

Program 1

Q Wap to convert Fahrenheit to Celsius in Java using formula given below

$$^{\circ}\text{C} = (^{\circ}\text{F} - 32) / (9/5)$$

```
package studentmtgtsystem;
// Q Wap to convert Fahrenheit to Celsius in Java using formula given below

// $^{\circ}\text{C} = (^{\circ}\text{F} - 32) / (9/5)$ 

import java.util.Scanner;

public class Conversion {

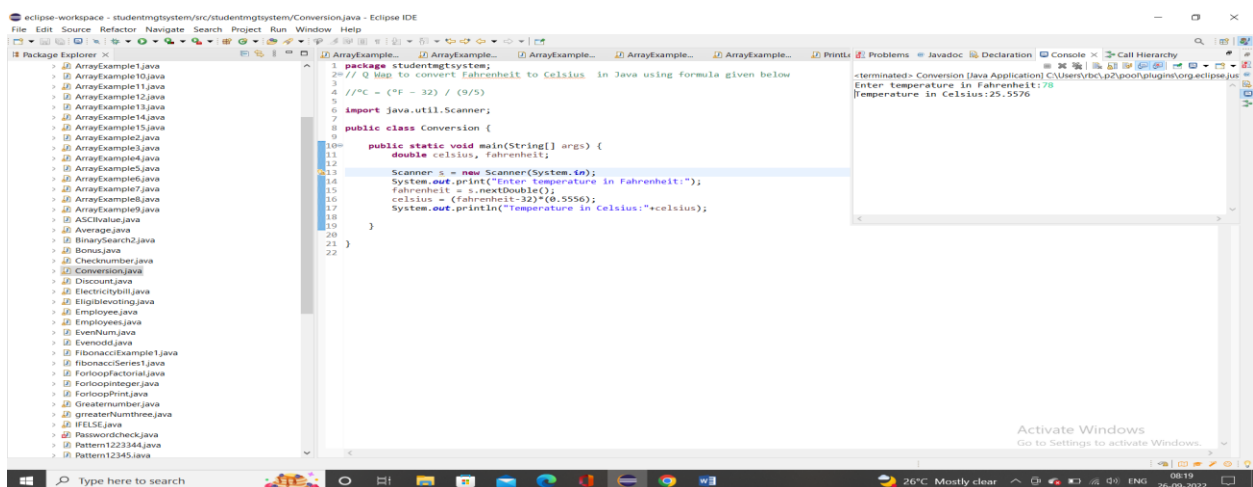
    public static void main(String[] args) {
        double celsius, fahrenheit;

        Scanner s = new Scanner(System.in);
        System.out.print("Enter temperature in Fahrenheit:");
        fahrenheit = s.nextDouble();
        celsius = (fahrenheit-32)*(0.5556);
        System.out.println("Temperature in Celsius:"+celsius);
    }
}
```

