

문제는 행[9] - 1, 3번 내용 정리 (미완성)

12/20 개강 SW개발자 양성과정 [손현지]

2021. 02. 03

문제 1

1. Product 클래스를 만들어보자

Random을 활용하여 임의의 상품과 가격들이 들어갈 수 있도록 만들도록 한다.

오늘 학습한 Get 을 활용하여 사용자가 URL을 요청한다.

오늘의 추천 상품을 누르면 POST쪽에서 알아서 상품이 들어가도록 만들고

추천된 상품의 결과를 JSON Viewer를 통해서 화면상에 출력하도록 만들어보자!

```

12 @Slf4j
13 //상품 정보는 많은 데이터 전송보다는 실제 HTML 페이지를 넘겨줘야 하는 상황이었지?
14 //RestController보다는 Controller를 사용하는 편이 좋을 것 같다.
15 @Controller
16 @RequestMapping("/9th/product")
17 public class Bank9Dash1Controller {
18
19     @GetMapping("/main")
20     public String boardMain(){
21         log.info("boardMain()");
22
23         return "9th/board/main";
24     }
25
26     //사용자가 URL을 요청할 수 있는 GetMapping
27     //상품의 정보가 들어와 JSON 형태로 정리되도록 하려면 responseBody가 필요하다
28     //근데 이게 Post쪽에 JSON 형태로 들어오도록 해야 맞는 건가?
29     //화면상에 JSON viewer로 출력되게 해야하니까 GetMapping에 하는 게 맞겠지?
30     @ResponseBody
31     @GetMapping("/productGet")
32     public String productGet(){
33         log.info("productGet()");
34
35         //사용자가 랜덤 추천 상품을 요청
36         //상품을 관리하는 ProductManager를 객체화
37         //MAX는 추천 상품 갯수
38
39         //public static final int로 상품 품목별로 번호를 매겨준다.
40         //0 = CLOTH, 1 = BOOK
41         RecommendManager p1 = new RecommendManager(TOTAL: 5, MAX: 2, ProductNumber.CLOTH);
42         //총 Total개의 상품 중에, MAX개의 추천 상품을, 상품 품목의 전역변수 넘버
43
44         return "9th/board/productViewInfo";
45     }
46 }

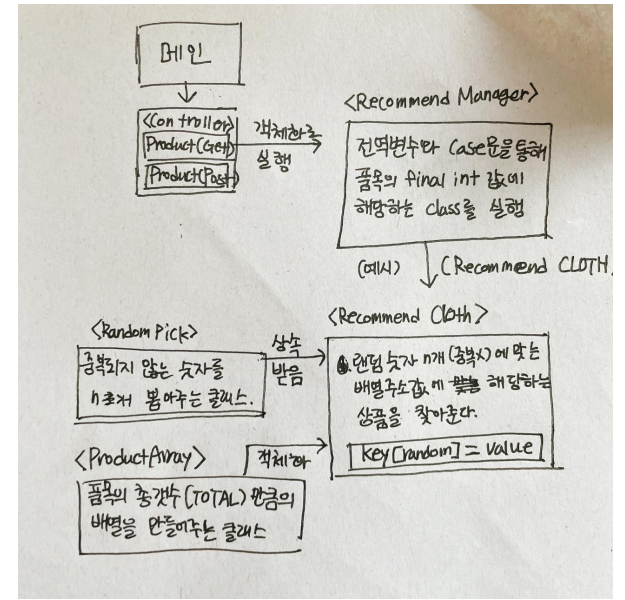
```

```

48 //URL 정보가 노출되지 않는 PostMapping
49 @PostMapping("/productPost")
50 public String productPost(){
51     log.info("productPost()");
52
53     return "9th/board/productPost";
54 }
55
56 }
57

