

2022.1.3 인지연

클래스 문제 1 배열

```
public class ProbBank4Ans1_2 {  
    public static void main(String[] args) {  
        // 2 by 2 이중 배열을 초기화해서 아무 값이나 넣어보세요.  
  
        MyArray myArray = new MyArray();  
        myArray.initMyArray(2);  
        myArray.printArray();  
    }  
}
```

문제, 이중배열로 값 출력하기

메인 클래스 값.
MyArray 클래스 객체 생성하여,
매소드 initMyArray에 2을 입력.
또 매소드 printArray를 통해 값을 출력하
도록 했다.

MyArray 클래스를 보러 가보자.

```

public class MyArray {
    int dimension;

1) final int D2 = 2;
    final int D3 = 3;

    MatrixD2 matD2;
    MatrixD3 matD3;

2) public void initMyArray (int dim) {
    dimension = dim;

    if (dim == D2) {
        matD2 = new MatrixD2();
        matD2.initMatrix();
    } else if (dim == D3) {
        matD3 = new MatrixD3();
        matD3.initMatrix();
    }
}

3) public void printArray () {
    if (dimension == D2) {
        matD2.printMatrix();
    } else if (dimension == D3) {
        matD3.printMatrix();
    }
}
}

```

MyArray클래스로 들어갔다.

1) 필요한 변수를 입력한다.

2) 아까 메인 클래스에서 입력했었던 initMyArray가 보인다.
그 안에 int dim 을 만들어서 값을 입력할 수 있게 한다.
Dim은 dimension과 동일하다. 라고 만들어놓고,
Dim =d2(즉 2)의 값이냐?? 라고 물어봤을때 그 값이 맞다면

matD2 = MatrixD2라는 객체를 또 만들어버렸다.
그 객체 안에서 initMatrix의 값을 끌어온다.

3) printArray 를 보자.

만약 dim =D2라면 또 MatrixD2여기 클래스로 들어가서
PrintMatrix를 출력해라고 한다.

어떻게따지면 비슷해보이는데
이렇게 각각의 값을 매소드로 넣으려고 하는걸까?
솔직히 정확하게는 잘 모르겠다.

자 이제 그럼, MatrixD2 요기 클래스로 한번 들어가보자

```

public class MatrixD2 {
1) final int RANGE = 10;
   final int DIM = 2;

   private int[][] mat;

2) public void initMatrix () {
   mat = new int[DIM][DIM];

   System.out.println("행렬 생성");

   for (int i = 0; i < DIM; i++) {
       for (int j = 0; j < DIM; j++) {
           mat[i][j] = (int) (Math.random() * RANGE);
       }
   }
}

3) public void printMatrix () {
   // 처음 루프 i = 0 -> 1
   for (int i = 0; i < DIM; i++) {
       // 루프 j = 0
       for (int j = 0; j < DIM; j++) {
           System.out.printf("%3d", mat[i][j]); // [0][0], [0][1]
           // x   y           [1][0], [1][1]
       }
       System.out.println();
   }
   System.out.println();
}
}

```

MatrixD2로 들어왔다.

1) 이 안에서 쓸 변수를 만든다.

그리고 이중 배열을 쓸거기 때문에 배열도 만들어준다.

2) initMatrix 매소드를 봐보자.

Mat이라는 배열에 각각 [2][2]씩을 부여했다.

그리고 for문을 통해서 (2번 (2번) -> 총 4번 반복)

Mat[i][j]를 통해 각각 랜덤 값을 각 배열 값 안에 넣는다.

Mat[0][0] = 4 / mat[0][1] = 8 / mat[1][0] = 5 / mat[1][1] = 2
이런식으로..

3) 그 다음에 그 값을 출력할 매소드도 하나 만든다.

이 안에 for문을 또 만드는 이유는

위의 값은 배열안에 값을 입력했다면,

지금 for문은 그 값을 출력하기 위해 반복을 통해 도출

Println은 이쁘게 보이려고 하는거임.

```
public class ProbBank4Ans1_2 {
    public static void main(String[] args) {
        // 2 by 2 이중 배열을 초기화해서 아무 값이나 넣어보세요.

        MyArray myArray = new MyArray();
        myArray.initMyArray(2);
        myArray.printArray();
    }
}
```

자, 그럼 다시 처음으로 돌아가서, 자세히 클래스를 구경하자.

