Problem One:

Write a program that initializes an array with ten random integers and then prints out the following:

* Every element at an even index;
* Every even element
* All elements in reverse order;
* Only the first and last elements;
* The minimum and maximum element
* The sum of all elements
* The alternating sum of all elements, where the alternating sum contains all elements at even index added, and the elements at odd index subtracted.

Solution:

import java.util.Random;

public class ProblemOne {

public static void main(String[] args){

Random rand = new Random();

int array[] = new int[10];

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

array[i]=rand.nextInt(20);

}

int max = array[0];

int min = array[0];

int sum=0;

int even=0;

System.out.println("Every Element at an Even Index:");

for(int j=0;j<array.length;j++){

if(j%2==0){

System.out.println("Index: " + j + ":\t" + array[j]);

even = even+array[j];

}

}

System.out.println("Every Even Element:");

for(int l=0;l<array.length;l++){

if(array[l]%2==0)

System.out.println("Index: " + l + ":\t" + array[l]);

}

System.out.println("All Elements in Reverse Order:");

for(int k=(array.length-1);k>=0;k--){

System.out.println("Index: " + k + ":\t" + array[k]);

}

for(int m=0;m<array.length;m++){

if(array[m]<min)

min = array[m];

if(array[m]>max)

max = array[m];

sum = sum + array[m];

}

int odd = even;

for(int n=0;n<array.length;n++){

if(n%2!=0){

odd = odd-array[n];

}

}

System.out.println("Max: " + max);

System.out.println("Min: " + min);

System.out.println("Sum: " + sum);

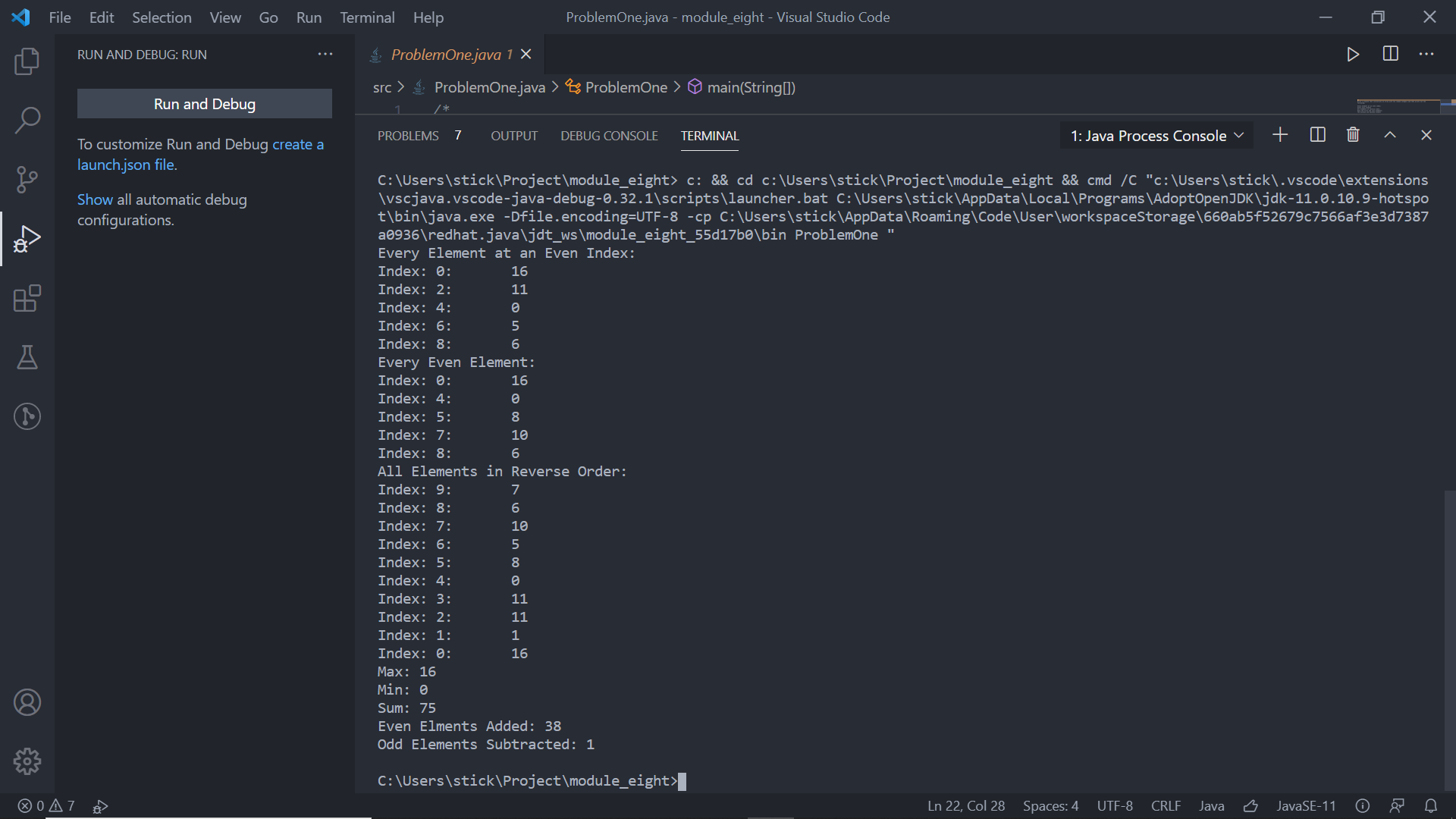
System.out.println("Even Elments Added: " + even);

