

Can We Place Any Statements Exactly Below break; And continue;

BREAK;

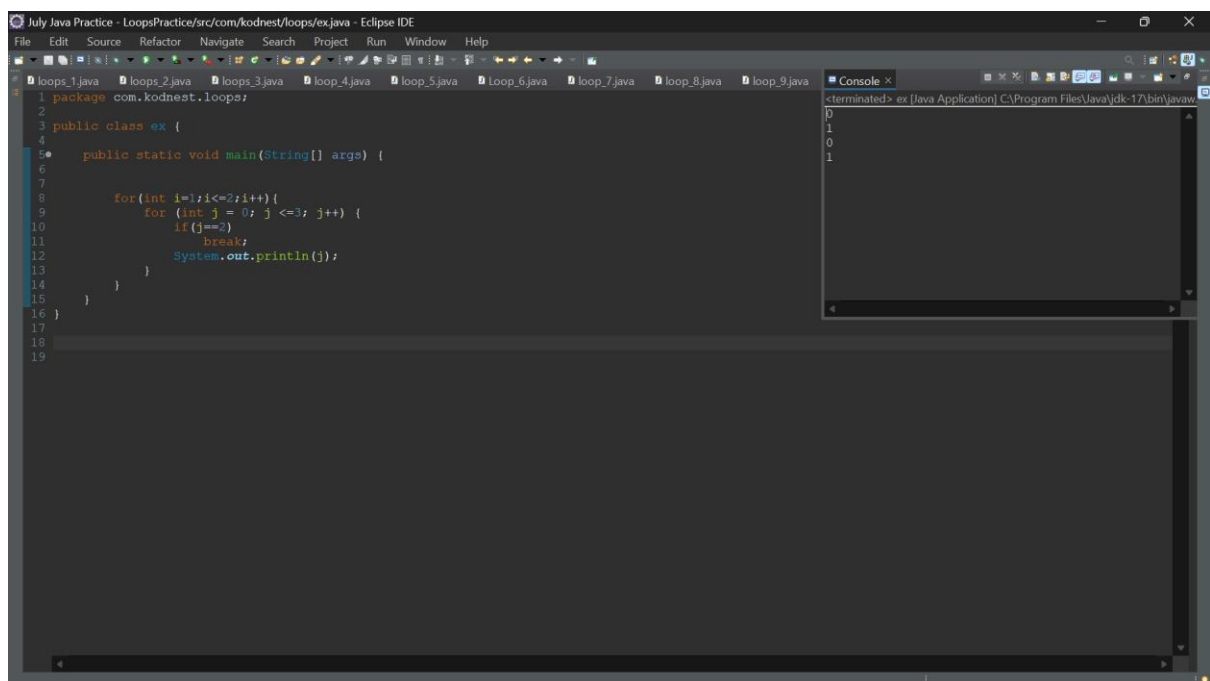
The break; statement is used to exit the innermost loop or switch statement it is placed in. Once the break; statement is encountered, the program will exit the loop or switch statement and continue executing the code after the loop or switch.

CONTINUE;

The continue; statement is used to skip the rest of the current iteration of the loop and proceed to the next iteration. It does not terminate the loop; instead, it just skips the remaining code in the current iteration and moves on to the next iteration.

You can place other statements after the break; and continue; statements, but those statements will not be executed within the loop iteration where the break; or continue; is encountered.

