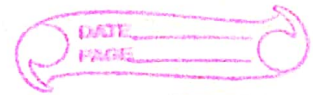


no overloading when return type
is different.



★ Method Overloading: same name different Parameter list

1) Same class

2) Parent-child class

1) Class A {

```
void max(int a, int b) {  
    if (a > b)  
        System.out.println(a);  
    else  
        System.out.println(b);
```

}

```
void max(int a, int b, int c) {
```

```
    if (a > b && a > c)  
        System.out.println(a);  
    else if (b > a && b > c)  
        System.out.println(b);  
    else if (c > a && c > b)  
        System.out.println(c);
```

```
}
```

```
}
```

```
class DemoMax {
```

```
    public static void main(String args[]) {
```

```
        A obj = new A();
```

```
        obj.max(10, 3);
```

```
        obj.max(70, 5, 1);
```

```
}
```

```
}
```

// Same class

```

2) class A {
    void max(int a, int b) {
        if (a > b)
            System.out.println(a);
        else
            System.out.println(b);
    }
}

```

```

class B extends A {
    void max(int a, int b, int c) {
        if (a > b && a > c)
            System.out.println(a);
        else if (b > a && b > c)
            System.out.println(b);
        else if (c > a && c > b)
            System.out.println(c);
    }
}

```

```

class DemoMax2 {
    public static void main(String args[]) {
        B obj = new B();
        obj.max(10, 5);
        obj.max(70, 5, 6);
    }
}

```

Parent-child
// Different class
^