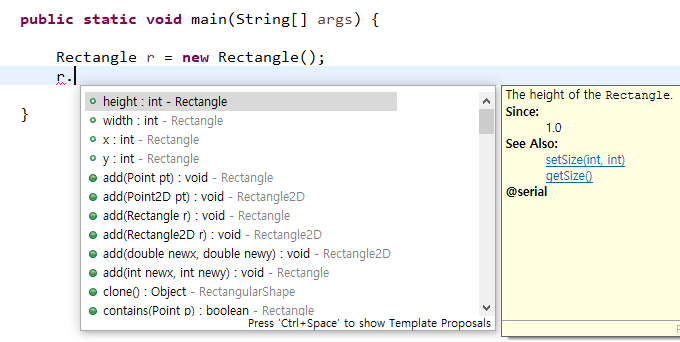
**자바지향프로그래밍2 실습과제**

**제3주 객체와 클래스 2**

* 클래스 선언문 바로 위에 그 클래스에 대한 설명을 주석으로 넣으세요.
* **main 메소드 외의 메소드에도 주석을 넣으세요.**

1. 표준 라이브러리에 들어 있는 java.awt.Rectangle 클래스를 이용해 봄으로써 표준 라이브러리 클래스 이용법을 다시 한번 학습합니다.

Rectangle 클래스가 intersection 메소드를 갖는다는 것은 Rectangle 객체에게 intersection 메소드를 호출할 수 있다는 의미입니다. 아래 그림에서 보듯이, 이클립스 편집창에서 Rectangle 인스턴스를 하나 만들어 r이라고 한 후, r과 .을 입력하면 r의 public 필드들과 public 메소드들이 모두 보여집니다. 그 중에 intersection이 있습니다.



java api 문서에서 Rectangle 클래스를 찾아 intersection 메소드 설명을 보세요. (구글에서 “java api”로 검색)

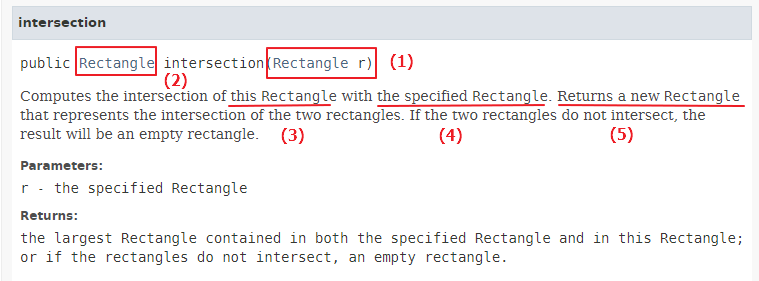
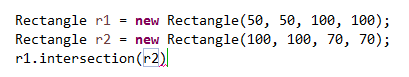
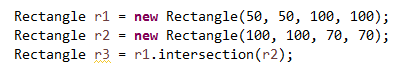


Figure 1 intersection api

위 그림 Figure 1의 (1)로부터, 이 메소드를 호출할 때는 Rectangle을 인자로 주어야 한다는 것을 알 수 있습니다. 그러니까 코드가 아래와 같은 모습이어야 합니다. r1에게 intersection 메소드를 호출하면서 r2를 인자로 주었습니다.

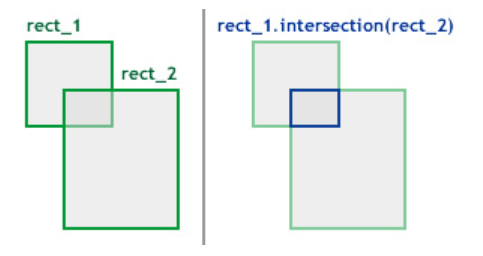


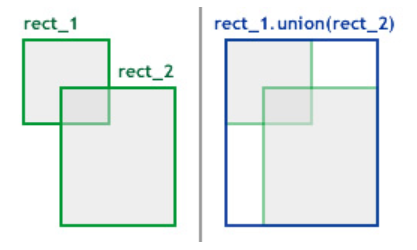
또, 위 그림 Figure 1의 (2)로부터 이 메소드는 Rectangle을 반환한다는 것을 알 수 있습니다. 그러니까 코드는 아래와 같은 모습이어야 합니다. Rectangle 타입 변수 r3를 선언하고, 반환되는 Rectangle을 r3에 저장합니다.



위 그림 Figure 1의 (3), this Rectangle은 intersection 메소드를 수행하는 r1을 말합니다. 위 그림 Figure 1의 (4), the specified Rectangle은 인자로 주어진 r2를 말합니다. 그리고 위 그림 Figure 1의 (5)는 **새로 만들어진** Rectangle 인스턴스가 반환된다고 말하고 있습니다.

Intersection과 union을 그림으로 보이면 아래와 같습니다.





아래 프로그램을 잘 읽어 보고 주석에 맞춰 괄호 안에 적당한 표현을 적어 넣으세요. 아래 프로그램은 "복사-붙여넣기(복붙)"를 할 수 있습니다. 완성된 프로그램을 강온실 실습과제 p1의 답으로 제출하세요.

import java.awt.Rectangle;

public class Rectangles {

public static void main(String[] args) {

// 상태변수 선언 및 초기화

Rectangle r1 = new Rectangle(50, 50, 100, 100);

Rectangle r2 = new Rectangle(100, 100, 70, 70);

Rectangle r3 = new Rectangle(300, 50, 100, 100);

Rectangle r4 = new Rectangle(350, 100, 70, 70);

Rectangle intersection, union; // 상태변수 선언

// intersection과 union을 구함

intersection = ( ); // r1과 r2의 intersection

union = ( ); // r3와 r4의 union

System.out.println("r1: " + r1);

System.out.println("r2: " + r2);

System.out.println("intersection: " + intersection);

System.out.println();

System.out.println("r3: " + r3);

System.out.println("r4: " + r4);

System.out.println("union: " + union);

}

}

<참고>

아래 프로그램을 이클립스에서 실행해 보세요. 아래 프로그램도 "복사-붙여넣기(복붙)"를 할 수 있습니다.

import java.awt.Color;

import java.awt.Graphics;

import java.awt.Rectangle;

import javax.swing.JFrame;

public class RectanglesView extends JFrame {

private static final long serialVersionUID = 1L;