System.out.println("Odd Elements Subtracted: " + odd);

}

}

Screenshot:



Problem Two:  
Write a program that reads a sequence of input values and displays a bar chart of the values in data, using asterisks, like this:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

You may assume that all values are positive. First figure out the maximum value in data. That value's bar should be drawn with 40 asterisks. Shorter bars should use proportionally fewer asterisks. Improve your program by adding caption to your bar. Prompt the user for the captions and data values. The output should look like this:

Egypt \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

France \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Norway \*\*\*\*\*\*\*\*\*

Germany \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Solution:

import java.util.Scanner;

public class ProblemTwo {

public static void main(String[] args){

Scanner input = new Scanner(System.in);

System.out.print("How big is the dataset?: ");

int dataSet = input.nextInt();

String caption[] = new String[dataSet];

int data[]=new int[dataSet];

int max=0;

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

System.out.print("Enter Caption: ");

caption[i]=input.next();

System.out.print("Enter Data for " + caption[i] + ": ");

data[i]=input.nextInt();

if(data[i]>max)

max=data[i];

}

Double multiplier = 40.0/max;

int asterisks;

for(int j=0;j<caption.length;j++){

asterisks = (int)(data[j]\*multiplier);

System.out.print(caption[j] + ": ");

for(int k=0;k<asterisks;k++){

System.out.print("\*");

}

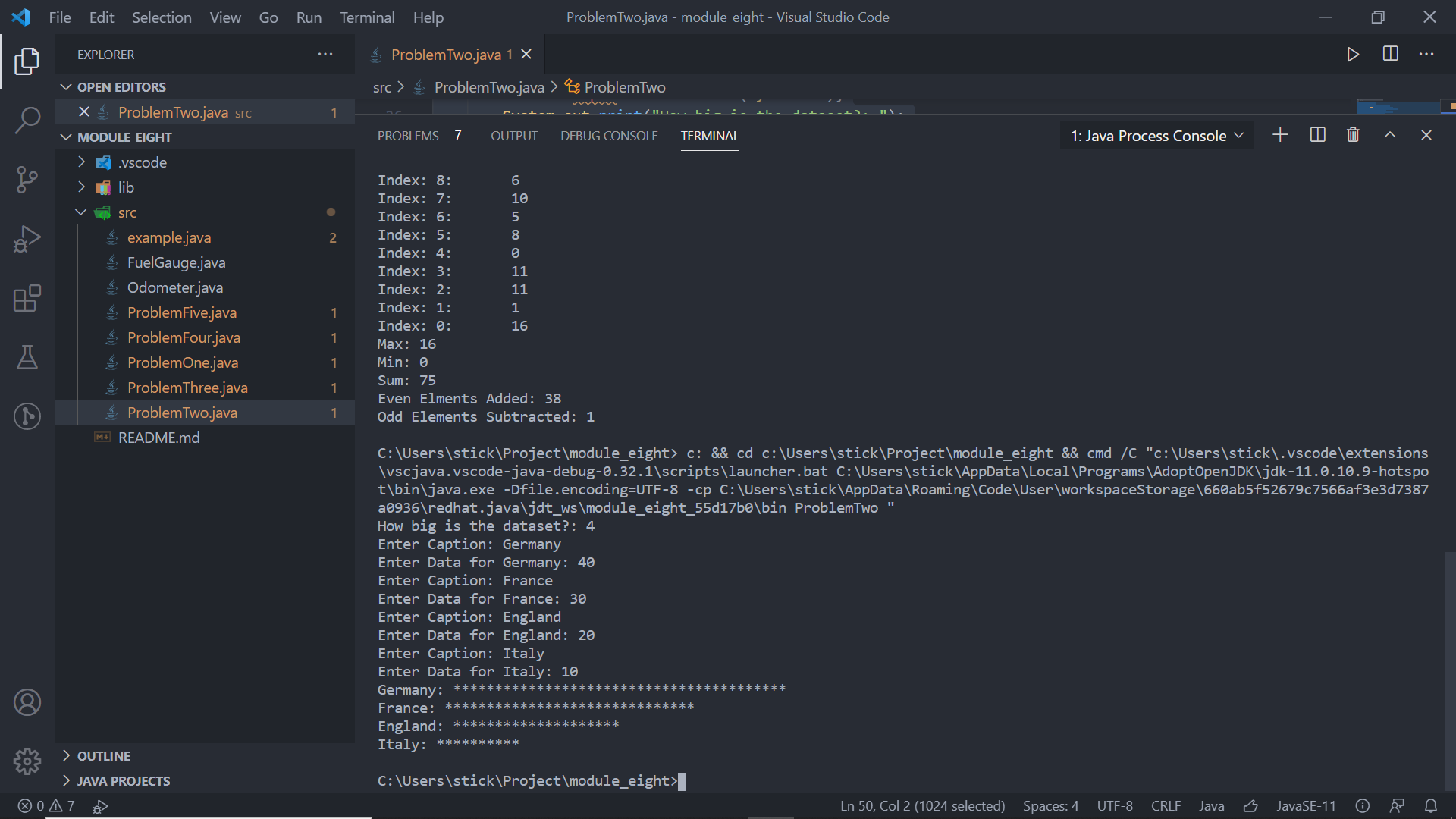
System.out.print("\n");

