Java의 정석定石 3판 - 연습문제 풀이

85

# Chapter

날짜와 시간 & 형식화

date, time and formatting

## [ 연습문제 ]

[10-1]

Calendar클래스와

SimpleDateFormat클래스를 이용해서

2010년의 매월 두 번째

일요일의 날짜를 출력하시오.

|  |
| --- |
| [실행결과] |
| 2010-01-10은 2번째 일요일입니다. |
| 2010-02-14은 2번째 일요일입니다. |
| 2010-03-14은 2번째 일요일입니다. |
| 2010-04-11은 2번째 일요일입니다. |
| 2010-05-09은 2번째 일요일입니다. |
| 2010-06-13은 2번째 일요일입니다. |
| 2010-07-11은 2번째 일요일입니다. |
| 2010-08-08은 2번째 일요일입니다. |
| 2010-09-12은 2번째 일요일입니다. |
| 2010-10-10은 2번째 일요일입니다. |
| 2010-11-14은 2번째 일요일입니다. |
| 2010-12-12은 2번째 일요일입니다. |

[10-2] 어떤 회사의 월급날이 매월 21일이다. 두 날짜 사이에 월급날이 몇 번있는지 계 산해서 반환하는 메서드를 작성하고 테스트 하시오.

import java.util.\*; import java.text.\*;

class Exercise10\_2 {

static int paycheckCount(Calendar from, Calendar to) {

**/\***

**(1)** 아래의 로직에 맞게 코드를 작성하시오**.**

1. **from** 또는 **to**가 **null**이면 **0**을 반환한다**.**
2. **from**와 **to**가 같고 날짜가 **21**일이면 **1**을 반환한다**.**
3. **to**와 **from**이 몇 개월 차이인지 계산해서 변수 **monDiff**에 담는다**.**
4. **monDiff**가 음수이면 **0**을 반환한다**.**
5. 만일 **from**의 일**(DAY\_OF\_MONTH)**이 **21**일이거나 이전이고

**to**의 일**(DAY\_OF\_MONTH)**이 **21**일이거나 이후이면 **monDiff**의 값을 **1** 증가시킨다**.**

1. 만일 **from**의 일**(DAY\_OF\_MONTH)**이 **21**일 이후고

**to**의 일**(DAY\_OF\_MONTH)**이 **21**일 이전이면 **monDiff**의 값을 **1** 감소시킨다**.**

**\*/**

return monDiff;

}

static void printResult(Calendar from, Calendar to) { Date fromDate = from.getTime();

Date toDate = to.getTime();

SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");

[연습문제]**/ch10/Exercise10\_2.java**

System.out.print(sdf.format(fromDate)+" ~ "

+sdf.format(toDate)+":"); System.out.println(paycheckCount(from, to));

}

public static void main(String[] args) { Calendar fromCal = Calendar.getInstance(); Calendar toCal = Calendar.getInstance();

fromCal.set(2010,0,1); toCal.set(2010,0,1); printResult(fromCal, toCal);

fromCal.set(2010,0,21); toCal.set(2010,0,21); printResult(fromCal, toCal);

fromCal.set(2010,0,1); toCal.set(2010,2,1); printResult(fromCal, toCal);

fromCal.set(2010,0,1); toCal.set(2010,2,23); printResult(fromCal, toCal);

fromCal.set(2010,0,23); toCal.set(2010,2,21); printResult(fromCal, toCal);

fromCal.set(2011,0,22); toCal.set(2010,2,21); printResult(fromCal, toCal);

}

}

|  |
| --- |
| [실행결과] |
| 2010-01-01 ~ 2010-01-01:0 |
| 2010-01-21 ~ 2010-01-21:1 |
| 2010-01-01 ~ 2010-03-01:2 |
| 2010-01-01 ~ 2010-03-23:3 |
| 2010-01-23 ~ 2010-03-21:2 |
| 2011-01-22 ~ 2010-03-21:0 |

[10-3] 문자열 “123,456,789.5”를 소수점 첫 번째 자리에서 반올림하고, 그 값을 만 단 위마다 콤마(,)로 구분해서 출력하시오.

data:123,456,789.5

반올림:123456790 만단위:1,2345,6790

[실행결과]

[10-4] 화면으로부터 날짜를 “2007/05/11”의 형태로 입력받아서 무슨 요일인지 출력하 는 프로그램을 작성하시오.

단, 입력된 날짜의 형식이 잘못된 경우 메세지를 보여주고 다시 입력받아야 한다.

날짜를 yyyy/MM/dd의 형태로 입력해주세요.(입력예:2007/05/11)

>>2009-12-12

날짜를 yyyy/MM/dd의 형태로 입력해주세요.(입력예:2007/05/11)

>>2009/12/12

입력하신 날짜는 토요일입니다.

[실행결과]

[10-5] 다음과 같이 정의된 메서드를 작성하고 테스트하시오.

메서드명 : getDayDiff

기 능 : yyyymmdd형식의 두 문자열을 넘겨받으면 두 날짜의 차이를 일(day)단위로 반환한다. 단, 첫 번째 날짜 빼기 두 번째 날짜의 결과를 반환한다.

만일 주어진 문자열이 유효하지 않으면 0을 반환한다. 반환타입 : int

매개변수 : String yyyymmdd1 - 시작날짜 String yyyymmdd2 - 끝 날짜

import java.util.\*;

class Exercise10\_5 {

**/\***

**(1) getDayDiff**메서드를 작성하시오**.**

**\*/**

public static void main(String[] args){ System.out.println(getDayDiff("20010103","20010101")); System.out.println(getDayDiff("20010103","20010103")); System.out.println(getDayDiff("20010103","200103"));

}

}

[연습문제]**/ch10/Exercise10\_5.java**

2

0

0

[실행결과]

[10-6] 자신이 태어난 날부터 지금까지 며칠이 지났는지 계산해서 출력하시오.

birth day=2000-01-01 today =2016-01-29

5872 days

[실행결과]

[10-7] 2016년 12월 넷째주 화요일의 날짜를 아래의 실행결과와 같은 형식으로 출력하시 오.

2016-12-27

[실행결과]

[10-8] 서울과 뉴욕간의 시차가 얼마인지 계산하여 출력하시오.

2016-01-28T23:01:00.136+09:00[Asia/Seoul]

2016-01-28T09:01:00.138-05:00[America/New\_York] sec1=32400

sec2=-18000

diff=14 hrs

[실행결과]