



Concepts of Programming

Day 5: Sep 2022

Introduction to Java

Kiran Waghmare

CDAC Mumbai

- Methods
- static
- Constructors

Types of variable:

1. Instance variable

- variables defined in class

```
class Book{
    //Instance variable
    String Title;
    String Author;
    int price;

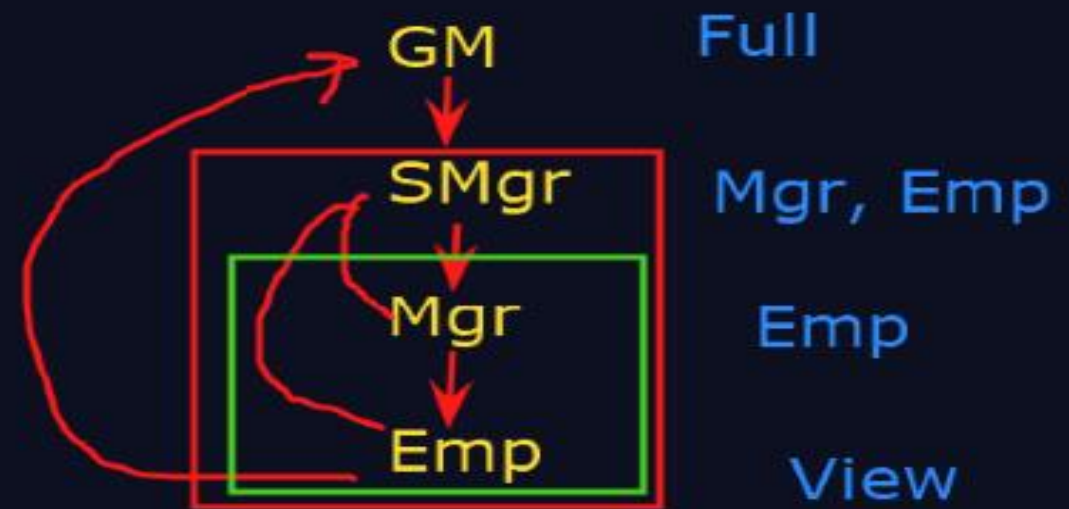
    //constructor
    //methods
    psvmmain()
    {
    }
}
```

Access specifier:

- private
-

2. Local variable

- variables used in methods, constructors and block



```
public class PredefinedMethod1{
```

Who can see what you share here? Recording On

```
    public static void main(String args[]){
```

```
        System.out.println("Square root = "+Math.sqrt(25));
```

```
    }
```

Command Prompt

```
C:\Test>javac PredefinedMethod1.java
```

```
C:\Test>java PredefinedMethod1  
Square root = 5.0
```

```
C:\Test>
```

```
public class InstanceMethod1{
```

Who can see what you share here? Recording On

```
void sayHello() ✓  
{  
    ✓ System.out.println("Hello CDAC !");  
}  
  
public static void main(String args[]){  
    InstanceMethod1 i1 = new InstanceMethod1();  
  
    //Method call  
    i1 sayHello();  
}  
}
```

A red line originates from the `i1 sayHello();` statement in the `main` method, loops around the right side of the code block, and points to the `sayHello()` method definition at the top. A green box highlights the entire `main` method block.

20 Who can see what you share here? Recording On

```
1 public class InstanceMethod1{
2
3
4     void sayHello()
5     {
6         System.out.println("Hello CDAC !");
7
8     }
9
10    public static void main(String args[]){
11
12        InstanceMethod1 i1 = new InstanceMethod1();
13
14        //Method call
15        i1.sayHello();
16
17        i1.sayHello();
18    }
19 }
```

```
public class InstanceMethod3{
```

```
void sayHello(String s)
```

```
{  
    String str;
```

```
    str = s;
```

```
    return s;  
}
```

```
public static void main(String args[]){
```

```
    InstanceMethod2 i1 = new InstanceMethod2();
```

```
    //Method call : calling od a function
```

```
    String s1 = i1.sayHello("Good morning");
```

```
    System.out.println(s1);  
}
```

main()

sayHello()

i1.sayHello()




```
1 class Test3{
2
3     static int x = 20; //static variable
4     int y = 30; //instance variable
5
6     //non static method
7     void display()
8     {
9         System.out.println(x);
10        System.out.println(y);
11        show();
12    }
13
14    //static method
15    static void show()
16    {
17        System.out.println(x);
18        //System.out.println(y); //Error
19    }
20
21    public static void main(String args[]){
22        Test3 t1 = new Test3();
23        t1.display();
24        show();
25    }
26 }
```

Who can see what you share here? Recording On

Command Prompt

C:\Test>javac Test3.java

Test3.java:17: error: non-static variable y
enclosed from a static context

System.out.println(y);

^

1 error

C:\Test>javac Test3.java

C:\Test>java Test3

20

30

20

20

C:\Test>

```
static int x = 20; //static variable  
int y = 30; //instance variable
```

```
//non static method  
public void display()  
{  
    System.out.println(x++);  
    System.out.println(y);  
    show();  
}
```

```
//static method  
public static void show()  
{  
    System.out.println(x);  
    //System.out.println(y); //Error  
}
```

```
public static void main(String args[]){  
    Test3 t1 = new Test3();  
    t1.display();  
}
```



public class Test5{

static int x = 20; //static variable

static int change()

{
int x = 10;
return x;
}

public static void main(String args[]){

System.out.println(x);

int a = change();

System.out.println(a);

System.out.println(x);

}

}

Who can see what you share here? Recording On

Command Prompt

C:\Test>javac Test4.java

C:\Test>javac Test5.java

C:\Test>java Test5

10

C:\Test>javac Test5.java

C:\Test>java Test5

20

10

20

C:\Test>