}

}

}

Screenshot:



Problem 3:

Design and implement an application that reads a sequence of up to 25 pairs of names and postal (ZIP) codes for individuals. Store the data in an object designed to store a first name (string), last name (string), and postal code (integer). Assume each line of input will contain two strings followed by an integer value, each separated by a tab character. Then, after the input has been read in, print the list in an appropriate format to the screen.

Solution:

import java.util.Scanner;

public class ProblemThree {

public static void main(String[] args){

Scanner value = new Scanner (System.in);

System.out.println("How many inputs?:");

int total = value.nextInt();

String arrayFirst[] = new String[total];

String arraySecond[] = new String[total];

int arrayZip[] = new int[total];

int count=0,countTwo=0,countThree=0;

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

Scanner input = new Scanner(value.next());

input.useDelimiter("\t");

while(input.hasNext()){

if(countTwo==0){

arrayFirst[count-countThree]=input.next();

countTwo++;

countThree++;

}

else if(countTwo==1){

arraySecond[count-countThree]=input.next();

countTwo++;

countThree++;

}

else{

arrayZip[count-countThree]=input.nextInt();

countTwo=0;

}

x=x-1;

}

input.close();

count++;

if(count%3==0){

x=(count/3)-1;

}

}

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

System.out.println("Person " + (i+1));

System.out.println("First Name: " + arrayFirst[i]);

System.out.println("Last Name: " + arraySecond[i]);