Result

Enter temperature in Fahrenheit:78

Temperature in Celsius:25.5576



Program 2

Q 2 wap to check a given number is armstrong or not i.e. $153 = 1*1*1 + 5*5*5 + 3*3*3$

```
package studentmgtsystem;

// wap to check a given number is armstrong or not i.e.  $153 = 1*1*1 + 5*5*5 + 3*3*3$ 

import java.util.Scanner;

public class ArmstrongNumber {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n,copy,remainder,sum=0;
        System.out.println("Enter the number");

        n=sc.nextInt();
        copy=n;
        while(copy!=0)
        {
            remainder=copy%10;
            sum=sum+remainder*remainder*remainder;
            copy=copy/10;
        }

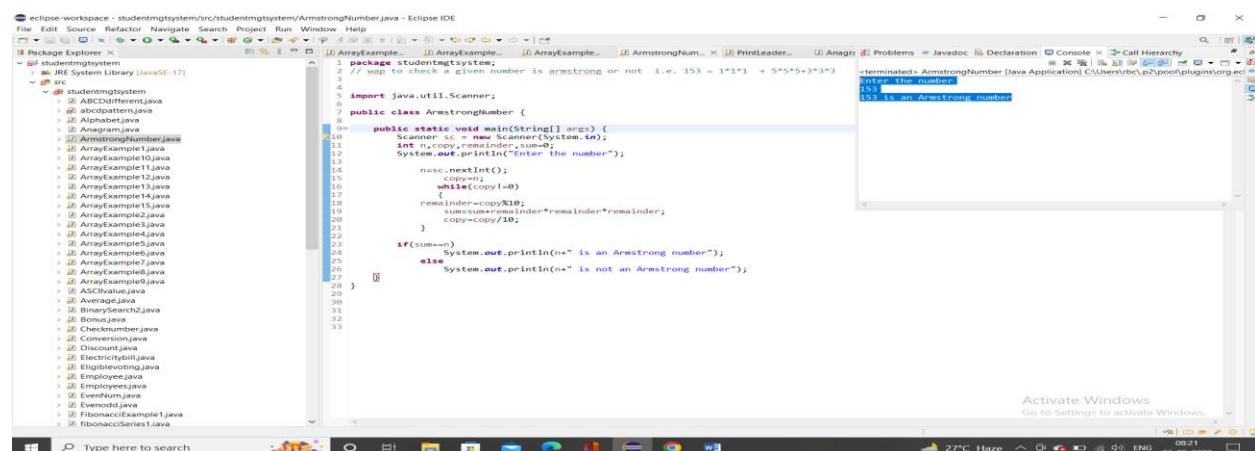
        if(sum==n)
            System.out.println(n+" is an Armstrong number");
        else
            System.out.println(n+" is not an Armstrong number");
    }
}
```

Result

Enter the number

153

153 is an Armstrong number



Program 3

```
package studentmgtsystem;
```

```
// Rajan went to a movie with his friends in a multiplex theatre and during break  
time he bought pizzas, puffs and cool drinks. Consider the following prices :
```

```
//Rs.100/pizza  
//Rs.20/puffs  
//Rs.10/cooldrink  
//Generate a bill for What Rajan has bought.
```

```
//Sample Input 1:
```

```
//Enter the no of pizzas bought:10  
//Enter the no of puffs bought:12  
//Enter the no of cool drinks bought:5
```

```
//Sample Output 1:
```

```
//Bill Details  
//No of pizzas:10  
//No of puffs:12  
//No of cooldrinks:5  
//Total price=1290
```

```
-----  
import java.util.Scanner;
```

```
public class ArrayExample15
```

```
{public static void main(String[] args) {
```

```
    int EachPizza = 100;
```

```
    int EachPuffs = 20 ;
```

```
    int EachCooldrink = 10 ;
```

```
    int Num1 = 0 , Num2 =0, Num3 =0 ,TotalPrize = 0;
```

```
    Scanner s= new Scanner (System.in);
```

```
    System.out.print("Enter the number of pizza bought = " );  
    Num1 =s.nextInt();
```

```
    System.out.print("Enter the number of Puffs bought = " );  
    Num2 =s.nextInt();
```

```
    System.out.print("Enter the number of Cooldrinks bought = " );  
    Num3 =s.nextInt();
```

```
    System.out.println("No of pizzas:"+ Num1);
```

```
    System.out.println("No of Puffs :" + Num2);
```

```
    System.out.println("No of Cooldrinks:"+ Num3);
```

```
{
```

