1a) int [] numArray = new int [15];

b) System.out.println(alpha[9]);

c) alpha[4] = 35;

d) alpha[8] = alpha[5] + alpha[12];

Question 2.

String [] daysOfTheWeek = {"Sunday", "Monday", "Tuesday", "Wednesday",  
 "Thursday", "Friday", "Saturday"};  
  
for(int i=0;i<daysOfTheWeek.length;i++){  
 System.*out*.println(daysOfTheWeek[i]);  
}

Question 3.

import java.util.Scanner;  
public class Main {  
 public static void main(String[] args) {  
 // Press Alt+Enter with your caret at the highlighted text to see how  
 // IntelliJ IDEA suggests fixing it.  
 int [] nums = new int[10];  
 Scanner console = new Scanner(System.*in*);  
 int sum=0;  
 int temp;  
 for (int i=0;i<nums.length;i++){  
 System.*out*.println("Please enter a number");  
 temp= console.nextInt();  
 nums[i]=temp;  
 sum+=temp;  
 }  
 System.*out*.println(sum);  
  
  
  
  
  
  
 }

Please enter a number

5

Please enter a number

5

Please enter a number

5

Please enter a number

5

Please enter a number

5

Please enter a number

5

Please enter a number

5

Please enter a number

5

Please enter a number

5

Please enter a number

5

50

Question 4.

public class Main {  
 public static void main(String[] args) {  
 // Press Alt+Enter with your caret at the highlighted text to see how  
 // IntelliJ IDEA suggests fixing it.  
 int [] list1=new int[5];  
 int [] list2=new int[5];  
 int num;  
 Scanner console = new Scanner(System.*in*);  
 for(int i=0;i<5;i++){  
 System.*out*.println("Please enter a number");  
 num=console.nextInt();  
 list1[i]=num;  
 }  
 for(int i=0;i<5;i++){  
 list2[i] = list1[i];  
 System.*out*.print(list2[i]+ " ");  
 }

Please enter a number

4

Please enter a number

3

Please enter a number

5

Please enter a number

1

Please enter a number

4

4 3 5 1 4

Question 5

public class Main {  
 public static void main(String[] args) {  
 // Press Alt+Enter with your caret at the highlighted text to see how  
 // IntelliJ IDEA suggests fixing it.  
 int [] temp={1,2,3,4,5};  
 int count=0;  
 int []reverse=new int[temp.length];  
  
 for(int i=temp.length-1;i>=0;i--){  
 reverse[count]=temp[i];  
 count++;  
 }  
 for(int i=0;i<5;i++){  
 System.*out*.print(reverse[i]+ " ");  
 }  
  
  
  
  
 }  
}

A black screen with white text

Description automatically generated

Question 6

import java.util.Scanner;  
public class Main {  
 public static void main(String[] args) {  
 // Press Alt+Enter with your caret at the highlighted text to see how  
 // IntelliJ IDEA suggests fixing it.  
 Scanner console = new Scanner(System.*in*);  
 System.*out*.println("Enter a set of numbers");  
  
 String listOfNums=console.nextLine();  
 String regex = "\s";  
 String[] group = listOfNums.split(regex);  
 int max=Integer.*MIN\_VALUE*;  
 int min=Integer.*MAX\_VALUE*;  
 int [] newGroup=new int[group.length];  
 for(int i=0;i<group.length;i++){  
 newGroup[i]= Integer.*valueOf*(group[i]);  
 if(newGroup[i]>max){  
 max=newGroup[i];  
 }  
 if(newGroup[i]<min){  
 min=newGroup[i];  
 }  
 }  
 System.*out*.println("The largest element of your array" +  
 "is " + max +" and the smallest element of your array is " + min);  
 }  
}

A black screen with white text

Description automatically generated

Question 7.

import java.util.Scanner;  
import java.util.ArrayList;  
public class Main {  
  
 public static void doAction(String action, ArrayList<Integer> nums, Scanner console, String listOfNums){  
 if(action.equals("display")){  
 for(int i=0; i<nums.size();i++){  
 System.*out*.print(nums.get(i)+ " ");  
 }  
 System.*out*.println(" ");  
 System.*out*.println("Please enter display, search, or sort ");  
 action = console.next();  
 *doAction*(action, nums, console, listOfNums);  
  