1) MyArray의 객체를 실행하고난 후, initMyArray에 값을 2 입력한다.



```
public void initMyArray (int dim) {
    dimension = dim;

    if (dim == D2) {
        matD2 = new MatrixD2();
        matD2.initMatrix();
    } else if (dim == D3) {
        matD3 = new MatrixD3();
        matD3.initMatrix();
    }
}
```

2) 2을 입력한 값은 initMyArray()안의 int dim = 2가 된다는 뜻이다!
그러면 if 문으로 2가 입력되어서 그 안에 있는 문장이 실행이 될 것.
그 안에 새로운 객체를 실행해서 새로운 객체의 **initMatrix**를 실행해버렸다.

그리고 메인 매소드에서 printArray를 했으니 여기서도 d2의 printMatrix로 들어가야한다.



```
public void printArray () {
    if (dimension == D2) {
        matD2.printMatrix();
    } else if (dimension == D3) {
        matD3.printMatrix();
    }
}
```

3) initMatrix로 들어가보자.
그 안에 2차 배열을 만들고, 크기 입력완료
For문 돌면서 각 배열 공간에 값 넣기 완료
그럼 아까 initMyArray의 값이 배열 [2][2]의 값이다! 라고 볼수 있다.

```
public void initMatrix () {
    mat = new int[DIM][DIM];

    System.out.println("행렬 생성");

    for (int i = 0; i < DIM; i++) {
        for (int j = 0; j < DIM; j++) {
            mat[i][j] = (int) (Math.random() * RANGE);
        }
    }
}

public void printMatrix () {
    // 처음 루프 i = 0 -> 1
    for (int i = 0; i < DIM; i++) {
        // 루프 j = 0
        for (int j = 0; j < DIM; j++) {
            System.out.printf("%3d", mat[i][j]);    //
            // x   y
        }
        System.out.println();
    }
    System.out.println();
}
```

PrintMatrix로도 들어가서, 아까 initMatrix에서 값 넣은 배열을 for문을 돌려서 값을 출력하는 작업이다.
즉 이것도 작업 후 다시 printArray로 돌아간다

값은 이렇게 나온다.

Mat[0][0] = 3 mat[0][1] = 2

mat[1][0]=2 mat[1][1]=8

%3d → 띄어쓰기 세 번 한다는 뜻

랜덤 출력이라 값이 달라진다.

행렬	생성
3	2
2	8

행렬	생성
3	6
1	7

클래스 문제 2 고양이

```
public class ProbBank4Ans3 {  
    public static void main(String[] args) {  
        // 고양이 클래스를 만들어주세요.  
  
        MyCat myCat = new MyCat();  
  
        myCat.initMyCat( age: 3, weight: 8.2f);  
        myCat.simulation();  
    }  
}
```

문제를 봐보자.

MyCat 객체생성했다.

그리고 myCat의 initMyCat매소드에 int형 하나, float형 값 하나 입력
또한 myCat객체의 simulation 매소드를 출력하도록 한다.

```

public class MyCat {
    int age;
1) float weight;
    final String defaultSound = "Meow";

    final int FEED = 1;
    final int TRAINING = 2;
    final int RANGE = TRAINING - FEED + 1;

    final int SCHEDULE_NUM = 3;

    int scheduleNum;

    int simulationCnt;

2) public void initMyCat (int age, float weight) {
    this.age = age;
    this.weight = weight;

    simulationCnt = 0;
}
3) private void feed () {
    System.out.println("고오급 사료를 먹습니다!");

    weight += 0.1f;

    System.out.println("운동 하세요!");
}
4) private void warningMessage () {
    System.out.println(defaultSound);

    if (weight > 8) {
        System.out.printf("운동 안하면 고양이 비만 됩니다: %.1f\n", weight);
    } else if (weight < 6) {
        System.out.printf("너무 많이 빠도 위험합니다: %.1f\n", weight);
    } else {
        System.out.printf("현재 고양이는 정상입니다: %.1f\n", weight);
    }
}
3) private void training () {
    weight -= 0.1f;

    System.out.println("몸무게를 조금 뺄것 같습니다!");
}

```

너무길군....

