

# Java for beginners

## Iterative control



Saurabh Shukla (MySirG)

# Agenda

- ① Iterative Control statements
- ② while
- ③ do-while
- ④ for
- ⑤ break
- ⑥ continue
- ⑦ nested loops
- ⑧ for-each loop

# Iterative Control Statements

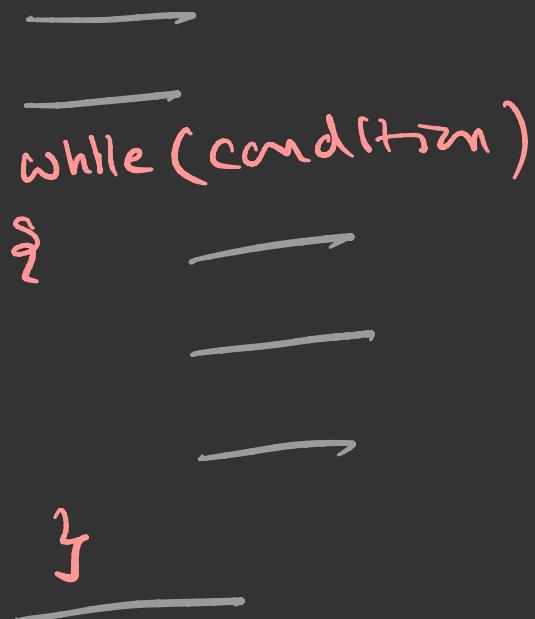
while

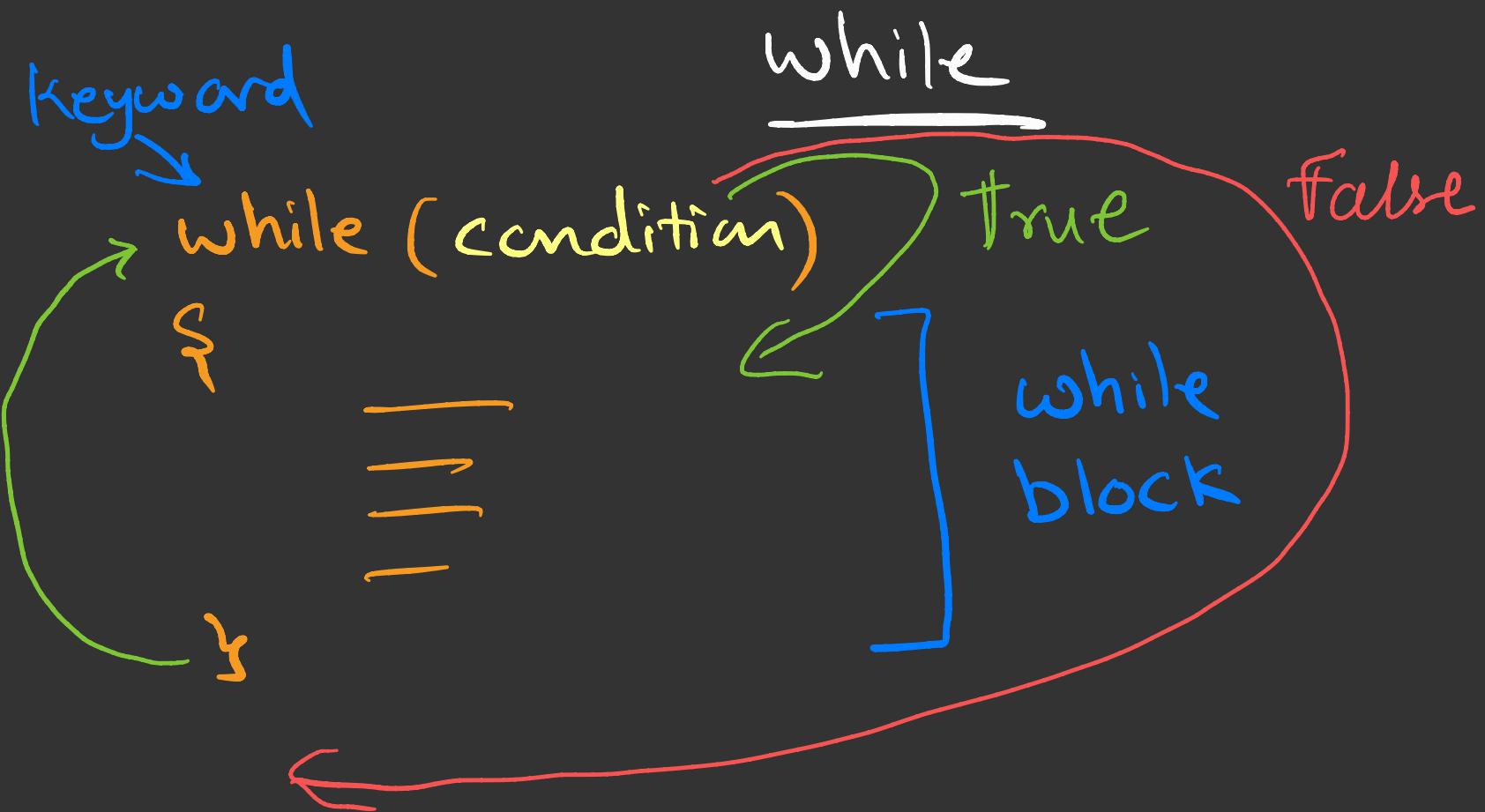
Loop

do while

for

for each





Condition must be evaluated as a boolean value (true or false)

Write a Java program to print MySirG  
5 times on the screen.

public class Example {

```
    public static void main(String[] args){  
        int i=1;  
        while(i<=5) {  
