



ClassArrayTest



Git:



ManagerTest.java ×

EqualsTest.java ×

ClassArrayTest.java ×

BigIntegerFibonacci.java ×

BigInteger.java ×

import java.util.Scanner;

public class EqualsTest { //1 문자열 비교에 쓰이는 equals()메소드

public static void main(String[] args) {

Scanner scan = new Scanner(System.in);

String str = scan.nextLine();

if (str.equals("네")) { // '문자' 비교는 == 연산자를 이용할 수 있으나, "문자열" 비교는 equals()를 이용

System.out.println("오 그래");

} else if (str.equals("아니오")) {

System.out.println("(처)맞는말");

System.out.println("맞는말");

} else {

System.out.println("무조건 동의하세요!");

}

}

"Kotlin" plugin update available

Update

Plugin Settings...

Ignore t

Windows 정품 인증

ClassArrayTest



ClassArrayTest



Git:



agerTest.java ×

EqualsTest.java ×

ClassArrayTest.java ×

BigIntegerFibonacci.java ×

BigInteger.java ×

```
import java.util.Scanner;
```

```
public class ScoresTest { //2 클래스와 배열을 혼합해서 사용하는 케이스
```

```
    final int MAX = 5;
```

```
    float sum;
```

```
    float mean;
```

```
    int randArr[];
```

```
    public ScoresTest () {
```

```
        // 5개의 배열을 만들고 랜덤값을 할당함
```

```
        sum = 0;
```

```
        randArr = new int[MAX];
```

```
        for (int i = 0; i < MAX; i++) {
```

```
            randArr[i] = (int)(Math.random() * 50 + 50);
```

```
        }
```

```
    public void calcMean () {
```

```
        for (int i = 0; i < MAX; i++) {
```

```
            sum += randArr[i];
```

```
        }
```

```
        mean = sum / (float)MAX;
```

```
    public int[] getRandArr() {
```

```
        return randArr;
```

"Kotlin" plugin update available

[Update](#)

[Plugin Settings...](#)

[Ignore this](#)

Windows 정품 인증

ClassArrayTest > main



ClassArrayTest



Git:

anagerTest.java × EqualsTest.java × ClassArrayTest.java × BigIntegerFibonacci.java × BigInteger.java ×

```
public class ClassArrayTest {  
    public static void main(String[] args) {  
        ScoresTest[] st;  
        Scanner scan = new Scanner(System.in);  
  
        System.out.print("몇 개의 학급이 있나요 ? ");  
  
        int num = scan.nextInt();  
  
        st = new ScoresTest[num];  
  
        float totalSum = 0;  
        float totalNumber = 0;  
  
        // 객체가 만들어진 모습을 그림으로 직관적 표현(배열의 인덱스는 0부터 시작임)  
        // -----  
        // | 객체1 | 객체2 | 객체3 | 객체4 | 객체5 |  
        // -----  
        //      [0]      [1]      [2]      [3]      [4]
```

```
        for (int i = 0; i < num; i++) {  
            st[i] = new ScoresTest();  
            st[i] = new ScoresTest();  
            st[i].calcMean();  
            totalSum = st[i].getSum();  
            totalNumber = i + 1;  
        }  
    }  
}
```

"Kotlin" plugin update available

[Update](#) [Plugin Settings...](#) [Ignore th](#)

Windows 정품 인증

main



ClassArrayTest



Git:

nagerTest.java × EqualsTest.java × ClassArrayTest.java × BigIntegerFibonacci.java × BigInteger.java ×

rt java.math.BigInteger;

```
ic class BigIntegerFibonacci {          //3 큰 수를 변수의 한계없이 증권가나 실제상황에서 유용할 수 있는 BigInteger타입
public static void main(String[] args) {}
```

```
    final int MAX = 5;
```

```
    BigInteger[] fibArr = new BigInteger[MAX];
```

```
// BigInteger 데이터타입:무한대에 가까운 수를 표현하기 위해 쓰임. 양수,음수 모두 쓰임.
```

```
fibArr[0] = new BigInteger( val: "100");
```

```
fibArr[1] = BigInteger.ONE;
```

```
// Q.BigInteger클래스의 변수가 ONE,ZERO,TEN이 있는데,
```

```
// 그냥 BigInteger("0"),BigInteger("1") 이런식으로 쓰면 되는데, 왜 이 3가지변수들을 제공하는지 이유가 궁금합니다.
```

```
for (int i = 2; i < fibArr.length; i++) {
```

```
    // BigInteger에서는
```

```
    // add(덧셈),subtract(뺄셈),mutiply(곱셈),devide(나눗셈),remainder(나머지)
```

```
    // 매서드를 통해 연산을 한다.
```

```
    fibArr[i] = fibArr[i - 1].add(fibArr[i - 2]);
```

```
    System.out.println("fibArr[" + i + "] = " + fibArr[i]);
```

```
}
```

"Kotlin" plugin update available

[Update](#) [Plugin Settings...](#) [Ignore th](#)

Windows 정품 인증

main



ClassArrayTest



Git:



anagerTest.java ×

EqualsTest.java ×

ClassArrayTest.java ×

BigIntegerFibonacci.java ×

BigInteger.java ×

```
fibArr[0] = new BigInteger(val: "100");  
fibArr[1] = BigInteger.ONE;  
// Q. BigInteger 클래스의 변수가 ONE, ZERO, TEN이 있는데,  
// 그냥 BigInteger("0"), BigInteger("1") 이런식으로 쓰면 되는데, 왜 이 3가지 변수들을 제공하는지 이유가 궁금합니다.
```

```
for (int i = 2; i < fibArr.length; i++) {  
    // BigInteger에서는  
    // add(덧셈), subtract(뺄셈), multiply(곱셈), divide(나눗셈), remainder(나머지)  
    // 매서드를 통해 연산을 한다.  
  
    fibArr[i] = fibArr[i - 1].add(fibArr[i - 2]);  
    System.out.println("fibArr[" + i + "] = " + fibArr[i]);  
}
```

```
System.out.println("피보나치 수열의 n번째항은 = " + fibArr[MAX - 1]);
```

```
BigInteger two = new BigInteger(val: "2");  
BigInteger veryBigNum = new BigInteger(val: "2374923749237482384238482");
```

```
System.out.println("2 - 2374923749237482384238482 = " +  
    two.subtract(veryBigNum));
```

"Kotlin" plugin update available

[Update](#)[Plugin Settings...](#)[Ignore t](#)

Windows 정품 인증