**DAY-01**

**Method**:Writing a function inside a class is called Method.

Method Statements:

-declarations

-defnitions

-calling

->declaration and definition should be outside the main class

->calling should be inside main class and main method

->Syntax:

function-type function-name(parameters){

defnitions

declarations

}

function-name() //function-call

->Function-types:

Except "Void" we have to return the data , for void we have to print the data.

->Except "Void" remaining all are Data-Types those are having some default value

import java.util.\*;

public class Main{

public static void main(String[] args){

Scanner sc = new Scanner(System.in);

int a = sc.nextInt();

int b = sc.nextInt();

int res;

function add(){

res = a+b;

System.out.println(res);

}

add();

}

}

**WITH PARAMETER JAVA:**

public class SimpleProgram {

public static void addNumbers(int num1, int num2) {

int sum = num1 + num2;

System.out.println("Sum: " + sum);

}

public static void main(String[] args) {

int number1 = 5;

int number2 = 10;

addNumbers(number1, number2); }

}

**INPUT/OUTPUT:**

Scanner Class:

Before Scanner Class we have "String buffer"

String Buffer:

String buffer will accept only strings and later we have to convert into targeted class

->We can't convert multiple times That's where Scanner class comes in:

->In scanner class we have multiple pre-defined methods

->Just creating an object for the scanner class we can use those pre-defined methods.

->creating objects for scanner class and calling those objects in scanner class.

->Create a class using class Keyword and name name of the class. the starting letter of the class should be capital(optional)

->For all the methods the name of thhe method should be small.

import java.util.\*;

public class Main{

public static void main(String[] args){

Scanner sc = new Scanner(System.in);

String s = sc.next();

System.out.println(s);

}

}

**CONDITIONAL STATEMENTS:**

Types:

i)if

ii)if-else

ii)switch case

->conditional statements are used to check the condition and to print the relevent block in constant time

->if-pseudo code:

public class Main{

public static void main(String[] args){

int n = 10;

if(n == 10){

System.out.println("hi");

}

}

}

->if-else Pseudo code:

public class Main{

public static void main(String[] args){

int n = 10;

if(n == 10){

System.out.println("hi");

}else{

System.out.println("Bye")

}

}

}

->switch-case pseudo code:

public class Main{

public static void main(String[] args){

int n = 10;

}

}

**CONTROL STATEMENTS:**

Two types of Control Statements:

i)Entry-control loops

ii)exit-control loops

\*Entry control loops:

->for loop and while loop

->for-loop Syntax:

for(start;end;diff){

//statements

}

for(intialization;condition;updation){

//statements

}

->while loop syntax:

start;

while(end){

diff;

//statements;

}

->when we know the range we prefer to use for-loop

->when we dont know the range we prefer to use while loop

//EXAMPLE:

int arr[] = {1,2,3,4,5};

for(int i = 0;i<n;i++){

return arr[i]

}

**NESTED LOOPS:**

**Nested Loops**

**There are 3 types of solutions for any problem**

**i)Brute-force**

**ii)Better**

**iii)optimal**

**->Brute-Forece: Fore Brute-Forec most of the solutions can be written in two(2) loops**

**Pattern printing:(Steps)**

**i):- How many rows you want to print(outer-loop)**

**ii):- How many coloumns you need to print(inner-loop)**

**iii):- What to print and what not to print**