MyCat 클래스 생성해서

1) 필요한 변수들 입력한다.

2) initMyCat에 int age, float weight를 적어서 메인클래스에서 값을 입력, 호출 가능하도록 했다.

simulationCnt는 왜 여기에 0을 써놓은거지??

3) Feed 매소드 : wieht 증가 // training 매소드 : weight 감소 - 매소드를 각각 만들어놨다.

4) warningMessage 매소드. If문 사용하여 toomuch하게 증가하거나 감소하면 값이 출력되도록 한다.

```

private int getRandomPattern () {
    return (int) (Math.random() * RANGE + FEED);
}

public void simulation () {
    boolean endFlag = false;
    int cnt = 0;

    while (true) {
        System.out.println("오늘은 무엇을 할까요 ? (하루 일과는 3가지를 선택합니다)");

        for (int i = 0; i < SCHEDULE_NUM; i++) {
            scheduleNum = getRandomPattern();

            switch (scheduleNum) {
                case FEED:
                    feed();
                    break;
                case TRAINING:
                    training();
                    break;
            }

            warningMessage();
            cnt++;

            if (cnt > 365) {
                System.out.println("냥이는 가족들과 함께 오래오래 행복하게 살았답니다 ^^");
                endFlag = true;
                break;
            } else if (weight > 12.0f) {
                System.out.println("병원 ^^");
                endFlag = true;
                break;
            } else if (weight < 5.1f) {
                System.out.println("병원 가요 ^^");
                endFlag = true;
                break;
            }
        }

        if (endFlag) {
            break;
        }
    }
}

```

왜 for문의 조건이 schedule num 즉 i<3인건 하루일과 3가지라고 되어있어서인건 알겠는데.. 실제 고양이가 움직이는건 feed와 traning뿐인게 맞는거죠?

for의 반복 조건은 3이라고 되어있지만, 3번 반복이 끝나고 While문 통해서 다시 3번 반복.. 이렇게 무한루프가 맞나요?

endFlag가 정확하게 어떤 의미인가요? True하면 값을 끝나게 되는게 맞는건가요? 그렇고하기엔, endFlag가 while문에서 활약 하는게 없는것같은데 While문에 거짓이 있다면 어떤 값이든 상관없이 끝나게되는건가요?

이것도 무슨뜻이였는지 잘 모르겠습니다 ^^

getRandomPattern매소드 생성 랜덤문 생성 -> 1또는 2
Simulation 생성/ 버전1)조금 복잡하게 생성

Cnt는 값을 하나씩 증가하면서 365까지간다.

While 시작 - true.

For문 반복, 언제까지? Schedule num(3)까지. 3번 반복

Schedulenum은 랜덤 값을 가진다. 그 값은 1 아님 2

Switch문에 넣어서, 1아니면 2로 증감이거나 감소거나 체크

경고메시지 넣기/12kg이상 또는5kg미만이면 경고 아님 정상
CNT++ 으로 값을 0부터 하나씩 증가하기

If 365까지 문제 없이 지속된다면 행복엔딩

If 12kg보다 많이 나가면 병원엔딩

If 5kg보다 덜나가면 병원엔딩

질문 : endFlag -> 기본값이 false니까 true로 바뀌면 Break를 하라는 뜻인건가 true로 왜 바꾸는거지.

여기서 주의해야할점.

If문 끝엔 다 true이고, break를 썼다.