**// 상태변수 선언 및 초기화**

**Rectangle r1 = new Rectangle(50, 50, 100, 100);**

**Rectangle r2 = new Rectangle(100, 100, 70, 70);**

**Rectangle r3 = new Rectangle(300, 50, 100, 100);**

**Rectangle r4 = new Rectangle(350, 100, 70, 70);**

**Rectangle intersection, union; // 상태변수 선언**

**// 생성자**

**RectanglesView() {**

**super("Rectangles"); // 창 제목 설정**

**setSize(500, 250); // 창 크기 설정**

**// intersection과 union을 구함**

**intersection = r1.intersection(r2);**

**union = r3.union(r4);;**

**System.out.println("r1: " + r1);**

**System.out.println("r2: " + r2);**

**System.out.println("intersection: " + intersection);**

**System.out.println();**

**System.out.println("r3: " + r3);**

**System.out.println("r4: " + r4);**

**System.out.println("union: " + union);**

**}**

public void paint(Graphics g) {

**// r1, r2를 그림 (draw Rectangle)**

**g.drawRect(r1.x, r1.y, r1.width, r1.height);**

**g.drawRect(r2.x, r2.y, r2.width, r2.height);**

**// Color 생성자 파라미터: red, green, blue, 투명도 (0~255)**

**g.setColor(new Color(100, 100, 100, 100));**

**g.fillRect(intersection.x, intersection.y, intersection.width, intersection.height); // fill Rectangle**

**g.setColor(Color.BLACK);**

**g.drawRect(intersection.x, intersection.y, intersection.width, intersection.height);**

**g.drawRect(r3.x, r3.y, r3.width, r3.height);**

**g.drawRect(r4.x, r4.y, r4.width, r4.height);**

**g.setColor(new Color(100, 100, 100, 100));**

**g.fillRect(union.x, union.y, union.width, union.height);**

**g.setColor(Color.BLACK);**

**g.drawRect(union.x, union.y, union.width, union.height);**

}

public static void main(String[] args) {

JFrame f = new RectanglesView();

f.setVisible(true);

}

}

RectanglesView는 강온실로 제출하지 않습니다.

1. 자바 표준라이브러리에 있는 클래스의 인스턴스를 구성하고 그 객체에게 메소드를 호출하는 연습입니다.

로또 복권 번호 추천 프로그램을 작성하세요. 0과 49를 포함하여 0과 49사이의 숫자 여섯 개를 무작위로 선택하여 이들을 차례로 출력하면 됩니다. 우연히 같은 숫자가 반복되어도 좋습니다. 이와 같이 범위에 0과 49를 포함하는 것을 inclusive라고 하고, 0과 49를 제외하는 것을 exclusive라고 합니다. 클래스의 이름은 LottoNumber로 하세요. Java.util.Random 클래스 객체를 이용하세요.

무작위로 정수를 선택하기 위해서는 Random 클래스를 사용합니다. Random 클래스 객체를 하나 만들어 그 객체에게 nextInt 메소드를 호출하면 무작위로 정수를 반환합니다. 무작위 수가 하나 더 필요하면 Random 객체를 새로 만들 필요 없이 이미 만들어 놓은 Random 객체에게 한 번 더 nextInt 메소드를 호출하면 됩니다.

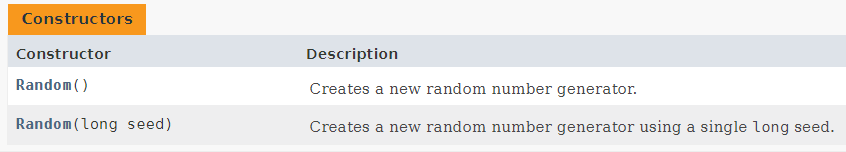
[API 문서](https://docs.oracle.com/en/java/javase/17/docs/api/index.html)에서 Random 클래스를 찾아 메소드 사용법을 읽어보세요. Random이 지원하는 nextInt 메소드는 아래와 같이 두 개가 있습니다(다중정의, overloading).

int nextInt(): 0 이상 Integer.MAX\_VALUE 이하 정수 난수 한 개를 반환

int nextInt(int n): 0 이상 n 미만 정수 난수 한 개를 반환

하나의 Random 객체에게 nextInt 메소드를 연이어 호출하면 그 때마다 난수가 만들어집니다.

[API 문서](https://docs.oracle.com/en/java/javase/17/docs/api/index.html)에서 Random 클래스의 생성자를 찾아 읽어보세요. Random 클래스는 아래와 같이 두 개의 생성자를 갖습니다.



두 번째 생성자를 이용하여 Random 객체를 생성하면 주어진 seed에 따라 난수열이 결정됩니다. 같은 seed를 갖는 두 Random 객체가 만들어내는 난수열은 같습니다. Seed가 다른 두 Random 객체는 서로 다른 난수열을 만들어냅니다.

아래는 LottoNumber프로그램의 실행화면 예입니다.







(참고 1)

난수를 발생시키는 방법은 두 가지가 있습니다.

1. Math 클래스의 static 메소드인 random() 메소드를 호출하는 방법

간단하고 기능이 적은 방법입니다. 간편하게 0.0 이상 1.0 미만 double 타입 난수를 만들 수 있습니다.

② Random 클래스 인스턴스를 구성한 후 이 객체에게 필요한 (인스턴스) 메소드를 호출하는 방법

이 문제에서 사용하는 방법입니다. 여러 가지 메소드가 준비되어 있어 필요에 따라 골라 호출할 수 있습니다. 강력한 기능을 지원합니다.

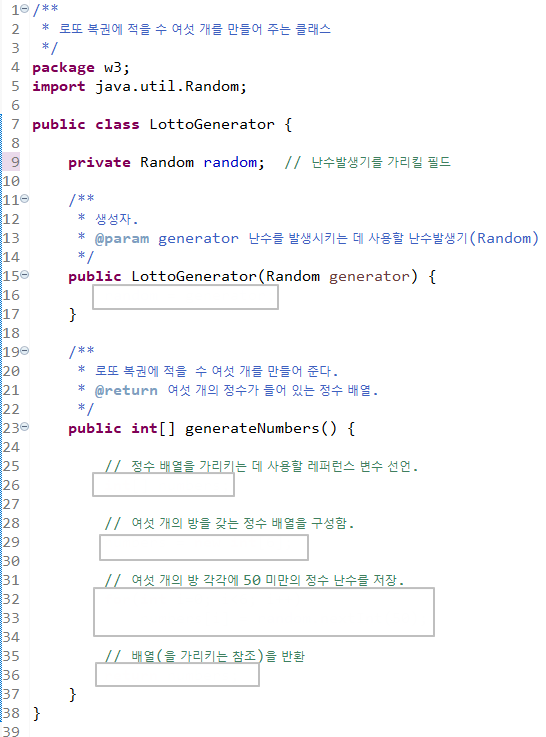
완성된 프로그램을 강온실 실습과제 p2의 답으로 제출하세요.

