Assignment-9

Submitted by

P.kundhana

Program 1:

Is nested simple if permitted?

Ans: Yes, java allows us to nest if statements within if statements. Nested if refers to if within an if statement. However, there can be any number of if statements within another if statement. The nested if statement comes under the category of the decision- making statements in java.

Example:

```
package nested_if;

public class Nested_if {
   public static void main(String[] args)
{
       int a=10;
        int b=20;

      if(a==10){

          if(b!=20){

System.out.println("Kodnest");
          }

          else{

System.out.println("omkar");
          }
     }

}
}
```

Output:

```
omkar
```

Program 2:

Is nested simple if else permitted?

Ans: Yes, java allows us to nest if else statements within if statements. Nested if else refers to if-else within an if-else statement. However, there can be any number of if-else statements within another if-else statement. The nested if-else statement comes under the category of the decision- making statements in java.

Example:

```
package Nested.IfElse;

public class Nested_IfElse {
    public static void main(String[] args)
{
        int n=24;
          if (n % 2 == 0){

                System.out.print("Even ");
                if (n % 6 == 0)
                {

                      System.out.println("and
divisible by 6");
                }
                else {

                      System.out.println("and
not divisible by 6");
                }
          }
          else {
            System.out.println("Odd ");

            if(n % 3 == 0)
            {

                System.out.println("and
divisible by 3");
            }
            else {

                System.out.println("and
not divisible by 3");
            }
        }
    }

}
```

Output:

```
Even and divisible by 6
```

Program 3:

Is nested if else permitted?

Ans: Yes, java allows us to nest if else statements within if statements. Nested if else refers to if-else within an if-else statement. However, there can be any number of if-else statements within another if-else statement. The nested if-else statement comes under the category of the decision- making statements in java.

Example:

```
package Nested.IfElse;

```

```
public class Nested_IfElse {
    public static void main(String[] args)
{
        int n=23;
         if  (n % 2 == 0){

              System.out.print("Even ");
              if  (n % 6 == 0)
              {

                   System.out.println("and
divisible by 6");
              }
              else {

                   System.out.println("and
not divisible by 6");
              }
         }
         else {

              System.out.println("Odd ");


         }
```

Output:

```
Odd
```

Program 4:

Is nested while permitted?

Ans: Yes, java allows us to nested while loop statements within while loop statements. A while loop inside another while loop is called nested while loop.

Example:

```
package Nested.whileLoop;

public class Nested_whileLoop {
    public static void main(String[] args)
{
    int i=1;
     while(i<=3)
     {
         System.out.println("\n"+i+"
"+"kodnest\n");

         int j=1;

         while(j<=4)
         {
             System.out.println(j+"
"+"omkar");
             j++;
         }

         i++;
     }

}
}
```

Output:

```
1 kodnest

1 omkar
```

```
2 omkar
3 omkar
4 omkar

2 kodnest

1 omkar
2 omkar
3 omkar
4 omkar

3 kodnest

1 omkar
2 omkar
3 omkar
4 omkar
```

Program 5:

Is nested do while loop permitted?

