**5) Methods in Java**

# Defining Methods

**Answer:-** In Java, a **method** is a block of code designed to perform a specific task. Methods are essential for organizing and reusing code in a structured way. Syntax:- accessModifier returnType methodName(parameters)

{

// Method body

}

# Method Parameters and Return Type

**Answer:-** In Java, **method parameters** and **return types** play a crucial role in determining how methods receive input and return output.

**Example :-**

## Method With Parameters and Return Type

public double calculateArea(double radius)

{

return 3.14 \* radius \* radius; }

## Method With No Parameters But a Return Type

public String getWelcomeMessage()

{

return "Hello, World!"; }

## Method With Parameters and No Return Type

public void printName(String name)

{

System.out.println("Name: " + name); }

# Method Overloading

**Answer:-** The two or more method name should be same in a single class but its behaviors(data types, arguments) are different i.e Method Overloading.

## Example :-

**class** Addition

{

**int** x,y; **public** **void** sum()

{ x=20; y=10;

System.***out***.println("First Sum is : "+(x+y));

}

**public** **void** sum(**int** a)

{

x=y=a;

System.***out***.println("Second Sum is : "+(x+y));

}

**public** **void** sum(**int** a,**int** b)

{ x=a; y=b;

System.***out***.println("Third Sum is : "+(x+y));

}

**public** **void** sum(**float** a)

{

x=y=(**int**)a;

System.***out***.println("Fourth Sum is : "+(x+y));

}

}

**public** **class** MethodOverloadingDemo

{

**public** **static** **void** main(String[] args)

{

Addition a1=**new** Addition(); a1.sum(); a1.sum(20); a1.sum(30, 30); a1.sum(10.04f);

}

}

# Static Methods and Variables

**Answer:-**

**Static Variables :**

Astatic variableis a variable that belongs to the class rather than instances of the class. It is shared across all objects of the class.

**Static Methods**

A static method is a method that belongs to the class and not to any specific object. It can be called without creating an instance of the class.

=====================================================================

==========================================================

===========

=====================================================================

==========================

=========================================

=