1. 위 p2번 문제에서는 main에서 Random 객체를 이용해 여섯 개의 난수를 만들었습니다. 이번에는 “로또 번호 생성기” 객체를 만들고 이 놈에게 여섯 개의 난수를 만들도록 일을 시켜 봅시다. 로또 번호 생성기의 이름은 LottoGenerator로 짓습니다.

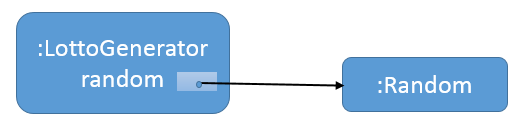
아래와 같이 LottoGenerator 클래스를 작성하세요.

* LottoGenerator 클래스는 로또 번호를 생성해 주는 generateNumbers라는 메소드를 갖습니다. LottoGenerator 인스턴스에게 generateNumbers 메소드를 호출하면 이 메소드는 로또 복권에 적어 넣을 여섯 개의 숫자들을 가진 int 배열을 반환합니다. **int 배열을 반환한다는 말은 int 배열을 가리키는 참조(reference)를 반환한다는 말입니다.**
* generateNumbers 메소드가 여섯 개의 난수를 만들어 내려면 Random 객체를 이용해야 합니다. 그러니까 LottoGenerator는 Random 객체를 하나 갖고 있어야 합니다. 이를 위해 LottoGenerator는 Random 타입 필드를 갖도록 하고 LottoGenerator 생성자에서 이 필드를 초기화해줍니다.

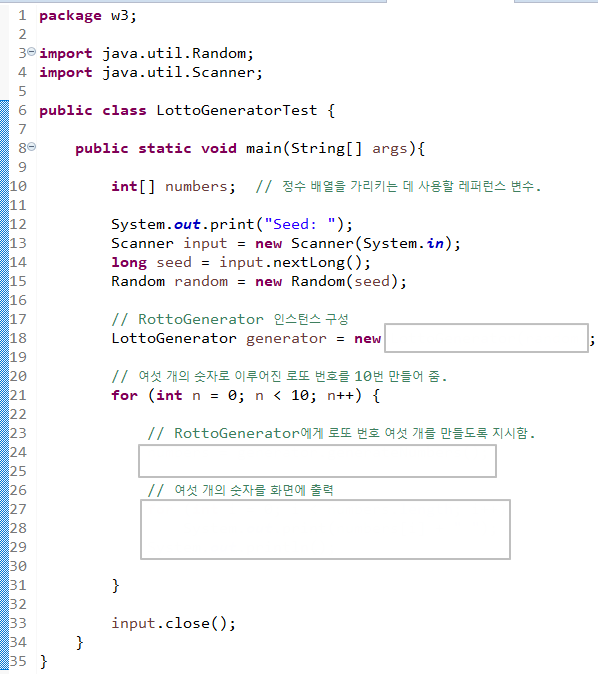
아래는 프로그램 뼈대입니다.



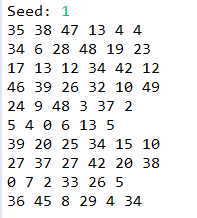
LottoGenerator의 객체 다이어그램 (object diagram)은 아래와 같습니다. **:**LottoGenerator는 Lottogenerator 인스턴스라는 의미입니다. random은 그 객체가 갖고 있는 필드(상태변수)입니다.

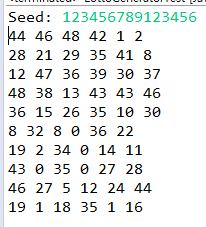


LottoGenerator 객체를 이용하여 여섯 개의 숫자로 이루어진 로또 번호를 10번 만들어 보여주는 main 메소드를 작성하세요. 클래스 이름은 LottoGeneratorTest로 지으세요.



아래는 실행화면 예입니다.

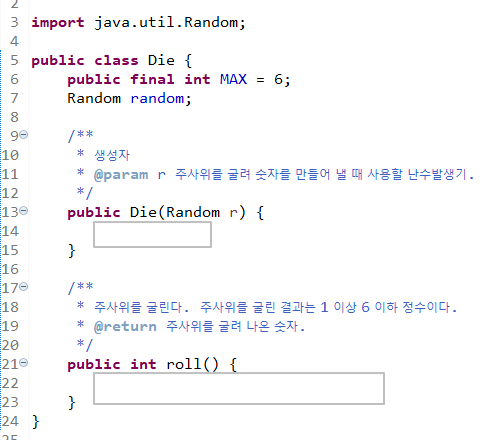




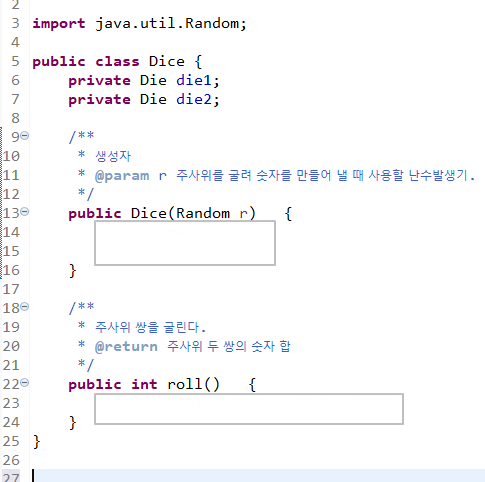
완성된 프로그램을 강온실 실습과제 p3의 답으로 제출하세요. 강온실에 제출할 때는 LottoGenerator와 LottoGeneratorTest 등 두 개의 클래스를 제출해야 합니다. 두 클래스 중 LottoGenerator 클래스 선언문에는 public을 붙이지 말고, main 메소드를 갖는LottoGeneratorTest클래스 선언문에는 public을 붙이세요.

1. 두 개의 주사위를 굴려 나오는 수의 합을 관찰하는 프로그램을 작성합니다. 이 과제에서는 주사위를 객체로 모델링합니다.

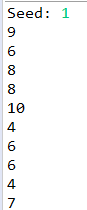
우선 한 개의 주사위를 나타내는 Die 클래스를 작성합니다. 이 클래스의 뼈대가 아래에 있습니다.

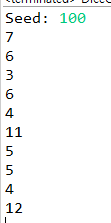


다음으로는 주사위 한 쌍을 나타내는 Dice클래스를 작성합니다. 영어 단어 주사위(die)의 복수형이 dice입니다. 아래 프로그램을 완성하세요. 아래 프로그램에서는 Dice클래스가 두 개의 Die 인스턴스들을 상태변수로 갖도록 했습니다. 생성자에서는 이 두 개의 상태변수 각각에 Die 객체를 만들어 저장해 둡니다. 두 개의 Die 객체는 같은 (하나의) 난수발생기를 사용하도록 하세요. Dice의 roll 메소드는 이 두 개의 주사위를 던져 나온 값을 반환합니다. Dice 클래스의 roll 메소드와 Die 클래스의 roll 메소드는 이름은 같지만 서로 다른 메소드입니다.



