

# java

## 14강\_객체와 메모리

메모리에 생성된 객체와 레퍼런스에 대해서 살펴보며, null에 대해서도 학습합니다.

---

14-1 메모리에서 객체 생성 (동적 생성)

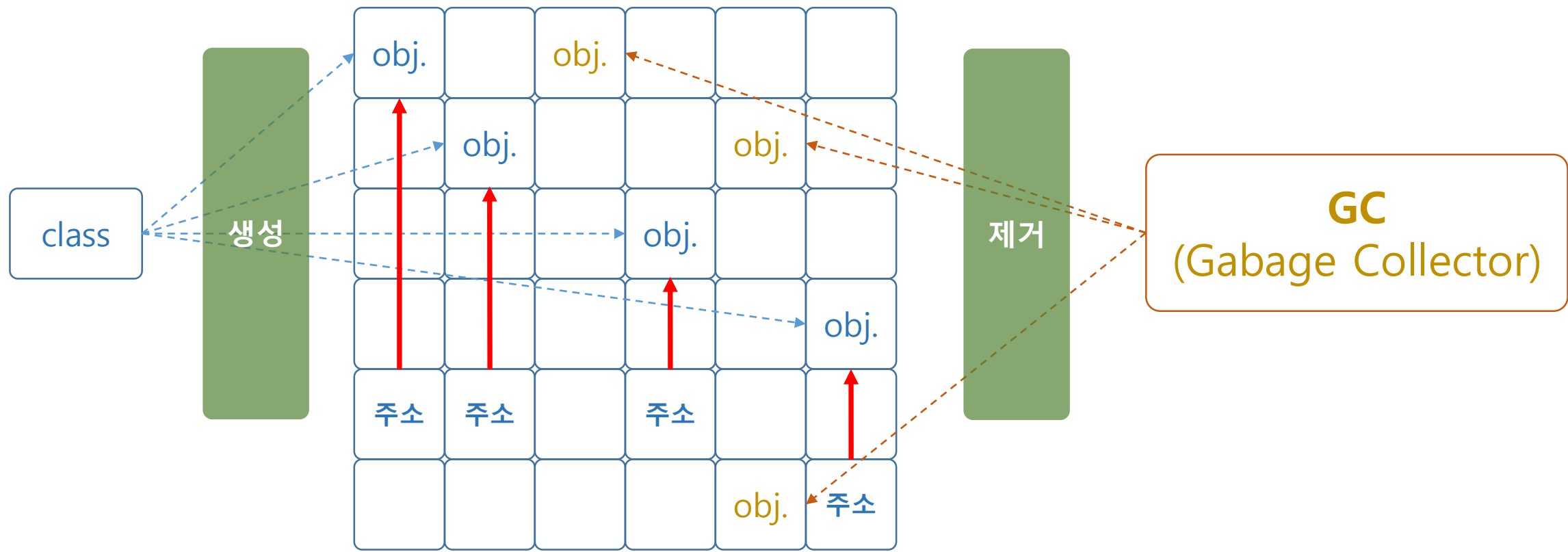
14-2 레퍼런스 (가리킨다)

14-3 자료형이 같아도 다른 객체

14-4 null과 NullPointerException

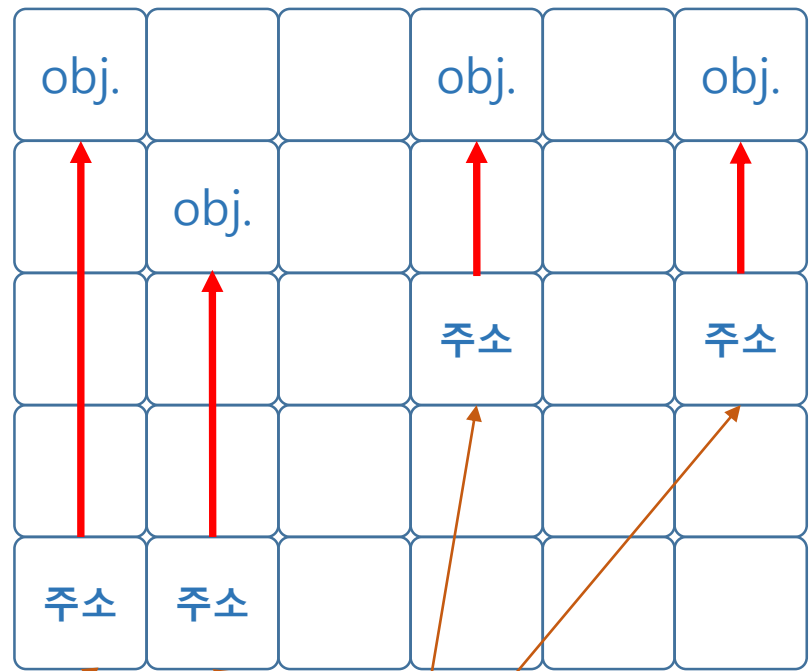
14-1 : 메모리에서 객체 생성(동적 생성)

