- Sequential
- Selection
- iteration

Iteration

loops

- while
- for
- do-while

Loops and files

Start of the loop	initialize loop condition (1)
	start as long as condition is true (2)
task ;	task;
	reset condition ; (3)
	end
end of the loop	continue with the rest of the program

while

```
initialize the loop condition
while( Boolean condition){
    task ;
     reset of condition;
}
```

Counting loop:

```
double   avg ;
int count = 0 , sum = 0;
while( count < 10){
    print("enter a number: ");
    num = KB.nextInt();
    sum += num ;
    count++; // count += 1 ;
}
avg = (double)sum /count;
```

count	sum	num	avg
0	0	10	
	10		
1	30	20	
2	31	1	
3	35	4	
9	95	5	
10	100		10.0

```
double   avg ;
int count = 0 , sum = 0;                    count       sum     num     avg
while( count < 10  && sum < 50){            0           0       10      25.0
      print("enter a number: ");            1           10      40
      num = KB.nextInt();                   2           50
      sum += num ;
      count++; // count += 1 ;
}
avg = (double)sum /count;


int count = 0 , sum = 0;                    count       sum     num     avg
while( count < 10  && sum < 50){            0           0       10
      print("enter a number: ");            1           10      -5
      num = KB.nextInt();                   2           55      45      27.5
      if ( num >= 0 ){
            sum += num ;
            count++; // count += 1 ;
      }
}
avg = (double)sum /count;

fileIn = new Scanner();
while( fileIn.hasNext() ){
    num = fileIn.nextInt();

      process num
}

fin = new Scanner(file1);   // initialize
while( fin.hasNext()){       //check of condition
     x = fin.next…()          // reset of condition
      process x ;
}
```

For loop: (counting loop)

```
++, -- operators
    ++  increment by 1 unit
     --   Decrement by 1 unit
int   n = 6, m = 0;
char  ch = 'c' ;
float   a = 2.2 ;
                                        n    m   ch   a
                                        6    0   c    2.2
n ++ ;    // n = n+1 ; n+= 1            7
m-- ; // m = m -1; m -=1                    -1
ch++ ; // ch +=1 ;                          d
ch-- ;                                      c
a++;                                             3.2
```

a--;

++n ; ++ch; ++m; --n, --ch ; --m;

++n ; pre-inc

n++ ; post-inc

m = n++; → m = n ; n++ ; 8 7

m= ++n ; → ++n ; m = n ; 9 9

2.2

++, -- pre-inc , pre-dec

()

-, + (Sign operator) → +n; -n ;

*,/,%

…. add, subtract, relational ops, boolean ops, -=, +=

=

++, -- post-inc or post-dec

For loop Syntax:

```
for(initialize cond. ;  check cond. ; reset cond.)  { // 3 steps are all together inside the ()
    task;
}
```

Example:

```
for(int i = 0 ; i < 10 ; i++)
    System.out.print(i+"   ");
```

Output:

0 1 2 3 4 5 6 7 8 9

i

0

1

2

3 ….. 9 10

```
for(int i = 9 ; i >= 0 ; i--)
    System.out.print(i+"   ");
```

9

8 -….. 0 -1

Output:

9 8 … 0

```
int  i ;
for ( i = 1 ; i <= 50 ; i++)
    if( i % 2 == 1)
        .print(i, "  ");
    else
        // process even number

print(i);
```

initialize→0 check cond. → 1 reset cond. →3

for(i = 1; i<=50 ; i += 2)

.print(i," "); // task → 2

```
for(int i = 9 ; i >= 0 ; i--)
    System.out.print(i+"   ");
print(i) ;  // error
```

9

8 -….. 0 -1

```
int  i = 0;                 // initialize
for (   ;  i <= 50 ; i++)   // check and reset
    task ;
```

```
int  i = 0;                 // initialize
for (   ;  i <= 50 ;  ){   // check
    task ;
    i++;                // reset
}
```

```
int  i  = 0;
while(i <= 50){
    task;
    i++;
}
```