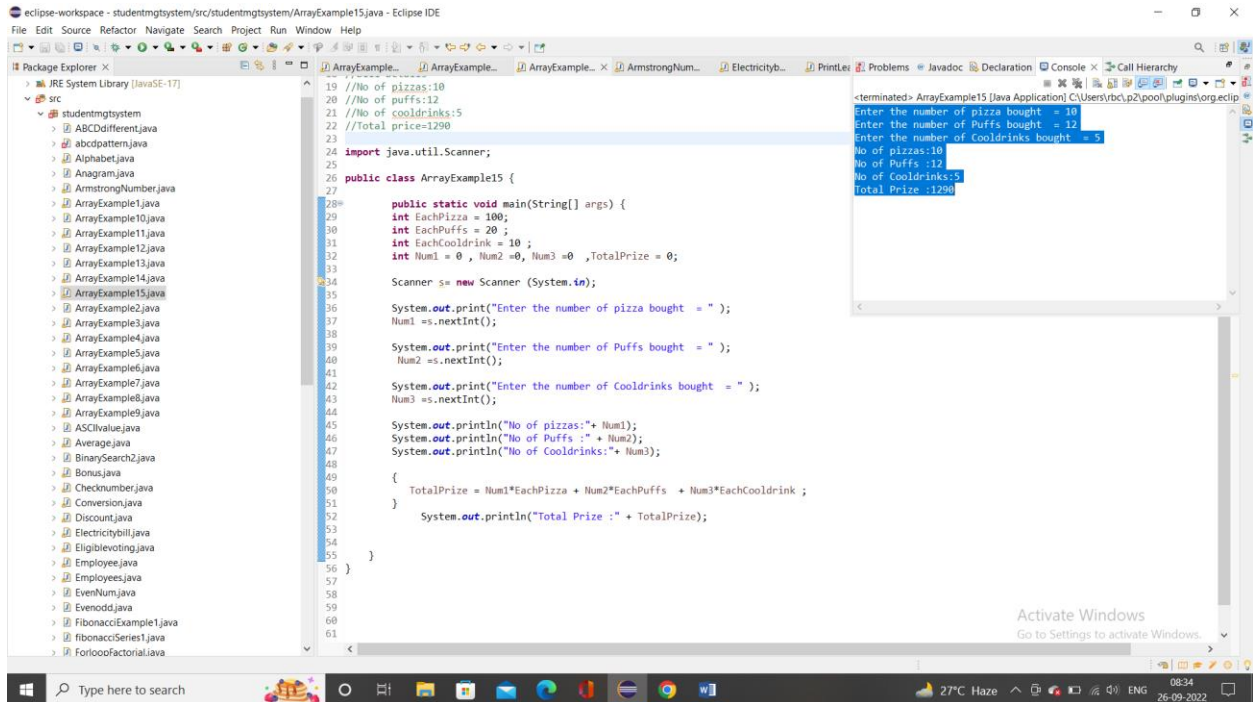
```
    TotalPrize = Num1*EachPizza + Num2*EachPuffs + Num3*EachCooldrink ;
```

```
}
```

```
    System.out.println("Total Prize :" + TotalPrize); }}
```

Result :

Enter the number of pizza bought = 10
Enter the number of Puffs bought = 12
Enter the number of Cooldrinks bought = 5
No of pizzas:10
No of Puffs :12
No of Cooldrinks:5
Total Prize :1290



The screenshot shows the Eclipse IDE interface. The Package Explorer on the left lists the project structure, including the 'src' folder and various Java files. The main editor displays the source code for 'ArrayExample15.java'. The code defines a public class with a main method that uses a Scanner to take user input for the number of pizzas, puffs, and cooldrinks bought. It then calculates the total price based on predefined rates (10 for pizza, 20 for puffs, 10 for cooldrinks) and prints the result. The Console on the right shows the program's execution, displaying the prompts and the user's input, followed by the calculated total price of 1290.

```
//No of pizzas:10
//No of puffs:12
//No of cooldrinks:5
//Total price:1290

import java.util.Scanner;

public class ArrayExample15 {

    public static void main(String[] args) {
        int EachPizza = 100;
        int EachPuffs = 20 ;
        int EachCooldrink = 10 ;
        int Num1 = 0 , Num2 =0, Num3 =0 ,TotalPrize = 0;

        Scanner s= new Scanner (System.in);

        System.out.print("Enter the number of pizza bought = ");
        Num1 =s.nextInt();

        System.out.print("Enter the number of Puffs bought = ");
        Num2 =s.nextInt();

        System.out.print("Enter the number of Cooldrinks bought = ");
        Num3 =s.nextInt();

        System.out.println("No of pizzas:"+ Num1);
        System.out.println("No of Puffs :"+ Num2);
        System.out.println("No of Cooldrinks:"+ Num3);

        {
            TotalPrize = Num1*EachPizza + Num2*EachPuffs + Num3*EachCooldrink ;
        }

        System.out.println("Total Prize :"+ TotalPrize);
    }
}
```