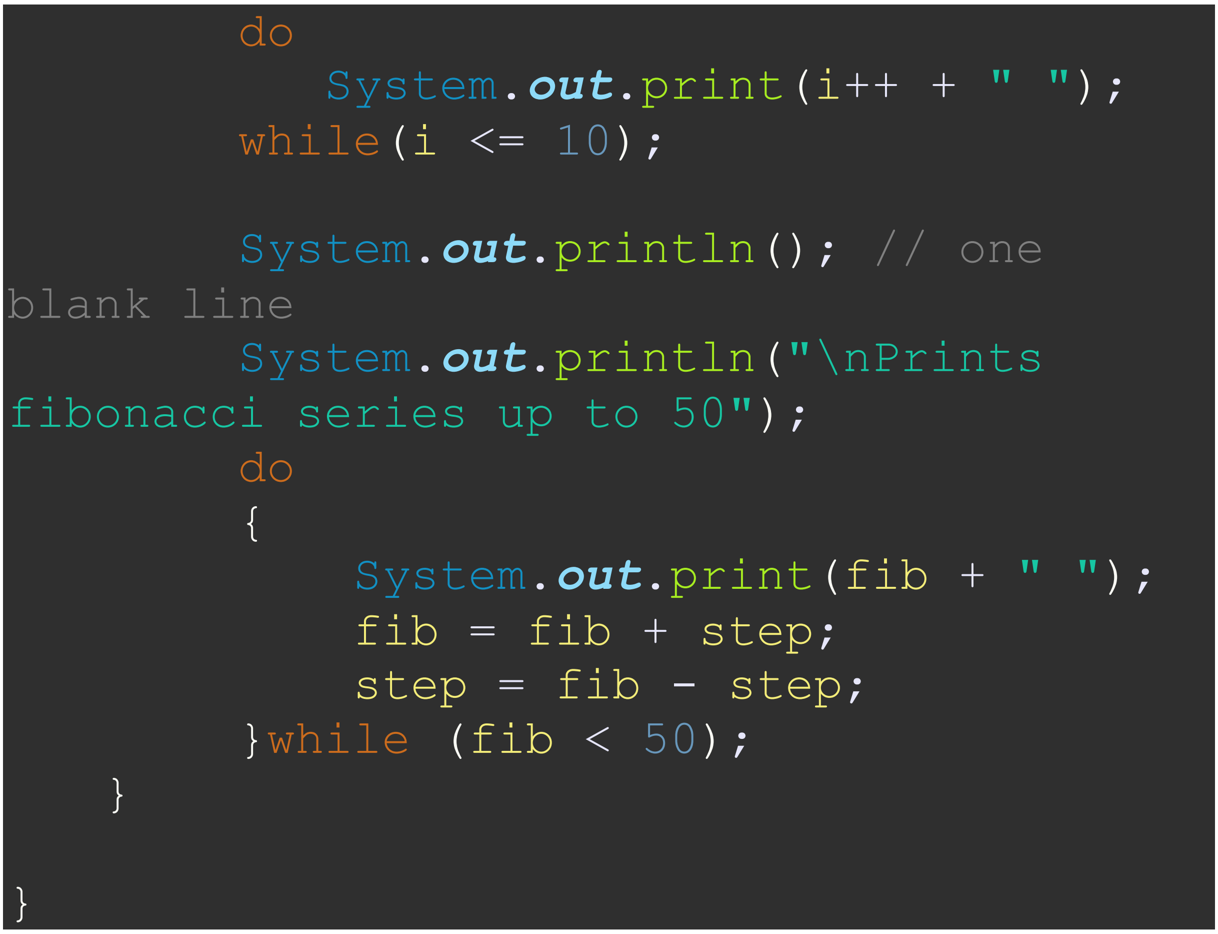
Ans: Yes, java allows us to nested do while loop statements within do while loop statements. A do-while loop inside another do-while loop is called nested do-while loop.

Example:

```
package Nested_doWhile;

public class Nested_doWhile {
    public static void main(String[] args)
{
        int fib = 0, step = 1, i = 1;

         System.out.println("Prints
counting from 1 - 10");
```

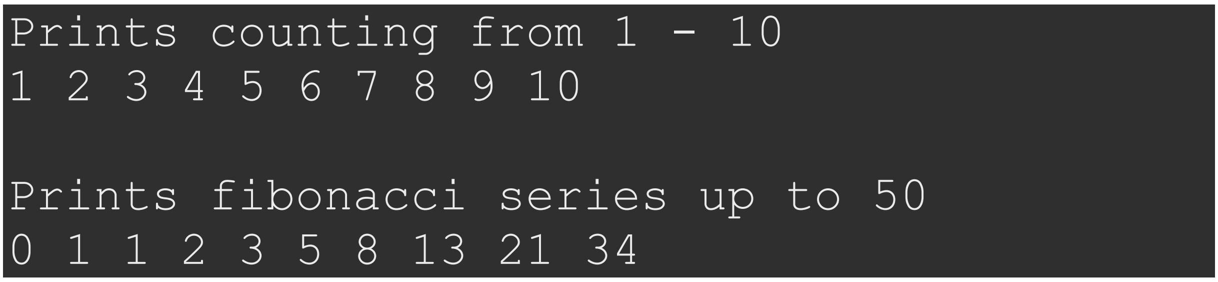


```
        do
           System.out.print(i++ + " ");
        while(i <= 10);

        System.out.println(); // one
blank line
        System.out.println("\nPrints
fibonacci series up to 50");
        do
        {
            System.out.print(fib + " ");
            fib = fib + step;
            step = fib - step;
        }while (fib < 50);
    }

}
```

Output:



```
Prints counting from 1 - 10
1 2 3 4 5 6 7 8 9 10

Prints fibonacci series up to 50
0 1 1 2 3 5 8 13 21 34
```