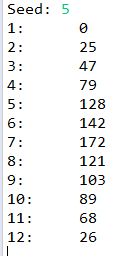
Dice에게 roll 메소드를 호출하여 그 결과(주사위 두 개 쌍을 던져 나온 숫자의 합)를 화면에 출력하는 작업을 10회 반복하는 main 메소드를 작성하세요. 클래스 이름은 DiceCasting1으로 지으세요. 실행 예는 아래와 같습니다.

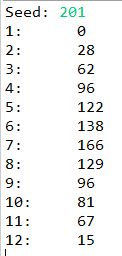




완성된 프로그램을 강온실 실습과제 p4의 답으로 제출하세요. 강온실에 제출할 때는 Die, Dice, DiceCasting1 등 세 개의 클래스를 제출해야 합니다. 세 클래스 중 Die와 Dice 클래스 선언문에는 public을 붙이지 말고, main 메소드를 갖는 DiceCasting1 클래스 선언문에는 public을 붙이세요.

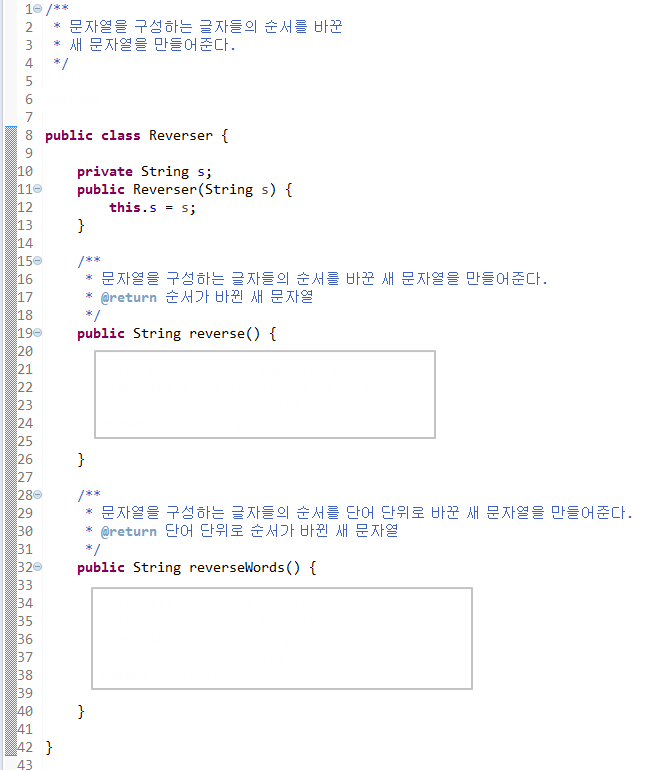
1. 위 p4 문제에서 Dice에게 roll을 호출하면 그 반환값은 2 이상 12 이하입니다. Dice에 roll을 1000번 호출하여 나온 값의 빈도수를 출력하는 main 메소드를 작성하세요. 클래스 이름은 DiceCasting2로 지으세요.   
   실행 예는 아래와 같습니다. 각 줄의 콜론(:) 바로 다음에는 탭 문자(‘\t’)를 넣으세요.





완성된 프로그램을 강온실 실습과제 p5의 답으로 제출하세요. 강온실에 제출할 때는 Die, Dice, DiceCasting2 등 세 개의 클래스를 제출해야 합니다. 세 클래스 중 Die와 Dice 클래스 선언문에는 public을 붙이지 말고, DiceCasting2 클래스 선언문에는 public을 붙이세요.

1. 문자열의 글자 순서를 바꿔 만들어진 새 문자열을 반환하는 프로그램을 작성하세요. 클래스 이름은 Reverser로 짓고 그 클래스에 인스턴스 메소드 두 개를 구현하세요. 구현할 메소드는 아래 그림과 같습니다.

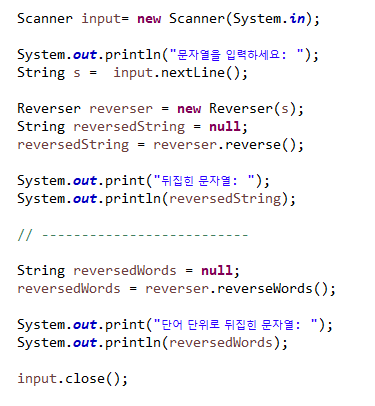


Reverser 인스턴스를 구성할 때는 문자열을 생성자 인자로 주어야 합니다. 그러면 그 문자열을 필드로 갖는 Reverser 인스턴스가 만들어집니다. Reverser 객체에게 reverse 메소드를 호출하면 그 객체가 갖고 있는 문자열의 순서를 바꾸어 반환합니다.

String은 immutable 클래스입니다. 즉, String 객체는 한 번 만들어지면 그 내용을 수정할 수 없습니다. reverse와 reverseWords 메소드는 기존 문자열을 수정하는 것이 아니라 새 문자열을 만들어 반환해야 합니다. 이 메소드들을 구현하기 위해서는 기존 문자열의 맨 끝 글자(단어)부터 시작하여 하나씩 차례로 글자(단어)를 읽고 그 글자(단어)를 새 문자열에 덧붙여 나가야 합니다. 이 때 새 문자열을 처음부터 String으로 만들면 한 글자(단어)를 덧붙일 때마다 새로운 String 객체를 구성하게 되어 비효율적이므로, String 대신 StringBuilder를 사용합니다. " StringBuilder 자바"를 키워드로 구글 검색을 하면 StringBuilder를 어떻게 사용하는지 보여주는 많은 예를 볼 수 있습니다. String 내의 몇 번째 글자를 읽을 때 사용하는 String 메소드는 charAt입니다. "charAt 자바"를 구글 검색하면 사용 예를 많이 볼 수 있습니다. 뒤집힌 내용의 StringBuilder가 완성되면 StringBuilder에게 toString 메소드를 호출하면 같은 내용을 갖는 String 객체가 반환됩니다.

String에 들어 있는 단어들을 추출하려면 어떻게 하는 것이 좋을까요? String의 메소드들 중 split이라는 메소드가 있습니다. 이 메소드는 string을 주어진 "구분자"를 기준으로 단어들을 추출하여 그 단어들을 String 배열로 반환합니다. "String split 자바"로 구글 검색을 해 보면 사용 예를 많이 볼 수 있습니다.