<terminated> ArrayExample15 (Java Application) C:\Users\rbcl\p2\pool\plugins\org.eclipse...
Enter the number of pizza bought = 10
Enter the number of Puffs bought = 12
Enter the number of Cooldrinks bought = 5
No of pizzas:10
No of Puffs :12
No of Cooldrinks:5
Total Prize :1290

Activate Windows
Go to Settings to activate Windows.

Type here to search 27°C Haze 08:34 26-09-2022

Program 4

```
package studentmgtsystem;
```

```
// Given an integer U denoting the amount of KWh units of electricity consumed, the task is to calculate the electricity bill with the help of the below charges:
```

```
import java.util.Scanner;
```

```
// 1 to 100 units - Rs. 10/unit  
// 100 to 200 units - Rs. 15/unit  
// 200 to 300 units - Rs. 20/unit  
// above 300 units - Rs. 25/unit  
//Examples:  
//Input: U = 250  
//Output: 3500  
//Explanation:  
//Charge for the first 100 units -  $10 \times 100 = 1000$   
//Charge for the 100 to 200 units -  $15 \times 100 = 1500$   
//Charge for the 200 to 250 units -  $20 \times 50 = 1000$   
//Total Electricity Bill =  $1000 + 1500 + 1000 = 3500$   
//Input: U = 95  
//Output: 950  
//Explanation:  
//Charge for the first 100 units -  $10 \times 95 = 950$   
//Total Electricity Bill = 950
```

```
public class Electricitybill {
```

```
    public static void main(String[] args) {
```

```
        int U = 0 ;
```

```
        Scanner s = new Scanner (System.in);
```

```
        System.out.print("the total KWh units of electricity consumed = " );
```

```
        U = s.nextInt();
```

```
        if ( U <= 100 )
```

```
        {
```

```
            float Output = U * 10 ;
```

```
            System.out.print("The amount of energy consumed = " + Output) ;
```

```
        }
```

```
        else if (U <= 200)
```

```
        {    float Output = 100*10 + (U-100)*15 ;
```

```
            System.out.print("The amount of energy consumed = " + Output) ;
```

```
        }
```

```
        else if (U <= 300)
```

```
        {
```

```
            float Output = 100*10 + (100 * 15) + (U-200)*20 ;
```

```
            System.out.print("The amount of energy consumed = " + Output) ;
```

```

    }

    else if (U>300)

    {
        float Output = 100*10 + (100 * 15) + (100)*20 + (U-300) * 25 ;
        System.out.print("The amount of energy consumed = " + Output) ;

    }

}

```

Result

the total KWh units of electricity consumed = 350

The amount of energy consumed = 5750.0

The screenshot shows the Eclipse IDE with the following components:

- Package Explorer:** Lists the project structure, including the `src` folder and various Java files like `ArrayExample1.java` through `ArrayExample15.java`, `Electricitybill.java`, and others.
- Editor:** Displays the `Electricitybill.java` file. The code is as follows:


```

21 //Charge for the first 100 units - 10*95 = 950
22 //Total Electricity Bill = 950
23
24 public class Electricitybill {
25
26     public static void main(String[] args) {
27         int U = 0 ;
28         Scanner s= new Scanner (System.in);
29         System.out.print("the total KWh units of electricity consumed = " );
30         U =s.nextInt();
31
32         if ( U<= 100 )
33         {
34             float Output = U * 10 ;
35
36             System.out.print("The amount of energy consumed = " + Output) ;
37
38         }
39
40         else if (U<=200)
41         {
42             float Output = 100*10 + (U-100)*15 ;
43
44             System.out.print("The amount of energy consumed = " + Output) ;
45
46         }
47
48         else if (U<=300)
49         {
50             float Output = 100*10 + (100 * 15) + (U-200)*20 ;
51             System.out.print("The amount of energy consumed = " + Output) ;
52
53         }
54
55         else if (U>300)
56         {
57             float Output = 100*10 + (100 * 15) + (100)*20 + (U-300) * 25 ;
58             System.out.print("The amount of energy consumed = " + Output) ;
59
60         }
61
62     }
63
64 }

