Guess the Output

* Snippet 1 –

```
public class NestedLoopOutput {
    public static void main (String args[]) {
        for (int i = 1; i<=3; i++){
            for (int j=1; j<=2; j++){
                System.out.print(i+" "+j+" ");
            }
            System.out.println();
        }
    }
}
```

i = 1 ⟶ 1 <= 3 ⟶
j = 1 ⟶ 1 <= 2 ⟶ 1 1
j = 2 ⟶ 2 <= 2 ⟶ 1 2
j = 3 ⟶ 3 <= 2 false

1	1	1	2
2	1	2	2
3	1	3	2

1 1
1 2
2 1
2 2
3 1
3 2

i = 2 ⟶ 2 <= 3 ⟶
j = 1 ⟶ 1 <= 2 ⟶ 2 1
j = 2 ⟶ 2 <= 2 ⟶ 2 2

i = 3 ⟶ 3 <= 3
j = 1 ⟶ 1 <= 2 ⟶ 3 1
j = 2 ⟶ 2 <= 2 ⟶ 3 2

o/p ⟶

1	1	1	2
2	1	2	2
3	1	3	2

Page No.
Date

Snippet 2 -

```
public class DecrementingLoop {
    psvm () {
        int total = 0;
        for (int i = 5; i > 0; i--) {
            total += i;      // total = total + i
            if (i == 3) continue;
            total -= 1;
        }
        System.out.println(total);
    }
}
```

Dry Run -

total = 0.

i = 5 → 5 > 0 → total = total + i = 0 + 5 = 5 .. 5--

~~total - total - i = 5 - 5 = 0~~

i = 4 → 4 > 0 → 4 + 4 = 8

~~5 + 5 = 10.~~

i = 3 → 3 > 0 → 10 + 3 = 13 → 13 - 1 = 12

+= t+i +=t-i

i = 5 → 5 > 0 → 0 + 5 = 5 → 5 - 1 = 4

i = 4 → 4 > 0 → 4 + 4 = 8 → 8 - 1 = 7

(i==3) i = 3 → 3 > 0 → 7 + 3 = 10 → skip.

i = 2 → 2 > 0 → 10 + 2 = 12 → 12 - 1 = 11

O/P → 11

Snippet 3 -

```
public class whileLoopBreak {
    public static void main (String. args[]) {
        int count = 0;
        while (count < 5) {
            System.out.println ( count + " ");
            count++;
            if (count == 3) break;
        }
        sop ( count);
    }
}
```

Dry Run -

count = 0

count < 5 → 0 < 5 → 0 → 1

1 < 5 → 1 → 2

2 < 5 → 2 → 3

(count == 3) 3 < 5 → 3 → 4 →

4 < 5 → 4 → 5

5 < 5 false.

o/p → 0 1 2 3

Snippet 4 -

```
public class Dowhileloop {
    p s v m () {
        int i=1;
        do {
            sop (i+" "); → 1
            i++; // 2
        }
        while (i<5); // 2<5
        sop (i);
    }
}
```

Dry Run -

	i	i++
i=1 →	1	1 → 2
i=2 →	2	2 → 3
	3	3
	4	4

0	P →	1	2	3	4

Snippet 5 →

```
public class conditionalLoopOutput {
    public static void main(String args[]) {
        int num = 1;
        for (int i=1; i<=4; i++) {
            if (i%2==0) {
                num += i;
            }
            else {
                num -= i;
            }
        }
        sop(num);
    }
}
```

DryRun –

num = 1

i=1 → 1<=4 → 1-1 = 0

i=2 → 2<=4 → 0+2 = 2

i=3 → 3<=4 → 2-3 = -1

i=4 → 4<=4 → -1+4 = 3

O/p → 3

Snippet 6 –

```
public class ID {
    psvm() {
        int x = 5;
        int y = ++x - x-- + --x + x++;
        sop(y);
    }
}
```

DryRun –

x = 5

++x = 6

x-- = 6

--x = 4

x++ = 4

y = 6 - 6 + 4 + 4

= 8

O/p → 8

Snippet 1 -

```
public class NestedIncrement{
    public static void main (String args []) {
        int a=10;
        int b=5;
        int result = ++a*b-- - --a + b++;
        Sop(result);
    }
}
```

Dry Run -

a=10 , b = 5

++a = 11 → increment before use

b-- = 5 → uses current value then decrement by 1.

--a = 10

b++ = 5

11 * 5 - 10 + 5

55 - 5

= 49

Snippet 8 -

```
public class LoopIncrement {
    p s v m () {
        int count = 0;
        for (int i=0; i<4; i++){
            count += i++ - ++i;
        }
        System.out.println(count);
    }
}
```

DryRun -

count = 0

i=0 → 0<4 → 0 + 0 - 2 = -2

i=2 → 2<4 → 2 - 4 = -2

i=4 → 4<4 false.

count += 0 - 2

count = count + i++ - ++i;

i=1

i=2

o/p → -4