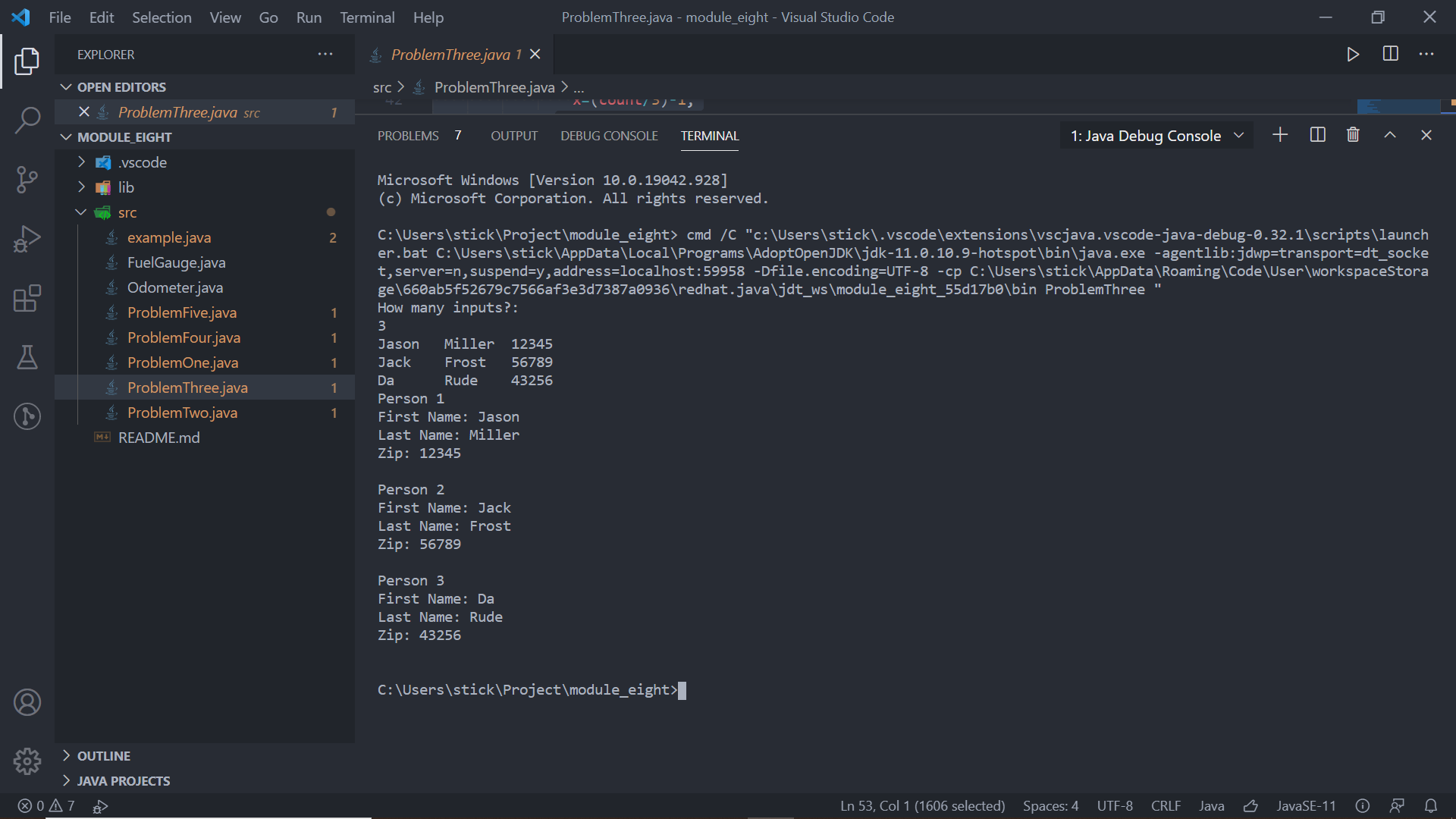
System.out.println("Zip: " + arrayZip[i] + "\n");

}

}

}

Screenshot:



Problem Four:

A theater seating chart is implemented as a two-dimensional array of ticket prices like presented below. Write a program that prompts the users to pick either a seat or a price. Mark sold seats by changing the price to zero. When a user specifies a seat, make sure it is available. When a user specifies a price, find any seat with that price. Verify that the price is among those offered by the theater.

10 10 10 10 10 10 10 10 10 10  
10 10 10 10 10 10 10 10 10 10  
10 10 10 10 10 10 10 10 10 10  
10 10 20 20 20 20 20 20 10 10  
10 10 20 20 20 20 20 20 10 10  
10 10 20 20 20 20 20 20 10 10  
20 20 30 30 40 40 30 30 20 20  
20 30 30 40 50 50 40 30 30 20  
30 40 50 50 50 50 50 50 40 30

Solution:

import java.util.Scanner;

public class ProblemFour{

public static void main(String[] args){

Scanner input = new Scanner(System.in);

int seats[][] = new int[][]{

new int[]{10,10,10,10,10,10,10,10,10,10},

new int[]{10,10,10,10,10,10,10,10,10,10},

new int[]{10,10,10,10,10,10,10,10,10,10},

new int[]{10, 10, 20, 20, 20, 20, 20, 20, 10, 10},

new int[]{10, 10, 20, 20, 20, 20, 20, 20, 10, 10},

new int[]{10, 10, 20, 20, 20, 20, 20, 20, 10, 10},

new int[]{20, 20, 30, 30, 40, 40, 30, 30, 20, 20},

new int[]{20, 30, 30 ,40, 50, 50, 40, 30, 30, 20},

new int[]{30, 40, 50, 50, 50, 50, 50 ,50, 40, 30},

};

char[] seatChar = {'A','B','C','D','E','F','G','H','I'};

boolean purchase = true, seatSel=true;

while(purchase){

System.out.println("Available Seats: ");

