Super Keyword in Java

- Super keyword in Java is a reference variable that refers to an immediate superclass object. In other words, it refers to the superclass of the current object.
- The keyword ‘super’ comes into the picture with the concept of inheritance in Java.
- Java super keyword always represents a superclass object. Whenever we create an object of subclass, an object of superclass is created implicitly, which is referred by super reference variable.

Super Keyword in Java

- ✓ The keyword “super” allows users to access members of a superclass in a subclass. In other words, if you want to call members of the super class explicitly from a subclass, then use the super keyword.

- ✓ We can apply super keyword with variables, methods, constructors of parent class. Hence, we can call immediate data members or member functions of the parent class.

- ✓ JVM always uses the super keyword to call the default constructor of super class implicitly from the subclass constructor.

Advantages of Super Keyword in Java

- ✓ This is especially useful when both the superclass and subclass have members with the same name.
- ✓ When a subclass constructor is invoked, the super() call is automatically added by the Java compiler if no explicit constructor call is made. This facilitates constructor chaining.
- ✓ When we override a method in a subclass, the super keyword is often used to call the overridden method of the superclass.
- ✓ We can use the super keyword to call specific constructors of the superclass. This is especially helpful when the superclass has multiple constructors with different parameter lists. By using super(arguments), we can choose which constructor to call.
- ✓ If a subclass and its superclass have members with the same name, using super helps in resolving ambiguity by explicitly indicating that we want to access the superclass's member.
- ✓ The super keyword helps to maintain properly inheritance hierarchy.

This Keyword in Java

- **this** keyword in Java is a reference variable that refers to the current class object.
- In other words, it holds the reference to the current class object or the same class object.
- The current class object can be referred by using this reference anywhere in the class.
- The keyword "**this**" in Java can be applied to instance variables, constructors, and methods.
- It is used only inside the member functions of the class. In other words, it can only be used within the non-static method of a class. This reference cannot be used outside the class.

The general syntax to use ‘this’ keyword in Java:

this.a; // It calls current class instance variable where a is an //instance variable.

this.msg(); // It calls current class instance method where msg() is an //instance method.

this(int a); // To call parameterized constructor of current class object.

Note:

this and **super** keyword cannot be used in a **static** method and **static initialization block**. This will give compile-time error because static members are not bound to instances.

Use of This keyword in Java

There are six usages of Java this keyword. They are as follows:

1. **this** reference can be used to refer to the current class instance variable.
2. **this** keyword is used to call the non-static method of the current class.
3. **this()** can be used to invoke the current class constructor.
4. **this** keyword can be used as a parameter in the method call.
5. The keyword "**this**" can be used as a parameter in the constructor call.
6. It can also be used to return the object of the current class from the method.

Difference between Super and This Keyword

SN	'this' Keyword	'super' Keyword
1.	"**this**" is a reference variable that contains current class objects.	"**super**" is a reference variable that contains immediate super class objects.
2.	Any member of the current class object from within an instance method or a constructor can be referred by using this keyword.	If the method overrides one of its superclass's method, the overridden method can be called through the use of super keyword.
3.	'**this**' keyword is used to call another constructor from within a constructor in the same class.	'**super**' keyword is used to call the superclass's constructor from within a constructor of the subclass.
4.	Java compiler never puts automatically **this()** keyword like **super().**	By default, the compiler automatically puts the **super()** keyword at first line inside the constructor.