```
- Console:** Shows the output of the program:


```

<terminated> Electricitybill [Java Application] C:\Users\ybc\p2\pool\plugins\org.eclipse.jdt.launcher\
the total KWh units of electricity consumed = 350
The amount of energy consumed = 5750.0

```

Program 5

Write a java program that define a sorted array of size N and an integer K, find the position at which K is

present in the array using binary search.

Example 1:

Input:

N = 5

arr[] = {1 2 3 4 5}

K = 4

Output: 3

Explanation: 4 appears at index 3.

```
package studentmgtsystem;
// Write a java program that define a sorted array of size N and an integer K, find
the position at which K is

//present in the array using binary search.

//Example 1:

//Input:
//N = 5
//arr[] = {1 2 3 4 5}
//K = 4
//Output: 3
//Explanation: 4 appears at index 3.
public class BinarySearch2 {

    public static void main(String[] args) {
        int[] array = new int[] { 1, 3, 5, 9, 12, 22, 38, 45 };
        int K = 22;
        int res = binarySearch(array, K);

        if (res >= 0) {
            System.out.println(K + " found at index: " + res);
        } else {
            System.out.println(K + " not found");
        }
    }

    private static int binarySearch(int[] array, int K) {
        int n = array.length;
        int low = 0;
        int high = n - 1;

        while (low <= high) {
            int mid = low + (high - low) / 2;
            // think: why not use (low + high) / 2 ?
        }
    }
}
```

```

        if (array[mid] < K) {
            low = mid + 1;
        } else if (array[mid] > K) {
            high = mid - 1;
        }

        else {
            // found K
            return mid;
        }

    }

    return -1;
}
}

```

Result

22 found at index: 5

The screenshot shows the Eclipse IDE with the following components:

- Package Explorer:** Lists various Java files in the 'studentmgmtsystem' package, including 'BinarySearch2.java' which is currently selected.
- Editor:** Displays the source code of 'BinarySearch2.java'. The code includes a package declaration, a comment describing the program, and a binary search implementation. The array being searched is {1, 3, 9, 12, 22, 38, 45} and the target value is 22.
- Console:** Shows the output of the program: '22 found at index: 5'.
- Bottom Bar:** Displays the system tray with the date '26-09-2022' and time '08:38'.

```

1 package studentmgmtsystem;
2 // Write a java program that define a sorted array of size N and an integer K, find the element
3 // present in the array using binary search.
4 //Example 1:
5 //Input:
6 //N = 5
7 //arr[] = {1 2 3 4 5}
8 //K = 4
9 //Output: 3
10 //Explanation: 4 appears at index 3.
11 public class BinarySearch2 {
12
13     public static void main(String[] args) {
14         int[] array = new int[] { 1, 3, 9, 12, 22, 38, 45 };
15         int K = 22;
16         int res = binarySearch(array, K);
17
18         if (res >= 0) {
19             System.out.println(K + " found at index: " + res);
20         } else {
21             System.out.println(K + " not found");
22         }
23     }
24
25     private static int binarySearch(int[] array, int K) {
26         int n = array.length;
27         int low = 0;
28         int high = n - 1;
29
30         while (low <= high) {
31             int mid = low + (high - low) / 2;
32             // think: why not use (low + high) / 2 ?
33
34             if (array[mid] < K) {
35                 low = mid + 1;
36             } else if (array[mid] > K) {
37                 high = mid - 1;
38             }
39         }
40
41         return low;
42     }
43 }

```


Program 6

write a java program and define an array, print all the elements which are leaders. A Leader is an element that is greater than all of the elements on its right side in the array.

