class DocumentUtilsOlder {
public static void WriteDoc(StringBuilder allSourceFilecontent,String targePath) throws IOException{
String allFIleContent=allSourceFilecontent.toString();
XWPFDocument doc = new XWPFDocument();
//创建一个段落
XWPFParagraph para = doc.createParagraph();
//一个XWPFRun代表具有相同属性的一个区域。
XWPFRun run = para.createRun();
run.setText(allFIleContent);
// run.setBold(true); //加粗
// run.setText("加入的内容");
OutputStream os = new FileOutputStream(targePath);
//把doc输出到输出流
doc.write(os);
doc.close();
}
public static void writeProject2Doc(String project, File targePath) throws IOException{
OutputStream os = new FileOutputStream(targePath);
XWPFDocument doc = new XWPFDocument();
XWPFParagraph para = doc.createParagraph();
// StringBuilder sb=new StringBuilder("");
ArrayList<File> projectFileList = ProjectUtils.getProjectFiles(project);
for(File file:projectFileList){
XWPFRun fileNameRun = para.createRun();
XWPFRun fileContentRun = para.createRun();
fileNameRun.setBold(true);
fileNameRun.setText(file.getName());
fileNameRun.setText("\r\n");
fileContentRun.setBold(false);
fileContentRun.setText(FileUtils.readFile(file));
}
doc.write(os);
doc.close();
System.out.println("OK");
}
}

# FileUtils.java

package cn.edu.scau.cmi.nouse;
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.InputStreamReader;
import java.io.OutputStreamWriter;
public class FileUtils {
public static String readFile(File sourceProjectFile) {
String fileContent = "";
try {
// File sourceProjectFile = new File(filePathAndName);
if(sourceProjectFile.isFile()&&sourceProjectFile.exists()){
InputStreamReader read = new InputStreamReader(new FileInputStream(sourceProjectFile),"UTF-8");
BufferedReader reader=new BufferedReader(read);
String line;
while ((line = reader.readLine()) != null) {
fileContent = fileContent+line+"\r\n";
}
read.close();
}
} catch (Exception e) {
System.out.println("读取文件内容操作出错");
e.printStackTrace();
}
return fileContent;
}
public static void writeFile(String filePathAndName, String fileContent) {
try {
File sourceFilePath = new File(filePathAndName);
if (!sourceFilePath.exists()) {
sourceFilePath.createNewFile();
}
OutputStreamWriter write = new OutputStreamWriter(new FileOutputStream(sourceFilePath),"UTF-8");
BufferedWriter writer=new BufferedWriter(write);
writer.write(fileContent);
writer.close();
} catch (Exception e) {
System.out.println("写文件内容操作出错");
e.printStackTrace();
}
}
}