

# [디지털 컨버전스] 스마트 콘텐츠와 웹 융합 응용SW 개발자 양성과정

강사 : 이상훈

학생 : 임초롱

# BigInteger : 무한 정수

링크 <https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day10/src/BigIntegerFibonacci.java>

```
1 import java.math.BigInteger;
2
3 public class BigIntegerFibonacci {
4     public static void main(String[] args) {
5
6         final int MAX = 5;
7
8         BigInteger[] fibArr = new BigInteger[MAX];
9
10        fibArr[0] = new BigInteger("100");
11
12        fibArr[1] = BigInteger.ONE;
13
14        for (int i = 2; i < fibArr.length; i++) {
15            ① fibArr[i] = fibArr[i - 1].add(fibArr[i - 2]);
16            System.out.println("fibArr[" + i + "] = " + fibArr[i]);
17        }
18
19
20
21        System.out.println("피보나치 수열의 n번째항은 = " + fibArr[MAX - 1]);
22
23        BigInteger two = new BigInteger("2");
24        BigInteger veryBigNum = new BigInteger("2374923749237482384238482");
25
26        System.out.println("2 - 2374923749237482384238482 = " +
27            two.subtract(veryBigNum));
28    }
29 }
```

Run: BigIntegerFibonacci

```
"C:\Program Files\Java\jdk-15.0.2\bin\java.exe" -javaagent
fibArr[2] = 101
fibArr[3] = 102
fibArr[4] = 203
피보나치 수열의 n번째항은 = 203
2 - 2374923749237482384238482 = -237492374923748238480
```

## < 개념 정리 >

### BigInteger :

엄청 큰 숫자를 처리하는 금융권 숫자 처리방식으로,  
무한 정수를 구현한 데이터 타입이다.

fibArr[0] = new BigInteger("100"); : fibArr[0] = 100  
fibArr[1] = BigInteger.ONE; : fibArr[1] = 1

BigInteger("100"); 와 BigInteger.ONE 로 표현할 수 있다.

BigInteger에서는 ( + , - , \* , / , % ) 가 아닌 다른 연산 방식을  
사용한다.

- 덧셈은 add()를 사용
- 뺄셈은 subtract()를 사용
- 곱셈은 multiply()를 사용
- 나눗셈은 divide()를 사용
- 나머지연산은 remainder를 사용

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

EX ① )    fibArr[2] = fibArr[2-1] + fibArr[2-1]  
            fibArr[2] = fibArr[1] + fibArr[0]  
            fibArr[2] = 1 + 100  
            fibArr[2] = 101

## equals : 문자열 비교 시 사용

링크 <https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day10/src/EqualsTest.java>

```
1  import java.util.Scanner;
2
3  ▶ public class EqualsTest {
4  ▶   public static void main(String[] args) {
5      Scanner scan = new Scanner(System.in);
6
7      String str = scan.nextLine();
8
9      if (str.equals("네")) {
10         System.out.println("오 그래");
11     } else if (str.equals("아니오")) {
12         System.out.println("맞는말");
13     } else {
14         System.out.println("무조건 동의하세요!");
15     }
16 }
17 }
```

문자열 사용 시 `scan.nextLine();` 를 쓰며,  
조건문에서 비교시 `equals`를 사용한다.

```
1  import java.util.Scanner;
2
3  ▶ public class Test {
4  ▶   public static void main(String[] args) {
5      Scanner scan = new Scanner(System.in);
6
7      int i = scan.nextInt();
8
9      if (i == 0) {
10         System.out.println("오 그래");
11     } else if (i > 0 ) {
12         System.out.println("맞는말");
13     } else {
14         System.out.println("무조건 동의하세요!");
15     }
16 }
17 }
```

정수형 사용 시 `scan.nextInt();` 를 쓰며,  
조건문에서 `==, >, <, <=, >=` 를 사용한다.

