★ LOOPS

* Loops helps to execute the same condition many times or we can say repetition.
* Loop means repetition of same work or condition.
* There are 3 types of loops :-
  i) while Loop
  ii) for Loop
  iii) do while loop

i) while Loop –

```
while (condition) {
    //do something
}
```

→ while statement जबतक Point करेगा जबतक while की condition True है।

ii) for Loop –

```
for (initialization; condition; updation) {
    //do something
}
```

iii) do-while Loop –

```
do {

} while (condition);
```

DRY RUN →

```
int i=1;
do {
    System.out.println("Hello world");
    i++;
} while (i <= 10);
```

• To important statements in the Loop are :-
  i) Break
  ii) Continue

i) Break Statement - to exit the loop.

#DRY RUN →
```
for (int i = 1; i <= 5; i++) {
    if (i == 3) {
        break;
    }
    System.out.println(i);
}
System.out.println("I am out of the Loop.");
```

* Keep entering numbers till users enter a multiple of 10:

```
do {
    Syop ("Enter Number:");
    int n = sc.nextInt();
    if (n % 10 == 0) {
        break;
    }
    Syop (n);
} while (true);
Syop ("EXIT! Number is multiple of 10.");
```

ii) Continue Statement - to skip an iteration.

#DRY RUN →
```
for (int i = 0; i <= 5; i++) {
    if (i == 3) {
        continue;
    }
    Syop (i);
}
```

* Display all numbers entered by user except multiples of 10.

```
do {
    Syop ("Enter Number: ");
    int n = sc.nextInt();
```

```
        if (n % 10 == 0) {
            continue;
        }
        Syop (n);
    } while (true);
```

* Check if a number is prime or not

```
Syop ("Enter number: ");
int n = sc.nextInt();
boolean isPrime = true;
for (int i = 2; i <= n-1; i++) {
    if (n % i == 0) {
        isPrime = false;
    }
}
if (isPrime == true) {
    Syop ("It is Prime number");
else {
    Syop ("It is Composite number");
}
```

For big numbers we ~~cho~~ can change a condition only i.e, in place of $i <= n-1$;

$$\Downarrow$$

$$i <= \sqrt{n};$$

$$\Downarrow$$

In Java, we $\sqrt{n}$ write as :-

```
i <= Math.sqrt(n);
```

QUESTIONS

Ques1.

```
public class ques1 {
    public static void main (String args[]) {
        for (int i=0; i<5; i++) {
            System.out.println ("Hello");
            i+=2;
        }
    }
}
```

i=0
i=2
i=2+2
i=4
i=4+2
i=6

Hello
Hello

Ques2. Write a program that reads a set of integers, & then prints the sum of the even & odd integers.

```
int num;
int choice;
int evenSum;
int oddSum;
do {
    System.out.print ("Enter the number: ");
    num = sc.nextInt();
    if (num %2 == 0) {
        evenSum += num;
    }else {
        oddSum += num;
    }
    System.out.println ("Do you want to continue? Press 1 for
                        YES or 0 for NO.");
    choice = sc.nextInt();
} while (choice == 1);
System.out.println ("Sum of even numbers: " + evenSum);
System.out.println ("Sum of odd numbers: " + oddSum);
```

* In Output, it will one num at a time, it is STACK like program.

Ques 3. Factorial program

```
int num;
int fact = 1;
System.out.print ("Enter the Number: ");
num = sc.nextInt();
for (int i=1; i<=num; i++) {
    fact = fact * i;
}
System.out.println ("Factorial: " + fact);
```

Ques 4. Multiplication of n number

```
System.out.print ("Enter the Number: ");
int num = sc.nextInt();
for (int i=1; i<=num; i++) {
    System.out.println (num + "x" + i + " = " + num*i);
}
```

Ques 5.

```
for (int i=0; i<=5; i++) {
    System.out.println ("i = " + i);
}
System.out.println ("After the Loop i = " + i);
```

* i is declared in the for loop.
* So, scope of variable i is limited to the for loop only.
* Display statement after the loop where variable i is used which means i is used out of scope.
* Program will not run & give error.
* To correct program, variable i needs to be declared outside the for loop.