Examples:

Example 1:

Input:

arr = [4, 7, 1, 0]

Output:

7 1 0

Explanation:

Rightmost element is always a leader. 7 and 1 are greater than the elements in their right side.

```
package studentmgtsystem;
//write a java program and define an array, print all the elements which are
leaders. A Leader is an element that is greater than all of the elements on its right
side in the array.
```

```
import java.util.Scanner;
```

```
//Examples:
```

```
//Example 1:
```

```
//Input:
```

```
// arr = [4, 7, 1, 0]
```

```
//Output:
```

```
// 7 1 0
```

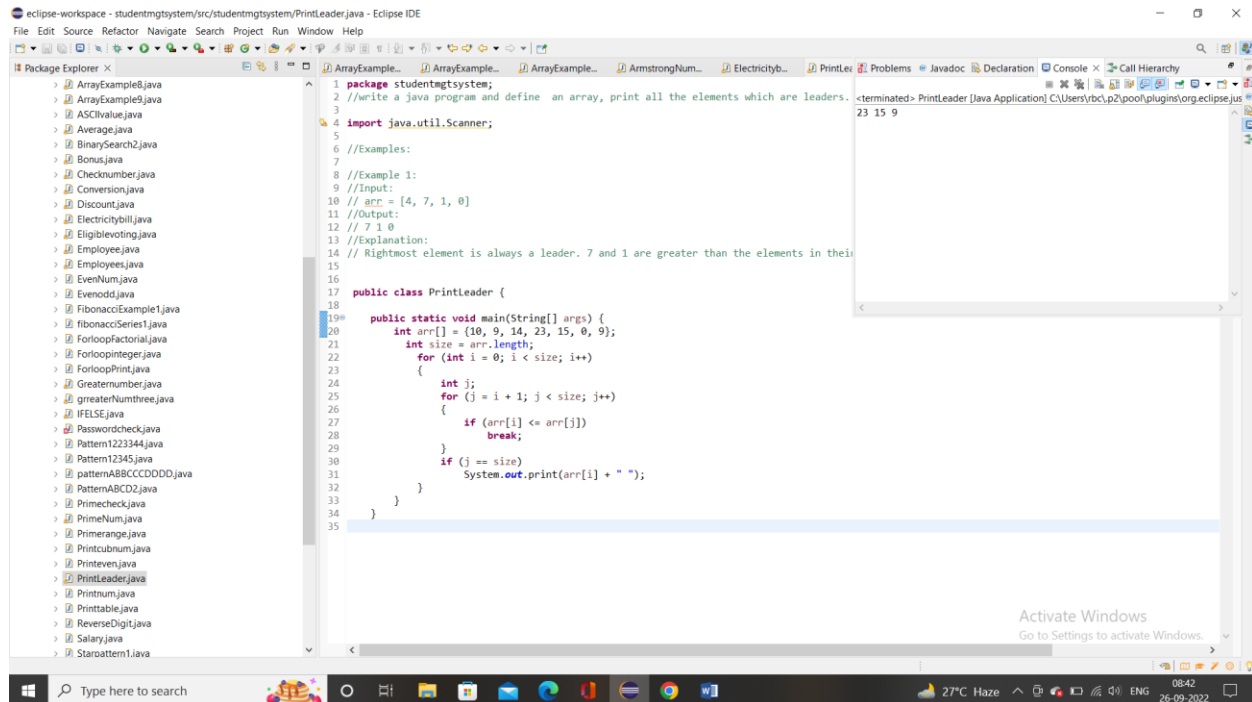
```
//Explanation:
```

```
// Rightmost element is always a leader. 7 and 1 are greater than the elements in
their right side.
```

```
public class PrintLeader {
```

```
    public static void main(String[] args) {
        int arr[] = {10, 9, 14, 23, 15, 0, 9};
        int size = arr.length;
        for (int i = 0; i < size; i++)
        {
            int j;
            for (j = i + 1; j < size; j++)
            {
                if (arr[i] <= arr[j])
                    break;
            }
            if (j == size)
                System.out.print(arr[i] + " ");
        }
    }
}
```