## ForEach 로 입력값 지정하기

링크 <https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day10/src/QnaAnswerTest.java>

```
1 // 시나리오.  
2 // 예로 특정 사업주와 관련된 평가의 평균치를 계산하고  
3 // 각 사업별로 가중치를 제각기 다르게 주는 케이스가 있다고 가정해보도록 한다.  
4 class MeanTest {  
5     float mean;  
6     int[] scores;  
7     int length;  
8  
9     @ public MeanTest (int[] arr) {  
10         length = arr.length;  
11         // 받는 배열 값의 길이를, length에 대입  
12         // int[] A = { 1, 2, 3 }; 로 받았음  
13         // length = 3  
14  
15         scores = new int[length];  
16         // int [] scores = new int[length];  
17  
18         for (int i = 0; i < length; i++) {  
19             scores[i] = arr[i];  
20             // scores[0] = arr[0] : 1  
21             // scores[1] = arr[1] : 2  
22             // scores[2] = arr[2] : 3  
23         }  
24     }  
25 }  
26
```

```
27 public void calcMean () {  
28     float sum = 0;  
29  
30     for (int i = 0; i < length; i++) {  
31         sum += scores[i];  
32         // sum = sum + scores[i];  
33         // sum = 0 + scores[0]  
34         // sum = scores[0] + scores[1]  
35         // sum = (scores[0] + scores[1]) + scores[2]  
36     }  
37  
38     mean = sum / (float)length;  
39     // mean = (scores[0] + scores[1]) + scores[2] / 3  
40 }  
41  
42 public void businessA() {  
43     mean *= 1.1;  
44 }  
45 public void businessB() {  
46     mean *= 1.3;  
47 }  
48 public void businessC() {  
49     mean *= 0.7;  
50 }  
51 public void businessD() {  
52     mean *= 3.2;  
53 }
```

## ForEach 로 입력값 지정하기

링크 <https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day10/src/QnaAnswerTest.java>

```
101 ▶ public class QnaAnswerTest {
102 ▶     public static void main(String[] args) {
103         // 역할과 책임(이 매서드는 이것을 수행한다는 - 역할이 명확해야하고
104         // 안전하게 작업이 진행된다는 책임이 분명해야함(데이터 무결성을 의미함)
105         int[] A = { 1, 2, 3 };
106
107         MeanTest mt = new MeanTest(A);
108         MeanTest mt2 = new MeanTest(A);
109
110     }
111 }
112
113 // new를 해서 객체가 만들어질때의 그림을 한 번 그려보겠습니다.
114
115 // ----- mt 객체
116 // | float mean; |
117 // | int[] scores; |
118 // | int length; |
119 // -----
120 // | MeanTest() |
121 // | calcMean() |
122 // | businessA() |
123 // | businessB() |
124 // | businessC() |
125 // | businessD() |
126 // -----
```

```
128 // 위의 정보는 오로지 mt 객체의 것임
129
130 // ----- mt2 객체
131 // | float mean; |
132 // | int[] scores; |
133 // | int length; |
134 // -----
135 // | MeanTest() |
136 // | calcMean() |
137 // | businessA() |
138 // | businessB() |
139 // | businessC() |
140 // | businessD() |
141 // -----
142
143 // 이 객체들은 서로 독립적이다!
144 // 나는 나, 너는 너!
```

Get : 리턴 타입이 있어야 한다.

Calc : 리턴 타입 없이 계산만 하는 매서드 사용시, 주로 사용한다.

- **역할과 책임** : 특정 매서드가 어떤 것을 수행한다 라는 역할이 명확해야 하고, 안전하게 작업이 진행된다는 책임이 분명해야 한다.
- New 를 통해 만들어진 객체들은 서로 독립적이다.