객체는 메모리에서 동적으로 생성되며, 객체가 더 이상 필요 없게 되면 GC(Gabage Collector)에 의해서 제거된다.



# 14-2 : 레퍼런스(가리킨다)

생성한 객체의 주소를 변수에 저장하는 것을 레퍼런스라고 한다.



```
ObjectClass obj1 = new ObjectClass();
ObjectClass obj2 = new ObjectClass();
ObjectClass obj3 = new ObjectClass();
```

```
System.out.println("obj1 ----> " + obj1);
System.out.println("obj2 ----> " + obj2);
System.out.println("obj3 ----> " + obj3);
```

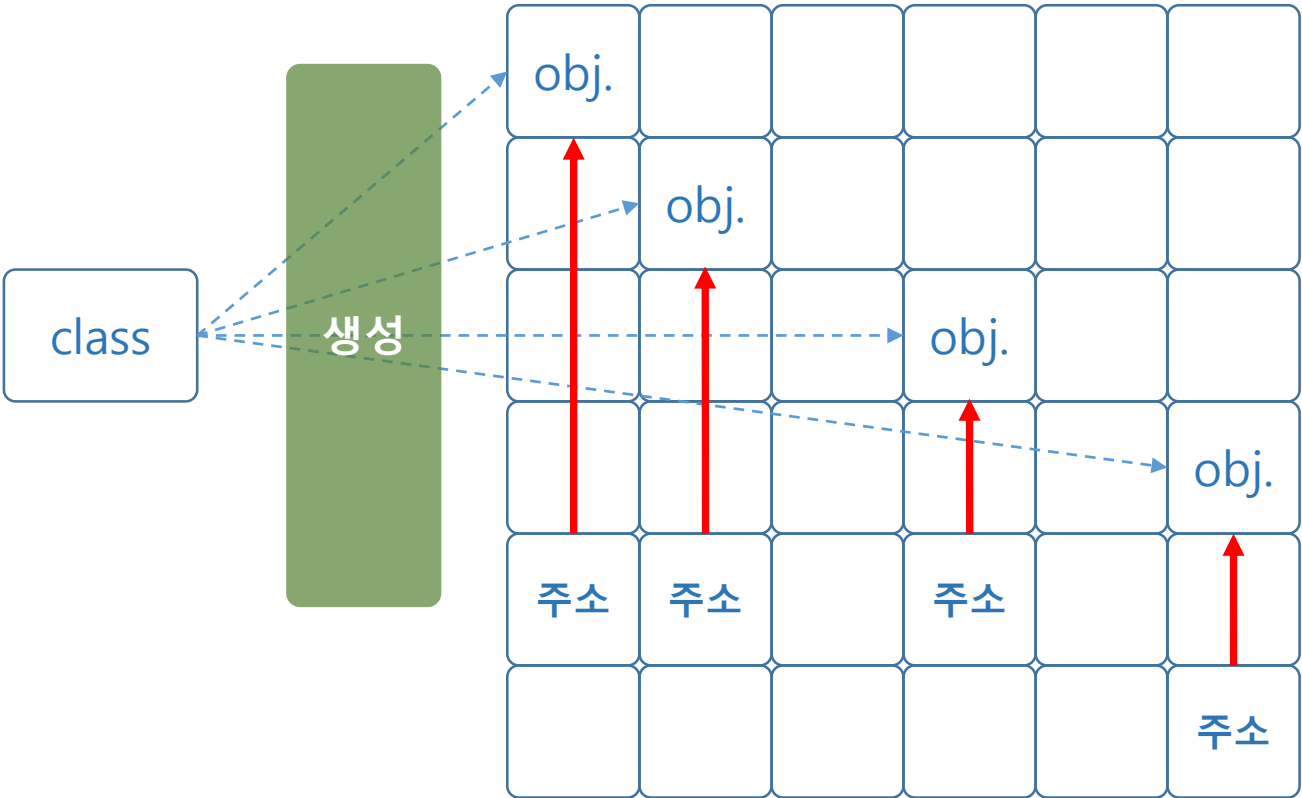


```
obj1 ----> lec14Pjt001.ObjectClass@7852e922
obj2 ----> lec14Pjt001.ObjectClass@4e25154f
obj3 ----> lec14Pjt001.ObjectClass@70dea4e
```

