**Advantages of labeled break and continue over normal break and continue in case of nested loop**

Suppose, labeled break and continue feature is not supported by java. Now, let a nested loop exists (with two for loops) and for the inner loop there are two conditions (condition 1 and condition 2). Suppose, if condition 1 is satisfied during the execution of the inner loop, we want to jump out of the whole nested loop (not only the inner loop, but outer loop too)

Then possible solution of it is:-

int flag=0;

for(initialization1;validity checking1;increment/decrement counter1)

{

for(initialization2;validity checking2;increment/decrement counter2)

{

if(condition1)

{

flag=1;

break;

}

if(condition2)

{

//certain operations performed

}

}

if(flag==1)

{

break;

}

}

In this case, the complexity of the loop structure as well as concept of the loop gets increased. But labeled break and continue provides us an easy way to deal with the situation like this.

**Here, I rewrite the same program using labeled break concept:**

int flag=0;

Loop1 :for(initialization1;validity checking1;increment/decrement counter1)

{ //providing a label to the loop as “Loop1”

Loop2: for(initialization2;validity checking2;increment/decrement counter2)

{

if(condition1)

{

flag=1;

break Loop1;

}

if(condition2)

{

//certain operations performed

}

}

}