

(디지털컨버전스) 스마트 콘텐츠와 웹 융합 응용 SW개발자 양성과정

-12일차 학습 및 질문 노트-

강사 - Innova Lee(이상훈)
gcccompil3r@gmail.com
학생 - Kyeonghwan Lee(이경환)
airtrade7@naver.com

▣ 생성자 호출

```

13 // Q: 생성자는 무조건 main()에서만 호출이 가능한가요 ?
14 // A: 어디서나 가능합니다.
15 class Students {
16     String name;
17     int score;
18
19     public Students (String name) {
20         score = (int)(Math.random() * 40 + 61);
21         this.name = name;
22     }
23
24     public String getName() {
25         return name;
26     }
27     public int getScore() {
28         return score;
29     }
30 }
31
32 class School {
33     final int MAX = 3;
34     final String[] names = {"아녀영", "안녀영", "안뇨웅"};
35     Students[] arr;
36
37     public School () {
38         arr = new Students[MAX];
39
40         for (int i = 0; i < MAX; i++) {
41             arr[i] = new Students(names[i]);
42         }
43     }
44     public void printStudentsInfo () {
45         for (int i = 0; i < MAX; i++) {
46             System.out.printf("학생 이름은 %s, 점수는 %d 입니다.\n",
47                 arr[i].getName(), arr[i].getScore());
48         }
49     }
50 }
51
52 public class ConstructorCallTest {
53     public static void main(String[] args) {
54         School school = new School();
55
56         school.printStudentsInfo();

```

생성자는 어디서든 호출이 가능하다

학생 이름은 아녀영, 점수는 67 입니다.
 학생 이름은 안녀영, 점수는 94 입니다.
 학생 이름은 안뇨웅, 점수는 68 입니다.

Process finished with exit code 0

▣ Prob43(Scanner 사용)

```

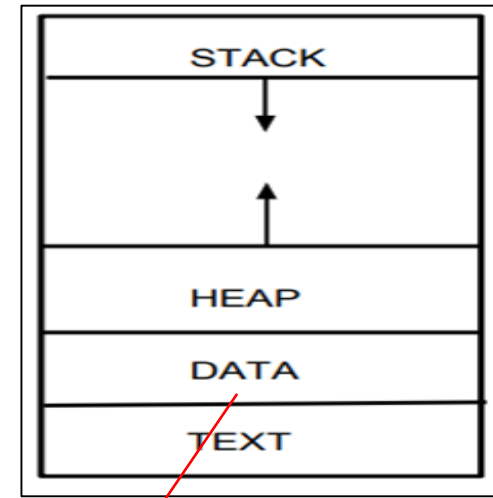
23 class Employee {
24     final float PERCENT = 0.01f;
25
26     String name;
27     int[] pay;
28     float[] incRate;
29
30     int curYear;
31
32     public Employee (String name, int year) {
33         this.name = name;
34         curYear = 0;
35
36         pay = new int[year];
37         incRate = new float[year];
38
39         pay[0] = (int)(Math.random() * 1101 + 2400); //초봉급 설정
40     }
41
42     // Convention(규약)
43     // 나중에 팀 프로젝트 할 때도 팀간에 컨벤션 설정이 중요
44     // 전부 다 가져오는 것은 표기를 어떻게 하자
45     // 날짜로 가져오는 것은 표기를 어떻게 하자
46     // 위와 같은 규약에 따라 그냥 지금 생각나는대로 작성한 부분
47     // calc_종류_변수 : 변수의 속성에 해당하는 종류 전체값을 전부 처리할
48     // calcYearPay : pay 변수의 속성 year에 해당하는 전체값을 모두 처리
49     // 이 규약은 만들기 나름이고 팀 프로젝트를 하는 경우엔 반드시 필요함
50     // (왜냐하면 내가 만든 매서드를 다른 사람이 사용할 수도 있기 때문)
51     public void calcYearPay () { //모든 년도의 합
52         for (int i = curYear + 1; i < pay.length; i++) {
53             pay[i] = pay[i - 1] + (int)(pay[i - 1] * getRandomIncrate() * PERCENT);
54         }
55     }
56     public void calcPay () { //한해만 계산
57         pay[curYear + 1] = pay[curYear] + (int)(pay[curYear] * getRandomIncrate() * PERCENT);
58         curYear++;
59     }
60     public void printPay () {
61         for (int i = 0; i < pay.length; i++) {
62             System.out.printf("%s님의 %d년차 연봉은 %d입니다.\n", name, i + 1, pay[i]);
63         }
64     }
65
66     public float getRandomIncrate () {
67         return (float)(Math.random() * 20 + 1);
68     }
69
70     public int[] getPay() {
71         return pay;
72     }
73     public int getYearPay(int year) {
74         return pay[year];
75     }

