//Student's Full name: Tasfique Enam

//Student's ID: 5886429

//Modification Date: 16/05/17

//Purpose of this file: driver class.

package assignment2;

/\*\*

\*

\* @author Tasfique

\*/

import java.io.\*;

import java.util.Scanner;

public class driver {

private static int size; //size is the static variable used for counter loop...

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

int i;

int choiceentry;

Student [] students = new Student[30]; //creating an array of 30 objects.

Scanner read = new Scanner(System.in);

for(i= 0;i<30;i++){

students[i] = new Student();//the objects

}

do{

System.out.println(); //display of menu

System.out.println("1. Read data from the file ");

System.out.println("2. Write data to the file ");

System.out.println("3. Display all student's information ");

System.out.println("4. Change student's data ");

System.out.println("5. Display student with the highest and the lowest nark ");

System.out.println("6. Display all the grades. ");

System.out.println("7. Display the average mark for the subject ");

System.out.println("0. To exit the program ");

System.out.println();

System.out.println("Please enter the number from your selection ");

choiceentry = read.nextInt();

switch(choiceentry){ //switch statement is used for the selection of the menu, to know which method to execute.

case 1: ReadFile(students);

break;

case 2: WriteFile(students);

break;

case 3: DisplayAll(students);

break;

case 4: ChangeData(students);

break;

case 5: DisplayHighLow(students);

break;

case 6: DisplayAllGrades(students);

break;

case 7: DisplayAvgMark(students);

break;

}

}while(choiceentry!=0);

}

public static int ReadFile(Student[]students)throws IOException{ //it is used to read student data from the text file.

String student\_name;

int i = 0;

int student\_id;

double practicalmark,exammark,assignmentmark;

File creating = new File("data.txt"); //filewriter avoids erasing a file that already exist.

Scanner read = new Scanner(creating); //creating is the file data.txt, it is directing to the txt file

while(read.hasNext()){

student\_name = read.next(); //reading text as name

student\_id = read.nextInt();//reading the text as id

students[i].setDetails(student\_name, student\_id);

practicalmark = read.nextDouble();

students[i].setPractical(practicalmark);

for(int y=0; y<4; y++){

assignmentmark = read.nextDouble();

students[i].setAssignment(assignmentmark, y); //it will read the next 4 lines as assignments marks

}

exammark = read.nextDouble(); //reading text for exam mark

students[i].setExam(exammark);

i++;

size++; //counter

}

System.out.println();

System.out.println("Your files has been read.");

System.out.println();

return size;

}

public static void WriteFile(Student[]array) throws IOException{ //void is used for not returning any values.

PrintWriter pw = new PrintWriter("data.txt"); // it writes new data to the txt file, and also overwrites.

for(int x=0; x<size; x++){

pw.println(array[x].getName());

pw.println(array[x].getID());

pw.println(array[x].getPracticalMark());

for(int y = 0; y<4; y++){

pw.println(y);

pw.println(array[x].getAsgMark(y)); //print writer for the new assignment mark

}

pw.println(array[x].getExamMark()); //print writer for the new exam mark

}

System.out.println();

System.out.println("Your data has been written ");

System.out.println();

}

public static void DisplayAll(Student[]array)throws IOException{ //to display all the student's data

for(int j=0;j<size;j++){

System.out.println();

System.out.println("Student name: "+array[j].getName());

System.out.println("Student ID: "+array[j].getID());

System.out.println("Practical Marks: "+array[j].getPracticalMark());

for(int i=0;i<4;i++){ //for loop for 4 different assignments.

System.out.println("Assignment mark " + (i+1) + " : "+array[j].getAsgMark(i));

}

System.out.println("Total Assignment Marks: " + array[j].getTotalAsgMark());

System.out.println("Exam mark: " + array[j].getExamMark());

System.out.println("Overall mark: " + array[j].getOvarallMark());

System.out.println("The Grade is : " + array[j].getGrade());

if(array[j].isTechnicalFail()){ //to check if the student has technically failed or not?

System.out.println("This student has Technically Failed. ");

}else{

System.out.println("This Student has passed. ");

}

}

}

public static void ChangeData(Student[]array){ //method for changing the data for students data

String student;

int option,assignment\_num;

double practical\_mark,assignment\_mark,exam\_mark;

Scanner read = new Scanner(System.in);

System.out.println("Enter name of the student, whose data you would like to change? ");

student = read.next();

for(int x=0;x<size;x++){

if(student.equals(array[x].getName())){

System.out.println("Which data would you like to change? "); //asking for which mark the user likes to change

System.out.println("1. Practical Mark ");

System.out.println("2. Assignment Mark ");

System.out.println("3. Exam Mark ");

System.out.println("0. Exit");

option = read.nextInt();

switch(option){ //switch statement for excuteting the option the user selected.

case 1:

System.out.println("Enter the new practical mark you like to change: ");

practical\_mark = read.nextDouble();

array[x].setPractical(practical\_mark);

break;

case 2:

