Name: Zacharie Happel

CST141-FA2023

Project for Week 08

Due: 10/29/2023

Overview of Approach:

I hope that I did this as the project intended. I created a Java file called Employees.java which, within the main method, orchestrated the ingestion of the data and created the EmployeeData objects.

As instructed, the compareTo() employs a comparison on the EmployeeData emplID instance variables. One issue that I encountered was in the original code that we obtained from the textbook, first there was a comparison between the two object’s deptNum. This resulted in a final list that was close to being sorted, but not entirely. Seeing that this was not a part of the comparison that our result was supposed to include and being that it was messing up the final output, I removed it so that just emplID was compared.

# Source Code for EmployeeData and Employee Records Classes:

**EmployeeData.java**

public class EmployeeData implements Comparable<EmployeeData> {

private String firstName; // First Name

private String lastName; // Last Name

private Integer emplID; // Employee ID

private Integer deptNum; // Department Number

EmployeeData(String firstName, String lastName, Integer emplID, Integer deptNum) {

this.firstName = firstName;

this.lastName = lastName;

this.emplID = emplID;

this.deptNum = deptNum;

}

@Override

public int compareTo(EmployeeData otherEmpl) {

//String fullName; // Full name, this employee

//String otherFullName; // Full name, comparison employee

int comparisonVal; // Outcome of comparison

// Compare based on department number first

//comparisonVal = deptNum.compareTo(otherEmpl.deptNum);

// If in same organization, use name

//if (comparisonVal == 0) {

//fullName = lastName + firstName;

//otherFullName = otherEmpl.lastName + otherEmpl.firstName;

comparisonVal = emplID.compareTo(otherEmpl.emplID);

//}

return comparisonVal;

}

@Override

public String toString() {

return lastName + " " + firstName +

" \tID: " + emplID +

"\t\tDept. #: " + deptNum;

}

}

**Employees.java**

import java.util.ArrayList;

import java.util.Scanner;

import java.io.FileInputStream;

import java.io.IOException;

import java.util.Collections;

// 0 if equal, -1 of comparableObjext les than otherComparable, 1 if comparable

public class Employees {

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

// Create ArrayList of object EmployeeData

ArrayList<EmployeeData> EmployeesList = new ArrayList<>();

int lineCount = 0;

// Make sure argument is a textfile

if (!(args[0].endsWith(".txt"))) throw new Exception("Invalid file name");

// Assign filename argument to variable, create file input bytestream and attach scanner

String fileName = args[0];

FileInputStream fileByteStream = new FileInputStream(fileName);

Scanner inFS = new Scanner(fileByteStream);

// While loop, while next line exists,

// Ingest line, split it on commas (,) and directly add as an EmployeeData object to EmployeesList

System.out.print("\nEmployees - Sorting EmployeeData Objects\n\n" + "Reading from file " + fileName + "...\n");

while(inFS.hasNext()) {

lineCount++;

String line = inFS.nextLine();

String[] lineArr = line.split(",");

EmployeesList.add(new EmployeeData(lineArr[0], lineArr[1], Integer.valueOf(lineArr[2]), Integer.valueOf(lineArr[3])));

//System.out.println("Line arr:" + lineArr.toString());

}

System.out.println("\nLine Count: " + lineCount + "\nEmployeeData objects amount: " + EmployeesList.size() + "\n\nUnsorted: ");

for (int i = 0; i <= EmployeesList.size() - 1; i++) {

System.out.println(EmployeesList.get(i));

}

Collections.sort(EmployeesList);

System.out.println("\n\nSorted: ");

for (int i = 0; i <= EmployeesList.size() - 1; i++) {

System.out.println(EmployeesList.get(i));

}

}

}

Screenshot(s) of Output:

