Write a program to find the sum of digits of a given number using a **while** loop. If the sum is greater than 15, break the loop

//to find the sum of digits of a given number using a while loop. //45

//If the sum is greater than 15, break the loop

import java.util.Scanner;

public class SumOfDigits

{

public static void main(String[] args)

{

Scanner scanner = new Scanner(System.in);

System.out.print("Enter a number: ");

int number = scanner.nextInt();

int sum = 0;

while (number > 0) {

int digit = number % 10;

sum += digit;

number /= 10;

if (sum > 15) {

break;

}

}

if (sum > 15) {

System.out.println("The sum of digits is greater than 15.");

} else {

System.out.println("The sum of digits is: " + sum);

}

}

}

1. Importing Scanner:

import java.util.Scanner;

This line imports the Scanner class, which allows us to take user input from the console.

2. Class and Main Method:

public class SumOfDigits { ... }

This defines a class named SumOfDigits.

public static void main(String[] args) { ... }

This is the main method, where the program execution begins.

3. Creating a Scanner Object:

Scanner scanner = new Scanner(System.in);

This creates a Scanner object to read input from the user.

4. Prompting for Input:

System.out.print("Enter a number: ");

This prints a message asking the user to enter a number.

5. Reading Input:

int number = scanner.nextInt();

This reads the integer entered by the user and stores it in the number variable.

6. Initializing Sum:

int sum = 0;

This initializes a variable sum to 0 to store the sum of digits.

7. While Loop:

while (number > 0) { ... }

This loop continues as long as number is greater than 0.

8. Extracting Last Digit:

int digit = number % 10;

This extracts the last digit of number using the modulo operator (%) and stores it in the digit variable.

9. Adding Digit to Sum:

sum += digit;

This adds the extracted digit to the sum variable.

10. Removing Last Digit:

number /= 10;

This removes the last digit of number by dividing it by 10.

11. Checking for Sum Threshold:

if (sum > 15) { break; }

If the sum becomes greater than 15, the loop is terminated using the break statement.

12. Printing Result:

if (sum > 15) { ... }

If the loop was broken due to the sum exceeding 15, this block prints a message stating that the sum is greater than 15.

else { ... }

Otherwise, this block prints the final sum of digits.

=======================================================================================================

Here's a breakdown of the code with examples:

1. Initializing the Sum:

int sum = 0;

This line creates a variable named sum and sets its initial value to 0. This variable will be used to store the sum of the digits of the input number.

2. While Loop:

while (number > 0) { ... }

This loop continues as long as the number is greater than 0. It repeatedly extracts and processes the digits of the number.

3. Extracting the Last Digit:

int digit = number % 10;

This line extracts the last digit of the number using the modulo operator (%). The modulo operator gives you the remainder of a division, so number % 10 gives you the last digit. For example, if number is 1234, then number % 10 would be 4.

4. Adding the Digit to the Sum:

sum += digit;

This line adds the extracted digit to the sum variable. For example, if sum was 0 and digit was 4, then sum would become 4 after this line.

5. Removing the Last Digit:

number /= 10;

This line removes the last digit from the number by dividing it by 10. For example, if number was 1234, then number /= 10 would make it 123.

6. Checking for Sum Threshold:

if (sum > 15) { break; }

This line checks if the sum has become greater than 15. If it has, the break statement is used to exit the loop immediately.

7. Printing the Result:

The code outside the loop checks the final value of sum and prints either the sum or a message indicating that the sum is greater than 15.