}
Result –
23, 15, 9



```
1 package studentmgtsystem;
2 //write a java program and define an array, print all the elements which are leaders.
3
4 import java.util.Scanner;
5
6 //Examples:
7
8 //Example 1:
9 //Input:
10 // arr = [4, 7, 1, 0]
11 //Output:
12 // 7 1 0
13 //Explanation:
14 // Rightmost element is always a leader. 7 and 1 are greater than the elements in their
15
16 public class PrintLeader {
17
18
19     public static void main(String[] args) {
20         int arr[] = {10, 9, 14, 23, 15, 0, 9};
21         int size = arr.length;
22         for (int i = 0; i < size; i++)
23         {
24             int j;
25             for (j = i + 1; j < size; j++)
26             {
27                 if (arr[i] <= arr[j])
28                     break;
29             }
30             if (j == size)
31                 System.out.print(arr[i] + " ");
32         }
33     }
34 }
35
```

Activate Windows
Go to Settings to activate Windows.

Program 7

Given two strings a and b consisting of lowercase characters. The task is to check whether two given strings are an anagram of each other or not. An anagram of a string is another string that contains the same characters, only the order of characters can be different. For example, abc and bca are an anagram of each other.

Example 1:

Input:a = cdacnoida, b = ciddacnoa

Output: YES

Explanation: Both the string have same characters with same frequency. So, both are anagrams.

```
package studentmgtsystem;
//Given two strings a and b consisting of lowercase characters. The task is to check
whether two given strings are an anagram of each other or not. An anagram of a string
is another string that contains the same characters, only the order of characters can
be different. For example, abc and bca are an anagram of each other.
```

```
//Example 1:
```

```
//Input:a = cdacnoida, b = ciddacnoa
//Output: YES
//Explanation: Both the string have same characters with
//           same frequency. So, both are anagrams.
```

```
import java.util.Arrays;
```

```
public class Anagram {

    public static void main(String[] args) {
        String str1 = "Silent";
        String str2 = "Listen";

        str1 = str1.toLowerCase();
        str2 = str2.toLowerCase();

        // check if length is same
        if(str1.length() == str2.length()) {

            // convert strings to char array
            char[] charArray1 = str1.toCharArray();
            char[] charArray2 = str2.toCharArray();

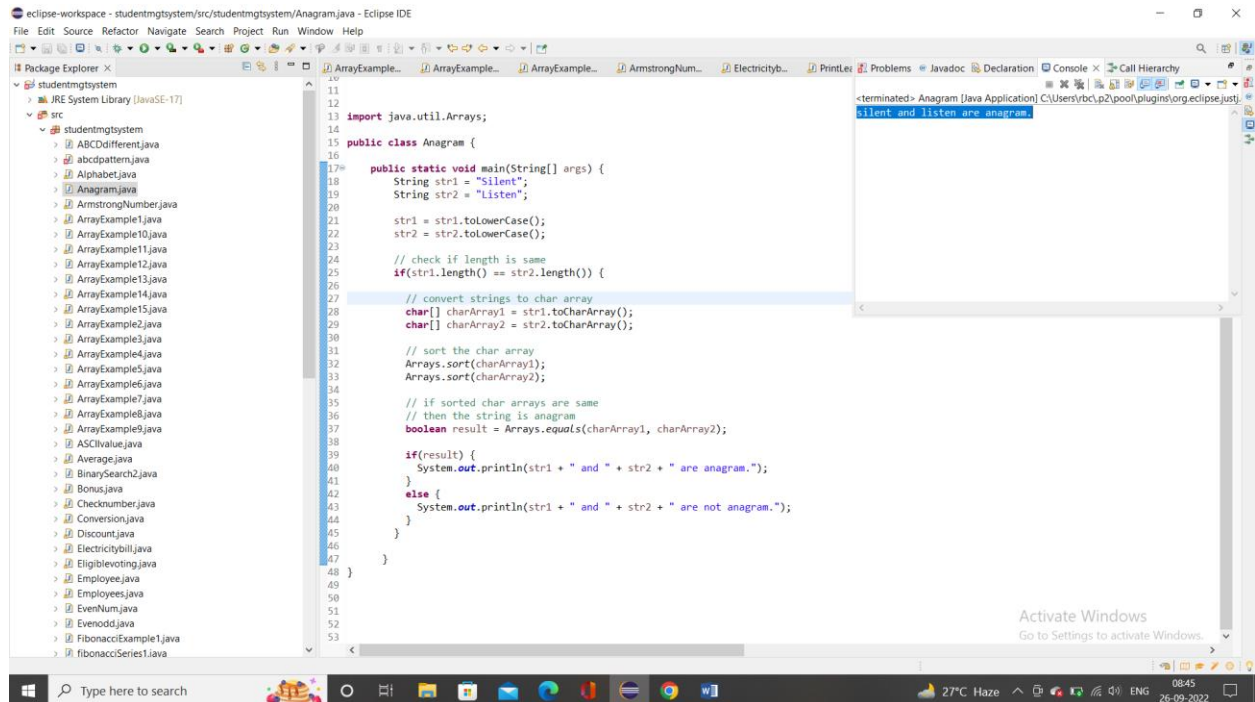
            // sort the char array
            Arrays.sort(charArray1);
            Arrays.sort(charArray2);

            // if sorted char arrays are same
            // then the string is anagram
            boolean result = Arrays.equals(charArray1, charArray2);

            if(result) {
                System.out.println(str1 + " and " + str2 + " are anagram.");
            }
            else {
                System.out.println(str1 + " and " + str2 + " are not anagram.");
            }
        }
    }
}
```