            System.out.println("MySirG");  
            i++;  
        }  
    }
```

MySirG  
MySirG  
MySirG  
MySirG  
MySirG

Write a Java program to print first 10 natural numbers.

public class Example {

    public static void main(String[] args) {

        int i = 1;

        while (i <= 10) {

            System.out.println(i);

            i++;

    }

}

}

Write a Java program to print first  
10 natural numbers in reverse order.

public class Example {

public static void main( String [] args ) {

int i = 1 ;

while( i <= 10 ) {

System.out.println( 11 - i );

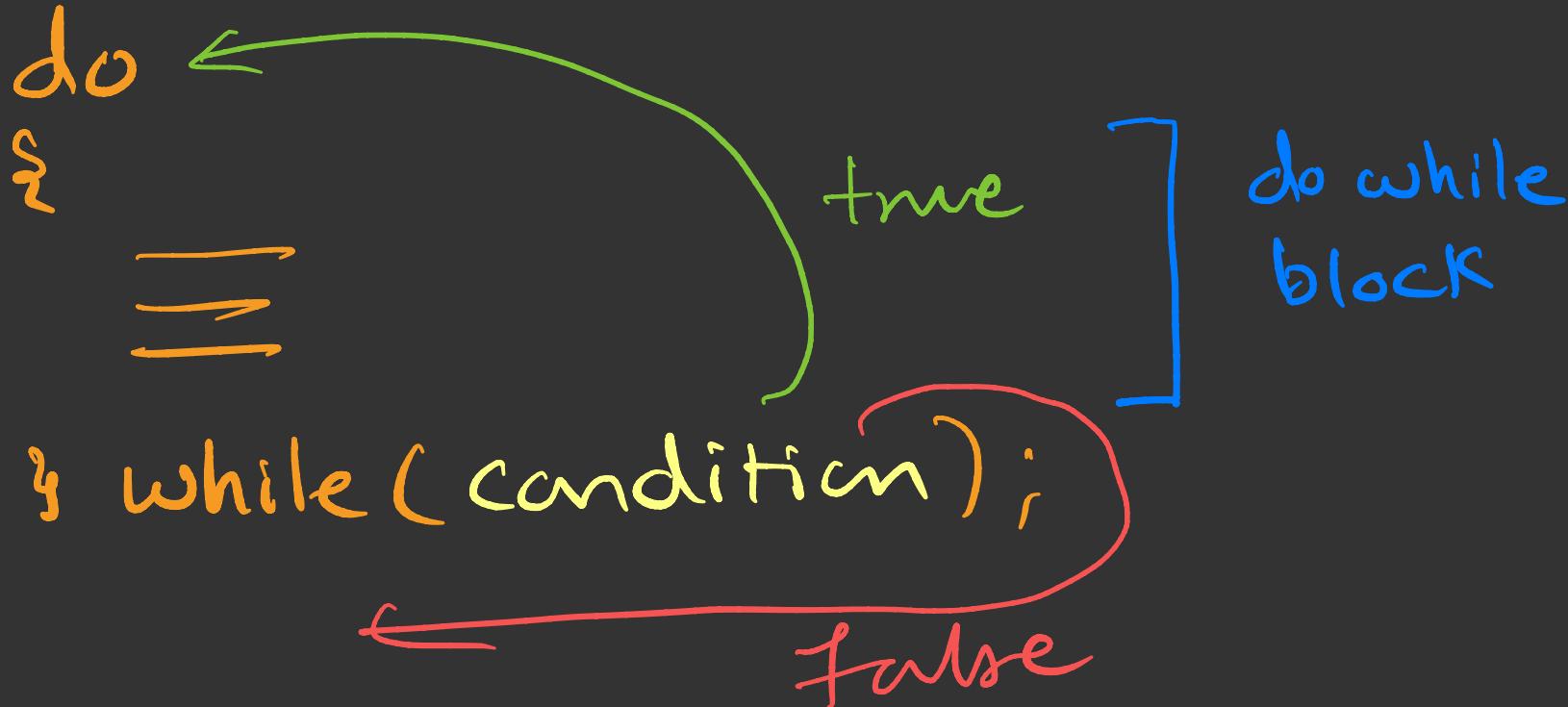
i++ ;