```

구상한 형태



```
RecommendManager.java
1 package com.example.homework.entity.order9;
2
3 import com.example.homework.entity.order9.ProductNumber;
4 import com.example.homework.entity.order9.RecommendBook;
5 import com.example.homework.entity.order9.RecommendCloth;
6
7 public class RecommendManager {
8
9     private RecommendCloth[] rCloth;
10    private RecommendBook[] rBook;
11
12    private final int CHOOSENUM = 3;
13    private int serviceCode;
14
15    public RecommendManager(int TOTAL, int MAX, int serviceCode){
16
17        this.serviceCode = serviceCode;
18        randomRecommend();
19    }
20
21    //recommendProduct를 실행 하면 랜덤 추천 상품이 MAX개 나오도록 만들자
22    //전역 변수를 사용해서 1번 상품(의류)나 2번 상품(책)이 나오게 해볼까
23    //쓰레드는 필요없고 상속까지 잘 활용하면 좋을 것 같다.
24    public void randomRecommend(){
25        switch(serviceCode){
26            //일반 배열이 아니라 MAP 배열을 쓰고 싶음...
27            case ProductNumber.CLOTH:
28                rCloth = new RecommendCloth[CHOOSENUM]; //총 3개짜리 배열 선언
29                break;
30
31            case ProductNumber.BOOK:
32                rBook = new RecommendBook[CHOOSENUM];
33                break;
34        }
35        //세 개를 뽑는 반복문은 어디에 들어가면 좋나...
36        //상품 중 3개를 뽑는 반복문이 들어있는 클래스를 상품 클래스들이 상속받게 만들까?
37    }
38 }
39
```

```
RecommendCloth.java
1 import com.example.homework.entity.order9;
2
3 public class RecommendCloth extends RandomPick {
4     //0~9(총 10개짜리 배열) 안에 각각의 Cloth 클래스들을 넣어주면 되나.
5     //상품은 10개 정도 넣을 거니까 배열 10개 주소값에 상품 10개 클래스 집어넣는 게
6     //그렇게 번거롭진 않겠지만...매번 이렇게 하나씩 넣어주는 게 좋은 방법일까...
7     //더 효율적인 방법이 있을텐데, 배웠던 거 같은데 기억이 안나네...
8
9    public RecommendCloth(int TOTAL, int MAX){
10        super(); //controller에서 추천 상품 갯수를 직접 받고 여기엔 변수명만 표시하도록 고치자
11
12        ProductArray pr = new ProductArray(TOTAL);
13        //TOTAL 값 만큼 배열을 만들어주는 메서드를 실행 시킨다.
14
15        //MAX만큼 돌아가는 반복문을 만들어서
16        //품목[randIdx](주소) = 랜덤상품 정보(값) 가 나오도록 만들어준다.
17        // -----> RandomPick에서 나온 랜덤 숫자
18
19        for(int i = 0; i < MAX; i++){
20            // Product[randIdx] = value
21        }
22    }
23 }
24
```

형광펜 친 부분...고민하다가 이쯤에서부터 막혔습니다.
ProductArray와 어떻게 구성해줘야할지 감이 오지 않더라구요.
ProductArray 클래스 부분에 후술할게요.

```

1 package com.example.homework.entity.order9;
2
3 import java.util.HashMap;
4 import java.util.Map;
5
6 //제품마다 배열 클래스를 만들어줄 것 없이
7 //이곳에서 TOTAL값 만큼 제품 배열을 만들어주고
8 //각각 Recommend제품 클래스에서 객체화를 통해 사용해도 좋을 것 같음.
9
10 //문제는 TOTAL값만큼 배열을 만들고, key에 번호를 부여하는 건 반복문으로 할 수 있겠는데
11 // value에 각각 다른 정보들을 넣는 건 어떻게 해야 좋을지 모르겠음.
12
13 public class ProductArray { //ProductArrayManager로 바꾸는 게 낫나?
14
15     //일단 자료형 갯수가 가변할 수 있으니 List 자료형을 쓰는 편이 좋을까?
16     //근데 하나하나 별도로 지정할 건 아니지 않나...
17     //하나하나 지정할 거면 자라리 MAP이 나올 것 같기도 한데... key값만 찾으면 편하니까
18     public ProductArray(int TOTAL){
19         Map<Integer,Class> product = new HashMap<>();
20         //Class에 클래스(=데이터 타입) 이름 적여줘야 한다.
21
22         //Cloth에 해당하는 정보가 저장된...클래스를 객체화 한다.
23         //20th 강의와 map-view 참고하자
24
25         for(int i = 0; i < TOTAL; i++){
26             // map.put("key" + i, product)
27         }
28     }
29 }
30

```

어차피 식이 중복될테니 품목별로 배열 만드는 클래스를 따로 만들어주지 않고 공통적으로 쓸 수 있는 ProductArray를 만들면 좋을 것 같습니다.

Controller에서 객체화 하면서 입력한 TOTAL 값만큼 루프를 돌며 반복문을 통해 배열주소값을 만들어줍니다.

