Java

Agenda

- this keyword

- Call By Value & Call By Reference

- Static variables & static methods

this keyword

```java
public class Calculator
{
int a, b;
void add(int a, int b)
{
a = a;
b = b;
}
void display() {
System.out.println(a);
System.out.println(b);
}
public static void main(String[] args)
{
Calculator  cal = new Student();
cal.add(10, 20);
cal.display();
}
}
```

Output:
0
0

```java
public class Calculator
{
int a, b;
void add(int a, int b)
{
this.a = a;
this.b = b;
}
void display() {
System.out.println(a);
System.out.println(b);
}
public static void main(String[] args)
{
Calculator cal = new Student();
cal.add(10, 20);
cal.display();
}
}
```

Output:
10
20

Call by Value & Call by Reference

```java
class ByVal
{
int x;
public void addition(int a)
{
x = a + 5;
}
}

public class CallByVal
{
public static void main(String args[])
{
ByVal b=new ByVal();
int x=10;
b.addition(x);
System.out.println(x);    // 10
}
}
```
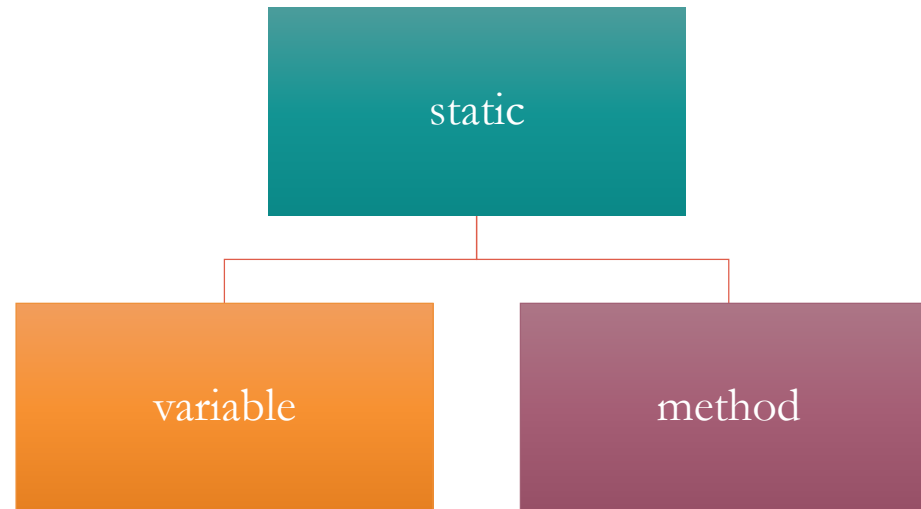
```java
public class ByRef
{
int x;
public void addition(ByRef a)
{
x = a.x + 5;
}
}

public class CallByRef
{
public static void main(String args[])
{
ByRef b=new ByRef();
b.x=10;
b.addition(b);
System.out.println(b.x);    //15
}
}
```
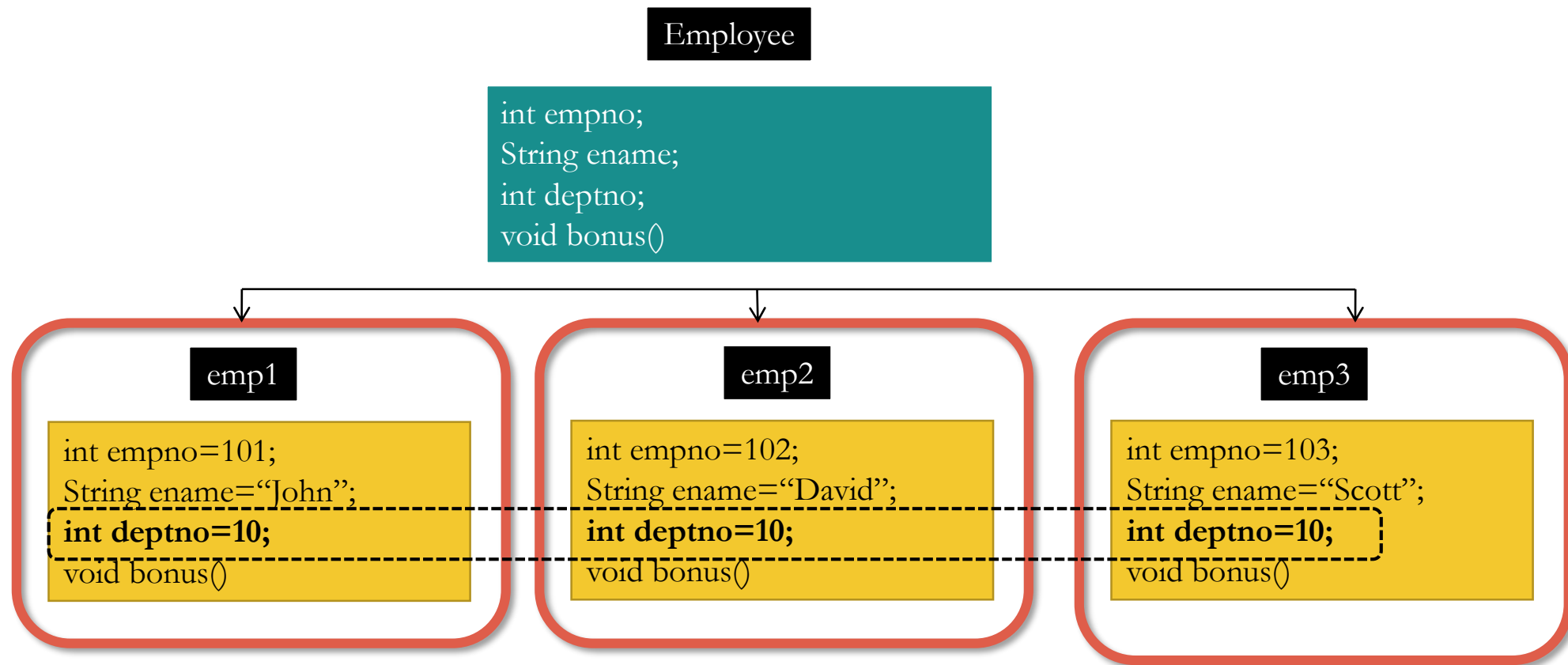
static

- The **static keyword** in java is used for memory management mainly.