키보드로부터 한 줄짜리 문장 하나를 입력 받고 Reverser의 두 메소드를 이용하여 이 문장의 순서를 뒤집어 출력하는 main 메소드를 작성하세요. 클래스 이름은 ReverserTest 로 지으세요.



위 프로그램의 실행 예는 아래와 같습니다.

Text

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

<참고>

문자열을 단어 단위로 나눌 때 단어와 단어 사이를 구분짓는 문자열을 구분자(delimiter)라고 합니다.

"boo:and:foo"를 split할 때,

구분자가 “:” 인 경우에는

{"boo", "and", "foo"}로 분해됩니다.

구분자가 “o” 인 경우에는

{"b", "", ":and:f"}로 분해됩니다. (끝에 나오는 빈 문자열들은 모두 버립니다!)

String의 split 메소드를 호출할 때 split("\\s+")과 같이 호출하면 한 개 이상 연속된 화이트스페이스를 구분자(delimiter)로 사용합니다.

" \t\n\r\f" <- 공백, 탭, 새줄, 캐리지리턴, 폼피드 - 요것들을 화이트스페이스라고 합니다.

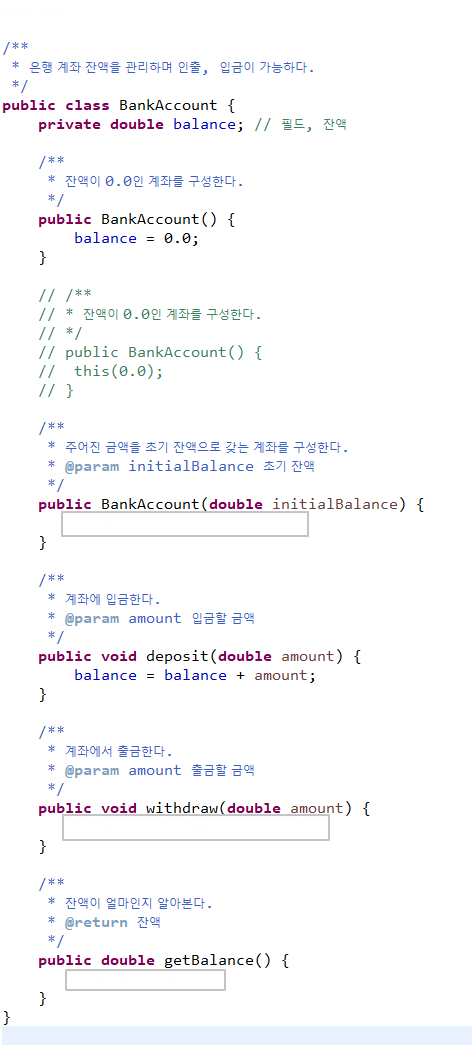
정규식에서 \s는 화이트스페이스 한 글자를 나타냅니다.

\s+는 화이트스페이스 글자가 하나 이상 연속해서 나타나는 패턴을 말합니다.

문자열 속에 ‘\’ 글자 자체를 적으려면 \\라고 적어야 합니다. \를 하나만 적으면 이 놈이 ‘\’로 인식되지 않고 그 다음에 오는 글자를 특수문자로 만들어주는 역할을 하게 되기 때문입니다. (가령 \n은 ‘\’와 ‘n’의 두 글자로 인식되지 않고 \n 전체가 새줄문자로 인식됩니다.)

완성된 프로그램을 강온실 실습과제 p6의 답으로 제출하세요.

1. 강의자료에 있는 BankAccount 클래스를 이용하는 응용프로그램을 작성하는 과제입니다. 아래 BankAccount 프로그램을 잘 읽어 보고 미완성 부분을 채워 넣으세요.



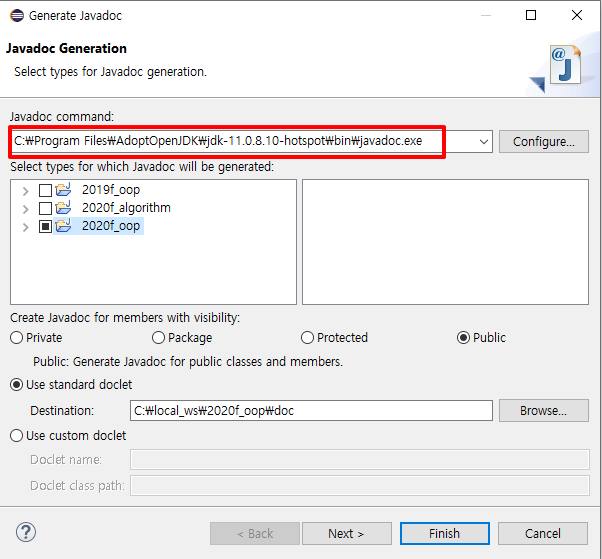
위 프로그램에서 첫 번째 생성자인 파라미터 없는 생성자는 자신의 다른 생성자를 호출하는 방식으로 구현할 수도 있습니다. (녹색으로 주석 처리된 부분처럼…)

또, main 메소드를 가지며 이름이 BankAccountTester라는 클래스를 작성하세요. 이 클래스의 main 메소드에서는 BankAccount 객체를 두 개 구성합니다. 첫 BankAccount 객체는 파라미터 없는 생성자를 사용하여 만들고, 다른 하나는 double 타입 파라미터 하나를 갖는 생성자를 이용하여 만들되 초기 잔고가1,000.0원이 되게 하세요. 그런 후 각각의 계좌로 500.0원을 입금하고 100.0원을 출금한 후 잔액을 화면에 프린트하도록 하세요. 출력 화면 예를 들면 아래와 같습니다.



또, javadoc을 이용하여 BankAccount 클래스의 사용설명서(api document)를 작성하세요. eclipse에서 javadoc을 사용하여 사용설명서를 만드는 방법은 아래와 같습니다.

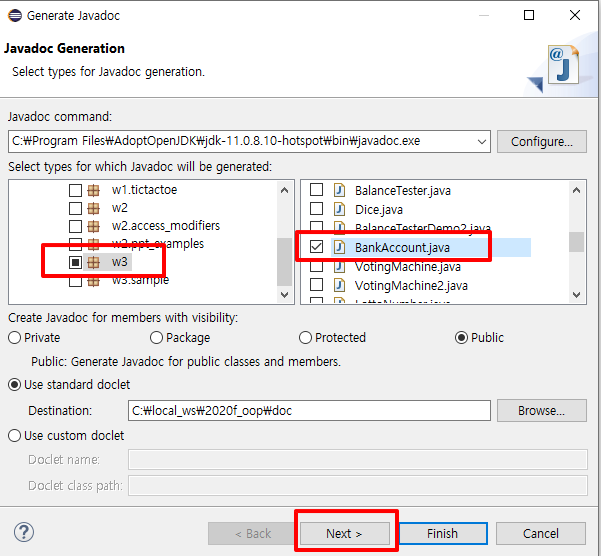
eclipse 맨 위에 있는 메뉴에서 Project --> Generate javadoc...을 선택하면 아래와 같은 대화창이 뜹니다.