 }else if(action.equals("search")){  
 System.*out*.println("Please enter an index ");  
 int index= console.nextInt();  
 if(index>=nums.size() || index<0){  
 System.*out*.println("Please try again. The index is out of bounds");  
 }else {  
 System.*out*.println("The element at index " + index + " is " + nums.get(index));  
 System.*out*.println("Please enter display, search, or sort ");  
 action = console.next();  
 *doAction*(action, nums, console, listOfNums);  
 }  
  
 }else if(action.equals("sort")){  
 int [] array =new int[nums.size()];  
 //bubble sort  
 for(int i=0;i<nums.size();i++){  
 for(int j=i;j<nums.size();j++){  
 if(nums.get(i)>nums.get(j)){  
 int temp=nums.get(j);  
 nums.set(j, nums.get(i));  
 //nums.get(j)=nums.get(i);  
 nums.set(i, temp);  
 // nums.get(i)=temp;  
 }  
 }  
  
 }  
 System.*out*.println("Please enter display, search, or sort ");  
 action = console.next();  
 *doAction*(action, nums, console, listOfNums);  
  
 }else if(action.equals("reset")){  
 nums.clear();  
 System.*out*.println("Please enter a set of numbers");  
 listOfNums=console.nextLine();  
 String regex = "\s";  
 String[] group = listOfNums.split(regex);  
 for (int i=0;i<group.length;i++){  
 nums.add(Integer.*valueOf*(group[i]));  
 }  
 System.*out*.println("Please enter reset, display, search, or sort ");  
 action = console.next();  
 *doAction*(action, nums, console, "temp");  
  
 }else{  
 //action is not valid  
 System.*out*.println("The action you tried is not valid, please try again");  
 System.*out*.println("Please enter display, search, or sort ");  
 action = console.next();  
 *doAction*(action, nums, console, "temp");  
  
 }  
  
 }  
  
 public static void main(String[] args) {  
 // Press Alt+Enter with your caret at the highlighted text to see how  
 // IntelliJ IDEA suggests fixing it.  
 Scanner console = new Scanner(System.*in*);  
 boolean arrayExists=false;  
 ArrayList<Integer>temp =new ArrayList<>();  
 if(!arrayExists){  
 System.*out*.println("Please enter a set of numbers");  
 String listOfNums=console.nextLine();  
 String regex = "\s";  
 String[] group = listOfNums.split(regex);  
 for (int i=0;i<group.length;i++){  
 temp.add(Integer.*valueOf*(group[i]));  
 }  
 System.*out*.println("Please enter reset, display, search, or sort ");  
 String action = console.next();  
 *doAction*(action, temp, console, listOfNums);  
  
 }  
 else System.*out*.println("Please enter reset, display, search, or sort");  
 String action = console.next();  
 *doAction*(action, temp, console, "temp");  
  
 //System.out.println("Enter a set of numbers");  
  
  
 }  
}

Please enter a set of numbers

-22 43 55 12 2 215 -58

Please enter reset, display, search, or sort

sort

Please enter display, search, or sort

display

-58 -22 2 12 43 55 215

Please enter display, search, or sort

search

Please enter an index

0

The element at index 0 is -58

Please enter display, search, or sort

10

The action you tried is not valid, please try again

Please enter display, search, or sort

-1

The action you tried is not valid, please try again

Please enter display, search, or sort

Question 8

import java.util.Scanner;  
import java.util.ArrayList;  
public class AscendDescend {  
  
  
 public static void main(String[] args) {  
 // Press Alt+Enter with your caret at the highlighted text to see how  
 // IntelliJ IDEA suggests fixing it.  
 Scanner console = new Scanner(System.*in*);  
 boolean arrayExists=false;  
 int[]temp1, temp2, answer;  
  
 System.*out*.println("Please enter a set of numbers");  
 String listOfNums = console.nextLine();  
 String regex = "\s";  
 String[] list1 = listOfNums.split(regex);  
 temp1 = new int[list1.length];  
 for (int i = 0; i < list1.length; i++) {  
 //temp1.add(Integer.valueOf(list1[i]));  
 temp1[i] = Integer.*valueOf*(list1[i]);  
 }  
  
 System.*out*.println("Please enter another set of numbers");  
 listOfNums = console.nextLine();  
 //String regex = "\s";  
 String[] list2 = listOfNums.split(regex);  
 temp2 = new int[list2.length];  
 for (int i = 0; i < list2.length; i++) {  
 //temp1.add(Integer.valueOf(list1[i]));  
 temp2[i] = Integer.*valueOf*(list2[i]);  
 }  
  
  
 answer = new int[temp1.length + temp2.length];  
  
 int ptr1 = 0;  
 int ptr2 = temp2.length-1;  
 int count = 0;  
  
 while (ptr1 < temp1.length && ptr2 >= 0) {  
 // System.out.println(temp1[ptr1]+ " " + temp2[ptr2]);  
  
 if (temp1[ptr1] > temp2[ptr2]) {  
 //ptr2 is smaller, take this value  
 answer[count] = temp2[ptr2];  
 ptr2--;  
 count++;  
 } else {  
 //ptr1 is smaller  
 answer[count] = temp1[ptr1];  
 ptr1++;  
 count++;  
 }  
  