System.out.println("Enter assignment number you would like to change: ");

assignment\_num = read.nextInt();

assignment\_num = assignment\_num - 1;

System.out.println("Enter the Assignment " + assignment\_num + " mark you would like to change: ");

assignment\_mark = read.nextDouble();

array[x].setAssignment(assignment\_mark, assignment\_num);

break;

case 3:

System.out.println("Please enter the new exam marks: ");

exam\_mark = read.nextDouble();

array[x].setExam(exam\_mark);

break;

case 0:

return;

}

}else{

System.out.println("New data successfully written ");

return;

}

}

}

public static void DisplayHighLow(Student[]array){ //method for displaying high and low

double low = 1000;

double high = 0;

String namelow = null;

String namehigh = null;

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

if( array[i].getOvarallMark()<array[i+1].getOvarallMark()){

if(array[i].getOvarallMark()<(low-array[i].getOvarallMark())){

low = array[i].getOvarallMark();

namelow = array[i].getName();

}

}

}

for(int j=0;j<size;j++){

if( array[j].getOvarallMark()>array[j+1].getOvarallMark()){

if(array[j].getOvarallMark()-high>0){

high = array[j].getOvarallMark();

namehigh = array[j].getName();

}

}

}System.out.println("The lowest mark is "+ namelow +" "+ low );

System.out.println("The highest mark is "+ namehigh +" "+ high );

}

public static void DisplayAllGrades(Student[]array){ //displaying all the grades for the student

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

array[i].getOvarallMark();//get ovarallmark for students, it is connected to each other because the grades needs ovarall mark.

System.out.println("The Grade for the Students " + array[i].getName()+ " are " + array[i].getGrade());

}

}

public static void DisplayAvgMark(Student[]array){ //displaying avg marks for the students.

double average;

double total=0;

for(int y=0;y<size;y++){

total = total + array[y].getOvarallMark(); //for loop for checkiong the overall mark.

}

average = total/size;

System.out.println("The average mark for all the students is " + average);//displays out the avg mark

}

}

//Student's Full name: Tasfique Enam

//Student's ID: 5886429

//Modification Date:

//Purpose of this file: Student class

package assignment2;

/\*\*

\*

\* @author Tasfique

\*/

public class Student {

//declaring attributes.

private String name;

private int id;

private double practical\_mark, exam\_mark;

private double [] assignment\_mark = new double [4];

double total\_assignment;

//default constructor for setting default values.

public Student(){

name = "";

id = 0;

practical\_mark = 0;

assignment\_mark[0]=0;

assignment\_mark[1]=0;

assignment\_mark[2]=0;

assignment\_mark[3]=0;

exam\_mark = 0;

}

//accessor methods

public void setDetails(String name, int id){

this.name = name;

this.id = id;

}

public void setPractical(double practical\_mark){

this.practical\_mark = practical\_mark; //assigning practical marks.

while (practical\_mark>10){ // if the mark is greater than 10, it will set the mark to 0

this.practical\_mark = 0;

}

}

public void setAssignment(double assignment\_mark, int assignment\_number){//assignment method

this.assignment\_mark[assignment\_number] = assignment\_mark;

if(this.assignment\_mark[assignment\_number]>10 || this.assignment\_mark[assignment\_number]<1){

this.assignment\_mark[assignment\_number] = 0;

}

}

public void setExam(double exam\_mark){ //set exam method

this.exam\_mark = exam\_mark;

if(exam\_mark>100 || exam\_mark <0){

this.exam\_mark = 0;

}

} //mutator methods.

String getName(){ //get name method

return name;

}

int getID(){ //get ID method

return id;

}

double getPracticalMark(){ //get practical method

return practical\_mark;

}

double getAsgMark(int i){ //get assignment mark method

return this.assignment\_mark[i];

}

double getTotalAsgMark(){ //total assignment mark method

double total\_assignment = 0;

for(int y=0;y<4;y++){

total\_assignment = total\_assignment + this.assignment\_mark[y];

}

return total\_assignment;

}

double getExamMark(){ //get exam mark method

return exam\_mark;

}

double getOvarallMark(){ //get overall mark method

double ovarall = practical\_mark + getTotalAsgMark() + (exam\_mark\*(0.5));

return ovarall;

}

String getGrade(){ //get grade method

String grade = null; //followed UOW grading scheme.

if(getOvarallMark()>=85 && getOvarallMark()<=100){

grade = "HD";

}

else if(getOvarallMark()>=75 && getOvarallMark()<=84){

grade = "D";

}

else if(getOvarallMark()>=65 && getOvarallMark()<=74){

grade = "C";

}

else if(getOvarallMark()>=50 && getOvarallMark()<=64){

grade = "P";

}

else if(getOvarallMark()<=49){

grade = "F";

}

return grade;

}

boolean isTechnicalFail(){ //technical fail method.

if(getOvarallMark()>=50 && exam\_mark<40){ //if the overall mark is greater than 50 but exam is less than 40

return true;

}

else{

return false;

}

}

}