```

```

78 public class ScannerClassRoomTest {
79     // static은 정적 변수로 메모리 레이아웃으로 치면 data 섹션에 잡힘
80
81     // -----
82     // | Stack |
83     // -----
84     // | Heap |
85     // -----
86     // | Data | <----- 이 영역에 잡히는 것은 전역 변수면서 프로그램 수행할
87     // -----
88     // | Text(Code) |
89     // -----
90
91     // 그렇기 때문에 static 으로 선언된 녀석들은 new가 없어도 사용이 가능한 것임
92     // 대표적으로 ? main() 매서드가 new 없이 사용했었고 static이 붙어있음
93     static final Scanner scan = new Scanner(System.in);
94
95     public static String inputName () {
96         System.out.print("학생 이름을 입력하세요: ");
97         return scan.nextLine();
98     }
99     public static int inputYear () {
100         Boolean isTrue = false;
101         int num;
102
103         do {
104             System.out.print("몇 년치를 계산할까요 ? ");
105             // return scan.nextInt();
106             // 예외 상황으로 인해
107             num = scan.nextInt();
108
109             if (num <= 0) {
110                 System.out.println("0 보다 큰 숫자를 입력해주세요!");
111                 isTrue = true;
112             }
113         } while (isTrue);
114
115         return num;
116     }

```



Static(정적변수)

-데이터 영역에 들어가는것은 전역 변수로서 어디서든 호출하면 사용이 가능하다.
-대표적으로 main()메서드가 static으로 new 없이 사용하고있다.

Static 매서드의 대표 예

굳이 특정한 클래스에 소속될 필요없이 필요하면 사용하는 경우를 유틸리티 매서드라고 하는데 이런 유틸리티 매서드들은 static매서드로 만들어서 사용한다. ex) Math.pow

do~while문은 while문과 비교해서 조건과 상관 없이 무조건 한 번 작업을 수행한다. 그 다음 조건이 참(true)이면 반복해서 작업을 수행 한다.

▣ Prob43(Scanner 사용)

```

public static float[] calcEmployeeMean(Employee[] emp) { // 클래스 배열을 가져온다
    int[] pay; // 배열을 리턴하고자 한다면 리턴 타입에 (데이터타입[])을 적는다.
    int length = emp[0].getPay().length;
    float mean[] = new float[length]; // 길이값만큼 평균 배열을 만듦

    // 몇 년간
    for (int i = 0; i < length; i++) {
        float sum = 0;

        for (int j = 0; j < emp.length; j++) {
            // j 번째 사람의 i 년차 연봉
            sum += emp[j].getYearPay(i);
        }

        mean[i] = sum / (float)emp.length;
    }

    return mean;
}

public static void main(String[] args) {
    final int YEAR = 10;

    System.out.print("몇 명의 학생 정보를 입력할까요 ? ");
    int studentNum = scan.nextInt();
    scan.nextLine();

    Employee[] emp = new Employee[studentNum];

    for (int i = 0; i < studentNum; i++) {
        emp[i] = new Employee(inputName(), YEAR);

        emp[i].calcYearPay();
        emp[i].printPay();
    }

    float[] mean = calcEmployeeMean(emp);

    for (int i = 0; i < mean.length; i++) {
        System.out.printf("%d년차 평균 연봉은 %f\n", i + 1, mean[i]);
    }
}

```

몇 명의 학생 정보를 입력할까요 ? 2

학생 이름을 입력하세요: 안

안님의 1년차 연봉은 3442입니다.

