**Conversion:**

**(Sample.java)**

**package practice.collectiontasks;**

**import java.io.BufferedReader;**

**import java.io.FileReader;**

**import java.io.IOException;**

**import java.util.ArrayList;**

**import java.util.HashMap;**

**import java.util.List;**

**import java.util.Map;**

**public class Sample {**

**public static HashMap<String, Student> mapFunction(String regNo,String subjectCode,String subjectName,String internalMarks,String externalMarks,String credits,HashMap<String, Student> hm2,String status)**

**{**

**HashMap<String, Student> hm1 = new HashMap<String, Student>();**

**hm1.putAll(hm2);**

**Student st = new Student();**

**List<String> subjectCodeList = new ArrayList<String>();**

**List<String> subjectNameList = new ArrayList<String>();**

**List<Double> marksList = new ArrayList<Double>();**

**List<Integer> creditsList = new ArrayList<Integer>();**

**List<String> statusList = new ArrayList<String>();**

**if(hm1.containsKey(regNo))**

**{**

**st=hm1.get(regNo);**

**subjectCodeList=st.getSubjectCodeList();**

**subjectNameList=st.getSubjectNameList();**

**marksList=st.getMarksList();**

**creditsList=st.getCreditsList();**

**statusList=st.getStatusList();**

**subjectCodeList.add(subjectCode);**

**subjectNameList.add(subjectName);**

**double marks = Double.parseDouble(internalMarks)+Double.parseDouble(externalMarks);**

**marksList.add(marks);**

**creditsList.add(Integer.parseInt(credits));**

**statusList.add(status);**

**st.setSubjectCodeList(subjectCodeList);**

**st.setSubjectNameList(subjectNameList);**

**st.setMarksList(marksList);**

**st.setCreditsList(creditsList);**

**st.setStatusList(statusList);**

**hm1.put(regNo, st);**

**}**

**else**

**{**

**subjectCodeList.add(subjectCode);**

**subjectNameList.add(subjectName);**

**double marks = Double.parseDouble(internalMarks)+Double.parseDouble(externalMarks);**

**marksList.add(marks);**

**creditsList.add(Integer.parseInt(credits));**

**statusList.add(status);**

**st.setSubjectCodeList(subjectCodeList);**

**st.setSubjectNameList(subjectNameList);**

**st.setMarksList(marksList);**

**st.setCreditsList(creditsList);**

**st.setStatusList(statusList);**

**hm1.put(regNo, st);**

**}**

**return hm1;**

**}**

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

**FileReader fr = new FileReader("D://notepad//checkdata.txt");**

**BufferedReader br = new BufferedReader(fr);**

**String line;**

**HashMap<String,HashMap<String, Student>> hm = new HashMap<String, HashMap<String,Student>>();**

**while((line=br.readLine())!=null)**

**{**

**String arr[] = line.split(" ");**

**StringBuffer sb = new StringBuffer();**

**int j=0;**

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

**if(!(arr[i].contains("0")||arr[i].contains("1")||arr[i].contains("2")||arr[i].contains("3")||arr[i].contains("4")||arr[i].contains("5")||arr[i].contains("6")||arr[i].contains("7")||arr[i].contains("8")||arr[i].contains("9")))**

**{**

**sb.append(arr[i]+" ");**

**j=i;**

**}**

**String subjectName=sb.toString();**

**String dept=null;**

**String subjectCode=arr[1];**

**String internalMarks = arr[j+1];**

**String externalMarks = arr[j+2];**

**String credits = arr[j+3];**

**String status="pass";**

**if(Integer.parseInt(credits)==0||Integer.parseInt(credits)==-1)**

**status="fail";**

**switch(arr[0].charAt(7))**

**{**

**case '2':**

**dept="EEE";**

**break;**

**case '3':**

**dept="MECH";**

**break;**

**case '4':**

**dept="ECE";**

**break;**

**case '5':**

**dept="CSE";**

**break;**

**}**

**HashMap<String, Student> hm1 = new HashMap<String, Student>();**

**String regNo=arr[0];**

**if(hm.containsKey(dept))**

**{**

**hm1=hm.get(dept);**

**hm1.putAll(mapFunction(regNo, subjectCode, subjectName, internalMarks, externalMarks, credits,hm1,status));**

**hm.put(dept, hm1);**

**}**

**else**

**{**

**hm1=mapFunction(regNo, subjectCode, subjectName, internalMarks, externalMarks, credits,hm1,status);**

**hm.put(dept, hm1);**

**}**

**}**

**//System.out.println(hm);**

**for(Map.Entry<String, HashMap<String, Student>> en:hm.entrySet())**

**{**

**String dept=en.getKey();**

**HashMap<String,Student> hm1=en.getValue();**

**for(Map.Entry<String, Student> en1:hm1.entrySet())**

**{**

**String regNo=en1.getKey();**

**Student st=en1.getValue();**

**List<String> subjectCodeList=st.getSubjectCodeList();**

**List<String> subjectNameList=st.getSubjectNameList();**

**List<Double> marksList=st.getMarksList();**

**List<Integer> creditsList=st.getCreditsList();**

**List<String> statusList = st.getStatusList();**

**if(statusList.contains("fail"))**

**System.out.println("Student Failed");**

**else**

**System.out.println("Student Passed");**

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

**{**

**System.out.println(dept+" "+regNo+" "+subjectCodeList.get(i)+" "+subjectNameList.get(i)+" "+marksList.get(i)+" "+creditsList.get(i)+" "+statusList.get(i));**

**}**

**}**

**}**

**br.close();**

**fr.close();**

**}**

**}**

**(Student.java)**

**package** practice.collectiontasks;

**import** java.util.List;

**public** **class** Student {

List<String> subjectCodeList;

List<String> subjectNameList;

List<Double> marksList;

List<Integer> creditsList;

List<String> statusList;

**public** List<String> getStatusList() {

**return** statusList;

}

**public** **void** setStatusList(List<String> statusList) {

**this**.statusList = statusList;

}

**public** List<String> getSubjectCodeList() {

**return** subjectCodeList;

}

**public** **void** setSubjectCodeList(List<String> subjectCodeList) {

**this**.subjectCodeList = subjectCodeList;

}

**public** List<String> getSubjectNameList() {

**return** subjectNameList;

}

**public** **void** setSubjectNameList(List<String> subjectNameList) {

**this**.subjectNameList = subjectNameList;

}

**public** List<Double> getMarksList() {

**return** marksList;

}

**public** **void** setMarksList(List<Double> marksList) {

**this**.marksList = marksList;

}

**public** List<Integer> getCreditsList() {

**return** creditsList;

}

**public** **void** setCreditsList(List<Integer> creditsList) {

**this**.creditsList = creditsList;

}

}