12.30 金 _ 복습

딥카피

- 동일한 객체를 복사한다
- '`실제 값`'을 새로운 메모리 공간에 복사

쉘로우카피

- 객체를 가르키는 참조 변수를 복사
- **얕은 복사**의 경우 주소 값을 복사하기 때문에, **참조하고 있는 실제값은 같습니다.**
- '`주소 값`'을 복사
- **복사한 객체가 변경된다면 기존의 객체도 변경이 되는 것**

[Java] - 깊은 복사(Deep Copy) vs 얕은 복사(Shallow Copy)

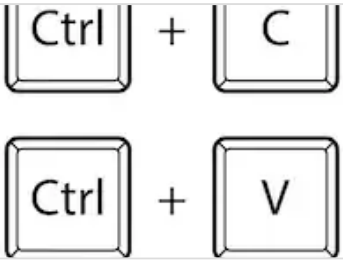
📎 Java 깊은 복사(Deep Copy)와 얕은 복사(Shallow Copy) 안녕하세요! 이번에 정리할 내용은 자바에서의 깊은 복사와 얕은 복사 입니다. 깊은 복사와 얕은 복사라는 개념은 평소에 접한적이 꽤 있었습니다. 하

https://zzang9ha.tistory.com/372

얕은복사 VS 깊은복사

자바로 개발을 하다보면 객체를 복사할 일이 있다. 이럴 때 나오는 개념이 얕은 복사(Shallow Copy)와 깊은 복사(Deep Copy) 개념인데, 두 개념의 차이를 간단하게 말하면 얕은 복사는 객체의 참조값(주소

https://rok93.tistory.com/entry/%EC%96%95%EC%9D%80%EB%B3%B5%EC%82%AC-VS-%EA%B9%8A%EC%9D%80%EB%B3%B5%EC%82%AC



책 읽고 자료를 정리 (머리속에 있는걸로)

1. 초기 스케치

최종적으로는 책을 안 보고 이해한 다음에 이재일의 생각으로 노트정리 해보기

오래되면 까먹으니까 다시 처음부터 반복해서 보기

오버로딩 적용 → 반환형은 예측이 불가능하다

```
package test;


class Bar {

  void prt(String argName) {
    System.out.println("Name : " + argName);
    }

//  int prt(String argName) {
//    System.out.println("Name : " + argName);
//    }
  // 반환형은 일치가 되어야한다. => 어떤 반환형이  올지 모르기 때문이다!!

  void prt(String argName , int argId) {
    System.out.println("Name : " + argName + ", ID: " +argId);
    }

}

public class test2 {
  public static void main(String[] args) {

    Bar b1 =new Bar();
    b1.prt("LSY");
    b1.prt("LSY,12");
  }

}
```

클래스

기 내용들이 원래 클래스들의 구성 요소

- 생성자
- 소멸자

- 멤버변수
- 멤버 메소드

Initialization Block (초기화 블록)

class 의 구성요소

3 번 정도 보기 6~8

Q. 1 초기화블록이뭐냐

Q. 2 스테틱 OR 인스턴스 초기화 블록이 뭐냐?

Q. 3 왜 쓰는지 ?

```
class Bar {
  {
    System.out.println("인스턴스 초기화 블록 호출");
  }
  static {
    System.out.println("스태틱 초기화 블록 호출");
  }

  Bar() {
    System.out.println("생성자 호출");
  }
}

public class MyProject {
  public static void main(string args[]) {
    Bar b1 = new Bar();
  }
}
```

회사에서 1개 프로젝트 끝날 때마다 6개월마다, 포트폴리오 만들면서 이력서 작성하기!!

계속해서 같은거 만 개발해선 안 된다 .