안님의 2년차 연봉은 3482입니다.

안님의 3년차 연봉은 4120입니다.

안님의 4년차 연봉은 4781입니다.

안님의 5년차 연봉은 5420입니다.

안님의 6년차 연봉은 6101입니다.

안님의 7년차 연봉은 6905입니다.

안님의 8년차 연봉은 7556입니다.

안님의 9년차 연봉은 7819입니다.

안님의 10년차 연봉은 8960입니다.

학생 이름을 입력하세요: 냉

냉님의 1년차 연봉은 2691입니다.

냉님의 2년차 연봉은 3238입니다.

냉님의 3년차 연봉은 3452입니다.

냉님의 4년차 연봉은 3751입니다.

냉님의 5년차 연봉은 3983입니다.

냉님의 6년차 연봉은 4137입니다.

냉님의 7년차 연봉은 4567입니다.

냉님의 8년차 연봉은 4940입니다.

냉님의 9년차 연봉은 5964입니다.

냉님의 10년차 연봉은 6155입니다.

1년차 평균 연봉은 3066.500000

2년차 평균 연봉은 3360.000000

3년차 평균 연봉은 3786.000000

4년차 평균 연봉은 4266.000000

5년차 평균 연봉은 4701.500000

6년차 평균 연봉은 5119.000000

7년차 평균 연봉은 5736.000000

8년차 평균 연봉은 6248.000000

9년차 평균 연봉은 6891.500000

10년차 평균 연봉은 7557.500000

Process finished with exit code 0

▣ Prob45(챌린지문제)

```
import java.util.Scanner;
// 1. 나와 컴퓨터 모두 1000만원을 가지고 있다.
// 2. 판돈을 걸 수 있고 배팅 비율은 Scanner를 통해 입력한다.
// 3. 둘중 1000만원 값이 0이 되는 경우에 게임이 종료된다.//or 연산을 통해 0값이 발생하면 게임종료
// 4. 주사위는 2개를 사용한다.
// 5. 숫자가 높은 사람이 이긴다.
// 6. 첫번째 주사위가 짝수가 나온경우에만 두 번째 주사위를 굴릴 수 있다.
// 7. 두번째 주사위는 1이 나오면 +3이된다./ 3이 나오면 상대방 주사위값을 -2 할 수 있다.
// 숫자 4가 나오면 내 주사위 값이 0이 된다.
// 숫자 6이 나오면 각자 만든 주사위값을 2배로 갖는다.(두배 적용의 경우 첫번째+두번째 주사위 모두 합산)
class Casino {
    final int myMoney = 1000;
    final int comMoney = 1000;
    int comDice1;
    int myDice1;
    int comDice2;
    int myDice2;

    Scanner scan;

    public Casino() {
        comDice1 = getRandDice();
        comDice2 = getRandDice();
        myDice1 = getRandDice();
        myDice2 = getRandDice();
    }

    public int getRandDice() {
        return (int) (Math.random() * 6 + 1);
    }

    public void Dicegame(){
        if(myDice1 % 2 != 0 && comDice1 % 2 != 0){
            if(myDice1 > comDice1) {
                System.out.println("당신의 승리입니다");
            }else if(myDice1 < comDice1) {
                System.out.println("패배 하였습니다.");
            }else{
                System.out.println("비겼습니다.");
            }
        }
    }

    public class Prob45 {
        public static void main(String[] args) {
            Casino cn = new Casino();
        }
    }
}
```

이것 저것 구현해야 되는게 많아서 어디서부터 손을 대서 풀어 나가야 될지 너무 어렵네요. 풀이 해주시면 복습 열심히 하도록 하겠습니다.