Nested Loop

- Definition and construct of Nested Loop
- Types of Nested Loop
- Pattern Programs

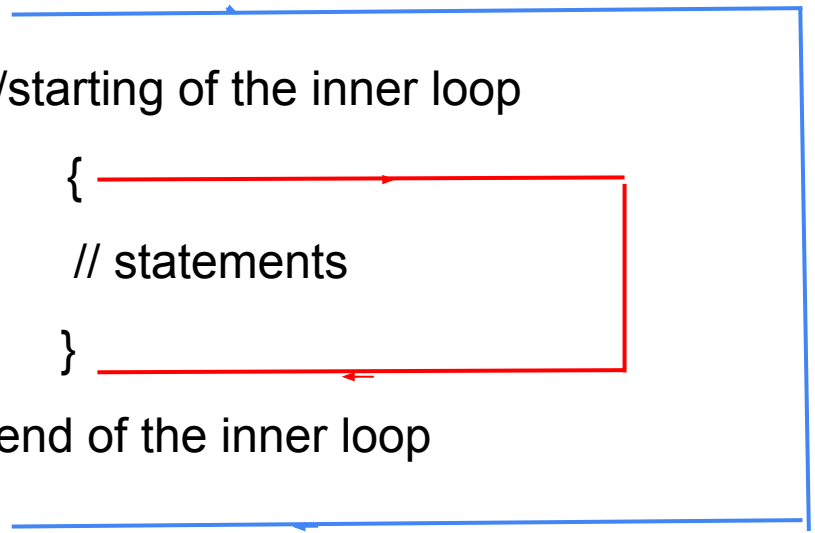
Definition of Nested Loop

A loop within another loop is called a “Nested Loop”.

Note:- In the nested loop the inner loop repeats a number of times for each repetition of the outer loop.

The construct of a nested loop

```
//starting of outer loop
{
  //starting of the inner loop
        {
         // statements
        }
 //end of the inner loop
}
//end of the outer loop
```



Types of Nested Loop

A nested loop can be designed in the following ways:

- ➔ Nested for loop
- ➔ Nested while loop
- ➔ Nested do-while loop

1. Nested for loop

When a for loop is used within another for loop ,it is called a nested for loop. A small program is illustrated as:

```
public class Pattern{
public static void main(String[] args){
        int a,b;
      for(a=1;a<=5;a++){
        for(b=1;b<=a;b++){
                System.out.print(b);
        }
        System.out.println()
}
}
```

Outer loop

Inner loop

OUTPUT

```
1
12
123
1234
12345
```

Note:- It is important to know that the inner loop is closed first and then the outer loop is closed. Similarly, the statements of inner loop are executed first which is followed by the outer loop.

2. Nested while loop

When a while loop is used within another while loop,it is called a nested while loop. An example is illustrated.

```
public class Pattern{

   public static void main (String[] args){

         int a=1;
       while(a<=5)
       {
         while(b<=a)
          {
            System.out.print(b);
            b++;
          }
       System.out.println();
       a++;

     }

}  }
```

Output
1
12
123
1234
12345

3.Nested do-while loop

When a do-while loop is used within another do-while, it is called a nested do while loop. A sample program is illustrated.

```
public class Pattern{
   public static void main (String[] args){
        int a=1;
      do
      {
          int b=1;
           do
           {
             System.out.println(b);
             b++;
             } while(b<=a);
          System.out.println();
          a++;
        } while(a<=5);
}
}
```

Output
1
12
123
1234
12345