Super Keyword in Java

The **super** keyword in Java is a reference variable which is used to refer immediate parent class object.

Usage of Java super Keyword

1. super can be used to refer immediate parent class instance variable.
2. super can be used to invoke immediate parent class method.
3. super() can be used to invoke immediate parent class constructor.

1) super is used to refer immediate parent class instance variable.

```
class Animal
{
String color="white";
}
class Dog extends Animal
{
String color="black";
void printColor()
{
System.out.println(color);
System.out.println(super.color);
}
}
class TestSuper1
{
public static void main(String args[])
{
Dog d=new Dog();
d.printColor();
}

}
```

2) super can be used to invoke parent class method

The super keyword can also be used to invoke parent class method. It should be used if subclass contains the same method as parent class. In other words, it is used if method is overridden.

```
class Animal
{
void eat()
{
System.out.println("eating...");
}
}
class Dog extends Animal
{
void eat()
{
System.out.println("eating bread...");
}
void bark()
{
System.out.println("barking...");
}
void work()
{
super.eat();
bark();
}
}
class TestSuper2
{
public static void main(String args[])
{
Dog d=new Dog();
d.work();
}

}
```

3) super is used to invoke parent class constructor.

The super keyword can also be used to invoke the parent class constructor. Let's see a simple example:

```java
class Animal
{
Animal()
{
System.out.println("animal is created");
}
}
class Dog extends Animal
{
Dog()
{
super();
System.out.println("dog is created");
}
}
class TestSuper3
{
public static void main(String args[])
{
Dog d=new Dog();
}

}
```

super example: real use

Let's see the real use of super keyword. Here, Emp class inherits Person class so all the properties of Person will be inherited to Emp by default. To initialize all the property, we are using parent class constructor from child class. In such way, we are reusing the parent class constructor.

```
class Person
{
int id;
String name;
Person(int id,String name)
{
this.id=id;
this.name=name;
}
}
class Emp extends Person
{
float salary;
Emp(int id,String name,float salary)
{
super(id,name);//reusing parent constructor
this.salary=salary;
}
void display()
{
System.out.println(id+" "+name+" "+salary);
}
}
class TestSuper5
{
public static void main(String[] args)
{
Emp e1=new Emp(1,"ankit",45000f);
e1.display();
}

}
```