# Java程序设计实验报告

**学号： 1190201421**

**姓名： 张瑞**

**专业： 工科试验班（计算机与电子通信）**

**班级： 19L0214**

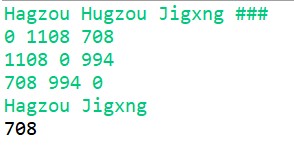
**哈尔滨工业大学**

一、实验内容

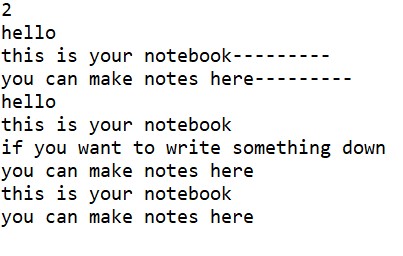
1. **使用容器类存储城市间的历程，并支持历程查询。**
2. **在我们上午实现的记事本程序的基础上，添加如下功能：“记录查询”功能，即输入一组关键字（String[] keywords）查询所有包括这组关键字的记录内容；“删除恢复”功能，即支持对最近n条删除的记录的恢复，恢复包括：内容恢复和存储位置恢复（插入到删除前的index）。**

二、实验运行结果

1.



2.



三、程序源代码

**package** distance;

**import** java.util.ArrayList;

**import** java.util.HashMap;

**import** java.util.Scanner;

**public** **class** CityDistance {

//创建一个顺序容器，用来存储城市的名字

**private** ArrayList<String> citylist = **new** ArrayList<String>();

//创建一个包含Hash表的Hash表。外部Hash表的前者表示第i个城市，内部Hash表示第j个城市及其与第i个城市之间的距离

**private** HashMap<String,HashMap<String,Integer>> distancelist = **new** HashMap<String,HashMap<String,Integer>> ();

**private** **static** Scanner *in* = **new** Scanner(System.***in***);

**public** **void** add() {

//读入城市名

String s;

**while**(**true**) {

s=*in*.next();

**if**(s.equals("###") == **true**) {//判断，当输入城市为###时，不再输入

**break**;

}

citylist.add(s);

}

//读入城市间距离

**for**(**int** i = 0;i < citylist.size();i++) {

HashMap<String,Integer> HM = **new** HashMap<String,Integer>();

**for**(**int** j = 0;j < citylist.size();j++){//依次录入各城市与第i个城市间距离

**int** d = *in*.nextInt();

HM.put(citylist.get(j), d);

}

distancelist.put(citylist.get(i), HM);

}

}

//查找两个城市之间的距离

**public** **int** getDistance(String start, String end){

**return** distancelist.get(start).get(end);

}

**public** **static** **void** main(String[] args) {

CityDistance citydistance = **new** CityDistance();

citydistance.add();

String start = *in*.next();

String end = *in*.next();

System.***out***.println(citydistance.getDistance(start, end));

}

}

**package** notebook;

**import** java.util.ArrayList;

**public** **class** Notebook {

**private** ArrayList<String> notebook = **new** ArrayList<String>();

**private** ArrayList<String> removelist = **new** ArrayList<String>();

**private** ArrayList<Integer> removenumber = **new** ArrayList<Integer>();

**public** **void** add(String note) {//添加至最后

notebook.add(note);

}

**public** **void** add(String note, **int** loc) {//添加至指定位置

notebook.add(loc, note);

}

**public** **int** getSize() {//获取记录条数

**return** notebook.size();

}

**public** String getNote(**int** index) {//获取指定记录

**return** notebook.get(index);

}

**public** **void** removeNote(**int** index) {//移除指定记录并记录其相关信息

removelist.add(notebook.remove(index));//按顺序记录移除内容

removenumber.add(index);//按顺序记录移除内容原本位置

}

**public** String[] list() {//列出所有记录

String[] a = **new** String[notebook.size()];

notebook.toArray(a);

**return** a;

}

**public** String[] search(String[] keywords) {//查找含有指定关键词的记录

String[] a = **new** String[notebook.size()];//先复制所有记录

notebook.toArray(a);

**for** (**int** i = 0; i < notebook.size(); i++) {//对复制的每条记录进行关键词检验

**for** (String k : keywords) {

**if**(notebook.get(i).contains(k)==**false**) {//若有关键词不包含于某条记录中，将该记录改为null，并退出关键词检验

a[i]=**null**;

**break**;

}

}

}

**return** a;//返回修改后的所有记录，即含所有关键词的记录

}

**public** **void** reset(**int** n) {//恢复最近删除的n条 记录

**for** (**int** i = removelist.size()-1; i >= removelist.size()-n; i--) {//从后往前依次恢复指定内容到指定位置

notebook.add(removenumber.get(i), removelist.get(i));

}

}

**public** **static** **void** main(String[] args) {

Notebook nb = **new** Notebook();

nb.add("hello");

nb.add("this is your notebook");

System.***out***.println(nb.getSize());

System.***out***.println(nb.getNote(0));

nb.removeNote(0);

System.***out***.println(nb.getNote(0)+"---------");

nb.add("you can make notes here");

nb.add("if you want to write something down", 1);

nb.removeNote(1);

System.***out***.println(nb.getNote(1)+"---------");

nb.reset(2);

String[] s = nb.list();

**for** (String k : s) {

System.***out***.println(k);

}

String[] t = {"note","you"};

String[] m = nb.search(t);

**for** (String k : m) {

**if**(k!=**null**)

System.***out***.println(k);

}

}

}