}

}

}

## do while



# Comparison between while and do-while

Entry control loop

while (condition)

{  
    
  }  
}



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

i <= 5	T	1
2 <= 5	T	2
3 <= 5	T	3
4 <= 5	T	4
5 <= 5	T	5
6 <= 5	F	

Exit control loop

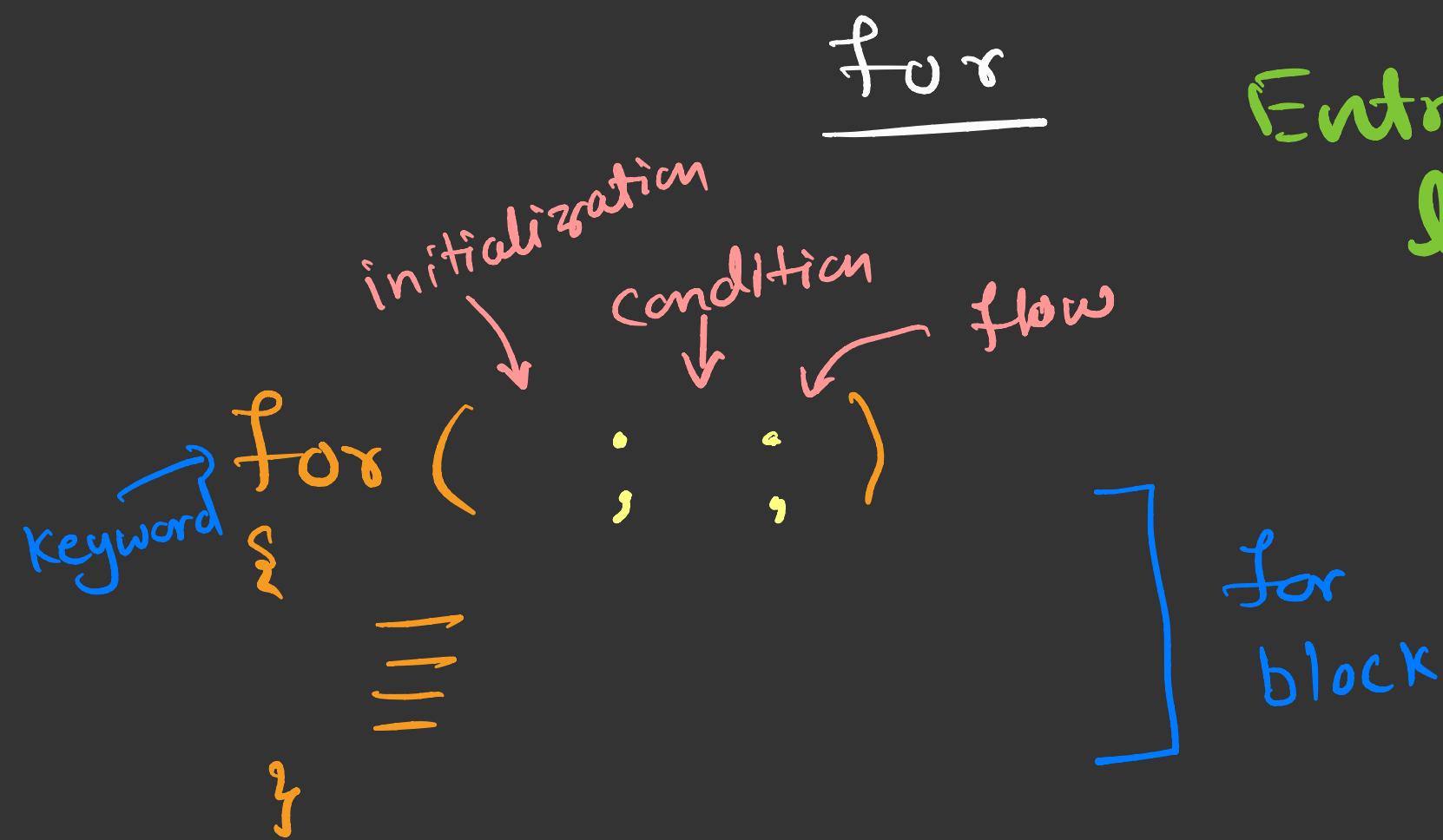
do  
{  
    
  }  
} while (condition);



int i=1;  
do  
{  
  System.out.println(i);  
  i++;  
} while (i<=5);

1		
2 <= 5	T	2
3 <= 5	T	3
4 <= 5	T	4
5 <= 5	T	5
6 <= 5	F	

# Entry control loop



Write a Java program to print first  
10 multiples of 5. (using for loop)

public class Example {

    public static void main(String[] args) {

        for (int i=1; i<=10; i++) {

            S.O.P.(5\*i);

}

}

?

## Nested Loop

```
while( )  
{
```

```
    for( ; ; )  
    {
```

```
        while( )  
        {  
            =  
        }
```

```
    while( )  
    {
```

```
}
```

```
    }  
}
```

## break

- The break is a control transfer statement.
- It is a keyword.
- It is used in the body of loop to terminate the execution of the loop.
- When the break statement is encountered inside a loop, the loop iteration stops and control moves to the first statement after the loop.
- break can also be used in the body of switch.

## Find Output of the code

```
int i;  
for (i=1 ; i<=10 ; i++)  
{    if (i==5)  
        break;  
    System.out.println("i=" + i);  
}  
System.out.println("outside loop");
```

i = 1  
i = 2  
i = 3  
i = 4  
outside loop

There is a need to break out from both the loops when sum of i and j becomes greater than 8.

```