  
 }  
 while (ptr1 < temp1.length) {  
 answer[count] = temp1[ptr1];  
 count++;  
 ptr1++;  
 }  
 while (ptr2 >=0) {  
 answer[count] = temp2[ptr2];  
 count++;  
 ptr2--;  
  
 }  
  
  
 for (int i=0;i<answer.length;i++){  
 System.*out*.print(answer[i] + " ");  
 }  
  
  
  
 }  
}

A screenshot of a computer code

Description automatically generated

Question 9

import java.util.Arrays;  
import java.util.Scanner;  
import java.util.ArrayList;  
public class TwoDimension {  
 public static String [] getAccurateArray(Scanner console, int m, int[][]matrix){  
 //String [] list1;  
  
 console.nextLine();  
 System.*out*.println("Please enter an array of values");  
 String listOfNums = console.nextLine();  
 String regex = "\s";  
 String[] list1 = listOfNums.split(regex);  
  
  
 while(list1.length!=matrix[0].length){  
 System.*out*.println("You need "+ matrix[0].length  
 + " elements in your array");  
  
 }  
 console.nextLine();  
 listOfNums = console.nextLine();  
 regex = "\s";  
 list1 = listOfNums.split(regex);  
  
  
 return list1;  
 }  
  
 public static void displayMatrix(int m, int n, int[][]matrix){  
 System.*out*.println("Display of Matrix");  
  
 for(int i=0;i<m;i++){  
 for(int j=0;j<n;j++){  
 System.*out*.print(matrix[i][j] + " ");  
 }  
 System.*out*.println(" ");  
 }  
  
 System.*out*.println("\_\_\_\_\_\_\_\_\_\_");  
  
 }  
 public static void addMatrix(int m, int n, int[][]matrix){  
 System.*out*.println("Sum Of Matrix");  
 int sum=0;  
 for(int i=0;i<m;i++){  
 for(int j=0;j<n;j++){  
 sum+=matrix[i][j];  
 }  
 }  
 //System.out.println(" ");  
 System.*out*.println(sum);  
 System.*out*.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_");  
  
 }  
 public static void rowWiseSum(int m, int n, int[][]matrix){  
 System.*out*.println("Row Wise Sum");  
 int rowSum=0;  
 for(int i=0;i<m;i++){  
 for(int j=0;j<n;j++){  
 rowSum+=matrix[i][j];  
 }  
 System.*out*.print(rowSum + " ");  
 rowSum=0;  
 }  
 System.*out*.println("\_\_\_\_\_\_\_\_\_\_\_");  
  
 //System.out.println(" ");  
 //return sum;  
 }  
 public static void columnWiseSum(int m, int n, int[][]matrix){  
 System.*out*.println("Column Wise Sum");  
  
 int [] colSum =new int[n];  
 Arrays.*fill*(colSum, 0);  
 for(int i=0;i<m;i++){  
 for(int j=0;j<n;j++) {  
 colSum[j] += matrix[i][j];  
 }  
 }  
 for(int i=0;i<colSum.length;i++){  
 System.*out*.print(colSum[i]+ " ");  
 }  
 //System.out.println(" ");  
 //return sum;  
 System.*out*.println("\_\_\_\_\_\_\_\_\_\_\_");  
  
 }  
 public static void transpose(int m, int n, int[][]matrix){  
 System.*out*.println("Transpose");  
 int [][] transposeMatrix= new int[n][m];  
 for(int i=0;i<m;i++){  
 for(int j=0;j<n;j++){  
 transposeMatrix[j][i]= matrix[i][j];  
 }  
 }  
 *displayMatrix*(n, m, transposeMatrix);  
  
  
  
 }  
  
  
 public static void main(String[] args) {  
 // Press Alt+Enter with your caret at the highlighted text to see how  
 // IntelliJ IDEA suggests fixing it.  
 Scanner console = new Scanner(System.*in*);  
 int [][] matrix;  
 int m,n;  
 int [] temp;  
  
