Darshana pubudu keerthirathna

ICM 106 OR23106564

PROGRAMMING FUNDAMENTALS WEEK – 08 ASSIGNMENT

**Notes:-**

I created tested this codes on single java file. I did not use methods because I had a limited time before submission. Rather than forces on the good structure, I focus on solving all the questions in given time.

Full codes can be found on following address.

<https://github.com/keerthimac/ICM106/blob/master/Programming%20Fundamentals/week_08/Assignment_08/source/Example.java>

The codes included in this document originally created on above file.

Thanks!

**Question 01**

import java.util.\*;

class Example{

public static void main(String args[]){

String[] months = {"January", "February", "March", "April", "May", "June", "July", "August",

"September", "October", "November", "December"};

int[] daysInMonth = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

//Create multidimensional array with Month count

int[][] patientCounts = new int[12][];

for (int i = 0; i < 12; i++){

patientCounts[i] = new int[daysInMonth[i]];

}

//Input Paitent count via Random numbers

Random r = new Random();

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

for (int j = 0; j < daysInMonth[i]; j++){

patientCounts[i][j]=r.nextInt(501);

}

}

// i Print for Given Dates

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

for (int j = 0; j < daysInMonth[i]; j++){

if((i==0&&j==16)||(i==2&&j==25)||(i==4&&j==0)||(i==6&&j==20)||(i==10&&j==29))

System.out.println("2021-"+(i+1)+"-"+(j+1)+" : "+patientCounts[i][j]);

}

}

//Program Continues…

**Question 02**

//ii Update for Given Dates ant Test

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

for (int j = 0; j < daysInMonth[i]; j++){

if(i==1&&j==15){

patientCounts[i][j]=78;

System.out.println(patientCounts[i][j]);

}else if(i==5&&j==4){

patientCounts[i][j]=202;

System.out.println(patientCounts[i][j]);

}else if(i==8&&j==8){

patientCounts[i][j]=181;

System.out.println(patientCounts[i][j]);

}else if(i==9&&j==9){

patientCounts[i][j]=178;

System.out.println(patientCounts[i][j]);

}else if(i==11&&j==24){

patientCounts[i][j]=84;

System.out.println(patientCounts[i][j]);

}

}

}

//Program Continues…

**Question 03**

//iii Print for Given Dates ant Test

System.out.println();

for (int i = 0; i <1; i++){

for (int j = 0; j < 10; j++){

System.out.println(

"Patient Count of 2021-"+(i+1)+"-"+(j+1)+" : "+patientCounts[i][j]

);

}

}

//Program Continues…

**Question 04**

//iv Find the total number of patients reported in the first week of 2021.

System.out.println();

int pt1stTenTotal = 0;

for (int i = 0; i <1; i++){

int total = 0;

for (int j = 0; j < 10; j++){

total+=patientCounts[i][j];

}

pt1stTenTotal=total;

}

System.out.println("Patient Count of first 10 days in 2021 : "+pt1stTenTotal);

//Program Continues…

**Question 05**

//v Find the total patient count reported in the first 10 days of March.

System.out.println();

int marchTenTotal = 0;

for (int i = 3; i <4; i++){

int total = 0;

for (int j = 0; j < 10; j++){

total+=patientCounts[i][j];

}

marchTenTotal=total;

}

System.out.println("Patient Count of first 10 days in 2021 March : "+marchTenTotal);

//Program Continues…

**Question 06**

//vi Find the total patient count reported in the last 10 days of October.

System.out.println();

int octTenTotal = 0;

for (int i = 2; i <3; i++){

int total = 0;

for (int j = 0; j < 10; j++){

total+=patientCounts[i][j];

}

octTenTotal=total;

}

System.out.println("Patient Count of first 10 days in 2021 October : "+octTenTotal);

//Program Continues…

**Question 07**

//vii Find the total patient count reported in June.

System.out.println();

int juneTotal = 0;

for (int i = 5; i <6; i++){

int total = 0;

for (int j = 0; j < daysInMonth[i]; j++){

total+=patientCounts[i][j];

}

juneTotal=total;

}

System.out.println("Patient Count of June : "+juneTotal);

//Program Continues…

**Question 08**

//viii Count the number of days in which more than 100 patients reported in January.