for (int i=1 ; i<=10; i++)  
{
```

```
    7 1 8  
    7 2 break  
    8 1 break  
    9 1 break  
    10 1 break  
    11 false
```

```
        for (int j=1 ; j<=5 ; j++)  
{
```

```
            if (i+j > 8)  
                break;
```

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

```
        }  
        6 1 7 5 1 6  
        6 2 8 5 2 7  
        6 3 break 5 3 8  
        break 5 4  
        4 5
```

i	j	sum
1	1	2
1	2	3
1	3	4
1	4	5
1	5	6
2	1	3
2	2	4
2	3	5
2	4	6
2	5	7
3	1	4
3	2	5
3	3	6
3	4	7
3	5	8

```
        4 1 5  
        4 2 6  
        4 3 7  
        4 4 8  
        break  
        4 5
```

## Labelled break

first :

```
for (int i=1 ; i<=10; i++)
```

```
{
```

```
    for (int j=1 ; j<=5 ; j++)
```

```
{
```

```
        if (i+j > 8)
```

```
            break first;
```

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

```
}
```

```
}
```

## Continue

- Continue is a keyword
- It is used only in the body of loop
- It is used to skip the current iteration

```
while (condition)
{
    ==
    if (...)

        continue;

    ==
}

{
```

```
for ( ; ; )
{
    ==
    ==
    continue;
    ==
}

}
```

## Find Output

```
for ( int i=1; i<=4; i++ ) {  
    for( int j=1; j<=5 ; j++) {  
        if ( j>i)  
            continue;  
        System.out.print( "*" );  
    }  
    System.out.println( );  
}
```

You can use labelled continue as well.

## for - each

for each loop is applicable in collection of values to access all the elements one by one from first element to the last element.

for ( type var : array)

{      ==  
      }

}