

Q1.

```
// a team of 10 members. For this create a program to find the BMI and display
the height,
// weight, BMI, and status of each individual
// Hint =>
// a. Take user input for the person's weight (kg) and height (cm) and store it
in the
// corresponding 2D array of 10 rows. The First Column stores the weight and
the second
// column stores the height in cm
// b. Create a Method to find the BMI and status of every person given the
person's height
// and weight and return the 2D String array. Use the formula BMI = weight /
(height *
// height). Note unit is kg/m^2. For this convert cm to meter
// c. Create a Method that takes the 2D array of height and weight as
parameters. Calls the
// user-defined method to compute the BMI and the BMI Status and stores in a 2D
String
// array of height, weight, BMI, and status.
// d. Create a method to display the 2D string array in a tabular format of
Person's Height,
// Weight, BMI, and the Status
// e. Finally, the main function takes user inputs, calls the user-defined
methods, and displays
// the result.
import java.util.*;

public class BodyMassIndex {
    // Method to calculate BMI and status
    public static String[] calculateBMIAndStatus(double weight, double heightCm)
    {
        double heightM = heightCm / 100.0;
        double bmi = weight / (heightM * heightM);
        String status;
        if (bmi <= 18.4) {
            status = "Underweight";
        } else if (bmi <= 24.9) {
            status = "Normal";
        } else if (bmi <= 39.9) {
            status = "Overweight";
        } else {
            status = "Obese";
        }
        return new String[] {
            String.format("%.2f", heightCm),
```

```

        String.format("%.2f", weight),
        String.format("%.2f", bmi),
        status
    };
}

// Method to process all persons and build result array
public static String[][] buildResultArray(double[][] data) {
    String[][] result = new String[data.length][4];
    for (int i = 0; i < data.length; i++) {
        result[i] = calculateBMIAndStatus(data[i][0], data[i][1]);
    }
    return result;
}

// Method to display the result in tabular format
public static void displayResults(String[][] results) {
    System.out.printf("%-10s %-10s %-10s %-15s\n", "Height(cm)",
"Weight(kg)", "BMI", "Status");

System.out.println("-----"
);
    for (int i = 0; i < results.length; i++) {
        System.out.printf("%-10s %-10s %-10s %-15s\n", results[i][0],
results[i][1], results[i][2], results[i][3]);
    }
}

// Main method
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    double[][] data = new double[10][2];
    for (int i = 0; i < 10; i++) {
        System.out.println("Enter details for Person " + (i + 1) + ":");
        System.out.print("Weight (kg): ");
        data[i][0] = sc.nextDouble();
        System.out.print("Height (cm): ");
        data[i][1] = sc.nextDouble();
    }
    String[][] results = buildResultArray(data);
    System.out.println("\nBMI Results:");
    displayResults(results);
    sc.close();
}
}

```

"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2025.2\lib\idea_rt.jar=49880" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8

-Dsun.stderr.encoding=UTF-8 -classpath "E:\JAVA
PROGRAMS\steparyansingh\out\production\week1" BodyMassIndex

Enter details for Person 1:

Weight (kg): 85

Height (cm): 174

Enter details for Person 2:

Weight (kg): 55

Height (cm): 156

Enter details for Person 3:

Weight (kg): 56

Height (cm): 168

Enter details for Person 4:

Weight (kg): 55

Height (cm): 170

Enter details for Person 5:

Weight (kg): 75

Height (cm): 170

Enter details for Person 6:

Weight (kg): 60

Height (cm): 169

Enter details for Person 7:

Weight (kg): 66

Height (cm): 167

Enter details for Person 8:

Weight (kg): 78

Height (cm): 180

Enter details for Person 9:

Weight (kg): 65

Height (cm): 180

Enter details for Person 10:

Weight (kg): 75

Height (cm): 170

BMI Results:

Height(cm)	Weight(kg)	BMI	Status
174.00	85.00	28.08	Overweight
156.00	55.00	22.60	Normal
168.00	56.00	19.84	Normal
170.00	55.00	19.03	Normal
170.00	75.00	25.95	Overweight
169.00	60.00	21.01	Normal
167.00	66.00	23.67	Normal
180.00	78.00	24.07	Normal