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

for(int k=0;k<seats[i].length;k++){

System.out.print(seatChar[i] + ((k+1) + ": ") + seats[i][k]+ " ");

}

System.out.println();

}

System.out.println("Would you like to select by seat or price? Enter 'exit' to exit");

seatSel=true;

while(seatSel){

String select = input.next();

select.toLowerCase();

if(select.equals("seat")){

System.out.println("Enter Seat ID");

String seatID = input.next();

seatID = seatID.toUpperCase();

for(int l=0;l<9;l++){

if(seatChar[l] == seatID.charAt(0)){

if(seatID.length()==2)

if(Character.isDigit(seatID.charAt(1)))

seats[l][(Character.getNumericValue(seatID.charAt(1)))-1] = 0;

if(seatID.length()==3)

if(Character.isDigit(seatID.charAt(1))&&seatID.charAt(2)=='0')

seats[l][(Integer.parseInt(String.valueOf(seatID.charAt(1)+""+seatID.charAt(2))))-1] = 0;

seatSel=false;

}

}

}

else if(select.equals("price")){

System.out.println("Enter a seat price");

int seatID = input.nextInt();

Boolean seatFound=false;

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

for(int k=0;k<seats[i].length;k++){

if(seats[i][k]==seatID){

seats[i][k]=0;

System.out.print("You have been assigned: " + seatChar[i] + ((k+1)));

seatFound=true;

break;

}

}

if(seatFound)

break;

}

if(!seatFound)

System.out.println("No seat available at that price");

seatSel=false;

}

else{

System.out.println("Invalid Entry. Enter 'price' to select by price, Enter 'seat' to seect by seat, or enter 'exit' to exit.");