System.out.println();

int patientCount = 0;

for (int i = 0; i <1; i++){

int count = 0;

for (int j = 0; j < daysInMonth[i]; j++){

if(patientCounts[i][j]>100){

count++;

}

}

patientCount=count;

}

System.out.println("Number of days Repoted more than 100 in January : "+patientCount);

//Program Continues…

**Question 09**

//ix Find the total patient count reported in June.

System.out.println();

for (int i = 0; i <12; i++){

for (int j = 0; j < 1; j++){

System.out.println("Patient Count 1st of "+months[i]+" : "+patientCounts[i][j]);

}

}

//Program Continues…

**Question 10**

//x Print the number of patients reported on the last day of each month with the month's name.

System.out.println();

for (int i = 0; i <12; i++){

for (int j = daysInMonth[i]-1; j > daysInMonth[i]-2; j--){

System.out.println("Patient Count Last day of "+months[i]+" : "+patientCounts[i][j]);

}

}

//Program Continues…

**Question 11**

//xi Print the number of patients reported on the last day of each month with the month's name.

System.out.println();

int[] totOfTen = new int[12];

for (int i = 0; i <12; i++){

int total = 0;

for (int j = 0; j < 10; j++){

total+=patientCounts[i][j];

}

totOfTen[i]=total;

}

//-----Printing Part----------

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

System.out.println("Total number of patients reported first 10 days in "+months[i]+" : "+totOfTen[i]);

}

//Program Continues…

**Question 12**

//xii Find the average number of patients reported on the 15th of every month.

System.out.println();

int totOfFifteen = 0;

for (int i = 0; i <12; i++){

int total = 0;

for (int j = 14; j < 15; j++){

total+=patientCounts[i][j];

}

totOfFifteen+=total;

}

System.out.println("Patient avarage 15th of every month : "+totOfFifteen/12);

//Program Continues…

**Question 13**

//xiii Find the total number of patients reported in 2021.

System.out.println();

int totAllYear = 0;

for (int i = 0; i <12; i++){

int total = 0;

for (int j = 0; j < daysInMonth[i]; j++){

total+=patientCounts[i][j];

}

totAllYear+=total;

}

System.out.println("Total Patients All year : "+totAllYear);

//Program Continues…

**Question 14**

//xiv Find the average number of patients reported in one day in 2021.

System.out.println();

int[] avgPerDay = new int[12];

for (int i = 0; i <12; i++){

int total = 0;

for (int j = 0; j < daysInMonth[i]; j++){

total+=patientCounts[i][j];

}

avgPerDay[i]=total/daysInMonth[i];

}

//-----Printing Part----------

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

System.out.println("Average number of patients reported in one day in "+months[i]+" : "+avgPerDay[i]);

}

//Program Continues…

**Question 15**

//xv Find the first date of patients reported more than 200.

System.out.println();

L1:for (int i = 0; i <12; i++){

for (int j = 0; j < daysInMonth[i]; j++){

if(patientCounts[i][j]>200){

System.out.println("First date of patients reported more than 200 is : 2021-"+(i+1)+"-"+(j+1));

break L1;

}

}

}

//Program Continues…

**Question 16**

//xvi Count the number of days in which more than 200 patients were reported.

System.out.println();

int patientTwoHun = 0;

for (int i = 0; i <12; i++){

int count = 0;

for (int j = 0; j < daysInMonth[i]; j++){

if(patientCounts[i][j]>200){

count++;

}

}

patientTwoHun+=count;

}

System.out.println("Number of days in which more than 200 patients were reported are : "+patientTwoHun);

//Program Continues…

**Question 17**

//xvii Print the total number of patients reported in each month with the name of the month.