레퍼런스

# 14-3 : 자료형이 같아도 다른 객체

자료형이 같아도 다른 공간에 존재하는 객체는 다른 객체이다.



```
ObjectClass obj1 = new ObjectClass();
ObjectClass obj2 = new ObjectClass();
ObjectClass obj3 = new ObjectClass();
```

```
if(obj1 == obj2) {
    System.out.println("obj1 == obj2");
} else {
    System.out.println("obj1 != obj2");
}
```

```
if(obj2 == obj3) {
    System.out.println("obj2 == obj3");
} else {
    System.out.println("obj2 != obj3");
}
```

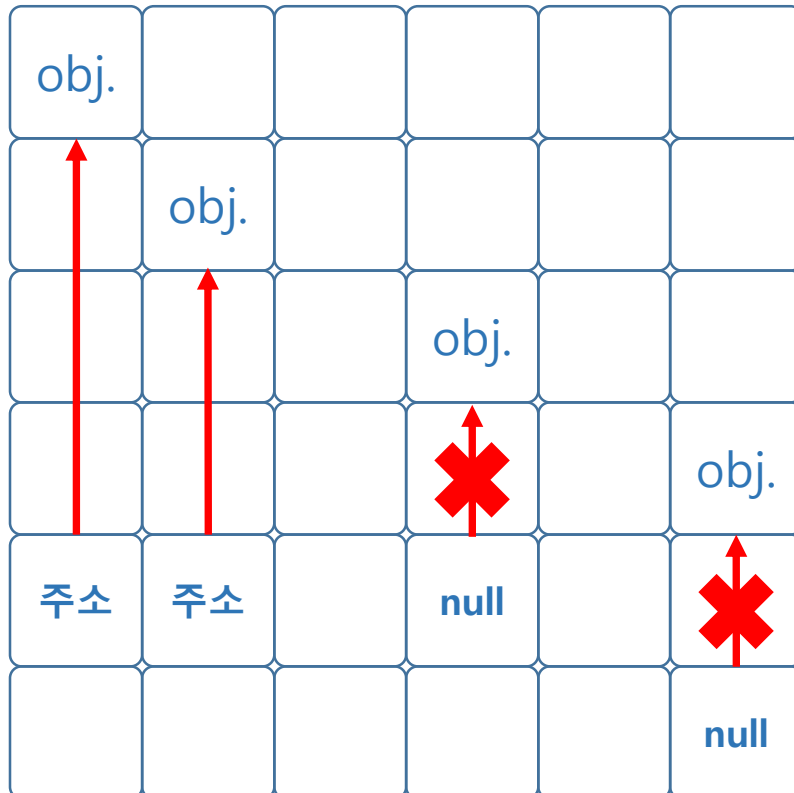
```
if(obj1 == obj3) {
    System.out.println("obj1 == obj3");
} else {
    System.out.println("obj1 != obj3");
}
```



obj1 != obj2  
obj2 != obj3  
obj1 != obj3

## 14-4 : null과 NullPointerException

레퍼런스에 null 저장되면 객체의 연결이 끊기며, 더 이상 객체를 이용할 수 없다.



```
System.out.println("obj1 ----> " + obj1);
obj1.getInfo();
```

```
obj1 = null;
System.out.println("obj1 ----> " + obj1);
obj1.getInfo();
```



```
obj1 ----> lec14Pjt001.ObjectClass@7852e922
-- getInfo method --
obj1 ----> null
Exception in thread "main" java.lang.NullPointerException
    at lec14Pjt001.MainClass.main(MainClass.java:44)
```