Properties & functions

```
class Batman1 {
        int rating; → 5
        string name;
    }
  checkHit ( ) {
```

```
int a=10, b=10
public static int sum (int a, int b)
        return a+b;
      }
   Psvm () {
        sum();
      }
```

```
class Emp{
    String Name;    ✓
    int id;    ✓

    Emp(String name, int id){
        this.name = name;    ✓
        this.id = id;    ✓
✓   }
    void display(){

    }
    psvm(){
        Emp e1;
→   Syso(e1.Name);  → X
```

Overloding

a b

12 13

+

syso (a+b) // 25

String a= 25 , String b =30

syso (a+b) // 2530

When you want to acheive the overloading

1. Arguments are different

2. Datatypes are also different.

3. We can change the order also

float int Double

✓

✗

Over a = new overload (23, 26);

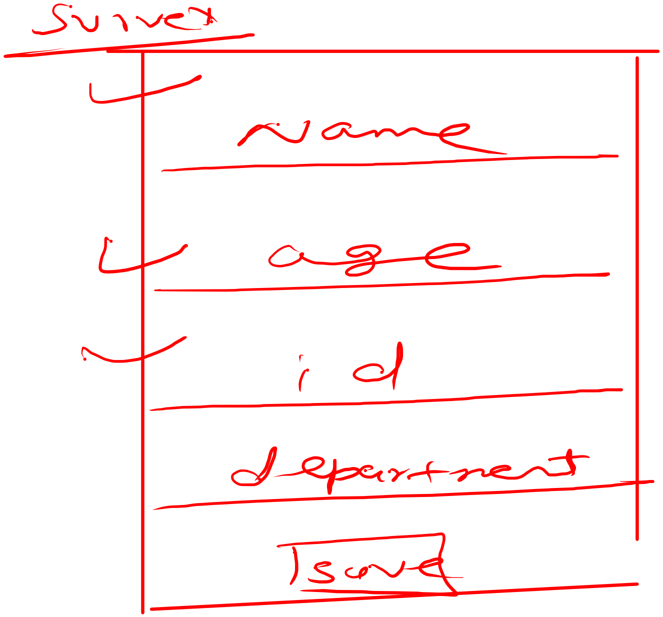
Over b = new overload (23, 26, "Ravi");