 System.*out*.println("Please enter a value for m");  
 m=console.nextInt();  
  
  
 System.*out*.println("Please enter a value for n");  
 n=console.nextInt();  
 matrix = new int[m][n];  
 console.nextLine();  
  
  
 for(int i=0;i<m;i++) {  
 String [] accurateArray;  
 System.*out*.println("Please enter an array of values");  
 String listOfNums = console.nextLine();  
 String regex = "\s";  
 String[] list1 = listOfNums.split(regex);  
  
 if(list1.length==matrix[0].length){  
 accurateArray=list1;  
 }  
 else {  
 accurateArray = *getAccurateArray*(console, m, matrix);  
 }  
 //temp = new int[accurateArray.length];  
 for (int j = 0; j < n; j++) {  
 //temp1.add(Integer.valueOf(list1[i]));  
 matrix[i][j] = Integer.*valueOf*(accurateArray[j]);  
 // temp1[i] = Integer.valueOf(list1[i]);  
 }  
 }  
  
 System.*out*.println(" ");  
 *displayMatrix*(m, n, matrix);  
 *addMatrix*(m, n, matrix);  
 *rowWiseSum*(m, n, matrix);  
 *columnWiseSum*(m, n, matrix);  
 *transpose*(m, n, matrix);  
  
 }  
}

Please enter a value for m

3

Please enter a value for n

2

Please enter an array of values

1 2

Please enter an array of values

3 -5

Please enter an array of values

9 -7

Display of Matrix

1 2

3 -5

9 -7

\_\_\_\_\_\_\_\_\_\_

Sum Of Matrix

3

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Row Wise Sum

3 -2 2 \_\_\_\_\_\_\_\_\_\_\_

Column Wise Sum

13 -10 \_\_\_\_\_\_\_\_\_\_\_

Transpose

Display of Matrix

1 3 9

2 -5 -7

\_\_\_\_\_\_\_\_\_\_

Process finished with exit code 0

Question 10

public static void leftRightDiagonalSum(int m, int n, int[][]matrix){  
 System.*out*.println("Left and Right diagonal");  
 int leftDiag=0;  
 int rightDiag=0;  
 if(m!=n){  
 System.*out*.println("Must be a square matrix!");  
  
 }else{  
 for(int i=0;i<m;i++){  
 for(int j=0;j<n;j++){  
 //left diagonal  
 if(i==j)leftDiag+=matrix[i][j];  
 //right diagonal  
 if(i+j==m-1)rightDiag+=matrix[i][j];  
  
 }  
 }  
 System.*out*.println("Right Diagonal:" +  
 rightDiag + " Left Diagonal: " + leftDiag );  
 }  
  
}

Please enter a value for m

2

Please enter a value for n

2

Please enter an array of values

1 2

Please enter an array of values

-9 5

Display of Matrix

1 2

-9 5

\_\_\_\_\_\_\_\_\_\_

Sum Of Matrix

-1

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Row Wise Sum

3 -4 \_\_\_\_\_\_\_\_\_\_\_

Column Wise Sum

-8 7 \_\_\_\_\_\_\_\_\_\_\_

Transpose

Display of Matrix

1 -9

2 5

\_\_\_\_\_\_\_\_\_\_

Left and Right diagonal

Right Diagonal:-7 Left Diagonal: 6

Process finished with exit code 0

Question 11

public static void matrixMultiplication(int m, int n, int[][] matrix){  
 System.*out*.println("Matrix Multiplication");  
 int[] multiplicationMatrix=new int[n];  
 Arrays.*fill*(multiplicationMatrix,1);  
 if(m!=4 && n!=6){  
 System.*out*.println("Matrix must be 4 X 6 ");  
  
 }else{  
 for(int i=0;i<m;i++){  
 for(int j=0;j<n;j++){  
 multiplicationMatrix[j]\*=matrix[i][j];  
 }  
 System.*out*.println(" ");  
 }  
 }  
 for(int i=0;i<multiplicationMatrix.length;i++){  
 System.*out*.println(multiplicationMatrix[i]+ " ");  
 }  
 System.*out*.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");  
}

Please enter a value for m

4

Please enter a value for n

6

Please enter an array of values

1 2 3 4 5 6

Please enter an array of values

12 5 6 -5 6 7

Please enter an array of values

6 -12 3 5 16 17

Please enter an array of values

7 -5 -16 7 8 5

Display of Matrix

1 2 3 4 5 6

12 5 6 -5 6 7

6 -12 3 5 16 17

7 -5 -16 7 8 5

\_\_\_\_\_\_\_\_\_\_

Sum Of Matrix

93

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Row Wise Sum

21 31 35 6 \_\_\_\_\_\_\_\_\_\_\_

Column Wise Sum

26 -10 -4 11 35 35 \_\_\_\_\_\_\_\_\_\_\_

Transpose

Display of Matrix

1 12 6 7

2 5 -12 -5

3 6 3 -16

4 -5 5 7

5 6 16 8

6 7 17 5

\_\_\_\_\_\_\_\_\_\_

Left and Right diagonal

Must be a square matrix!

