

[디지털 컨버전스]  
스마트 콘텐츠와  
웹 융합 응용 SW  
개발자 양성과정

10회차 수업  
2021/05/20 금요일

강사 : 이상훈  
학생 : 김원석

```
import java.math.BigInteger;

public class BigIntegerFibonacci {
    public static void main(String[] args) {
        final int Max = 5;
        BigInteger[] fibArr = new BigInteger[Max];
        // 여기서 final로 고정된 값을 정해준다.
        // BigInteger 사용하기 위해 문법을 불러온다.
        // new BigInteger[Max]; 여기서 배열안에 Max는 파이널 고정값을 말하는 것 같다.

        fibArr[0] = new BigInteger( val: "100");
        // <- 여기서 new의 이미지는 확실히 잘 모르겠지만, "100을 넣어준 의미는" 출력값을 보니 100자리 10진수를 뽑기 위한 것 같다
        fibArr[1] = BigInteger.ONE;
        // 결과적으로 배열 1인덱스 안에는 숫자 1이 들어간다는 뜻이다.
        for (int i = 2; i < fibArr.length; i++) {
            fibArr[i] = fibArr[i - 1].add(fibArr[i - 2]);
            System.out.println("fibArr[" + i + "] = " + fibArr[i]);
            // 설정값은 배열에 (인덱스) 2부터 돌아가고 거기에 값은 -1과 -2를 더한 값이라고 설정한다.
            // 출력은 i번 후위연산에 fibArr[i] 값을 컴파일 하게 한다.
        }

        BigInteger two = new BigInteger( val: "2");
        BigInteger verBigNum = new BigInteger( val: "1234567890");

        System.out.println("2 - 1234567890 = " + two.subtract(verBigNum));
        // two.subtract에서 two를 넣는 이유는 뭘까?
        // 위의 예제에선 .add할 때는 변수명이 없어서 그 앞을 안 쓴 것인가?
```

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

```
class ScoresTest {
```

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

```
    float sum;
```

```
    float mean;
```

```
    int randArr[];
```

```
    public int getMAX() { return MAX; }
```

```
    public float getSum() { return sum; }
```

```
    public float getMean() { return mean; }
```

```
    public int[] getRandArr() { return randArr; }
```

```
    public ScoresTest() {
```

```
        sum = 0;
```

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

```
        // 여기서 랜덤arr는 5개의 인덱스를 만드는데 사용하는것 같다.
```

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

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

```
            // 여기서 randArr i = 숫자 50부터 50까지의 값을 구해달라는 이야기이다.
```

```
        }
```

```
    }
```

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

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

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

```
        } mean = sum / (float)MAX; // 평균값을 구하는 오버로딩 함수..? 이렇게 표현해야 하는건가?
```

```
    }
```

```
}
```

```
public class ClassArrayTest {
```

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

```
        ScoresTest st[];
```

```
        Scanner scan = new Scanner(System.in);
```

```
        System.out.print("몇 학급이 있나? ");
```

```
        int num = scan.nextInt();
```

```
        st = new ScoresTest[num];
```

```
        float a = 0;
```

```
        float b = 0;
```

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

```
            st[i] = new ScoresTest(); //생성자를 호출하는거?
```

```
            st[i].calcMean();
```

```
            float tmpSum = st[i].getSum();
```

```
            a += tmpSum; // 각 학급에 합을 구하는 문법
```

```
            b += st[i].getMAX(); // 이건 뭐에 필요한지 모르겠다..
```

```
            System.out.println("각 객체별 합산 값 + " + tmpSum);
```

```
            System.out.println("각 반의 평균 = " + st[i].getMean());
```

```
        }
```

```
        System.out.println("최종 계산된 전체 평균은 = " + (a / b));
```

```
    }
```

```
}
```

```
import java.util.Scanner;

public class EqualsTest {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("말씀하세요 = ");
        String tell = scan.nextLine();
        if(tell.equals("네")) {
            System.out.println("알겠어요");
        } else if(tell.equals("아니오")){
            System.out.println("다시 알아보께요!");
        } else {
            System.out.println("잘못된 입력어 입니다.");
        }
    }
}

} // 이해가 되었다.
```

```

class ForEachTestClass {
    int arr[];
    Scanner scan;

    public ForEachTestClass(int[] input) {
        int len = input.length; // 집어놓을 데이터의 길이를 작성
        int i = 0; // i는 0부터 시작한다는 전제
        arr = new int[len]; // 배열 공식은 데이터를 입력하는것에 길이 일것이다.
        for (int data : input) {
            arr[i++] = data;
        }
    }

    public ForEachTestClass() {
        scan = new Scanner(System.in);
        System.out.print("몇 개를 입력하시겠습니까 ? ");
        int num = scan.nextInt();
        arr = new int[num];

        for (int i = 0; i < arr.length; i++) {
            System.out.print("입력할 값을 적어주세요: ");
            arr[i] = scan.nextInt();
        }
    }

    public void printArr() {
        for (int data : arr) {
            System.out.println("입력값= " + data);
        }
    }
}

public class ForEachTest {
    public static void main(String[] args) {
        ForEachTestClass a = new ForEachTestClass();
        a.printArr();
    }
}

```

위에서 입력했던 값을  
출력해주기 위해  
Foreach를 사용하여  
값을 표출

```

class Salary {
    int[] yearSalary;
    int len;
    char[] employees;
    public Salary () {
        // 일단 전체적인 layout을 구상해보자.
        // 직원 열명이 있다고 한다. 직원을 생성해보자!
        // 여기서 빨간색으로 error가 나고 있다. 왜 오류가 나는 걸까?

        char employees [] = {a, b, c, d, e, f, g, h, i, j};
    }

    // 연봉의 범위를 설정해보자
    public Salary(int moneyArr[]){
        len = moneyArr.length;
        yearSalary = new int[len];

        for (int i = 0; i < len; i++) {
            len = (int)(Math.random()* 1100 + 2400);
            yearSalary[i] = moneyArr[i];
        }

        // 연봉 인상률은 이렇게 설정하는것이 맞나?
        public void inCrease(){
            int inCrease = (int)(Math.random() * (20 + 1)/100);
        }
    }
}

```

```

public class SalaryTest {
    public static void main(String[] args) {
        //랜덤 연봉 적용을 해보자
        //어떤 회사에 직원이 10명 있다.
        //10명의 이름은 적당히 지어주도록 한다.
        //이들의 시작 연봉은 2400 ~ 3500 으로 랜덤하게 지정한다.
        //또한 연봉 인상률은 1% ~ 20% 사이의 랜덤값을 가지게 한다.
        //10 년후의 각 직원들의 연봉을 출력하도록 프로그래밍해보자!
        //또한 연별로 평균 연봉값을 계산해보도록 한다.

        Salary num = new Salary();
    }
}

```

- 43번 클래스 배열 문제