Classes and Objects

Syntax

```
class classname
{   type instancevariable1;
    type instancevariable2;    ….
    type instancevariablen;
    type method1(parameter list)
    {   …..    }
     type method 2(parameter list)
    {   …..    }
……
 type method n(parameter list)
    {   …..    }
}
Class testclassname
{
Public static void main(String[] args){…….}
}
```

Example

```
public class student {
int id;
String name;
void print()
{
System.out.println(id+name);
}
}
 class test
{
public static void main(String[]
args)
{student s=new student();
s.print();  //prints 0 null
}  }
```

Example

```
public class student
{
int id;
String name;
student()
{
System.out.println("Default cons");
}
student(int a)
{
System.out.println("one arg cons");
id=a;
name="sri";
}
student(int a,String b)
{
System.out.println("two arg cons");
id=a;
name=b;
}
void print()
{System.out.println(id+" " +name);
}}
class test
{
public static void main(String[] args)
{student s=new student();
s.print();
student s1=new student(1);
s1.print();
student s2=new student(1,"sri");
s2.print();
} }
```

Output:

```
Default cons
 0 null
 one arg cons
 1 sri
 two arg cons
 1 sri
```

This keyword

- automatically passed as implicit reference to the current Object whose Method or constructor is being invoked.
- Whenever we access any member though this pointer only we are accessing (But its implicit)

- Eg.

  id=a; is equivalent to

  this->id=a;

```
public class student
{
int id;
String name;
student()
{
System.out.println("Default cons");
}
student(int  id)
{
System.out.println("one arg cons");
id=id;
name="sri";
}
student(int id, String name)
{
System.out.println("two arg cons");
id=id;
name=name;
}
void print()
{
System.out.println(id+" " +name);
}}
 class test
{
public static void main(String[] args)
{
student s=new student();
s.print();
student s1=new student(1);
s1.print();
student s2=new student(1,"sri");
s2.print();
}
}
```

Output:

Default cons
0 null
one arg cons
0 sri
two arg cons
0 null

```
public class student {
int id;
String name;
student()
{
System.out.println("Default cons");
}
student(int id)
{
System.out.println("one arg cons");
this.id =id;
this.name="sri";
}
student(int id,String name)
{
System.out.println("two arg cons");
this.id=id;
this.name=name;
}
void print()
{
System.out.println(id+" " +name);
}
}
 class test
{
public static void main(String[] args)
{
student s=new student();
s.print();
student s1=new student(1);
s1.print();
student s2=new student(1,"sri");
s2.print();
}
}
```

Output:

```
Default cons
 0 null
 one arg cons
 1 sri
 two arg cons
 1 sri
```