

# 수업 복습 노트

강사 : 이상훈  
학생 : 여인준

-11일차 수업-

## 수업 복습(피드백 적용)

```
3  class Employee{
4
5      Scanner scan;
6      private String name;
7      private int pay;
8      //상승한 연봉을 담을수 있는 배열 만들고 -> 상승한 연봉은 배열에 저장
9      int[] increasedPay;
10     //연봉상승률을 담을수 있는 배열을 만든다.
11     int[] increasedRatio;
```

```
19     public void setYearArr(int year){
20         increasedPay = new int[year];
21         increasedRatio = new int[year];
22     }
```

1. 연봉 상승률과 증가된 연봉을 담을 배열을 Employee 클래스에 객체변수로 추가했다.
2. 배열의 크기를 정하는 year은 다른 클래스에서 할당하기때문에 메소드화 했다.

## 수업 복습(피드백 적용)

```
33 class PayCalculator{
34     double increase;
35     int randRatio;
36     int year;
37
38     Employee[] em;
39
40
41     PayCalculator(Employee[] em,int year){
42         this.em = em;
43         this.year = year;
44     }
45 }
```

1. 연봉상승률(randRatio)과 몇년에 관해 계산을하는지에 대한 연도(year) 변수를 PayCalculator객체변수로 추가했다.
2. 생성자에서 연도 변수에 관한 값을 받아서 초기화한다.

## 수업 복습(피드백 적용)

```
47 //직원들 1년 인상률과 상승한 연봉, 연봉상승률 배열에 담기
48 private void employeesIncrease(){
49     for(int i=0;i<em.length;i++){
50         //생성자에서 받은 year로 연봉상승 배열생성
51         em[i].setYearArr(year);
52         //연봉상승률과 상승률을 연봉상승률 배열에 담음
53         randRatio =(int)(Math.random()*20+1);
54         em[i].increasedRatio[i] = randRatio;
55         //연봉증가량과 증가량을 증가된연봉 배열에 담음
56         increase = em[i].getPay()*(randRatio*0.01);
57         em[i].setPay(em[i].getPay()+(int)increase);
58         em[i].increasedPay[i] = em[i].getPay();
59         //출력리
60         System.out.printf("%s의 연봉률 상승폭은 %d%%이며 상승된 연봉은 %d 입니다.\n",em[i].getName(),em[i].increasedRatio[i],em[i].increasedPay[i]);
61     }
62 }
```

Employee 클래스에서 만든 연봉상승에 관한 배열을 employeesIncrease메소드에서 처리하도록 했다.  
EmployeesIncrease메소드에서는 각 직원마다 연봉상승에 관한 계산을 수행하는 메소드이다.

- 1.PayCaluculator생성자에서 초기화한 year로 배열의 크기를 정한다.
- 2.객체변수 randRatio값(연봉상승률)을 정의하고, 직원의 연봉상승률 배열(increaseRatio)에 넣는다.
- 3.increase(상승한연봉)값을 계산하고, set메소드로 상승한 연봉값을 직원의 연봉에 할당한다음, get메소드로 상승한 연봉값을 직원의 상승한연봉 배열(increasePay)에 넣는다.

## 수업 복습(피드백 적용)

```
총 몇명의 직원이 있습니까 ? 3
직원 이름을 입력해주세요 : 레이너
레이너님의 연봉은 2776
직원 이름을 입력해주세요 : 제라툴
제라툴님의 연봉은 3076
직원 이름을 입력해주세요 : 케리건
케리건님의 연봉은 2811
0년차 연봉 평균은2887
레이너의 연봉률 상승폭은 2%이며 상승된 연봉은 2831 입니다.
제라툴의 연봉률 상승폭은 14%이며 상승된 연봉은 3506 입니다.
케리건의 연봉률 상승폭은 18%이며 상승된 연봉은 3316 입니다.
1년차 연봉 평균은3217
레이너의 연봉률 상승폭은 7%이며 상승된 연봉은 3029 입니다.
제라툴의 연봉률 상승폭은 15%이며 상승된 연봉은 4031 입니다.
케리건의 연봉률 상승폭은 8%이며 상승된 연봉은 3581 입니다.
2년차 연봉 평균은3547
레이너의 연봉률 상승폭은 11%이며 상승된 연봉은 3362 입니다.
제라툴의 연봉률 상승폭은 9%이며 상승된 연봉은 4393 입니다.
케리건의 연봉률 상승폭은 3%이며 상승된 연봉은 3688 입니다.

Process finished with exit code 0
```

정상작동 되는걸 확인.

## 복습 후 느낀점

### 느낌점.

1.기능에 따라 클래스에 적절히 변수나 메소드를 배치하는 작업이 생각보다 많이 어렵구나 생각했다.

처음 연봉상승률(increasedRatio)배열 과 상승된연봉(increasedPay)배열 변수를 PayCalculator클래스 즉, 계산에 관련된 클래스에 추가했는데 코드를 작성해보니 개개인 직원에 관한 연봉 배열이 작성이 안돼서 코드진행이 안됐다.

생각을 잠시 해보니 연봉에관한 정보는 계산에 관련된 클래스가 아니라, 직원 개개인이 가지고 있는게 옳다고 판단해 Employee클래스에 추가했다. 생각보다 훨씬 더 객체 관련프로그램은 현실세계와 닮았다고 생각했다. 클래스의 특성과 역할을 정의하는게 정말로 중요한것같다.