180.00	65.00	20.06	Normal
170.00	75.00	25.95	Overweight

Q2.

```
// Write a program to find the frequency of characters in a string using unique
characters and
// display the result
// Hint =>
// a. Create a method to Find unique characters in a string using the charAt()
method and
// return them as a 1D array. Use Nested Loops to find the unique characters in
the text
// b. Create a method to find the frequency of characters in a string and
return the characters
// and their frequencies in a 2D array. The logic used here is as follows:
// i. Create an array to store the frequency of characters in the text. ASCII
values of
// characters are used as indexes in the array to store the frequency of each
character.
// There are 256 ASCII characters
// ii. Loop through the text to find the frequency of characters in the text
// iii. Call the uniqueCharacters() method to find the unique characters in the
text
// iv. Create a 2D String array to store the unique characters and their
frequencies.
// v. Loop through the unique characters and store the characters and their
frequencies
// c. In the main function take user inputs, call user-defined methods, and
displays result.
```

```
import java.util.*;

public class CharacterFrequencyASCII {
    // Method to find unique characters using charAt()
    public static char[] uniqueCharacters(String text) {
        int len = 0;
        try {
            while (true) {
                text.charAt(len);
                len++;
            }
        } catch (IndexOutOfBoundsException e) {}
        char[] unique = new char[len];
        int uniqueCount = 0;
        for (int i = 0; i < len; i++) {
            char c = text.charAt(i);
            boolean isUnique = true;
            for (int j = 0; j < i; j++) {
                if (text.charAt(j) == c) {
```

```

        isUnique = false;
        break;
    }
}
if (isUnique) {
    unique[uniqueCount++] = c;
}
}
char[] result = new char[uniqueCount];
for (int i = 0; i < uniqueCount; i++) {
    result[i] = unique[i];
}
return result;
}

// Method to find frequency of characters and return as 2D String array
public static String[][] characterFrequencies(String text) {
    int[] freq = new int[256];
    int len = 0;
    try {
        while (true) {
            text.charAt(len);
            len++;
        }
    } catch (IndexOutOfBoundsException e) {}
    for (int i = 0; i < len; i++) {
        freq[text.charAt(i)]++;
    }
    char[] unique = uniqueCharacters(text);
    String[][] result = new String[unique.length][2];
    for (int i = 0; i < unique.length; i++) {
        result[i][0] = String.valueOf(unique[i]);
        result[i][1] = String.valueOf(freq[unique[i]]);
    }
    return result;
}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter a string: ");
    String text = sc.nextLine();

    String[][] freqArr = characterFrequencies(text);
    System.out.printf("%-10s %-10s\n", "Character", "Frequency");
    System.out.println("-----");
    for (int i = 0; i < freqArr.length; i++) {
        System.out.printf("%-10s %-10s\n", freqArr[i][0], freqArr[i][1]);
    }
    sc.close();
}

```

```
}  
}
```

```
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA  
2025.2\lib\idea_rt.jar=49960" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8  
-Dsun.stderr.encoding=UTF-8 -classpath "E:\JAVA  
PROGRAMS\steparyansingh\out\production\week1" CharacterFrequencyASCII  
Enter a string: Hello everyone it's nice using IntelliJ IDE  
Character Frequency
```

```
-----  
H      1  
e      6  
l      4  
o      2  
      6  
v      1  
r      1  
y      1  
n      4  
i      4  
t      2  
'      1  
s      2  
c      1  
u      1  
g      1  
l      2  
J      1  
D      1  
E      1
```

Q3.

```
// Write a program to find the frequency of characters in a string using nested  
loops and  
// display the result  
// Hint =>  
// a. Create a method to find the frequency of characters in a string and  
return the characters  
// and their frequencies in a 1D array. The logic used here is as follows:  
// i. Create an array to store the frequency of each character in the text and  
an array to  
// store the characters in the text using the toCharArray() method  
// ii. Loops to Find the frequency of each character in the text and store the  
result in a  
// frequency array. For this use a Nested Loop with an Outer loop to iterate  
through
```

```

// each character in the text and initialize the frequency of each character to
1. And an
// Inner loop to check for duplicate characters. In case of duplicate increment
the
// frequency value and set the duplicate characters to '0' to avoid counting
them again.
// iii. Create a 1D String array to store the characters and their frequencies.
For this
// Iterate through the characters in the text and store the characters and
their
// frequencies
// b. Finally, the main function takes user inputs, calls the user-defined
methods, and displays
// the result.

import java.util.*;

public class CharacterFrequencyNestedLoops {
    // Method to find frequency of characters using nested loops
    public static String[] characterFrequencies(String text) {
        char[] chars = text.toCharArray();
        int[] frequencies = new int[chars.length];
        String[] result = new String[chars.length];

        for (int i = 0; i < chars.length; i++) {
            if (chars[i] != '0') {
                frequencies[i] = 1;
                for (int j = i + 1; j < chars.length; j++) {
                    if (chars[i] == chars[j]) {
                        frequencies[i]++;
                        chars[j] = '0';
                    }
                }
                result[i] = String.valueOf(chars[i]) + ": " + frequencies[i];
            }
        }
        return result;
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String text = sc.nextLine();

        String[] freqArr = characterFrequencies(text);
        System.out.println("Character | Frequency");
        System.out.println("-----");
        for (String frequency : freqArr) {
            if (frequency != null) {

