While loop

```
initialization
while(condition){
upgradation;
}
```

```
for(int i=0;i<5;i++){
S.o.pln(i)
}
```

→ 0 1 2 3 4

```
int i=0;
while(i<5){
```

```
System.out.println(i);
i++;
}
```

do while loop

```
initialization
do {
// statement
//upgradation
} while (condition);
```

```
int i = 7;
do {
s.o.pln(i);
i++;
} while (i<6);
```

Output

→ 7

While

```
initialization
while(condition) {
//statement
upgradation;
}
```

```
int i = 7 ← false
while (i<6) {
s.o.pln(i);
i++;
}
```

Output

Fibonacci Series

↳ current value is sum of previous

0 1 1 2 3 5 8 13 21 34

0 1 1 2 3 5 8 13 21 34 55 .....
1 2 3 4 5 6 7 8 9 10 11

a=0, b=1;
sum = a+b → 3rd value.

a=b; → 1
b=sum → 1

4th → sum = a+b = 2
a=b → 1
b=sum → 2

5th
sum = a+b
= 1+2 = 3
a=b;
b=sum

```
int fibonacciValue(int n) {
if(n==1)
 return 0;
if(n==2)
   return 1;
 int a=0; -> 1st fibonacci value
 int b=1; ->2nd fibonacci value
 int sum=0;
for(i=3;i<=n;i++){
    sum=a+b;
    a=b;
    b=sum;
   }
S.O.P(sum);
```

n=5.

i=3. sum=1, a=1, b=1

i=4 sum=1+1=2, a=1, b=2

i=5 ← sum=1+2=3, a=2, b=3

Nth Fibonacci Number?

1 1 2 3 5 8 13 21

a = sp = 0

b = fp = 1

3rd sum = fp + sp;

= 1 + 0

= 1

n = 1

↓

return 0

n = 2

return 1

```
for(int i=3; i<=n; i++){
  sum = fp + sp;
  fp = sum;
  sp = fp;
}
```