

[Workshop #2]

❖ Workshop #2-1

다음 실행코드는 은행의 계좌클래스(Account)에서 객체를 생성하고 돈을 입금 및 출금한 내용이다. 그리고 각각의 메서드를 사용해서 필요한 작업을 했다.

각 메서드들을 보고 원본 클래스를 정의하여보라(반드시 접근제한자를 포함해야 한다. interest를 제외한 모든 멤버변수는 Encapsulation을 적용하여라.)

```
public class AccountTest {
    public static void main(String[] args){
        Account customer1=new Account("111-222-33333333","최은희",20000);
        Account customer2=new Account("555-666-77777777","남매월",100000);
        System.out.println("기본 적립금");
        printAccount(customer1);
        printAccount(customer2);
        System.out.println("한번의 입출금");
        customer1.deposit(1000000);
        customer2.withdraw(30000);
        printAccount(customer1);
        printAccount(customer2);
        System.out.println("이자율의 계산");
        Account.interest=0.05;
        customer1.addInterest(); //저축된 금액=원금+ 원금*이자율
        customer2.addInterest();//저축된 금액=원금+ 원금*이자율
        printAccount(customer1);
        printAccount(customer2);
    }
    static void printAccount(Account customer){
        System.out.println("계좌번호:"+customer.getAccountNo());
        System.out.println("예금주이름:"+customer.getAccountName());
        System.out.println("잔액:"+customer.getBalance());
        System.out.println();
    }
}
```

[Workshop #2]

❖ Workshop #2-2

SubscriberTest실행 클래스의 형태를 보고 생성자를 포함하는 클래스를 정의해보라.

```
public class SubscriberInfoTest {
    public static void main(String[] args)
    {
        SubscriberInfo obj1, obj2,obj3;
        obj1=new SubscriberInfo("박유신","1017db","daewi");
        obj2=new SubscriberInfo("백곰","baekom","white",
"200-000-0000","north pole");
        obj3=new SubscriberInfo();
        printSubscriberInfo(obj1);
        printSubscriberInfo(obj2);
        printSubscriberInfo(obj3);
        obj1.setAddress("교직원1숙사");
        obj1.setPhoneNo("133-1234-5678");
        printSubscriberInfo(obj1);
    }
    static void printSubscriberInfo(SubscriberInfo obj){
        System.out.println("이름:"+obj.name);
        System.out.println("아이디:"+obj.id);
        System.out.println("패스워드:"+obj.password);
        System.out.println("전화번호:"+obj.phoneNo);
        System.out.println("주소:"+obj.address);
        System.out.println();
    }
}
```

[Workshop #2]

❖ Workshop #2-3

다음은 인터넷 게시판의 게시글을 표현하는 클래스이다.

```
public class BBSItem {
    int seqNo;                //일련번호 필드
    String writer;            //작성자 필드
    String writtenDate;       //작성일자 필드
    String title;             //제목필드
    String content;           //내용필드
    BBSItem(int seqNo, String writer, String writtenDate,
              String title, String content){//생성자
        this.seqNo=seqNo;
        this.writer=writer;
        this.writtenDate=writtenDate;
        this.title=title;
        this.content=content;
    }
}
```

일련번호에 해당하는 seqNo 필드의 값을 생성자 파라미터로 받는 것이 아니라, 새로운 객체가 생성될 때마다 자동으로 붙여지게 하려고 한다. 처음으로 생성되는 BBSItem 객체에는 1, 두번째로 생성되는 BBSItem 객체에는 2, 이런 식으로 일련번호가 붙여지도록 이 클래스를 수정하여라.