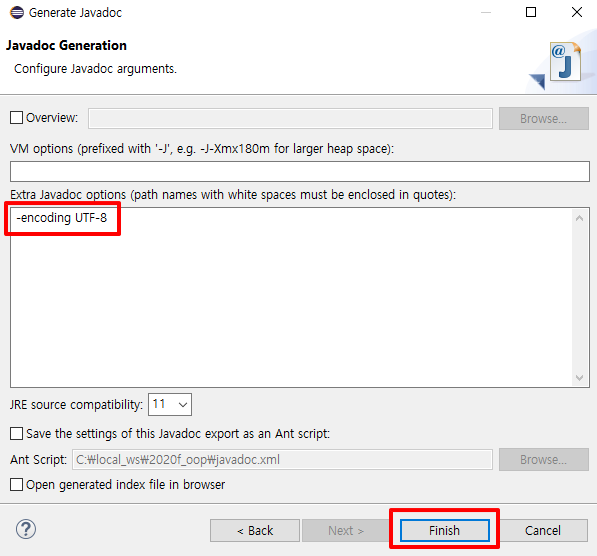
빨간 네모와 같이 설정되어 있으면 javadoc 도구가 제대로 설정되어 있는 것입니다.

빨간 메모 부분이 비어 있다면 설정이 필요합니다. 그 오른쪽에 있는 Configure...를 눌러 해당 파일을 찾아 설정해 주세요.

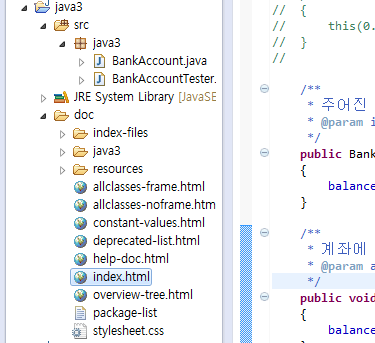
그리고 아래 그림과 같이 api document를 만들어낼 대상 소스파일을 선택해 주고 Next> 버튼을 누릅니다.



그 다음 창에서는 아무것도 하지 말고 Next> 버튼을 누릅니다. 그 다음 나타나는 창에서 아래 그림과 같이 소스코드의 encoding을 UTF-8로 설정해 주고 Finish 버튼을 누릅니다.



그러면 javadoc이 실행되고 아래와 같이 document가 만들어지는데 index.html을 클릭하면 편집창에서 api 문서를 볼 수 있습니다. 소스코드에 입력한 주석들이 어떻게 api 문서에 나타나는지 잘 살펴보세요.



이렇게 만든 api 문서를 과제결과로 제출할 필요는 없습니다. 소스코드 파일들만 강온실 실습과제 p7의 답으로 제출하세요. 강온실에 제출할 때는 BankAccount, BankAccountTester 등 두 개의 클래스를 제출해야 합니다. 두 클래스 중 BankAccount 클래스 선언문에는 public을 붙이지 말고, BankAccountTester클래스 선언문에는 public을 붙이세요.

1. 스태틱(static) 필드를 이용하는 문제입니다. 강의자료에 나오는 예입니다. 강의자료를 참고하여 프로그램을 작성하세요. 위 p7번 문제의 BankAccount 클래스를 수정하여 NumberedBankAccount 클래스를 작성하세요. 이 클래스는 아래와 같은 특징을 갖도록 하세요.

* 각 NumberedBankAccount 객체는 계좌번호(accountNumber)를 갖는다. 각 NumberedBankAccount의 계좌번호는 객체가 구성될 때 자동으로 부여되며 100번부터 오름차순으로 부여된다.
* 아래와 같은 메소드를 갖는다.

public int getAccountNumber()      // 계좌번호를 반환

public String toString()

             // 계좌번호와 잔액을

             // “NumberedBanksAccount[accountNumber=1001, balance=2500.0]" 형식으로 반환

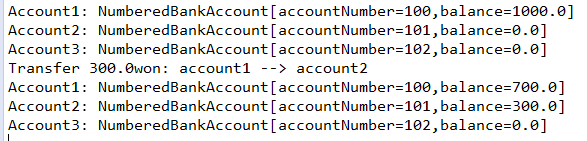
public void transfer(double amount, NumberedBankAccount other)

             // other에게 amount만큼 계좌이체한다(송금한다.)



그리고 적당한 main 메소드를 갖는 NumberedBankAccountTest 프로그램을 작성하여 NumberedBankAccount의 동작을 확인하세요. main에서 초기 잔고가 각각 1000.0, 0.0, 0,0인 세 개의 NumberedBankAccount들을 생성하고 이들을 출력합니다. 그리고 첫 계좌에서 두 번째 계좌로 300.0원을 이체한 후 세 계좌를 다시 출력합니다.

NumberedBankAccountTest의 실행 결과가 아래와 같게 하세요.



NumberedBankAccount와 NumberedBankAccountTest 등 두 개의 클래스를 제출하세요.

1. 인기투표를 하는 데 쓰는 기계를 만들어 봅니다. 사람들이 떡복기와 순대 중 어떤 것을 더 좋아하는지 투표합니다. 이를 위해 투표기가 필요합니다. 투표기 클래스 이름을 VotingMachine이라고 합시다. 이 클래스는 아래와 같은 메소드를 가지면 될 것입니다.

public void voteForTkeokk()           // 떡복기에 투표

public void voteForSun()               // 순대에 투표

public void clear()                        // 지금까지의 투표 내용을 모두 삭제 (reset)

