

자바 입문 과제_3

박은종

static

학생 인스턴스가 새로 생성될때마다 학생에게 새로운 아이디를 순차적으로 부여하도록 구현하세요

```
public class StudentTest {  
  
    public static void main(String[] args) {  
  
        Student studentLee = new Student();  
        studentLee.setStudentName("이순신");  
        System.out.println(Student.getSerialNum()); //serialNum 값 가져오기 위해 get 메서드 호출  
                                                // 클래스 이름으로 직접 호출  
        System.out.println(studentLee.studentName + " 학번:" + studentLee.studentID);  
  
        Student studentSon = new Student();  
        studentSon.setStudentName("손흥민");  
        System.out.println(Student.getSerialNum());  
        System.out.println(studentSon.studentName + " 학번:" + studentSon.studentID);  
  
    }  
}  
  
// 1001  
// 이순신 학번:1001  
// 1002  
// 손흥민 학번:1002  
# static변수 #static메서드 #클래스변수 #클래스메서드 #static변수는 언제 생성되는가?
```


Singleton Pattern

- ◆ 다음 코드가 수행되도록 Company를 구현하세요

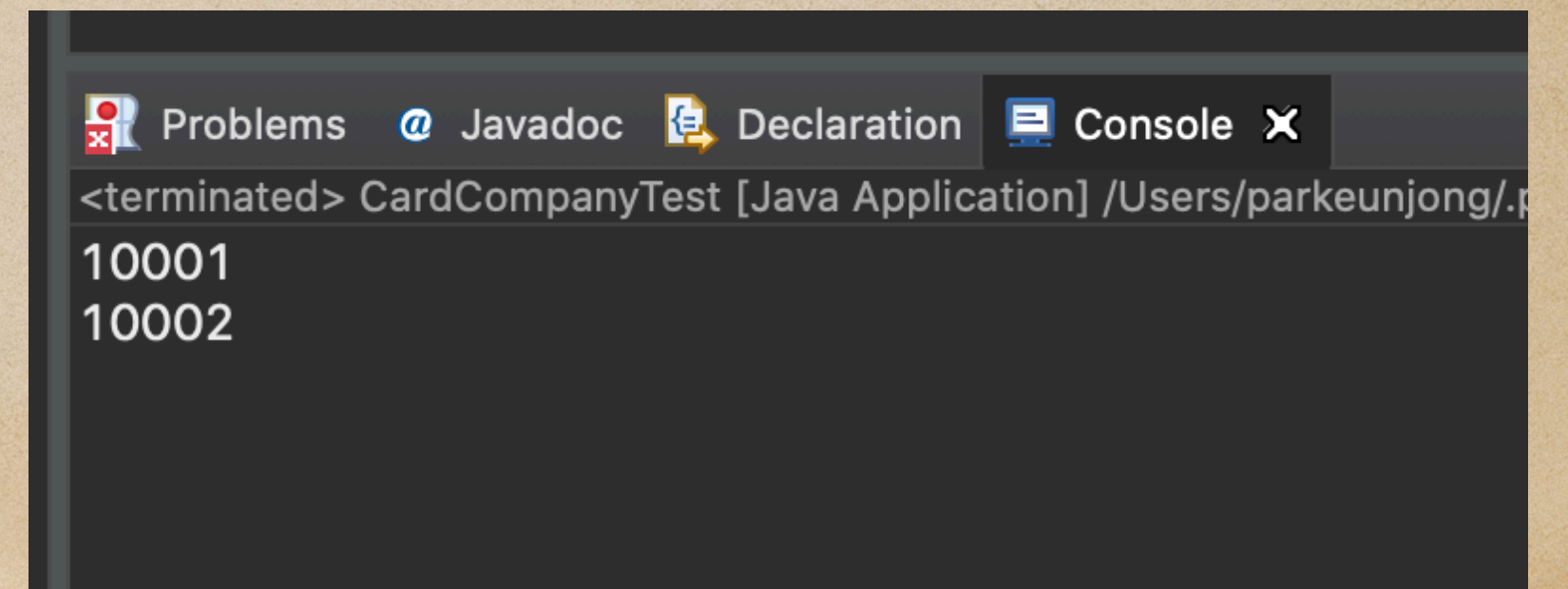
```
public class CompanyTest {  
  
    public static void main(String[] args) {  
  
        Company myCompany1 = Company.getInstance();  
        Company myCompany2 = Company.getInstance();  
  
        Company company = new Company() // 오류  
        System.out.println( myCompany1 == myCompany2 );  
    }  
}
```