Break썼다는건..반복문 for문을 끝낸다? 끝내면 while문 다시 돌아간다.


```

private boolean checkSimulation () {
    boolean checkSim = false;

    simulationCnt++;
    System.out.printf("%d 일차!", simulationCnt);

    if (simulationCnt > 365) {
        System.out.println("냥이는 가족들과 함께 오래오래 행복하게 살았답니다 ^^");
    } else if (weight > 12.0f) {
        System.out.println("병원 ㅠ");
    } else if (weight < 5.1f) {
        System.out.println("병원 가요 ㅠ");
    } else {
        checkSim = true;
        return checkSim;
    }

    return checkSim;
}

public void simulation2 () {
    boolean endFlag = false;
    int cnt = 0;

    while (checkSimulation()) {
        System.out.println("오늘은 무엇을 할까요 ? (하루 일과는 3가지를 선택합니다)");

        for (int i = 0; i < SCHEDULE_NUM; i++) {
            scheduleNum = getRandomPattern();

            switch (scheduleNum) {
                case FEED:
                    feed();
                    break;
                case TRAINING:
                    training();
                    break;
            }

            warningMessage();
        }
    }
}

```

질문 : While문이 끝나면 while문 소가로도 자동으로 출력이 되는건가요?
Simulation2메소드에 checksimulation출력이 안나오는데 출력이 되는것이 궁금합니다.

다른 방법의 시뮬레이션 2번이다.
다른점은, 시뮬레이션 체크(결말)을 다른곳에 썼기때문에 가독성이 좋아진다.

결국 이 뜻은.
Boolean false로 값을 막아둔다. If문을 하면서 필요한부분 즉 해피앤딩 아님 병원행 말고 값을 반복해야할 때 Else에는 true로 바꿔서 while문이 계속 반복된다는 것을 표현할수있다.

while이 true로 다시 실행해서 checksimulation이 실행될때 시작 값을 false로 잡아서 다시 막아둔다~
그러다가 해피/병원행일때는 return은 그래도 false로 끝나기 때문에 While문은 이제 그만하기!!

- 1.While문 시작할때 -> checksimulation 매소드 들어가기.(시작은 false) simulationCnt값이 결과값이 나오냐? 아니 → true로 바꾸기.
- 2.결국 checksimulation의 결과값은 true이기 때문에 while문 실행 가능 아래 내용 진행 가능.
3. 만약 simulationCnt가 365이상이거나 병원행이라면, boolean이 true로 바꾸지 못하고 false로 끝난다. While문에서도 false로 나오기 때문에 While 반복이 끝나게 된다. 결과값은 checksimulation의 값 나오게된다?

클래스 문제 3 학생

```
public class ProbBank4Ans5_6_7_9_10 {  
    public static void main(String[] args) {  
        // 학생 클래스를 만들어봅시다.  
  
        Student student = new Student();  
        student.initStudents( name: "나배달", age: 20, major: "배달");  
  
        // 어 ? 어떻게 println()에 student 객체를 넣으니  
        // 알아서 내부의 내용이 나올까 ?  
        // toString()을 만들면 알아서 println() 사용시 정보가 맵핑되어 출력되게 만들어진다.  
        System.out.println(student);  
        // toString()이 없으면 Student@1b6d3586 형태의 메모리 주소값이 나온다.  
  
        // 평균을 구하는 7번 문제의 경우엔 고려해야할 사항이 추가로 있다.  
        // 평균의 종류가 몇 개 있을까 ??? ㅋㅋㅋ  
        // 산술 평균, 기하 평균, 조화 평균, 중간값, 4분위수상의 평균, 빈도값 (mean: 평균)  
        // 우선 편의상 그냥 결합시키자 ... (정석은 분리시켜야함)  
    }  
}
```

문제는 은근 간단하다.

학생 클래스를 만들고, 그거에 대한 과목 이름 적기
값도출해서 평균/분산/표준편차 만들기

그런데 몇 개 봐야할 것이 있다.

내용이 복잡하니까 toString 먼저 알고 가자...

클래스 문제 3 학생 – toString 사용

Student class

```
@Override
public String toString() {
    return "Student{" +
        "name=" + name + "\" +
        ", age=" + age +
        ", major=" + major + "\" +
        ", score=" + score +
        '}'
}
```

score class

```
@Override
public String toString() {
    return "Score{" +
        ", range=" + range +
        ", score=" + Arrays.toString(score) + "\n" +
        ", scoreName=" + Arrays.toString(scoreName) +
        ", mean=" + mean +
        ", var=" + var +
        ", stdDev=" + stdDev +
        '}'
}
```

```
Student{name='나배달', age=20, major='배달', score=Score{, range=71, score=[72]
, scoreName=[국어어], mean=72.0, var=0.0, stdDev=0.0}}
```

toString생김새
클래스 내에 있는 변수들의 값을 retur해준다.
인텔리제이는 고맙게 toString만 써도 값이 다 나오게 도와준다.
이렇게 하면 데이터타입 이런거 상관없이 값을 다 나오게 해준다.