```

```

        System.out.println(frequency);
    }
}
sc.close();
}
}

```

"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2025.2\lib\idea_rt.jar=49982" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath "E:\JAVA PROGRAMS\steparyansingh\out\production\week1" CharacterFrequencyNestedLoops
Enter a string: Hello evveryone it's lovely using IntelliJ IDE
Character | Frequency

```

-----
H: 1
e: 6
l: 6
o: 3
: 6
v: 3
r: 1
y: 2
n: 3
i: 3
t: 2
': 1
s: 2
u: 1
g: 1
I: 2
J: 1
D: 1
E: 1

```

Process finished with exit code 0

Q4.

```

// Write a program to find the frequency of characters in a string using the
charAt() method and
// display the result
// Hint =>
// a. Create a method to find the frequency of characters in a string using the
charAt() method

```



```

// and return the characters and their frequencies in a 2D array. The logic
used here is as
// follows:
// i. Create an array to store the frequency of characters in the text. ASCII
values of
// characters are used as indexes in the array to store the frequency of each
character.
// There are 256 ASCII characters
// ii. Loop through the text to find the frequency of characters in the text
// iii. Create an array to store the characters and their frequencies
// iv. Loop through the characters in the text and store the characters and
their
// frequencies
// b. In the main function take user inputs, call user-defined methods, and
displays result.

```

```

import java.util.*;

```

```

public class CharFrequency {
    // Method to find frequency of characters and return as 2D array
    public static Object[][] getCharFrequencies(String text) {
        int[] freq = new int[256]; // ASCII
        int len = text.length();
        // Count frequency
        for (int i = 0; i < len; i++) {
            freq[text.charAt(i)]++;
        }
        // Count unique characters
        int uniqueCount = 0;
        for (int i = 0; i < len; i++) {
            if (freq[text.charAt(i)] > 0) {
                uniqueCount++;
                freq[text.charAt(i)] = -freq[text.charAt(i)]; // Mark as
processed
            }
        }
        Object[][] result = new Object[uniqueCount][2];
        int idx = 0;
        for (int i = 0; i < len; i++) {
            char c = text.charAt(i);
            if (freq[c] < 0) {
                result[idx][0] = c;
                result[idx][1] = -freq[c];
                freq[c] = 0; // Reset to avoid duplicate
                idx++;
            }
        }
        return result;
    }
}

```

```

// Method to display character frequencies
public static void displayFrequencies(Object[][] freqArr) {
    System.out.println("Character | Frequency");
    System.out.println("-----");
    for (Object[] row : freqArr) {
        System.out.println("      " + row[0] + "      |      " + row[1]);
    }
}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter a string: ");
    String text = sc.nextLine();
    Object[][] freqArr = getCharFrequencies(text);
    displayFrequencies(freqArr);
    sc.close();
}
}

```

"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA
 2025.2\lib\idea_rt.jar=50063" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8
 -Dsun.stderr.encoding=UTF-8 -classpath "E:\JAVA
 PROGRAMS\steparyansingh\out\production\week1" CharFrequency
 Enter a string: Hello Everyone and welcome back to another program
 Character | Frequency

```

-----
H | 1
e | 6
l | 3
o | 6
 | 7
E | 1
v | 1
r | 4
y | 1
n | 3
a | 4
d | 1
w | 1
c | 2
m | 2
b | 1
k | 1
t | 2