- We can apply java static keyword with variables and methods

- The static can be:
  - variable (also known as class variable)
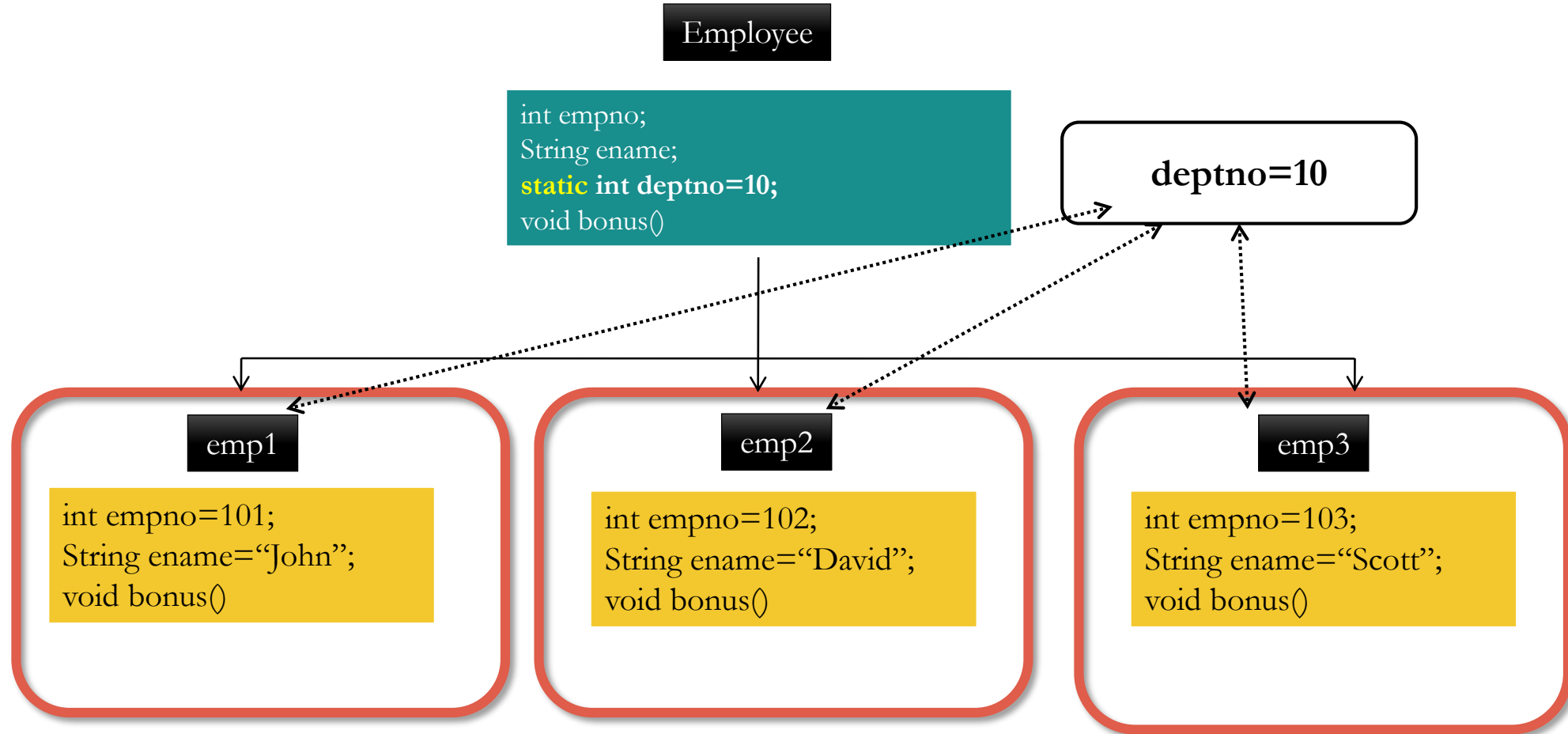 - method (also known as class method)

static

Employee

```
int empno;
String ename;
int deptno;
void bonus()
```

emp1

```
int empno=101;
String ename="John";
int deptno=10;
void bonus()
```

emp2

```
int empno=102;
String ename="David";
int deptno=10;
void bonus()
```

emp3

```
int empno=103;
String ename="Scott";
int deptno=10;
void bonus()
```

- Every object occupy certain amount of space in memory.

- Objects are independent. So if modify a variable value in obj1 , that will not reflect other objects variable values.

static

static variables and methods

	static		Non-static	
	Variables	Methods	Variables	Methods
static methods	✔	✔	✔	✔
Non-static methods	✔	✔	✔	✔

✔ **Direct Access**

✔ **Through Object**

System.out.println()

class Test
{
static String s="Welcome";
}

Test.s.length();

class System
{
static PrintStream out;
}

System.out.println();