****

**面向对象程序设计上机实验报告**

实验题目4：数据结构的综合运用

**学院名称 智能与计算学部**

**专 业 软件工程**

**学生姓名 陈昊昆**

**学 号 3021001196**

**年 级 2021级**

**班 级 3班**

**时 间 2022年 4月10日**

1. **实验目的**

熟悉Eclipse/MyEclipse；

熟悉Java中各种类的综合使用；

1. **实验内容**

现在所有学生的名单students.txt（为了避免乱码问题，文件中不包含汉字），其中每行为一个学生信息，包括学号、姓名、班级，以tab符号分割(\t)，学院要求所有同学把自己一寸照片发给辅导员，图片命名规则为 “学号.jpg”。请用编程的方式帮助辅导员解决如下问题：

1. 找出那些同学的照片没有交照片；
2. 在目标目录下每个班级建立一个子目录，把上交的同学的照片，统一按 学号\_姓名.jpg 方式拷贝到各自班级目录下，原来的文件不要删除.
3. **程序实现**

// 从txt中读取后存入String

**private** String readFromTxt(String filename) **throws** Exception {

Reader reader = **null**;

**try** {

StringBuffer buf = **new** StringBuffer();

**char**[] chars = **new** **char**[1024];

reader = **new** InputStreamReader(**new** FileInputStream(filename), "UTF-8");

**int** readed = reader.read(chars);

**while** (readed != -1) {

buf.append(chars, 0, readed);

readed = reader.read(chars);

}

**return** buf.toString();

} **finally** {

*close*(reader);

}

}

// 关闭输入输出流

**private** **static** **void** close(Closeable inout) {

**if** (inout != **null**) {

**try** {

inout.close();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

// 从输入流读取内容并写入到另外一个流

**public** **static** **int** *CHUNK\_SIZE* = 4096;

**public** **static** **void** copyIO(InputStream in, OutputStream out)

**throws** IOException {

**byte**[] buf = **new** **byte**[*CHUNK\_SIZE*];

**int** len = in.read(buf);

**while** (len != -1) {

out.write(buf, 0, len);

len = in.read(buf);

}

}

// 复制文件

**public** **static** **void** copyFile(String fsrc, String fdest) **throws** IOException {

InputStream in = **null**;

OutputStream out = **null**;

**try** {

in = **new** FileInputStream(fsrc); // 从fsrc中输入

out = **new** FileOutputStream(fdest, **true**); // 输出到fdes中

*copyIO*(in, out);

} **finally** {

*close*(in);

*close*(out);

}

}

**public** Set<String> copyToTargetDirAndReturnNoExist(String studentListFile,String srcDir,String target) **throws** Exception {

// 存放学生id与其班级 姓名 是否提交照片的对应关系

Map<String, String[]> map = **new** HashMap<String, String[]>();

// 存放所有的出现的班级

Set<String> classset = **new** HashSet<String>();

// 存放未提及照片的学生id

Set<String> notsubset = **new** HashSet<String>();

String idsample = "";

String totalstr = readFromTxt(studentListFile);

String indistu[] = totalstr.split("\r\n");

// 存入学生id与其班级 姓名 是否提交照片的对应关系

**for**(**int** i = 0; i < indistu.length; i++) {

String splitindistu[] = indistu[i].split("\t");

classset.add(splitindistu[2]);

idsample = splitindistu[0];

// 默认未提交照片

String info[] = {splitindistu[1], splitindistu[2], "0"};

map.put(splitindistu[0], info);

}

**char** []charidsample = idsample.toCharArray();

// 按照班级集合 构造每个班级的目录

File classdirs = **new** File(target);

**if** (!classdirs.exists()) {

classdirs.mkdir();

}

Iterator it = classset.iterator();

**while**(it.hasNext()) {

String classnum = (String)it.next();

File classdir = **new** File(target, classnum);

**if** (!classdir.exists()) {

classdir.mkdir();

}

}

File picdir = **new** File(srcDir);

// 打开目录 读取文件

String[] picfiles = picdir.list();

**for** (**int** i = 0; i < picfiles.length; i++) {

File picfile = **new** File(picdir, picfiles[i]);

String picname = picfile.getName();

**char** checkid[] = picname.toCharArray();

// 检查照片目录里的每个文件是否为所需的名字和格式

**boolean** flag = **true**;

**for**(**int** u = 0; u < 3; u++) {

**if**(charidsample[u] == checkid[u]) {

flag = flag || **true**;

}

**else** flag = **false**;

}

**if**(flag == **true** && checkid[checkid.length - 1] == 'g'

&& checkid[checkid.length - 2] == 'p' && checkid[checkid.length - 3] == 'j'

) {

flag = **true**;

}

// 如果是所需的名字和格式

**if**(flag) {

StringBuffer trueid = **new** StringBuffer("");

**for**(**int** u = 0; u < idsample.length(); u++) {

trueid.append(checkid[u]);

}

String tid = trueid.toString();

**try** {

// 修改map中value的值

String info[] = map.get(tid);

info[2] = "1";

map.put(tid, info);

// 构造文件 复制文件

String despath = target + info[1] + "/";

File desdir = **new** File(despath);

String despicname = tid + "\_" + info[0] + ".jpg";

File despic = **new** File(desdir, despicname);

**if** (!despic.exists()) {

despic.createNewFile();

}

despicname = despath + despicname;

String srcpicname = srcDir + tid + ".jpg";

*copyFile*(srcpicname, despicname);

} **catch**(Exception e) {

**continue**;

}

; }

}

// 遍历map中的学生 检查其value 将未提交照片的学生id加入未提交集并返回

Iterator itt = map.keySet().iterator();

**while**(itt.hasNext()) {

String key = (String)itt.next();

String flag = map.get(key)[2];

**if** (flag.equals("0")) {

notsubset.add(key);

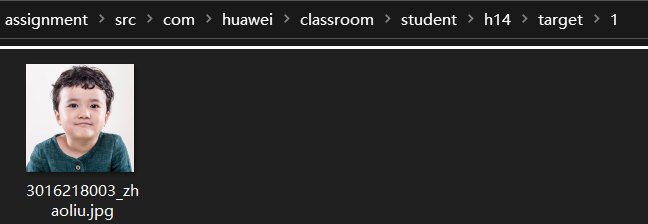
}

}

**return** notsubset;

}

1. **实验结果**



1. **实验中遇到的问题及解决方法**

这次实验中，我学会了读入文件和写入文件，实现了文件从一处向另外一处的复制。同时，我还学会了通过对文件扩展名的识别，在读入时判断文件的类型。也巩固了对map的理解和运用。

问题：复制了的非所需的文件

解决：提前检查源文件的格式和文件名