Super and This

Thursday, August 28, 2025 1:37 PM

◇ Definition of super in Java

The **super** keyword in Java is a **reference variable** used to refer to the **immediate parent class object**.
It is mainly used for **inheritance** to access parent properties, methods, and constructors.

◇ Uses of super

1. Access Parent Class Variables

When the child class has a variable with the same name as the parent, super is used to avoid ambiguity.

```
public void setP(Parent obj){  1 usage  new *
    super.name = obj.name ;
    super.bodyType = obj.bodyType ;
}
```

2. Call Parent Class Methods

If the child overrides a method, super can be used to call the parent's version.

```
public void showP(){  1 usage  new *
    super.show();
}
```

3. Call Parent Class Constructor

super() is used inside a child constructor to call the parent constructor.
👉 It must be the **first statement** in the child constructor.

```
// constructors
public Child(String name , String bodyType){  1 usage  new *
    super();
    this.name = name ;
    this.bodyType = bodyType ;
}
```

◇ Definition of this in Java

The **this** keyword in Java is a **reference variable** that refers to the **current object of the class**.
It is mainly used to **differentiate instance variables from local variables**, to **call methods/constructors of the same class**, and to **pass the current object as a parameter**.

◇ Uses of this

1. Access Current Class Variables

When local variables (like constructor or method parameters) have the same name as instance variables, this is used to avoid ambiguity.

```
public Child(String name , String bodyType){  1 usage  new *
    super();
    this.name = name ;
    this.bodyType = bodyType ;
}
```

2. Call Current Class Methods

this can be used to call another method of the same class.

```
public void setP(Parent obj){  1 usage  new *
    super.name = obj.name ;
    super.bodyType = obj.bodyType ;
    this.showP();
}
```

3. Call Current Class Constructor (Constructor Chaining)

this() is used to call another constructor in the same class.

👉 It must be the **first statement** inside the constructor.

```
public Child(String name , String bodyType){  1 usage  new *
    this() ;
    this.name = name ;
    this.bodyType = bodyType ;
}
```

Notes :

1. When the parent method is called by super keyword then the variables used in the method is of parent
2. Changing in child obj , parent variables then there is no change in other child obj parent variables
3. Only one constructor can be called explicitly through super or this
4. Super and this can call any constructor inside a constructor itself only
5. Every child constructor by default call super() , means it by default call the no-arg constructor of the parent class