```

h		1
p		1
g		1

Process finished with exit code 0

Q5

```
// Write a program to find the first non-repeating character in a string and
show the result
// Hint =>
// a. Non-repeating character is a character that occurs only once in the
string
// b. Create a Method to find the first non-repeating character in a string
using the charAt()
// method and return the character. The logic used here is as follows:
// i. Create an array to store the frequency of characters in the text. ASCII
values of
// characters are used as indexes in the array to store the frequency of each
character.
// There are 256 ASCII characters
// ii. Loop through the text to find the frequency of characters in the text
// iii. Loop through the text to find the first non-repeating character in the
text by checking
// the frequency of each character
// c. In the main function take user inputs, call user-defined methods, and
displays result.

import java.util.*;

public class NonRepeatingChar {
    // Method to find the first non-repeating character
    public static char findFirstNonRepeatingChar(String text) {
        int[] freq = new int[256]; // ASCII
        int len = text.length();
        // Count frequency
        for (int i = 0; i < len; i++) {
            freq[text.charAt(i)]++;
        }
        // Find first non-repeating
        for (int i = 0; i < len; i++) {
            if (freq[text.charAt(i)] == 1) {
                return text.charAt(i);
            }
        }
        return '\0'; // If none found
    }
}
```

```

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter a string: ");
    String text = sc.nextLine();
    char result = findFirstNonRepeatingChar(text);
    if (result != '\0') {
        System.out.println("First non-repeating character: " + result);
    } else {
        System.out.println("No non-repeating character found.");
    }
    sc.close();
}
}

```

"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA
 2025.2\lib\idea_rt.jar=50069" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8
 -Dsun.stderr.encoding=UTF-8 -classpath "E:\JAVA
 PROGRAMS\steparyansingh\out\production\week1" NonRepeatingChar
 Enter a string: Hello and welcome back to yet another episode of coding
 First non-repeating character: H

Process finished with exit code 0

Q4

```

// Find unique characters in a string using the charAt() method and display the
result
// Hint =>
// a. Create a Method to find the length of the text without using the String
method length()
// b. Create a method to Find unique characters in a string using the charAt()
method and
// return them as a 1D array. The logic used here is as follows:
// i. Create an array to store the unique characters in the text. The size is
the length of
// the text
// ii. Loops to Find the unique characters in the text. Find the unique
characters in the text
// using a nested loop. An outer loop iterates through each character and an
inner loop
// checks if the character is unique by comparing it with the previous
characters. If the
// character is unique, it is stored in the result array
// iii. Create a new array to store the unique characters
// c. Finally, the main function takes user inputs, calls the user-defined
methods, and displays

```

```

// the result.

import java.util.*;

public class UniqueChar {
    // Method to find the length of the text without using String.length()
    public static int getLength(String text) {
        int count = 0;
        try {
            while (true) {
                text.charAt(count);
                count++;
            }
        } catch (IndexOutOfBoundsException e) {
            // End of string
        }
        return count;
    }

    // Method to find unique characters using charAt()
    public static char[] findUniqueChars(String text) {
        int len = getLength(text);
        char[] unique = new char[len];
        int uniqueCount = 0;
        for (int i = 0; i < len; i++) {
            char c = text.charAt(i);
            boolean isUnique = true;
            for (int j = 0; j < i; j++) {
                if (text.charAt(j) == c) {
                    isUnique = false;
                    break;
                }
            }
            if (isUnique) {
                unique[uniqueCount++] = c;
            }
        }
        // Create a new array with only the unique characters
        char[] result = new char[uniqueCount];
        for (int i = 0; i < uniqueCount; i++) {
            result[i] = unique[i];
        }
        return result;
    }

    // Method to display unique characters
    public static void displayUniqueChars(char[] chars) {
        System.out.print("Unique characters: ");
        for (char c : chars) {

```

```

        System.out.print(c + " ");
    }
    System.out.println();
}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter a string: ");
    String text = sc.nextLine();
    int len = getLength(text);
    System.out.println("Length (without using length()): " + len);
    char[] uniqueChars = findUniqueChars(text);
    displayUniqueChars(uniqueChars);
    sc.close();
}
}

```

"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA
 2025.2\lib\idea_rt.jar=50106" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8
 -Dsun.stderr.encoding=UTF-8 -classpath "E:\JAVA
 PROGRAMS\steparyansingh\out\production\week1" UniqueChar
 Enter a string: Hello world I am here
 Length (without using length()): 21
 Unique characters: H e l o w r d I a m h

Process finished with exit code 0