Feedback

✓ ✓

1 ✓ ✓ ✓

111 ✓ ✓ ✓



(String Name){

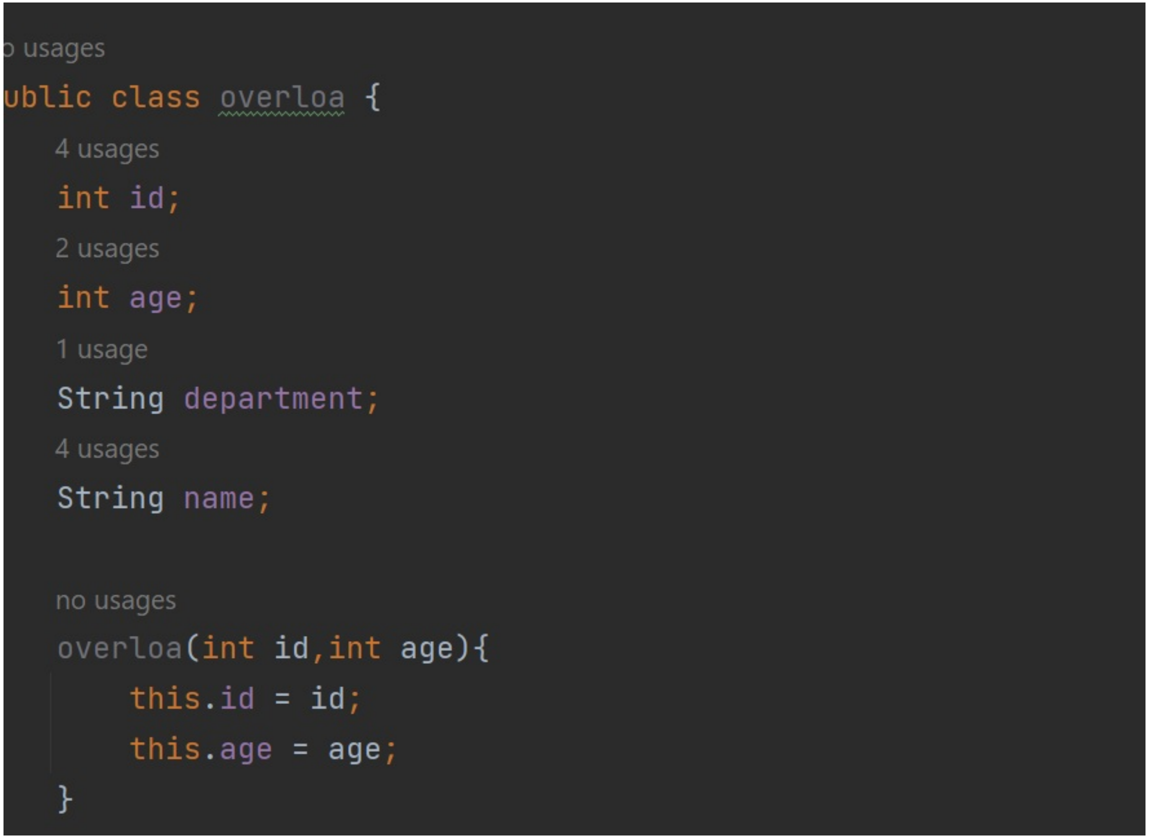
}

(String Name, int age){

}

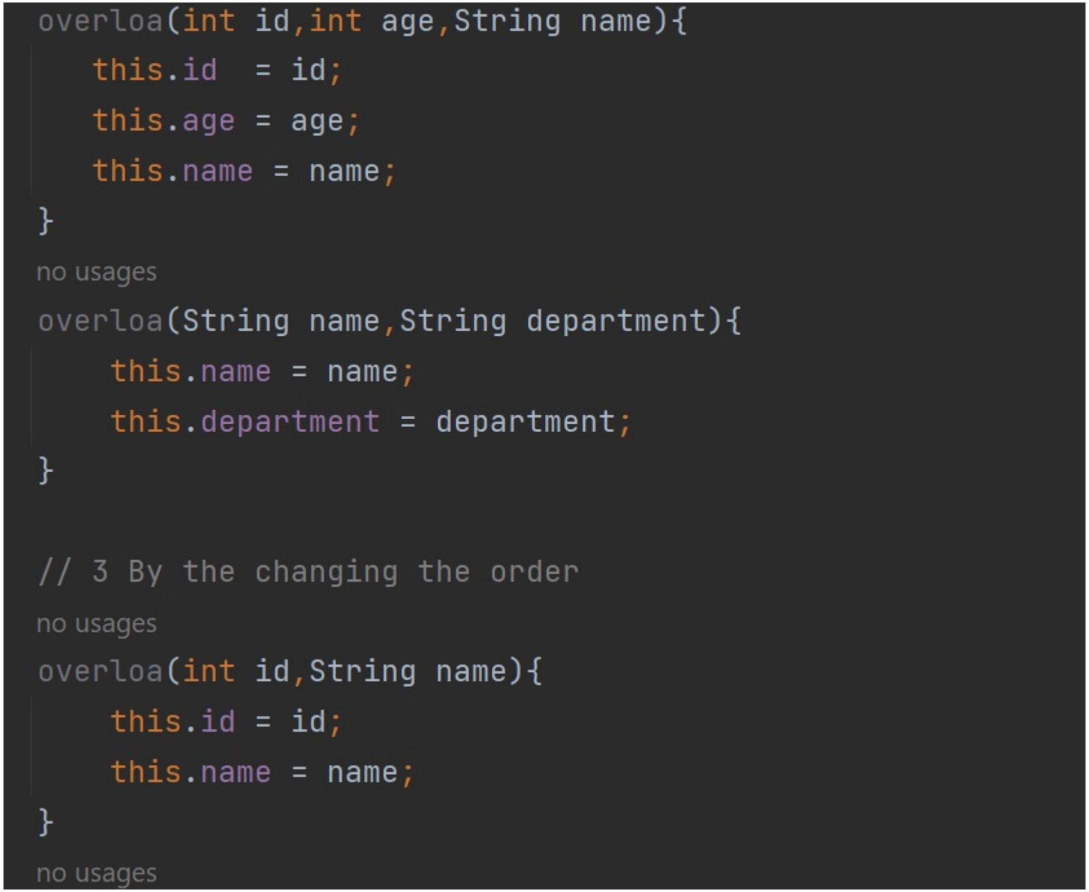
(String Name, int age, int id){

}



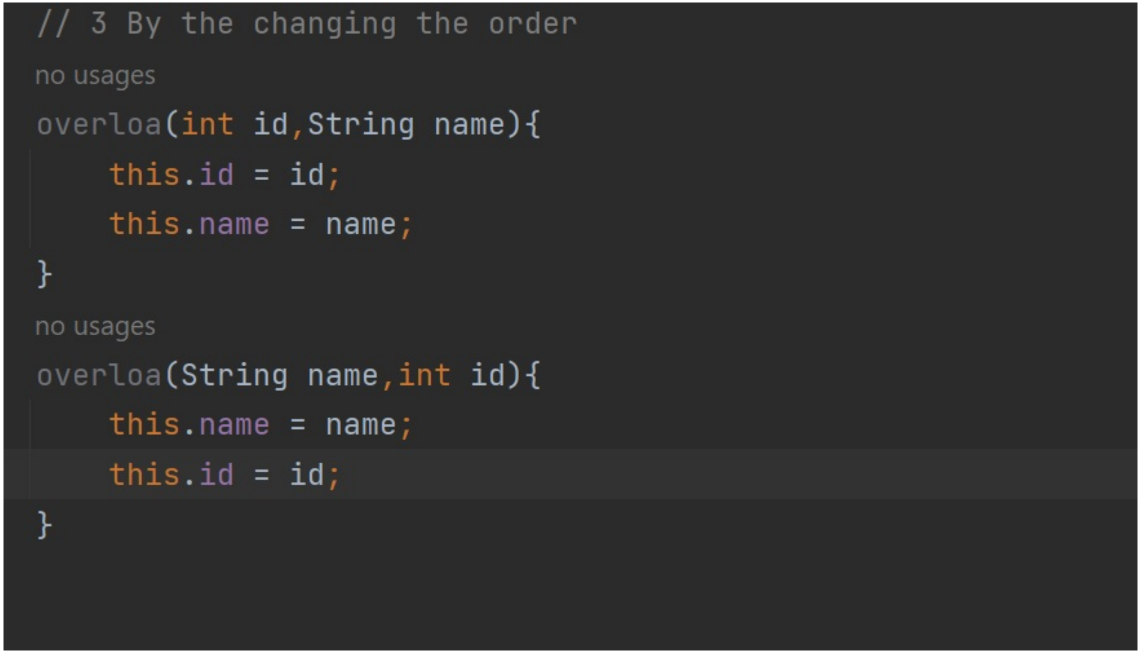
```
o usages
ublic class overloa {
    4 usages
    int id;
    2 usages
    int age;
    1 usage
    String department;
    4 usages
    String name;

    no usages
    overloa(int id,int age){
        this.id = id;
        this.age = age;
    }
```



```
overloa(int id,int age,String name){
   this.id  = id;
   this.age = age;
   this.name = name;
}
no usages
overloa(String name,String department){
    this.name = name;
    this.department = department;
}

// 3 By the changing the order
no usages
overloa(int id,String name){
    this.id = id;
    this.name = name;
}
no usages
```



```
// 3 By the changing the order
no usages
overloa(int id,String name){
    this.id = id;
    this.name = name;
}
no usages
overloa(String name,int id){
    this.name = name;
    this.id = id;
}
```