EXAMPLE FOR BREAK:

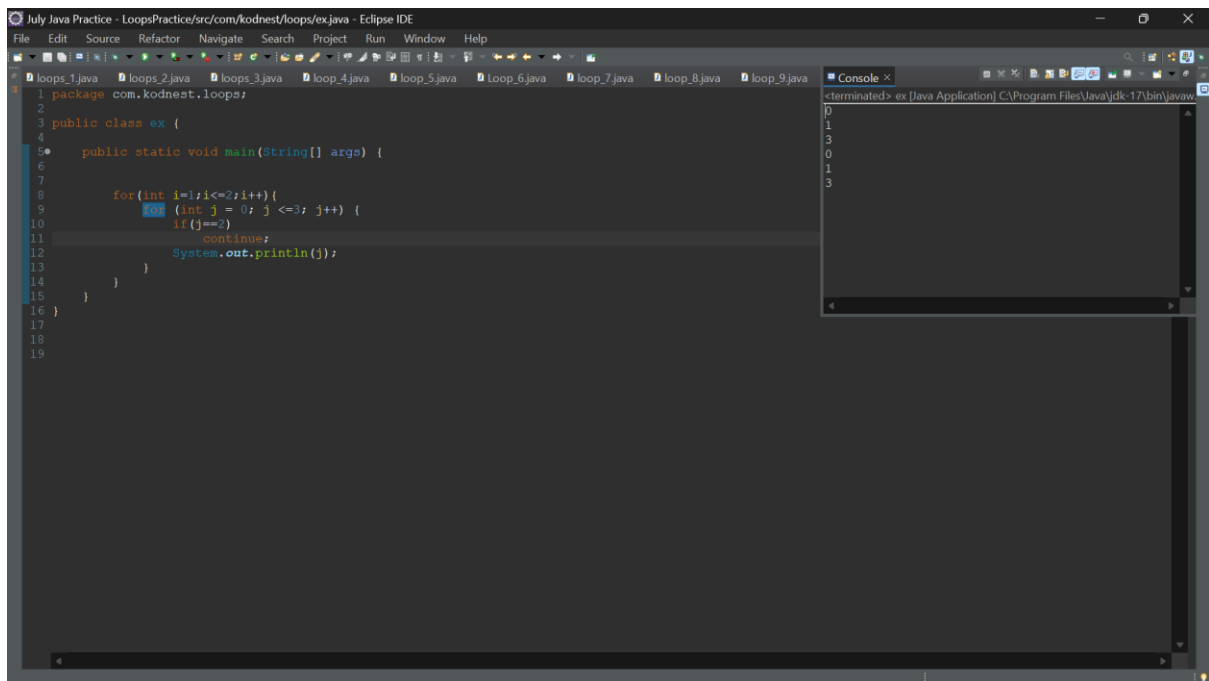


The screenshot shows the Eclipse IDE with a Java project named 'July Java Practice'. The editor displays a file 'ex.java' with the following code:

```
1 package com.kodnest.loops;
2
3 public class ex {
4
5     public static void main(String[] args) {
6
7         for(int i=1; i<=2; i++){
8             for(int j = 0; j <=3; j++) {
9                 if(j==2)
10                     break;
11                 System.out.println(j);
12             }
13         }
14     }
15 }
16
17
18
19
```

The console output shows the execution of the program, displaying the values 0, 1, and 0, which correspond to the values of j in the inner loop for i=1 and i=2. The program terminates after the first iteration of the inner loop for each value of i, as indicated by the 'break;' statement.

EXAMPLE FOR CONTINUE:



```
July Java Practice - LoopsPractice/src/com/kodnest/loops/ex.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
loops_1.java loops_2.java loops_3.java loops_4.java loops_5.java loops_6.java loops_7.java loops_8.java loops_9.java
1 package com.kodnest.loops;
2
3 public class ex {
4
5     public static void main(String[] args) {
6
7         for(int i=1; i<=2; i++){
8             for(int j = 0; j <=3; j++) {
9                 if(j==2)
10                    continue;
11                 System.out.println(j);
12             }
13         }
14     }
15 }
16 )
17
18
19
```

```
<terminated> ex [Java Application] C:\Program Files\Java\jdk-17\bin\javaw
0
1
3
0
1
3
```

In this example, after encountering the break; statement, the loop will exit and the statement after the loop will be executed. Similarly, after encountering the continue; statement, the current iteration will be skipped, and the loop will continue with the next iteration. The statements after the loops will still be executed as normal.