Result

silent and listen are anagram.



The screenshot shows the Eclipse IDE interface. On the left, the Package Explorer displays the project structure for 'studentmgmtsystem'. The main editor window shows the source code for 'Anagram.java'. The code defines a public class 'Anagram' with a 'main' method that takes a string array 'args'. It initializes two strings, 'str1' and 'str2', with the values 'Silent' and 'Listen' respectively. Both strings are converted to lowercase. The code then checks if the lengths of the two strings are equal. If they are, it converts each string into a character array, sorts the arrays, and compares them using 'Arrays.equals'. If the arrays are equal, it prints 'silent and listen are anagram.'. Otherwise, it prints 'silent and listen are not anagram.'. The Console window on the right shows the output of the program: 'silent and listen are anagram.'. The Windows taskbar at the bottom shows the system clock as 08:45 on 26-09-2022.

```
10
11
12
13 import java.util.Arrays;
14
15 public class Anagram {
16
17     public static void main(String[] args) {
18         String str1 = "Silent";
19         String str2 = "Listen";
20
21         str1 = str1.toLowerCase();
22         str2 = str2.toLowerCase();
23
24         // check if length is same
25         if(str1.length() == str2.length()) {
26
27             // convert strings to char array
28             char[] charArray1 = str1.toCharArray();
29             char[] charArray2 = str2.toCharArray();
30
31             // sort the char array
32             Arrays.sort(charArray1);
33             Arrays.sort(charArray2);
34
35             // if sorted char arrays are same
36             // then the string is anagram
37             boolean result = Arrays.equals(charArray1, charArray2);
38
39             if(result) {
40                 System.out.println(str1 + " and " + str2 + " are anagram.");
41             }
42             else {
43                 System.out.println(str1 + " and " + str2 + " are not anagram.");
44             }
45         }
46     }
47 }
48
49
50
51
52
53
```

<terminated> Anagram [Java Application] C:\Users\rbcl.p2\pool\plugins\org.eclipse.just...
silent and listen are anagram.

Activate Windows
Go to Settings to activate Windows.

Type here to search

27°C Haze 08:45 26-09-2022