그런데, 메인클래스에서는 student만 적었는데,
Score의 toString도 출력되네요.
toString하면 무조건 출력인건가요??
아니면 student가 score매소드를 가지고있어서 되는건가요

```

public class ProbBank4Ans5_6_7_9_10 {
    public static void main(String[] args) {
        // 학생 클래스를 만들어봅시다.

        Student student = new Student();
        student.initStudents( name: "나배달", age: 20, major: "배달");

        // 어 ? 어떻게 println()에 student 객체를 넣으니
        // 알아서 내부의 내용이 나올까 ?
        // toString()을 만들면 알아서 println() 사용시 정보가 맵핑되어 출력되게 만들어진다.
        System.out.println(student);
        // toString()이 없으면 Student@1b6d3586 형태의 메모리 주소값이 나온다.

        // 평균을 구하는 7번 문제의 경우엔 고려해야할 사항이 추가로 있다.
        // 평균의 종류가 몇 개 있을까 ??? ㅋㅋㅋ
        // 산술 평균, 기하 평균, 조화 평균, 중간값, 4분위수상의 평균, 빈도값 (mean: 평균)
        // 우선 편의상 그냥 결합시키자 ... (정석은 분리시켜야함)
    }
}

```

Student 의 객체 사용.
initStudents로 string, int string 값 입력하기

그리고 출력에 student를 넣으면
toString 값이 출력된다!?

그럼, initStudents 매소드를 보러 가보자

```

public class Student {
    String name;
    int age;
    String major;

    Score score;

    public void initStudents (String name, int age, String
major) {
        this.name = name;
        this.age = age;
        this.major = major;

        score = new Score();
        score.initScore();
    }

    @Override
    public String toString() {
        return "Student{" +
            "name=" + name + "\" +
            ", age=" + age +
            ", major=" + major + "\" +
            ", score=" + score +
            "}";
    }
}

```

Student 클래스 생성,
학생하면 생각나는것 : 이름 / 나이 / 전공

initStudent에 string, int, string형 적어서
메인메소드에서 값 도출 할 수 있도록 체크

보면, 그 안에 생성자끼리 값 맞춰주는데 마지막에
Score에 대한 새로운 메소드를 생성한다.
그래서 이번 클래스에 쓴 메소드가 initStudent밖에
안썼는데도 Score 메소드 사용이 가능.

그럼 score 메소드 한번 구경하러가보자.

```
public class Score {
    // 학생 클래스에 수학, 영어, 국어 점수를 입력 받도록
    // 개조합니다.
    // 6번 문제를 푸는데 있어 국영수만 처리하면 되지만
    // 실제로 점수라는 클래스는 보다 범용적으로 확장이 가능한
    // 녀석이므로
    // 확장성을 고려한 설계를 진행하도록 한다.
```

```
final int ARR_MAX = 10;
final int SCORE_MAX = 100;
final int SCORE_MIN = 30;
```

```
int range;
```

```
int[] score;
String[] scoreName;
Scanner scan;
```

```
float mean;
float var;
float stdDev;
```

```
int cnt;
```

```
public void initScore () {
    scan = new Scanner(System.in);
    cnt = 0;

    addSubject();
    // 편의상 랜덤값을 활용하도록 함
    inputScore();
    // 평균 계산
    calcMean();
    // 분산 계산
    calcVariance();
    // 표준편차 계산
    calcStdDev();
}
```

시작하기에 앞서, 우리가 해야할 것들이 좀 있다.

각 과목의 값을 저장 후에 출력하기 위해 배열 필요
과목의 점수는 랜덤으로 생성한다.

