

retain: 해당오브젝트의 reference Count를 1증가시킨다.

release : 해당오브젝트의 reference Count값을 1감소시킨다.

- 예제

```
- (NSString *)stringWithInteger:(NSInteger)number {

// object를 retain합니다.
NSString *returnValue = [[NSString alloc] initWithInteger:number];

//[returnValue release]; // object를 release합니다.
return returnValue;
```

이렇게 alloc 하면 retain되어 referenceCount가 1이 증가한다.

그 다음 [returnValue release]로 -1감소시켜 memory leak가

않지만 release가 된 오브젝트를 리턴하면 그 오브젝트를 받아 접근한다면 에러가 발생합니다.(메소드 내에서 이
미 리턴 전release했기 때문에)

그래서 저 [returnValue release];를 지우고

호출받고 호출한 곳에서 release하는 방식으로 할 수 있다.

```
-(void)returnValueWithString{
//호출해서 오브젝트를 쓴다 reference Count=1;
    NSString *returnValue =[self stringWithInteger:10];
    NSLog(@"%@값입니다."return);
//다쓰고 release 하면 reference Count =0
    [returnValue realese];

}
```

이것은 너무 불편하니 나온게 autorelease이다.

```
- (NSString *)stringWithInteger:(NSInteger)number {

// object를 retain합니다.
NSString *returnValue = [[NSString alloc] initWithInteger:number];

//[returnValue autorelease]; // object를 release합니다.
return returnValue;
```