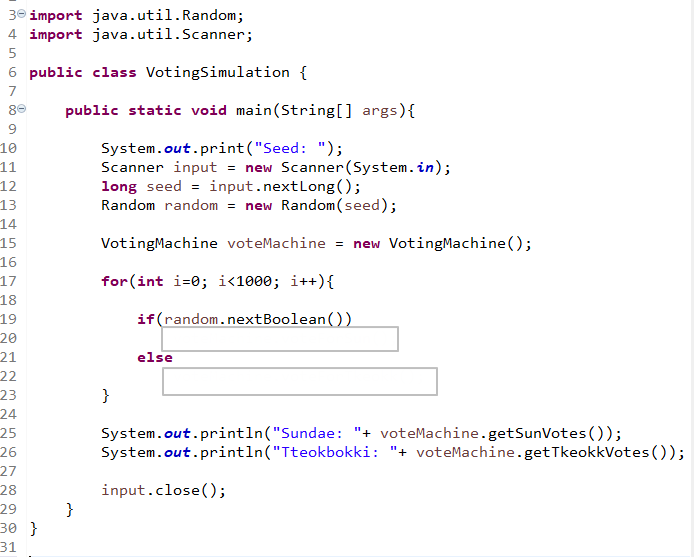
public int getTkeokkVotes()            // 떡복기 득표수를 반환

public int getSunVotes()                // 순대 득표수를 반환

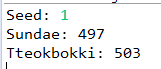
VotineMachine이 한 대 마련되면 순대를 좋아하는 사람은 voteForSun을 호출하고, 떡볶이를 좋아하는 사람은 voteForTkeokk을 호출합니다. 최종 득표를 보려면 getTkeokkVotes와 getSunVotes를 호출하면 됩니다.

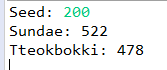
VotineMachine 클래스의 생성자와 위 메소드들을 구현하여 클래스를 완성하세요. 인스턴스 필드도 필요에 따라 추가하세요. 생성자에서는 인스턴스 필드를 적절히 초기화합니다.

VotingSimulation이라는 클래스를 작성하고 여기에 main 메소드를 두어 VotingMachine의 기능을 아래 예와 같이 난수를 이용해 시뮬레이션해 봄으로써 VotingMachine을 테스트하세요. 아래 프로그램에서는 1000명의 사람들이 50%의 확률로 떡뽁이와 순대를 선택한다고 가정하고 이런 무작위 선택을 모의실험하기 이해 Random에게 nextBoolean 메소드를 호출했습니다.



 프로그램 실행 예는 아래와 같습니다.





1. 위 p9 VotingMachie을 복사하여 VotingMachine2로 저장하고 VotingMachine2가 아래와 같은 세 개의 메소드를 갖도록 하세요. (clear를 제외한 나머지 기존 메소드들은 모두 지우세요.)

* public void vote(String pick)     // pick에 투표 (pick은 “Tteokbokki” 혹은 “Sundae”), “Tteokbokki”도 “Sundae”도 아닌 경우 아무 일도 하지 않음.
* public void clear()                   // 지금까지의 투표 내용을 모두 삭제
* public int getVotes(String pick)  // pick의 득표수를 반환 ((pick은 “Tteokbokki” 혹은 “Sundae”), “Tteokbokki “도 “Sundae”도 아닌 경우 0을 반환

VotingSimulation2이라는 클래스를 작성하고 여기에 main 메소드를 두어 VotingMachine2의 기능을 테스트하세요. 출력은 위 p9와 같아야 합니다.

1. 아래 지시에 따라 Counter 클래스를 작성하세요.

하나의 필드(상태변수) value를 갖는다. value는 Counter의 현재 값을 나타낸다. value는 정수 값을 갖는다.

네 개의 메소드를 갖는다.

        countUp – value 값을 1만큼 증가시킨다.

        countDown - value 값을 1만큼 감소시킨다.

        reset – value 값을 0으로 만든다.

        getValue – 현재의 value 값을 반환한다.

두 개의 생성자를 갖는다.

        파라미터 없는 기본 생성자 – value 값을 0으로 초기화한다.

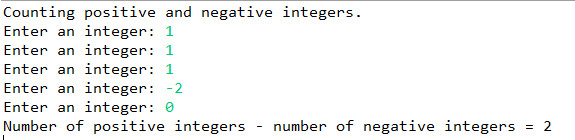
        파라미터 하나를 갖는 생성자 – 주어진 값으로 value를 초기화한다.

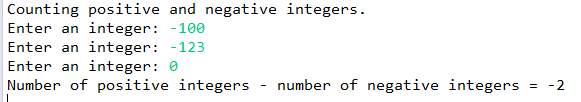
main 메소드를 갖는다. main 메소드에서는 아래와 같은 작업을 한다.

초기값이 0인 Counter 객체를 하나 구성한다.

숫자 0이 입력될 때까지 키보드로부터 정수들을 읽어 들인다.

키보드로부터 읽어 들인 정수들 중 양수의 갯수에서 음수의 갯수를 뺀 수를 출력한다. (키보드로부터 양수가 읽힐 때마다 Counter의 countUp() 메소드를 호출하고, 음수가 읽힐 때마다 countDown 메소드를 호출한다. 0이 입력되면 Counter의 getValue() 메소드를 호출하고 반환되는 값을 출력한다.)

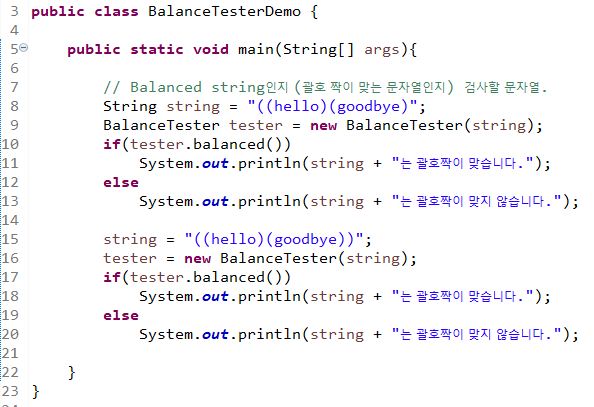




1. 아래 지시에 따라 BalanceTester 클래스를 작성하세요.

BalanceTester는 문자열에 포함되어 있는 괄호의 짝이 맞는지 검사해 줍니다.

우선BalanceTester를 활용하는 아래 예를 보세요.



위 프로그램의 실행 결과는 아래와 같습니다.



BalanceTester은 괄호의 짝이 맞는지 판별하기 위해 문제 p.11에서 만든 Counter을 이용합니다. 기본적으로는 주어진 문자열을 차례로 살펴 여는 괄호가 나타나면 countUp, 닫는 괄호가 나타나면 countDown하고 마지막에 counter이 0의 값을 갖는지 살펴보면 됩니다.

 BalanceTester클래스는 하나의 생성자와 하나의 메소드를 갖습니다.

생성자: BalanceTester(String str)

BalanceTester 인스턴스를 구성한다. BalanceTester는 주어진 문자열 str을 필드에 저장한다.

메소드: boolean balanced()

괄호의 짝이 맞는가? str에 있는 괄호가 짝이 맞으면 true를, 그렇지 않으면 false를 반환한다.