## ForEach + Scanner 케이스

링크 <https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day10/src/ForEachTest.java>

```
1  import java.util.Scanner;
2  class ArrPrintTest{
3      int [] arr;
4      int i = 0;
5
6
7  @ public ArrPrintTest(int [] inputArr) {
8
9      int len = inputArr.length;
10     arr = new int[len];
11     // int i = 0;
12     // 수업때는 i값이 한번만 있으면 오류가 안났는데
13     // 제가 코드 작성한 이후로 아래에도 i값을 주지않으면 오류가 나서 클래스에
14     // int i = 0; 를 선언했습니다.
15
16     for(int data : inputArr) {
17         arr[i++] = data;
18     }
19 }
20
21 public void scanPrint() {
22     Scanner scan = new Scanner(System.in);
23     System.out.print("몇 개를 입력하시겠습니까?");
24     int num = scan.nextInt();
25     // int i = 0;
26     arr = new int [num];
27
28     for(i = 0 ; i < num; i++) {
29         System.out.print("숫자를 입력하세요");
30         arr[i] = num;
31     }
32 }
33
```

```
34     public void printArr() {
35         for(int data : arr) {
36             System.out.println("입력값 = " + data);
37         }
38     }
39
40 }
41
42
43 public class ForEachTest2 {
44     public static void main(String[] args) {
45         int [] ap = {1, 2, 3, 4, 5, 6, 7};
46         ArrPrintTest apt = new ArrPrintTest(ap);
47         apt.printArr();
48
49         // ArrPrintTest apt2 = new ArrPrintTest();
50         // apt2.printArr();
51
52         // new ArrPrintTest()부분에서 ()안에 빨간줄 오류가 생기며
53         // 입력값을 받을 수 없습니다.
54         // 이유가 무엇일까요?
55
56     }
57 }
58
```

## class 배열 예제

링크 <https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day10/src/ClassArrayTest.java>

```
1  import java.util.Scanner;
2
3  class ScoresTest {
4      final int MAX = 5;
5
6      float sum;
7      float mean;
8      int randArr[];
9
10     public ScoresTest () {
11         // 5개의 배열을 만들고 랜덤값을 할당함
12         System.out.println("생성자 호출!");
13         sum = 0;
14         randArr = new int[MAX];
15
16         for (int i = 0; i < MAX; i++) {
17             randArr[i] = (int)(Math.random() * 50 + 50);
18         }
19     }
20
21     public void calcMean () {
22         for (int i = 0; i < MAX; i++) {
23             sum += randArr[i];
24         }
25
26         mean = sum / (float)MAX;
27     }
28
29     public int[] getRandArr() { return randArr; }
30
31     public float getSum() { return sum; }
32
33     public float getMean() { return mean; }
34
35     public int getMAX() { return MAX; }
36
37 }
38
39
40 }
```

// 시나리오 : 각 반에 5객체가 있다. 학급의 개수를 지정하여 학급 당 객체 별 합산값, 학급평균을 구한 후, 전체 학급의 평균을 구한다.

고정된 숫자는 전부 대문자로 표기하는 것이 관습이다.

final int MAX = 5;

Int randArr[] = new int [MAX] : heap배열 / []안에 개수

ScoresTest()

For문을 사용하여

randArr[0] ~ randArr[4]일때 값이 각각 랜덤 지정된다.

랜덤값은 (int)(Math.random() \* 50 + 50) 으로

randArr[i] = 50 이상 100미만 (50 ~99사이의 값)

calcMean()

Sum = sum + randArr[i]

Sum = 0 + randArr[0]

Sum = randArr[0] + randArr[1]

Sum = (randArr[0] + randArr[1] )+ randArr[2]

Sum = (randArr[0] + randArr[1] + randArr[2] ) + randArr[3]

Sum = (randArr[0] + randArr[1] + randArr[2] + randArr[3] )  
+ randArr[4]

따라서 sum은 총 합계가 되고, mean은 평균이 된다.

## class 배열 예제

링크 <https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day10/src/ClassArrayTest.java>

```
42 ▶ public class ClassArrayTest {
43 ▶     public static void main(String[] args) {
44         // 클래스 <==> 커스텀 데이터타입(우리가 커스텀하여 만들 수 있는 데이터타입)
45         ScoresTest st[];
46         Scanner scan = new Scanner(System.in);
47
48         System.out.print("몇 개의 학급이 있나요 ? ");
49
50         int num = scan.nextInt();
51
52         // 클래스 형식의 커스텀 데이터타입으로 만들어진 배열을 num 개수만큼 만듭니다.
53         // 그리고 st라는 변수명이 이 배열 메모리 공간을 관리합니다.
54         st = new ScoresTest[num];
55
56         float totalSum = 0;
57         float totalNumber = 0;
58
59         // -----
60         // | 객체1 | 객체2 | 객체3 | 객체4 | 객체5 |
61         // -----
62         // [0]   [1]   [2]   [3]   [4]
63
64         for (int i = 0; i < num; i++) {
65             st[i] = new ScoresTest(); // 이 부분을 통해 객체1, 객체2, ... 객체5가 생성됨
66             st[i].calcMean();
67
68             float tmpSum = st[i].getSum();
69             totalSum += tmpSum;
70             totalNumber += st[i].getMAX();
71             System.out.println("각 객체별 합산값 = " + tmpSum);
72             System.out.println("각 반의 평균 = " + st[i].getMean());
73         }
74
75         System.out.println("최종 계산된 전체 평균은 = " + (totalSum / totalNumber));
76     }
77 }
```

ScoresTest st[];

st = new ScoresTest[num]; 으로 따로 선언하였다.

