

2024년 상반기 K-디지털 트레이닝

# java.base 모듈

[KB] IT's Your Life



- 다음 조건을 만족하는 Student 클래스를 정의하세요.
  - o 멤버 필드를 초기화하는 생성자
  - o 멤버 필드에 대한 Getter/Setter 메서드
  - o equals(), hashCode(), toString() 메서드

```
package ch12.sec03.exam02;

public class Student {
   private int no;
   private String name;
}
```

## Student.java

```
package ch12.sec03.exam02;
public class Student {
 private int no;
 private String name;
 public Student(int no, String name) {
   this.no = no;
   this.name = name;
 public int getNo() { return no; }
 public String getName() { return name; }
```

## Student.java

```
@Override
public int hashCode() {
 int hashCode = no + name.hashCode();
 return hashCode;
@Override
public boolean equals(Object obj) {
 if(obj instanceof Student target) {
   if(no == target.getNo() && name.equals(target.getName())) {
     return true;
 return false;
```

#### ○ 다음과 같이 운영할 수 있도록 Member클래스를 record로 정의하세요.

```
package ch12.sec03.exam04;
public class RecordExample {
 public static void main(String[] args) {
   Member m = new Member("winter", "눈송이", 25);
   System.out.println(m.id());
   System.out.println(m.name());
   System.out.println(m.age());
   System.out.println(m.toString());
   System.out.println();
   Member m1 = new Member("winter", "눈송이", 25);
   Member m2 = new Member("winter", "눈송이", 25);
   System.out.println("m1.hashCode(): " + m1.hashCode());
   System.out.println("m2.hashCode(): " + m2.hashCode());
   System.out.println("m1.equals(m2): " + m1.equals(m2) );
```

## Member.java

```
package ch12.sec03.exam04;
public record Member(String id, String name, int age) {
}
```

- 🤍 Lombok 라이브러리를 다운로드 하세요.
- 프로젝트에 lib 폴더에 복사하세요.
- ☑ Intellij에 Lombok 플러그인을 설치하세요.
- ☑ Lombok 라이브러리를 프로젝트에 추가하세요.
- ☑ 프로젝트에 어노테이션 프로세싱(annotation processing) 기능을 활성화 하세요.

#### 다음 조건을 만족하는 Member 클래스를 정의하세요.

- o Lombok을 이용하여 정의함
- o 기본 생성자, 전체 필드를 초기화하는 생성자,
- o Getter/Setter 메서드
- o equals(), hashCode(), toString() 메서드
- o 위의 메서드들이 제대로 생성되었는지 IntelliJ의 Structure 기능으로 확인하세요.

```
package ch12.sec03.exam05;

public class Member {
  private String id;
  private String name;
  private int age;
}
```

## Member.java

```
package ch12.sec03.exam05;
import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;
@Data
@NoArgsConstructor
@AllArgsConstructor
public class Member {
 private String id;
 private String name;
 private int age;
```

○ 1부터 1000000까지의 합을 구하는 프로그램을 작성하고, 계산결과와 계산에 소요된 시간을 nano 초 단위로 출력하세요.

```
package ch12.sec04;

public class MeasureRunTimeExample {
   public static void main(String[] args) {
   }
}
```

## MeasureRunTimeExample.java

```
package ch12.sec04;
public class MeasureRunTimeExample {
 public static void main(String[] args) {
   long time1 = System.nanoTime();
   int sum = 0;
   for(int i=1; i<=1000000; i++) {
    sum += i;
   long time2 = System.nanoTime();
   System.out.println("1~1000000까지의 합: " + sum);
   System.out.println("계산에 " + (time2-time1) + " 나노초가 소요되었습니다.");
```

```
1~1000000까지의 합: 1784293664
계산에 4161100 나노초가 소요되었습니다.
```

- ☑ 다음 조건을 만족하는 프로그램을 작성하세요.
  - ㅇ 운영체제 명 출력
  - ㅇ 로그인 사용자 명과 홈 디렉토리 출력
  - o 시스템의 속성 정보 전체 출력

```
package ch12.sec04;
import java.util.Properties;
import java.util.Set;

public class GetPropertyExample {
   public static void main(String[] args) {
   }
}
```

## **☑** GetPropertyExample.java

```
package ch12.sec04;
import java.util.Properties;
import java.util.Set;
public class GetPropertyExample {
 public static void main(String[] args) {
   //운영체제와 사용자 정보 출력
   String osName = System.getProperty("os.name");
   String userName = System.getProperty("user.name");
   String userHome = System.getProperty("user.home");
   System.out.println(osName);
   System.out.println(userName);
   System.out.println(userHome);
                                            Windows 10
                                            C:₩Users₩a
```

## **☑** GetPropertyExample.java

```
//전체 키와 값을 출력
System.out.println("----");
System.out.println(" key: value");
System.out.println("----");
Properties props = System.getProperties();
Set keys = props.keySet();
for(Object objKey : keys) {
 String key = (String) objKey;
 String value = System.getProperty(key);
 System.out.printf("%-40s: %s\n", key, value);
```

♡ 다음 프로그램의 실행결과를 적어보고, 실제 결과와 비교하세요.

