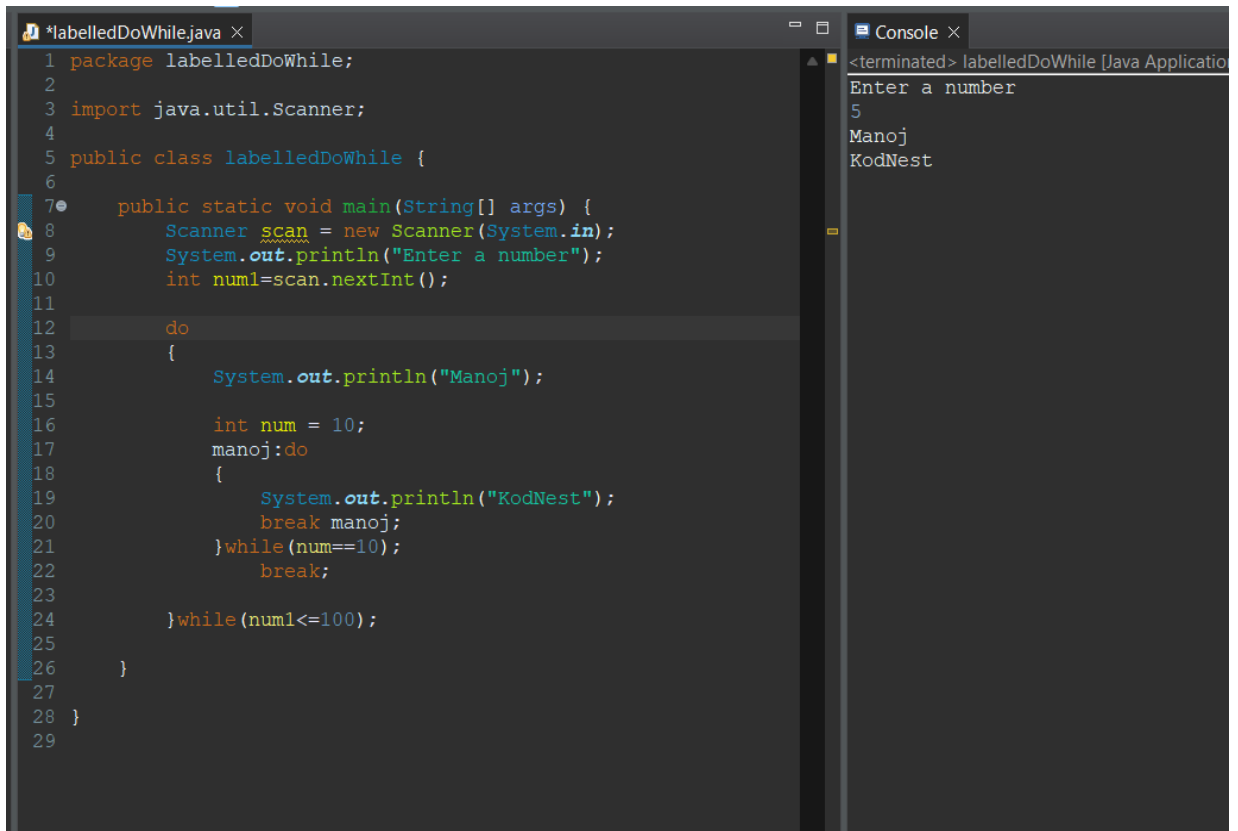


Labelled Do While

- A **label** is a valid variable name that denotes the name of the loop to where the control of execution should jump.
- To label a loop, place the label before the loop with a colon at the end. Therefore, a loop with the label is called a **labeled loop**.
- Placing the label/naming the do while loop is said to be **labelled do while loop**.
- Labelled loops are not commonly used in Java and are typically only necessary when you have **nested loops** and **need to specify which loop you want to break or continue**

Example:



```
1 package labelledDoWhile;
2
3 import java.util.Scanner;
4
5 public class labelledDoWhile {
6
7     public static void main(String[] args) {
8         Scanner scan = new Scanner(System.in);
9         System.out.println("Enter a number");
10        int num1=scan.nextInt();
11
12        do
13        {
14            System.out.println("Manoj");
15
16            int num = 10;
17            manoj:do
18            {
19                System.out.println("KodNest");
20                break manoj;
21            }while(num==10);
22            break;
23
24        }while(num1<=100);
25
26    }
27
28 }
29
```

Console Output:

```
<terminated> labelledDoWhile [Java Application]
Enter a number
5
Manoj
KodNest
```

In the above example I have labelled the do-while loop with name man whenever I want to break the loop I can break the loop by calling with label of the loop.

T.MANOJ