예를 들면, “((hello)(goodbye))”는 balanced이고,  “((a)(b)(())”는 balanced가 아니다.

괄호를 하나도 갖지 않는 문자열은 balanced인 것으로 간주한다.

문자열의 괄호가 짝이 맞는지 알아보려면 문자열을 왼쪽부터 오른쪽으로 차례로 스캔하면서 왼쪽 괄호가 나타날 때마다 countUp, 오른쪽 괄호가 나타날 때마다 countDown한다. 최종적으로 counter 값이 0이고, 문자열을 스캔하는 동안 **counter 값이 음수가 된 적이 없으면** 괄호가 짝이 맞는 것으로 판단하면 된다.

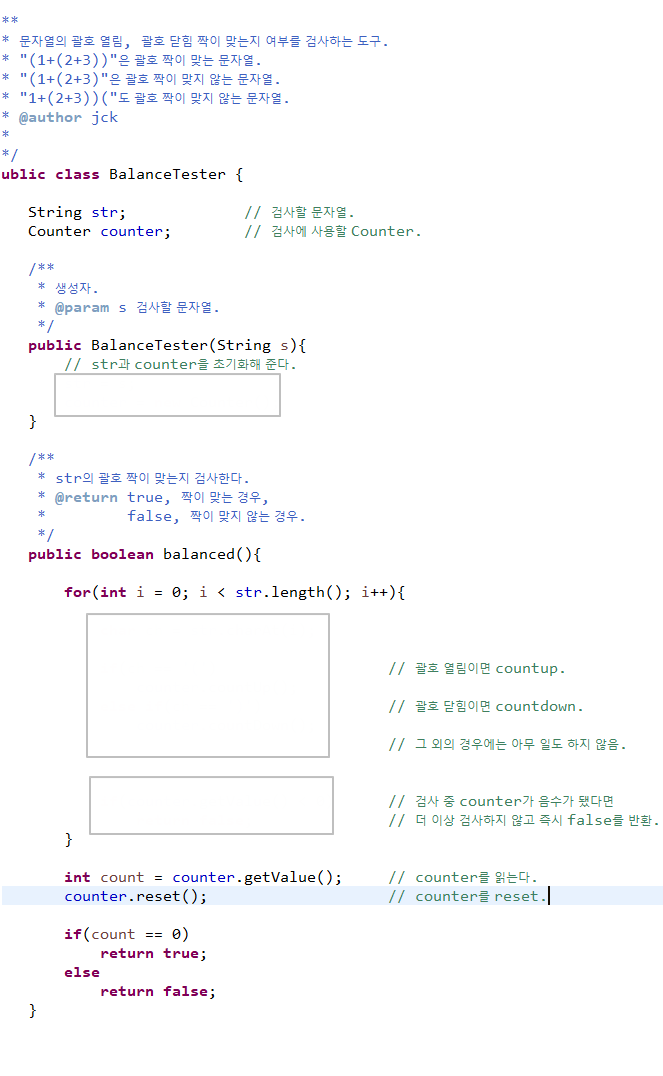
아래와 같은 경우는 최종 counter 값이 0이지만 스캔 도중에 counter 값이 음수가 된 적이 있으므로 괄호 짝이 맞지 않는 것으로 판별한다.

              (a))b(

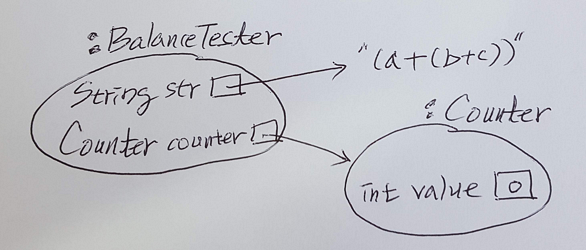
그러니까 문자열을 스캔하는 동안 counter 값이 **음수가 되면** 스캔을 멈추고 **즉시 false를 리턴하면 된다.**

BalanceTester는 Counter를 하나 가지고 있어야 합니다. 그러니까 BalanceTester는 필드로서 문자열 하나와 Counter 하나를 가져야 합니다.

BalanceTester 클래스의 뼈대는 아래와 같습니다.



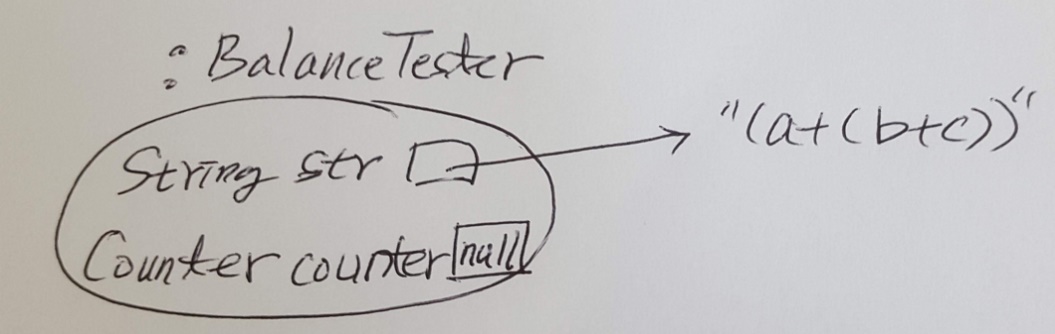
BalanceTester 객체가 구성되면 그 모양은 아래 그림과 같아야 합니다.



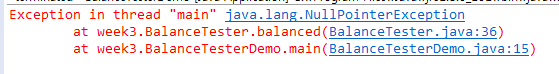
BalanceTester는 str과 counter 등 두 개의 상태변수를 필드로 갖습니다.

BalanceTester의 생성자가 이 두 개의 상태변수를 적절히 설정해 주어야 합니다.

만약, 아래 <그림 A>처럼 counter가 초기화되지 않고 기본값인 null로 남아 있다면 프로그램을 실행할 때 <그림 B>와 같은 에러가 날 것입니다. BalanceTester.java counter에게 메소드를 호출하는데 counter에 null이 들어 있어 아무 객체도 가리키지 않고 있기 때문에 발생하는 에러입니다.



<그림 A>



<그림 B>

BalanceTester 클래스를 테스트하는 BalanceTesterDemo2 클래스를 작성하세요.

아래는 BalanceTesterDemo2 실행 예입니다.









강온실에는 BalanceTesterDemo2, BalanceTester, Counter 등 세 개의 클래스를 제출하세요. Counter 클래스에도 main 메소드가 있지만, BalanceTesterDemo2만 public으로 선언하고, 나머지 두 클래스는 public으로 선언하지 않으면, BalanceTesterDemo2에 있는 main이 실행됩니다.

끝.