**ABCEF** 

- ☑ 다음 조건을 만족하도록 프로그램을 수정하세요.
  - o split 메서드를 이용하여 "홍길동&이수홍,박연수"에서 이름 문자열만 추출하여 출력
  - o String Tokenizer를 이용하여 "홍길동/이수홍/박연수" 에서 이름 문자열만 추출하여 출력

```
package ch12.sec05;
import java.util.StringTokenizer;
public class StringTokenizerExample {
 public static void main(String[] args) {
   String data1 = "홍길동&이수홍,박연수";
   String data2 = "홍길동/이수홍/박연수";
```

## StringTokenizerExample.java

```
package ch12.sec05;
import java.util.StringTokenizer;
public class StringTokenizerExample {
 public static void main(String[] args) {
   String data1 = "홍길동&이수홍,박연수";
   String[] arr = data1.split("&\,");
   for(String token : arr) {
    System.out.println(token);
   System.out.println();
   String data2 = "홍길동/이수홍/박연수";
   StringTokenizer st = new StringTokenizer(data2, "/");
                                                         홍길동
   while (st.hasMoreTokens()) {
                                                         이수홍
    String token = st.nextToken();
                                                         박연수
    System.out.println(token);
                                                         홍길동
                                                         이수홍
                                                         박연수
```

◎ 현재 날짜 데이터를 "2024.06.15 15:22:20"와 같은 포맷으로 출력하세요.

```
package ch12.sec08;

public class DateExample {
  public static void main(String[] args) {
  }
}
```

## **☑** DateExample.java

```
package ch12.sec08;
import java.text.*;
import java.util.*;
public class DateExample {
 public static void main(String[] args) {
   Date now = new Date();
   String strNow1 = now.toString();
   System.out.println(strNow1);
   SimpleDateFormat sdf = new SimpleDateFormat("yyyy.MM.dd HH:mm:ss");
   String strNow2 = sdf.format(now);
   System.out.println(strNow2);
```

```
Thu Jan 18 12:22:20 KST 2024
2024.01.18 12:22:20
```

♥ Calendar 클래스를 이용하여 현재 날짜, 시간기준으로 다음과 같이 출력하세요.

```
package ch12.sec08;
import java.util.*;
public class CalendarExample {
   public static void main(String[] args) {
   }
}
```

```
2024년 6월 18일
화요일 오후
0시 25분 0초
```

## **☑** CalendarExample.java

```
package ch12.sec08;
import java.util.*;
public class CalendarExample {
 public static void main(String[] args) {
   Calendar now = Calendar.getInstance();
   int year = now.get(Calendar.YEAR);
   int month = now.get(Calendar.MONTH) + 1;
   int day = now.get(Calendar.DAY OF MONTH);
   int week = now.get(Calendar.DAY_OF_WEEK);
   String strWeek = null;
   switch(week) {
     case Calendar.MONDAY: strWeek = "월"; break;
     case Calendar.TUESDAY: strWeek = "화"; break;
     case Calendar.WEDNESDAY: strWeek = "수"; break;
     case Calendar.THURSDAY: strWeek = "목"; break;
     case Calendar.FRIDAY: strWeek = "금"; break;
     case Calendar.SATURDAY: strWeek = "토"; break;
                             strWeek = "일";
     default:
```

## Calendar Example. java

```
int amPm = now.get(Calendar.AM_PM);
String strAmPm = null;
if(amPm == Calendar.AM) {
 strAmPm = "오전";
} else {
 strAmPm = "오후";
int hour = now.get(Calendar.HOUR);
int minute = now.get(Calendar.MINUTE);
int second = now.get(Calendar.SECOND);
System.out.print(year + "년 ");
System.out.print(month + "월 ");
System.out.println(day + "일 ");
                                             2024년 1월 18일
System.out.print(strWeek + "요일 ");
                                             목요일 오후
System.out.println(strAmPm + " ");
                                             0시 25분 0초
System.out.print(hour + "시 ");
System.out.print(minute + "분 ");
System.out.println(second + "초 ");
```