System.out.println();

int[] totForEachMo = new int[12];

for (int i = 0; i <12; i++){

int total = 0;

for (int j = 0; j < daysInMonth[i]; j++){

total+=patientCounts[i][j];

}

totForEachMo[i]=total;

}

//-----Printing Part----------

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

System.out.println("Total number of patients reported in "+months[i]+" : "+totForEachMo[i]);

}

//Program Continues…

**Question 18**

//xviii Days have the number of patients reported in more than 200 of each month

System.out.println();

int[] totTwHunEachMo = new int[12];

for (int i = 0; i <12; i++){

int count = 0;

for (int j = 0; j < daysInMonth[i]; j++){

if(patientCounts[i][j]>200)

count++;

}

totTwHunEachMo[i]=count;

}

//-----Printing Part----------

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

System.out.println("Total number days reported more than 200 in "+months[i]+" : "+totTwHunEachMo[i]);

}

//Program Continues…

**Question 19**

//xix highest number of patients count in April

System.out.println();

int maxApril = 0;

for (int i = 3; i <4; i++){

int max = patientCounts[i][0];

for (int j = 1; j < daysInMonth[i]; j++){

if(patientCounts[i][j]>max)

max=patientCounts[i][j];

}

maxApril=max;

}

//-----Printing Part----------

System.out.println("highest number of patients count in April is : "+maxApril);

//Program Continues…

**Question 20**

//xx highest number of patients count in 2021

System.out.println();

int maxForYear = 0;

for (int i = 0; i <12; i++){

for (int j = 1; j < daysInMonth[i]; j++){

if(patientCounts[i][j]>maxForYear)

maxForYear=patientCounts[i][j];

}

}

//-----Printing Part----------

System.out.println("highest number of patients count in 2021 is : "+maxForYear);

//Program Continues…

**Question 21**

//xxi minimum number of patients count in 2021 reported

System.out.println();

int[] totForEachMo1 = new int[12];

for (int i = 0; i <12; i++){

int total = 0;

for (int j = 0; j < daysInMonth[i]; j++){

total+=patientCounts[i][j];

}

totForEachMo1[i]=total;

}

//-----Printing Part----------

int minMonth = totForEachMo1[0];

int monIndex = 0;

for (int i = 0; i < totForEachMo1.length; i++){

if(minMonth>totForEachMo1[i]){

minMonth = totForEachMo1[i];

monIndex = i;

}

}

System.out.println("minimum number of patients count in 2021 reported On "+months[monIndex]);

//Program Continues…

**Question 22**

//xxii number of days that have passed to the highest number of patients reported.

System.out.println();

int maxPatient = 0;

int maxday=0;

int maxMonth=0;

int count=0;

for (int i = 0; i <12; i++){

for (int j = 0; j < daysInMonth[i]; j++){

if(patientCounts[i][j]>maxPatient){

maxPatient=patientCounts[i][j];

maxMonth=i;

maxday =j;

}

}

}

int countDays=0;

L2:for (int i = 0; i <12 ;i++){

for (int j = 0; j < daysInMonth[i]; j++){

if(i==maxMonth&&j==maxday){

break L2;

}

countDays++;

}

}

System.out.println(countDays+"number of days that have passed to the highest number of patients reported.");

//Program Continues…

**Question 23**

//xxiii highest number of patients reported in each month

System.out.println();

int[] maxPatientCount = new int[12];

for (int i = 0; i <12; i++){

int maxPerMonth = 0;

for (int j = 0; j < daysInMonth[i]; j++){

if(patientCounts[i][j]>maxPerMonth){

maxPerMonth=patientCounts[i][j];

}

}

maxPatientCount[i]=maxPerMonth;

}

//-----Printing Part----------

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

System.out.println("highest number of patients reported in "+months[i]+" : "+maxPatientCount[i]);

}

//Program Continues…

**Question 24**

//xxiv Minimum number of patients reported in each month

System.out.println();

int[] minPatientCount = new int[12];

for (int i = 0; i <12; i++){

int minPerMonth = 500;

for (int j = 0; j < daysInMonth[i]; j++){

if(patientCounts[i][j]<minPerMonth){

minPerMonth=patientCounts[i][j];

}

}

minPatientCount[i]=minPerMonth;

}

//-----Printing Part----------

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

System.out.println("Minimum number of patients reported in "+months[i]+" : "+minPatientCount[i]);

}

//Program Continues…

**Question 25**

//xxv number of patients reported in November in descending order

System.out.println();

int[] novPatientCount = new int[30];

int[] novDates = new int[30];

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

novPatientCount[i]=patientCounts[10][i];

novDates[i]=i;