(float) 실수 형태로 totalSum과 totalNumber 을 각각 지정.

st[i] = new ScoresTest()

i값은 int num = scan.nextInt()로 지정한 학급의 개수  
지정한 학급의 개수만큼 객체가 생성된다.

st[i].calcMean();

지정한 학급의 개수만큼 각 반의 랜덤점수 합산값, 각반의 평균값.

float tmpSum = st[i].getsum()

각 반의 합계값을 얻는다.

totalSum = totalSum + tmpSum

총 합계 = 0 + 각 반의 합계값

totalNumber = totalNumber + st[i].getMax()

totalNumber = 0 + i값(지정한 학급개수) 마다 MAX값



## 43번 : 클래스 배열 문제

링크 <https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day10/src/ClassArrayTest.java>

```
1 // 직원 10명에게 각각 이름을 부여함
2 // 랜덤연봉 2400 ~3500 , 연봉 인상률 1% ~20% 랜덤
3 // 10년 후 각 직원들의 연봉은?
4 // 연별로 평균 연봉값은?
5
6 class AnnualIncome {
7     // 직원 10명에게 알파벳 이름을 부여함
8     final String peopleName [] = {"A","B","C","D","E","F","G","H","I","J"};
9     final int MAX = 10;
10    int income;
11    int increase;
12    float sum;
13    float mean;
14
15    public void peopleIncome() {
16        for(int i =0; i < peopleName.length; i++) {
17            // 랜덤연봉 2400 ~ 3500
18            income = (int)(Math.random() * 1100 + 2400);
19            // 직원에 따른 랜덤 연봉 및 랜덤 연봉 인상률 출력은 가능하나,
20            // 그 값을 활용하여 평균값스 10년후 연봉들에
21            // income, increase를 사용하여 코드를 만들어내기 어렵습니다.
22            //
23            // income을 단순 int형으로 사용하여 재사용이 불가능한 것일까요?
24            // income[i]로 배열로서 income[0] = (int)(Math.random() * 1100 + 2400);
25            // 출력하여 재사용 하려 했으나
26            // 클래스에서 int income []; 선언 후,
27            // incomep[] = (int)(Math.random() * 1100 + 2400);를 사용해도 오류가 뜨는게
28            // 배열을 잘 적용하지 못하고, 내용에 따라 코드 작성을 하는데 어려움을 느낍니다.
29
30            // 연봉 인상률
31            increase = (int)(Math.random() * 20 + 1 );
32            System.out.printf("%s의 연봉은 %d만원이고, 매년 %d퍼센트 연봉이 인상된다.\n",
33                ,peopleName[i],income,increase);
```

```
38    public void tenYearIncome(){
39        // 10년 후 각 직원들의 연봉 :
40        // [0] = 랜덤초봉
41        // [1] = 랜덤 초봉 + (랜덤초봉 * 랜덤 연봉 인상률)
42        // [2] = [1] + ([1] * 랜덤 연봉 인상률)
43        // [3] = [2] + ([2] * 랜덤 연봉 인상률) ...
44        // [i] 가 9가 되었을때 각 직원들의 연봉
45
46        for(int i =0; i < MAX; i++ ) {
47        }
48    }
49
50
51    // 연별로 평균 연봉값
52    public void annualIncomeMean(){
53        for(int i = 0; i < MAX; i++ ) {
54            ;
55        }
56        mean = sum / (float) MAX;
57    }
58
59 }
60
61 public class Day10_Quiz43 {
62     public static void main(String[] args) {
63         AnnualIncome pi = new AnnualIncome();
64         pi.peopleIncome();
65
66         // pi.annualIncomeMean();
67     }
68 }
```