# .project

# Client.java

//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");
// }
//
//}

# CmiClient.java

package cn.edu.scau.cmi.implcode;
import java.io.File;
import java.util.ArrayList;
public class CmiClient {
public static void main(String[] args) throws Exception {
String project="D:\\Workspace\\Jee\\CopyProjectFile2WordDocument";
File targePath=new File("D:\\Workspace\\Jee\\CopyProjectFile2WordDocument\\testResult.docx");
DocumentUtils.writeProject2Doc(project, targePath);
}
}

# DocumentUtils.java

package cn.edu.scau.cmi.implcode;
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;
public class DocumentUtils {
/\*\*
\* word整体样式
\*/
private static CTStyles wordStyles = 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
wordStyles = template.getStyle();
} catch (FileNotFoundException e) {
e.printStackTrace();
} catch (IOException e) {
e.printStackTrace();
} catch (XmlException e) {
e.printStackTrace();
}
}
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);
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(wordStyles);
// XWPFParagraph paraFileName = doc.createParagraph();
// XWPFParagraph paraFileContent = doc.createParagraph();
// StringBuilder sb=new StringBuilder("");
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("OK");
}
}

# DocumentUtilsOlder.java

package cn.edu.scau.cmi.implcode;
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;
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.implcode;
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();
}
}
}

# ProjectUtils.java

package cn.edu.scau.cmi.implcode;
import java.io.File;
import java.util.ArrayList;
public class ProjectUtils {
private static ArrayList<File> projectFiles = new ArrayList<File>();
// 获取指定目录下所有的非.class文件和非.jar文件的文件数组
public static ArrayList<File> getProjectFiles(String dirpath) {
File directory = new File(dirpath);// 建立当前目录中文件的File对象
if (directory.exists()) {
File[] files = directory.listFiles();// 取得目录中所有文件的File对象数组
for (int i = 0; i < files.length; i++) {
File file = files[i];
if ( file.isFile()){
if(file.getName().endsWith(".java")
|| file.getName().endsWith(".jsp")
|| file.getName().endsWith(".html")
|| file.getName().endsWith(".xml")
|| file.getName().endsWith(".xsd")
|| file.getName().endsWith(".css")
|| file.getName().endsWith(".properties")
|| file.getName().endsWith(".map")
|| file.getName().endsWith(".project")
|| file.getName().endsWith(".springBeans")) {
projectFiles.add(file);
}
}
if (file.isDirectory()) {
getProjectFiles(file.getPath());
}
}
}
return projectFiles;
}
}