\_\_\_\_\_\_\_\_\_\_\_

Upper Half of the Matrix

Must be a square matrix!

Matrix Multiplication

504

600

-864

-700

3840

3570

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Process finished with exit code 0

Question 12

public static void upperHalf(int m, int n, int[][] matrix){  
 System.*out*.println("Upper Half of the Matrix");  
 if(m!=n){  
 System.*out*.println("Must be a square matrix!");  
  
 }else{  
 for(int i=0;i<m;i++){  
 for(int j=0;j<n;j++){  
 if(i<=j){  
 System.*out*.print(matrix[i][j]+ " ");  
 }else {  
 System.*out*.print(" ");  
 }  
 }  
 System.*out*.println(" ");  
 }  
  
 }  
}

Please enter a value for m

4

Please enter a value for n

4

Please enter an array of values

1 2 3 4

Please enter an array of values

5 6 7 4

Please enter an array of values

8 7 6 5

Please enter an array of values

6 5 4 5

Display of Matrix

1 2 3 4

5 6 7 4

8 7 6 5

6 5 4 5

\_\_\_\_\_\_\_\_\_\_

Sum Of Matrix

78

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Row Wise Sum

10 22 26 20 \_\_\_\_\_\_\_\_\_\_\_

Column Wise Sum

20 20 20 18 \_\_\_\_\_\_\_\_\_\_\_

Transpose

Display of Matrix

1 5 8 6

2 6 7 5

3 7 6 4

4 4 5 5

\_\_\_\_\_\_\_\_\_\_

Left and Right diagonal

Right Diagonal:24 Left Diagonal: 18

\_\_\_\_\_\_\_\_\_\_\_

Upper Half of the Matrix

1 2 3 4

6 7 4

6 5

5

Question 13

public static void displayMiddle(int m, int n, int[][]matrix){  
 System.*out*.println("Displaying Middle Row and Column");  
 String middleRow="";  
 String middleCol="";  
 int middle;  
  
 int[] multiplicationMatrix=new int[n];  
 Arrays.*fill*(multiplicationMatrix,1);  
 if(m!=n){  
 System.*out*.println("Must be a square matrix");  
 }else if(m%2==0) {  
 System.*out*.println("Matrix must be of odd dimensions");  
 }else{  
 middle=m/2;  
 for(int i=0;i<m;i++){  
 for(int j=0;j<n;j++){  
 if(i==middle){  
 //middle row  
 middleRow+=" "+ matrix[i][j];  
 }  
 if(j==middle){  
 //middle column  
 middleCol+=" "+matrix[i][j];  
 }  
  
 }  
 }  
 }  
 System.*out*.println("Middle row: " + middleCol);  
 System.*out*.println("Middle column: " + middleRow);  
 System.*out*.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");  
  
} Please enter a value for m

3

Please enter a value for n

3

Please enter an array of values

4 5 6

Please enter an array of values

6 7 8

Please enter an array of values

5 4 3

Display of Matrix

4 5 6

6 7 8

5 4 3

\_\_\_\_\_\_\_\_\_\_

Sum Of Matrix

48

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Row Wise Sum

15 21 12 \_\_\_\_\_\_\_\_\_\_\_

Column Wise Sum

15 16 17 \_\_\_\_\_\_\_\_\_\_\_

Transpose

Display of Matrix

4 6 5

5 7 4

6 8 3

\_\_\_\_\_\_\_\_\_\_

Left and Right diagonal

Right Diagonal:18 Left Diagonal: 14

\_\_\_\_\_\_\_\_\_\_\_

Upper Half of the Matrix

4 5 6

7 8

3

Matrix Multiplication

Matrix must be 4 X 6

1

1

1

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Displaying Middle Row and Column

Middle row: 5 7 4

Middle column: 6 7 8

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Process finished with exit code 0

Question 14

A screenshot of a computer program

Description automatically generated

A black background with white text

Description automatically generated