#singleton pattern #객체의 유일성

카드 발급하기

- ◆ 다음 코드가 실행되도록 클래스를 완성하세요

```
public class CardCompanyTest {  
  
    public static void main(String[] args) {  
  
        CardCompany company = CardCompany.getInstance();  
  
        Card myCard = company.createCard(); //메서드에서 Car 생성  
        Card yourCard = company.createCard();  
  
        System.out.println(myCard.getCardNumber()); //10001 출력  
        System.out.println(yourCard.getCardNumber()); //10002 출력  
  
    }  
}
```



The screenshot shows an IDE's console window with tabs for Problems, Javadoc, Declaration, and Console. The Console tab is active, displaying the output of the Java application. The output consists of two lines: "10001" and "10002", which correspond to the card numbers printed in the code. The window title bar indicates the application is "CardCompanyTest [Java Application]" and the user is "parkeunjong".

```
<terminated> CardCompanyTest [Java Application] /Users/parkeunjong/.p  
10001  
10002
```


변수의 종류와 그 유효성

지역 변수, 인스턴스 변수, static 변수의 scope와 life cycle, 생성되는 메모리, 언제 소멸되는지... 등등에 대해 알아봅시다.

객체 배열

다음과 같이 배열을 복사한 경우 기존 배열과 복사한 배열의 내용이 같이 변하게됩니다.
이유를 설명하고, 각각 다른 값을 가질 수 있도록 프로그램을 구현하세요

```
public class ObjectCopy2 {  
  
    public static void main(String[] args) {  
        Book[] bookArray1 = new Book[3];  
        Book[] bookArray2 = new Book[3];  
  
        bookArray1[0] = new Book("태백산맥", "조정래");  
        bookArray1[1] = new Book("데미안", "헤르만 헤세");  
        bookArray1[2] = new Book("어떻게 살 것인가", "유시민");  
        System.arraycopy(bookArray1, 0, bookArray2, 0, 3);  
  
        for(int i=0; i<bookArray2.length; i++){  
            bookArray2[i].showBookInfo();  
        }  
  
        bookArray1[0].setBookName("나목");  
        bookArray1[0].setAuthor("박완서");  
  
        System.out.println("=== bookArray1 ===");  
        for(int i=0; i<bookArray1.length; i++){  
            bookArray1[i].showBookInfo();  
        }  
  
        System.out.println("=== bookArray2 ===");  
        for(int i=0; i<bookArray2.length; i++){  
            bookArray2[i].showBookInfo();  
        }  
    }  
}  
#객체배열사용하기 #얕은복사, 깊은복사
```

```
<terminated> ObjectCopy2 [Java Application] C:\Users\Adminis  
태백산맥, 조정래  
데미안, 헤르만 헤세  
어떻게 살 것인가, 유시민  
=== bookArray1 ===  
나목, 박완서  
데미안, 헤르만 헤세  
어떻게 살 것인가, 유시민  
=== bookArray2 ===  
나목, 박완서  
데미안, 헤르만 헤세  
어떻게 살 것인가, 유시민
```


ArrayList 사용하기

이름이 Lee 이고 학번이 1001번인 학생과 이름이 Kim 이고 학번이 1002번인 두 학생이 있습니다.

Lee는 국어, 수학 두 과목을 수강하였고, 국어점수가 100점, 수학점수가 50점입니다.

Kim은 국어, 수학, 영어 세 과목을 수강하였고, 국어가 70점, 수학이 85점, 영어가 100점입니다.

Student 클래스와 Subject 클래스를 생성하고 ArrayList를 활용하여 Student에 Subject를 배열 변수로 저장합니다. 두 학생의 대한 각과목의 성적과 그 총점을 다음과 같이 출력해보세요

#ArrayList 클래스 메서드 #javaDoc #enhanced for문


```

public class StudentTest {

    public static void main(String[] args) {
        Student studentLee = new Student(1001, "Lee");

        studentLee.addSubject("국어", 100);
        studentLee.addSubject("수학", 50);

        Student studentKim = new Student(1002, "Kim");

        studentKim.addSubject("국어", 70);
        studentKim.addSubject("수학", 85);
        studentKim.addSubject("영어", 100);

        studentLee.showStudentInfo();
        System.out.println("=====");
        studentKim.showStudentInfo();
    }
}

```

```

<terminated> StudentTest (2) [Java Application] C:\Users\Administ
학생 Lee의 국어 과목 성적은 100입니다.
학생 Lee의 수학 과목 성적은 50입니다.
학생 Lee의 총점은 150 입니다.
=====
학생 Kim의 국어 과목 성적은 70입니다.
학생 Kim의 수학 과목 성적은 85입니다.
학생 Kim의 영어 과목 성적은 100입니다.
학생 Kim의 총점은 255 입니다.

```