}

//-----Sorting----------------

for (int i = novPatientCount.length-1; i>0; i--){

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

if(novPatientCount[j]<novPatientCount[j+1]){

int temp = novPatientCount[j];

novPatientCount[j] = novPatientCount[j+1];

novPatientCount[j+1]=temp;

int tempIndex = novDates[j];

novDates[j] = novDates[j+1];

novDates[j+1]=tempIndex;

}

}

}

//-----Printing Part----------

System.out.println("number of patients reported in November decending");

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

System.out.println("Nov "+(novDates[i]+1)+" - "+novPatientCount[i]);

}

//Program Continues…

**Question 26**

//xxvi number of patients reported in August in ascending order with date.

System.out.println();

int[] augPatientCount = new int[31];

int[] augDates = new int[31];

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

augPatientCount[i]=patientCounts[7][i];

augDates[i]=i;

}

//-----Sorting----------------

for (int i = augPatientCount.length-1; i>0; i--){

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

if(augPatientCount[j]>augPatientCount[j+1]){

int temp = augPatientCount[j];

augPatientCount[j] = augPatientCount[j+1];

augPatientCount[j+1]=temp;

int tempIndex = augDates[j];

augDates[j] = augDates[j+1];

augDates[j+1]=tempIndex;

}

}

}

//-----Printing Part----------

System.out.println("number of patients reported in August acending");

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

System.out.println("Aug "+(augDates[i]+1)+" - "+augPatientCount[i]);

}

//Program Continues…

**Question 27**

//xxvii total number of patients reported in each month in descending order with the name of month.

System.out.println();

int[] totalForEachMonth = new int[12];

int[] totalMo = new int[12];

for (int i = 0; i <12; i++){

int total = 0;

for (int j = 0; j < daysInMonth[i]; j++){

total+=patientCounts[i][j];

}

totalForEachMonth[i]=total;

totalMo[i]=i;

}

//-----Sorting----------------

for (int i = totalForEachMonth.length-1; i>0; i--){

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

if(totalForEachMonth[j]<totalForEachMonth[j+1]){

int temp1 = totalForEachMonth[j];

totalForEachMonth[j] = totalForEachMonth[j+1];

totalForEachMonth[j+1]=temp1;

int tempIndex1 = totalMo[j];

totalMo[j] = totalMo[j+1];

totalMo[j+1]=tempIndex1;

}

}

}

//-----Printing Part----------

System.out.println();

System.out.println("Number of patients reported for Each Month decending");

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

System.out.println(months[totalMo[i]]+" - "+totalForEachMonth[i]);

}

**Question 28**

//xxviii & xxix Graph

System.out.println();

System.out.println("-------------------------------------------------------------------------------------------------------------");

System.out.println("\t\t\t\tCOVID-19 patient in 2021");

System.out.println("-------------------------------------------------------------------------------------------------------------");

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

System.out.printf("%-8s","");

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

System.out.printf("%-10s",months[i]);

}

System.out.println();

//Print Table

for (int i = 0; i <31 ; i++){

System.out.printf("%-8d",i+1);

for (int j = 0; j <12 ; j++){

if ((i==28&&j==1)||(i==29&&j==1)||(i==30&&j==1)||(i==30&&j==3)||(i==30&&j==5)||(i==30&&j==8)||(i==30&&j==10)){

System.out.printf("%-10s","");

}else{

System.out.printf("%-10s",patientCounts[j][i]);

}

}

System.out.println();

}

//Find Total

System.out.println();

int[] ptTotal = new int[12];

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

int total = 0;

for (int j = 0; j <daysInMonth[i] ; j++){

total+=patientCounts[i][j];

};

ptTotal[i]=total;

}

//Print Total

System.out.println();

System.out.printf("%-8s","Total");

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

System.out.printf("%-10s",ptTotal[i]);

}

//Find Avarage

System.out.println();

int[] ptAvarage = new int[12];

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

int total = 0;

for (int j = 0; j <daysInMonth[i] ; j++){

total+=patientCounts[i][j];

};

ptAvarage[i]=total/daysInMonth[i];

}

//Print Avarage

System.out.printf("%-8s","Avarage");

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

System.out.printf("%-10s",ptAvarage[i]);

}

}

}