}

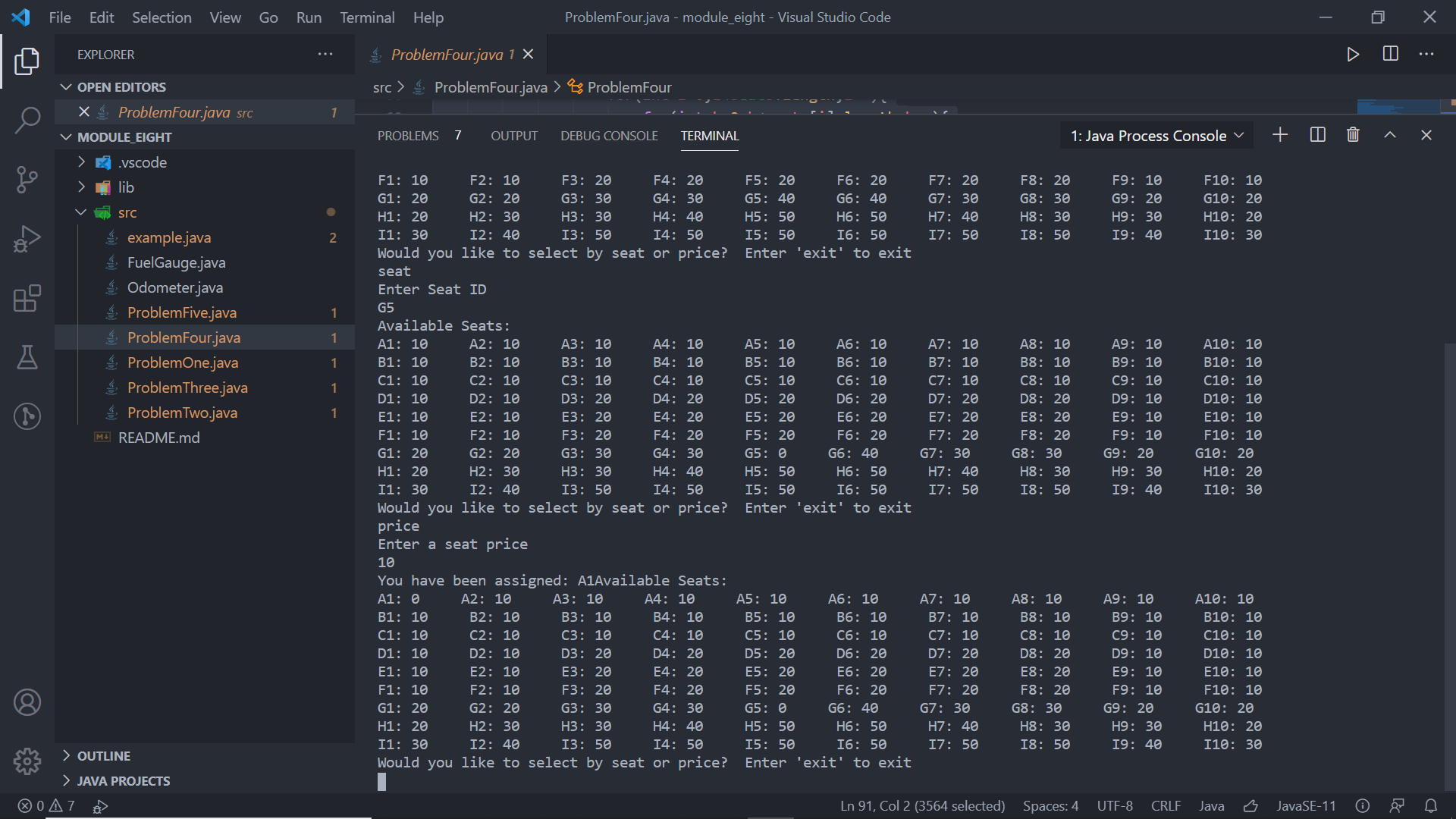
}

}

}

}

Screenshot:



Problem 5:

Write a program that creates a two-dimensional array initialized with test data. Use any primitive data type that you wish. The program should have the following methods:

* getTotal. This method should accept a two-dimensional array as its argument and return the total of all the values in the array.
* getAverage. This method should accept a two-dimensional array as its argument and return the average of all the values in the array.
* getRowTotal. This method should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The method should return the total of the values in the specified row.
* getColumnTotal. This method should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a column in the array. The method should return the total of the values in the specified column.
* getHighestInRow. This method should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The method should return the highest value in the specified row of the array.
* getLowestInRow. This method should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The method should return the lowest value in the specified row of the array.

Demonstrate each of the methods in this program.

Solution:

import java.util.Random;

import java.util.Scanner;

public class ProblemFive {

public static void main(String[] args){

Scanner input = new Scanner(System.in);

Random rand = new Random();

System.out.println("Let's Generate an Array");

System.out.print("How many rows? ");

int rows = input.nextInt();

System.out.print("How many columns? ");

int columns=input.nextInt();

int[][] array=new int[rows][columns];

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

for(int k=0;k<array[i].length;k++){

array[i][k]=rand.nextInt(10);

}

}

System.out.println("Here's your array...");

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

for(int k=0;k<array[i].length;k++){

System.out.print(array[i][k]+ " ");

}

System.out.println();

}

System.out.println("The total of all the values: "+ ProblemFive.getTotal(array));

System.out.println("The average of all the values: "+ ProblemFive.getAverage(array));

System.out.println("Enter a row number...");

int rownum=input.nextInt();

System.out.println("The total of all the values in that row is: "+ ProblemFive.getRowTotal(array, rownum));

System.out.println("Enter a column number...");

rownum=input.nextInt();

System.out.println("The total of all the values in that column is: "+ ProblemFive.getColumnTotal(array, rownum));

System.out.println("Enter a row number...");

rownum=input.nextInt();

System.out.println("The lowest value in that row: "+ ProblemFive.getLowestInRow(array, rownum));

System.out.println("Enter a row number...");

rownum=input.nextInt();

System.out.println("The highest value in that row: "+ ProblemFive.getHighestInRow(array, rownum));

}

public static int getTotal(int[][] a){

int total=0;

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

for(int k=0;k<a[i].length;k++){

total=total+a[i][k];

}

}

return total;

}

public static int getAverage(int[][] a){

int average=0;

int count=0;

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

for(int k=0;k<a[i].length;k++){

average=average+a[i][k];

count++;

}

}

average=average/count;

return average;

}

public static int getRowTotal(int[][] a, int b){

int total=0;

for(int i=0;i<a[b].length;i++){

total = total + a[b][i];

}

return total;

}

public static int getColumnTotal(int a[][], int b){

int total=0;

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

total=total+a[i][b];

}

return total;

}

public static int getHighestInRow(int a[][], int b){

int total=a[b][0];

for(int i=0;i<a[b].length;i++){

if(a[b][i]>total)

total = a[b][i];

}

return total;

}

public static int getLowestInRow(int a[][], int b){

int total=a[b][0];

for(int i=0;i<a[b].length;i++){

if(a[b][i]<total)

total = a[b][i];

}

return total;

}

}

Screenshot:

