



# ( 디지털 컨버전스 ) 스마트 콘텐츠와 웹 융합 응용 SW개발자 양성과정

훈련기간 : 2021.05.07 ~ 2021.12.08

# 객체배열

## 클래스 ScoresTest

ScoresTest생성자:  
5개의 배열을 만들고 50~100의랜덤값을 할당한다

calcMean메서드 :  
배열의 각 값을 더해서 sum변수에 저장하고 mean이라는 평균 변수를 저장한다.

## 클래스 ClassArrayTest

ScoresTest st[] 객체 배열을 입력값만큼 생성한다

ScoresTest st[];  
st[i] = new ScoresTest() : 각 객체 생성  
st[i] = calcMean() : 각 평균값 호출

객체배열을통해 각 평균값과 합산값 출력

```
import java.util.Scanner;

class ScoresTest {
    final int MAX = 5;

    float sum;
    float mean;
    int randArr[];

    public ScoresTest () {
        // 5개의 배열을 만들고 랜덤값을 할당함
        System.out.println("생성자 호출!");
        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; }
    public float getSum() { return sum; }
    public float getMean() { return mean; }
    public int getMAX() { return MAX; }
}
```

```
public class ClassArrayTest {
    public static void main(String[] args) {
        // 클래스 <=== 커스텀 데이터타입(우리가 커스텀하여 만들 수 있는 데이터타입)
        ScoresTest st[];
        Scanner scan = new Scanner(System.in);

        System.out.print("몇 개의 학급이 있나요 ? ");

        int num = scan.nextInt();

        // 클래스 형식의 커스텀 데이터타입으로 만들어진 배열을 num 개수만큼 만듭니다.
        // 그리고 st라는 변수명이 이 배열 메모리 공간을 관리합니다.
        st = new ScoresTest[num];

        float totalSum = 0;
        float totalNumber = 0;

        // -----
        // | 객체1 | 객체2 | 객체3 | 객체4 | 객체5 |
        // -----
        //   [0]   [1]   [2]   [3]   [4]

        for (int i = 0; i < num; i++) {
            st[i] = new ScoresTest(); // 이 부분을 통해 객체1, 객체2, ... 객체5가 생성됨
            st[i].calcMean();

            float tmpSum = st[i].getSum();
            totalSum += tmpSum;
            totalNumber += st[i].getMAX();
            System.out.println("각 객체별 합산값 = " + tmpSum);
            System.out.println("각 반의 평균 = " + st[i].getMean());
        }

        System.out.println("최종 계산된 전체 평균은 = " + (totalSum / totalNumber));
    }
}
```

# 무한 정수 클래스

주의 : 상수의 경우는 대문자로 표기해야 한다.

BigInteger는 무한정수를 구현한 데이터 타입이다.

## BigInteger클래스 내부 메소드

1. 뺄셈은 subtract()
2. 곱셈은 multiply()
3. 나눗셈은 divide()
4. 나머지연산은 remainder()

BigInteger에서는 add 매서드를 통해 연산을 해야한다.

출력값 :  
subtract메소드로 veryBigNum의 값을 2 뺀.

원리 : int + int + int + int 필요할때마다 계속 동적할당해서 추가한다.

```
public class BigIntegerFibonacci {
```

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

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

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

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

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

```
            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));
```

```
    }
```

## 출력값

```
fibArr[2] = 101
```

```
fibArr[3] = 102
```

```
fibArr[4] = 203
```

```
피보나치 수열의 n번째항은 = 203
```

```
2 - 2374923749237482384238482 = -237492374923748238480
```

# ForEach문

배열에서 하나씩정수를 받아와 출력한다.

ForEachTestClass메서드에서 배열을 입력받아  
해당 배열을 ForEach문을 통해 arr 배열에  
입력  
받는다.

printArr메서드에서  
입력받은 값은 하나씩 출력한다.

```
class ForEachTestClass {  
    int[] arr;  
    Scanner scan;  
  
    public ForEachTestClass (int[] inputArr) {  
        int len = inputArr.length;  
        int i = 0;  
  
        arr = new int[len];  
  
        for (int data : inputArr) {  
            arr[i++] = data;  
        }  
    }  
  
    public void printArr () {  
        for (int data : arr) {  
            System.out.println("입력값 = " + data);  
        }  
    }  
  
    public class ForEachTest {  
        public static void main(String[] args) {  
            int[] testArr = { 1, 2, 3, 4, 5 };  
  
            ForEachTestClass fetc = new ForEachTestClass(testArr);  
            fetc.printArr();  
        }  
    }  
}
```

## 출력값

입력값	=	1
입력값	=	2
입력값	=	3
입력값	=	4
입력값	=	5

# 객체 배열 문제

- 1. 랜덤 연봉 적용을 해보자
- 2. 10명의 이름 설정
- 3. 이들의 시작 연봉은 2400 ~ 3500 으로 랜덤지정
- 4. 연봉 인상률은 1% ~ 20% 사이의 랜덤설정
- 5. 10 년후의 각 직원들의 연봉을 출력
- 6. 연별로 평균 연봉값을 계산해보도록 한다.

1번째 작성했을때 사용할 변수를 설정하지않고 무작정 코드만 적었다.  
클래스변수가 너무 많이 작성되서인지 도중에 헛갈려서 의식의 흐름대로 작성되었고 결과적으로 출력값에서 오류가발생했다.  
헛갈리는 부분에 주석 조작 적지않고 생성자에 다 만들어서 수정이 어려웠다.

2번째 작성했을때 이해를하고 작성해서 빠르게 작성됐다 클래스 변수를 조금 줄이긴 했는데 메서드 분리가 적어서 코드가 오히려 길어졌다.

## 1번째 작성

```
class Salaryman {
    float randomMoney;
    float random_Rate;
    float random_RateMoney;
    float toTal = 0;
    float arrRecord[] = new float[10];
    float sum = 0;

    public Salaryman() {
        RandomBox();
        toTal = randomMoney;
        System.out.printf(" 1년차 연봉 : %.0f \n",randomMoney);
        arrRecord[0] = randomMoney;
        sum +=randomMoney;

        for (int i = 1; i < 10; i++) {
            Random_Rate();
            toTal += randomMoney*random_Rate;
            System.out.printf( i + 1 +" 년차 인상률 : %.1fP  연봉 : %.0f 만원 \n ", (random_Rate*100), toTal);

            sum += toTal; //합계 저장
            arrRecord[i] = toTal; // 배열에 저장
        }

        //랜덤 조봉 2400~3500
        public void RandomBox() { randomMoney = (float)(Math.random() *1100)+ 2400; }

        //랜덤 인상을 범위 1~20%
        void Random_Rate() {
            random_Rate = ((float)(Math.random() * 19 + 1))/100;
        }

        public float[] getArrRecord() {
            return arrRecord;
        }

        public float getSum() {
            return sum;
        }

        public float getToTal() {
            return toTal;
        }
    }
}
```

## 2번째 작성

```
class Money {
    float sum = 0;
    int random_Money;
    float random_rate;
    int year_Money[] = new int[10];

    //생성자
    public Money(){
        Random_Money();
        Random_rate();
        Constructor();
    }

    //랜덤 생성
    public void Constructor() {
        System.out.printf("1 년차 연봉 : %d만원\n", random_Money);
        year_Money[0] = random_Money;
        sum += random_Money;

        for (int i = 1; i < year_Money.length; i++) {
            int sum_Year = (int)(random_Money*random_rate)+random_Money; //연봉 인상금액
            sum += sum_Year; // 연봉 합계에 넣는 변수
            System.out.printf((i+1) +" 년차 연봉 : %d만원\t 연봉인상률 : %.1fP\t 인상액: %d만원 \n"
                ,sum_Year, random_rate*100,(int)(random_Money*random_rate)); //출력 부분
            Random_rate(); // 랜덤 인상을 생성
            random_Money =sum_Year; //N년차으로 연봉 초기화
            year_Money[i] = sum_Year; //N년차 연봉 배열에 저장
        }

        System.out.printf("연봉 10년간 평균 : %.0f만원\n",sum/year_Money.length);

        //인상률
        float Random_rate() {
            random_rate = (float)(Math.random() * 20 + 1)/100;
            return random_rate;
        }

        //랜덤 돈
        int Random_Money() {
            random_Money = (int) (Math.random() * 1100 + 2400);
            return random_Money;
        }

        public float getSum() {
            return sum;
        }

        public int[] getYear_Money() {
            return year_Money;
        }
    }
}
```

1~9년차 출력값은 길어서 잘랐습니다.

1번째 출력 값

이름 : 문씨	
1년차 연봉	: 2659
2년차 인상률	: 5.1P 연봉 : 2793만원
3년차 인상률	: 4.3P 연봉 : 2909만원
4년차 인상률	: 17.4P 연봉 : 3371만원
5년차 인상률	: 5.4P 연봉 : 3515만원
6년차 인상률	: 11.4P 연봉 : 3817만원
7년차 인상률	: 7.7P 연봉 : 4022만원
8년차 인상률	: 18.0P 연봉 : 4500만원
9년차 인상률	: 5.9P 연봉 : 4657만원
10년차 인상률	: 1.4P 연봉 : 4694만원
평균 연봉	: 3694만원
사원 10년 평균 연봉	: 4285만원
조회하고자하 하는 년도 평균 연봉	: 5
5년차 평균 연봉 조회	: 4219만원

오류

2년차 출력값 = 2793만원  
2년차 2659X0.051+2659 = 2794.609만원

3년차 출력값 = 2909만원  
3년차 2793X0.043+2793=2913.099만원

나머지 평균 연봉 값은 맞지만 연봉값만 다르다.

1번째 코드작성에서 형변환을 잘못된 부분이있기때문이다.

질문 : 소수점을 반올림하는 방법이 궁금합니다.

1번째 작성

```
public class Quiz {
    public static void main(String[] args) {
        Salaryman ss[] = new Salaryman[10];
        float sum;
        float tmpSum = 0;
        float PeopleSum = 0;
        int totalYear = 0;
        float ts_sum = 0;
        float selectSum = 0;
        String[] Name = {"김씨","이씨","박씨","전씨","신씨","서씨","고씨","주씨","장씨","문씨"};
        //사원 10명 평균연봉 출력
        for (int i = 0; i < 10; i++) {
            System.out.printf(" 이름 : %s\n\n",Name[i]);

            ss[i] = new Salaryman();
            System.out.println();

            sum = ss[i].getSum();
            tmpSum += sum;
            totalYear += ss[i].arrRecord.length;
            PeopleSum += tmpSum;

            System.out.printf(" 평균 연봉 : %.0f만원 \n\n", tmpSum/10);
            tmpSum= 0;
        }
    }
}
```

```
//사원 10명의 10년 평균 연봉
System.out.printf(" 사원 10년 평균 연봉 : %.0f만원\n\n",PeopleSum/ totalYear);
```

```
//년도 별 연봉조회
Scanner sc = new Scanner(System.in);
System.out.printf(" 조회하고자하 하는 년도 평균 연봉 : ");
int num = sc.nextInt();
for (int i = 0; i < 10; i++) {
    selectSum = ss[i].arrRecord[num-1];
    ts_sum += selectSum;
}
System.out.printf(" %d년차 평균 연봉 조회 : %.0f만원" ,num,ts_sum/10);
```

2번째 작성

```
public class Salary {
    public static void main(String[] args) {
        //이름 생성
        String r_Name[] = {"김", "이", "박", "전", "오", "서", "임", "마", "신", "왕"};
        Money mo[] = new Money[r_Name.length];
        float sum = 0;
        int total_Year = 0;
        float selectSum = 0;

        for (int i = 0; i < r_Name.length; i++) {
            System.out.printf("이름 : %s\n", r_Name[i]); //이름
            mo[i] = new Money(); //클래스 배열 생성
            total_Year += mo[i].year_Money.length; // 연도 수 총합
            sum += mo[i].getSum(); //모든 연봉 합계
        }

        System.out.printf("모든 사원 10년 평균 연봉 : %.0f만원 \n", sum/total_Year);

        int j = 0;
        int cnt = 0;
        while (j < 10) {
            for (int i = 0; i < 10; i++) {
                selectSum += mo[i].year_Money[j]; //모든 사원 N년차 연도 평균연봉 출력
            }
            cnt++; // 5개 출력이후에 엔터
            if (cnt == 6) {
                System.out.println();
            } System.out.printf(" "+(j + 1) + "년차 평균 연봉 : %.0f", selectSum / mo.length); //N년차 연봉 출력
            selectSum = 0; //N년차 연봉 초기화
            j++;
        }
    }
}
```

2번째 출력 값

이름 : 왕

1년차 연봉	: 3172만원				
2년차 연봉	: 3594만원	연봉인상률	: 13.3P	인상액	: 422만원
3년차 연봉	: 4161만원	연봉인상률	: 15.8P	인상액	: 567만원
4년차 연봉	: 4693만원	연봉인상률	: 12.8P	인상액	: 532만원
5년차 연봉	: 5378만원	연봉인상률	: 14.6P	인상액	: 685만원
6년차 연봉	: 5892만원	연봉인상률	: 9.6P	인상액	: 514만원
7년차 연봉	: 6911만원	연봉인상률	: 17.3P	인상액	: 1019만원
8년차 연봉	: 7877만원	연봉인상률	: 14.0P	인상액	: 966만원
9년차 연봉	: 8541만원	연봉인상률	: 8.4P	인상액	: 664만원
10년차 연봉	: 9948만원	연봉인상률	: 16.5P	인상액	: 1407만원

연봉 10년간 평균 : 6017만원

모든 사원 10년 평균 연봉 : 5502만원

1년차 평균 연봉	: 3127	2년차 평균 연봉	: 3574	3년차 평균 연봉	: 4028	4년차 평균 연봉	: 4545	5년차 평균 연봉	: 4976
6년차 평균 연봉	: 5463	7년차 평균 연봉	: 6073	8년차 평균 연봉	: 6930	9년차 평균 연봉	: 7730	10년차 평균 연봉	: 8577

정상

4년차 출력값 = 4693만원

4년차 연봉:3594\*0.15.8 = 532만원  
532 +4161 = 4693만원

## 메서드 연습

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

## 출력값

```
=====
1번 : (u_b)
2번 : (*^v^*)
3번 : (^ v ^)
=====
캐릭터를 선택해주세요 :
1
당신의 캐릭터
1번 : (u_b)
컴퓨터 체력 설정하기 :
20
사람 체력 설정하기 :
20
진행하려면 아무키나 입력 :
컴퓨터의 공격턴
첫번째 숫자 : 6, 두번째 숫자 : 6
*****데미지 두배***** 24
유저의 남은체력 : -4
[999]===== (u_b);;
진행하려면 아무키나 입력 :
유저의 공격턴
첫번째 숫자 : 1, 두번째 숫자 : 5
컴퓨터의 남은체력 : 14
(u_b)===== [^--];;
유저가 져요
```

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

//로또 번호는 총 4개입니다 중복값도 발생하며 범위는 1~9 입니다.
//메서드로 나눠서 작성
```

## 출력값

```

로또 번호는 총 4개입니다 중복값도 발생하며 범위는 1~9 입니다.
자재로 번호를 입력하세요
1
자재로 번호를 입력하세요
2
자재로 번호를 입력하세요
3
자재로 번호를 입력하세요
4
입력한번호 : 1 2 3 4
[1, 2, 3, 4]
번호를 수정하시겠습니까? (Y/N) :
y
당신의 번호입니다.
[1, 2, 3, 4]
바꿀 열의 번호 : 4
바꿀 번호 : 2
[1, 2, 3, 2]
번호를 수정하시겠습니까? (Y/N) :
n
[1, 2, 3, 2]당신의 번호입니다.
추첨 로또번호는 :
1 번째 번호는 = 1
2 번째 번호는 = 4
3 번째 번호는 = 4
4 번째 번호는 = 5
이번 당첨 번호 입니다.[1, 4, 4, 5]
5만원

```

메서드 쪼개는 연습을했다.  
코드는 대부분 `System.out.print`로 구성되었다.

메서드로 기능별로 나눠서 하니깐 오류발생했을때 어디일 것 같다는 느낌이 직관적으로오고 재활용도가 높아진다.