그담에 평균 / 분산 / 표준편차 계산 할것이 많다.
아무래도 평균값은 소수점이 필요하기 때문에 float사용

```

public void setRange () {
    range = SCORE_MAX - SCORE_MIN + 1;
}

public void inputScore () {
    score = new int[cnt];

    setRange();

    for (int i = 0; i < cnt; i++) {
        score[i] = (int) (Math.random() * range + SCORE_MIN);
    }
}

public boolean checkInput () {
    System.out.print("계속 입력하시겠습니까 ? (y/n): ");

    while (true) {
        String yOrN = scan.nextLine();

        if (yOrN.equals("n")) {
            return false;
        } else if (yOrN.equals("y")) {
            System.out.print("다음 과목을 입력하세요: ");
            return true;
        } else {
            System.out.print("다시 입력하세요! (y/n): ");
        }
    }
}

```

setRange로 랜덤문 기초공사
inputScore로 score배열에 점수 랜덤으로 받기

Checkinput 매소드
→ 값을 계속 받을 수 있을지 체크하기위한
Boolean 형 매소드!

While true로 시작하면서, 값이 y인지 n인지에 따라서
While이 계속할지 말지를 선택할 수 있다.
N은 false로 while중단.
Y는 true로 while 반복
Else도 리턴은 안했지만, 어차피 while처음으로 돌아가서
True로 다시 시작할 수 있다.

```

public void addSubject () {
    // 실제로는 Java의 Collection에 있는 Map이나 Vector를 사용해야 하는데
    // 여기서 직접 자료구조를 구현하는것도 예바고
    // 그렇다고 문제를 안 풀수도 없으니 일종의 꼼수로서
    // 백업 배열을 하나 만들어서 Java의 Collection 역할을 하게 만든다.
    String[] backup = new String[ARR_MAX];

    System.out.print( " 다루고 싶은 과목을 입력하세요: " );

    // 보편적으로 지저분한 코드는 어느 시점에 루프를 멈출지를 결정하는 부분에서
    // 만들어진다.
    // 그러므로 루프를 언제 멈출지 판정하는 부분을 매서드화하면
    // 상대적으로 깔끔한 코드를 얻을 수 있다.
    Do {
        backup[cnt++] = scan.nextLine();

        // 현재 다시 입력하세요를 구현하는 부분에서 boolean 변수가 추가됨에 따라
        // 코드의 가독성을 저해하는 부분이 있어
        // 기본적인 로직 작성후 매서드화 시키는 부분이 좋다 판단하여 삭제 처리
    } while (checkInput());

    scoreName = new String[cnt];

    for (int i = 0; i < cnt; i++) {
        scoreName[i] = backup[i];
    }
}

```

Backup 배열을 사용해서 ARR_MAX(10)임의 지정 배열 크기 만들었다.

Do ~ while문 사용! 깔끔하게 보기가 좋다.
:일단 한번 시작해! 그리고 while문에서 true냐 False냐에 따라 반복여부 확인.

While문의 ()에는 checkinput 아까봤던 boolean형 매소드를 넣었다. False가 나올때까지 무한 반복!

하면서 scorename 배열에 cnt만큼의 크기를 가지고 For문으로 scorename의 값과 backu의 값을 일치시킨다.
아, 이럴려고 backup 배열을 만들었구나.


```

public void calcMean () {
    mean = 0;

    for (int i = 0; i < cnt; i++) {
        mean += score[i];
    }

    mean /= cnt;
}

public void calcVariance () {
    var = 0;

    for (int i = 0; i < cnt; i++) {
        var += Math.pow(score[i] - mean, 2);
    }

    var /= cnt;
}

public void calcStdDev () {
    stdDev = (float) Math.sqrt(var);
}

@Override
public String toString() {
    return "Score{" +
        ", range=" + range +
        ", score=" + Arrays.toString(score) + "\n" +
        ", scoreName=" + Arrays.toString(scoreName) +
        ", mean=" + mean +
        ", var=" + var +
        ", stdDev=" + stdDev +
        '}';
}

```

나머지는 순서대로
합산 후 평균구하기 / 분산 / 표준편차 수하는 내용이다.

크게 문제없을듯.

toString은 return하여 값 출력!