문제1. 번거로움을 줄이기 위해 이런 방식으로 배열 주소값을 만들어 주었는데 value값은 하나하나 클래스를 다 따로 만들어서 손수 입력해줘야 할까요?

- 클래스 갯수가 너무 많아진다는 문제점도 있는 것 같습니다. 한 클래스 내에 정보만 입력하여 모아둘 방법은 없을까요?
- 문자열로 쪽 {"1번옷", "2번옷", "3번 옷"} 이렇게 나열하는 방식이 있는 건 알겠는데 상품명 뿐만 아니라 가격도 함께 표시될 수 있도록 하고 싶었어요. 선생님께서 보여주신 예시에서 key 하나당 ID와 PW를 같이 볼 수 있었던 것 처럼요...

문제2. 또, RecommendCloth 클래스에 위와 같은 value 정보를 하나하나 입력해주면 보기에 좋지 않은 방법이 아닐까 고민이 되었습니다.

- ProductArray -(상속)→ Cloth배열의 value 정보를 모아둔 클래스 -(객체화)→RecommendCloth
- ProductArray -(객체화)→ RecommendCloth Cloth의 value정보를 모아둔 클래스 -(객체화)→ RecommendCloth(두 개의 클래스를 객체화 시킴)

둘 중 어떤 방법이 더 나은 방법인지도 고민이 됩니다... 아니면 또 다른 방법이 있을까요?

```

1 package com.example.homework.entity.order;
2
3 import lombok.Data;
4
5 import java.util.ArrayList;
6 import java.util.Collections;
7 import java.util.List;
8
9 //Total 갯수만큼 있는 배열 중에서 MAX 갯수만큼 배열[i] 주소값을 랜덤으로 뽑아준다.
10 //이때 랜덤 주소값 MAX개 중에 중복은 없어야 한다.
11 @Data
12 public class RandomPick {
13
14     private int pickNum;
15
16     private int TOTAL; //품목의 총 갯수
17     private int MAX; //최대로 뽑을 랜덤추천상품
18     //이 둘은 값이 언제든 바뀔 수 있으니 여기에서 값을 final로 지정X
19
20     private boolean isAlloc = true;
21
22     private int randIdx = 0;
23
24     //인자의 숫자(3)만큼 연산하면 된다.
25     public void RandomPick(int TOTAL, int MAX) {
26
27         boolean[] randBox = new boolean[TOTAL]; //TOTAL값은 RecommendCloth에서 취어나
28         int[] selectIdx = new int[MAX];
29
30         this.TOTAL = MAX;
31         this.MAX = MAX;

```

RandomPick.java

```

32
33 //로또박스를 응용하면 되지 않을까...
34 //왜냐면 중복 상품도 뜨면 안되는 거긴 하니까 중복 거를 수도 있고 관심을 것 같다.
35 for(int i = 0, allocCnt = 0; i < MAX; i++){ //반복문 1 : MAX 값만큼 랜덤 숫자를 뽑아준다.
36     while(isAlloc){
37         pickNum = (int)(Math.random() * TOTAL);
38
39         isAlloc = false;
40
41         //루프마다 랜덤 숫자를 새로 뽑아서 randArr[랜덤]을 MAX갯수만큼 뽑아준다
42         for(int j = 0; j < allocCnt; j++){
43             if(selectIdx[j] == randIdx){
44                 isAlloc = true;
45                 break;
46             }
47         }
48     }
49     randBox[randIdx] = true;
50     selectIdx[allocCnt++] = randIdx;
51
52     isAlloc = true;
53 }
54 //boolean[] randBOX = true;로 반복문 나온 녀석들을...어떻게 변형해줘야하지?
55
56 //근데 이게 막상 또 작성하고 보니 헷갈린다... 어차피 스레드 3개로 나눠 돌리는 게 아니라면
57 //스레드로 만들 필요가 없지 않나...
58 //스레드로 돌릴 때는 중복 발생을 어떻게 막아야 하는지도 잘 모르겠음.
59 //그냥 스레드 없이 가보자... 일단...
60 }
61
62
63

```

전체 TOTAL값 중에서 MAX갯수만큼 랜덤으로 (중복 없이)숫자를 뽑아주는 lotto 반복문을 이용해보기로 했습니다.

Recommend 상품 클래스들이 모두 공통적으로 이 클래스를 상속받도록 합니다.

문제 3

3. 일전에 우리가 넓이 계산하던 문제를 함께 응용해보는 문제입니다.

웹상에 정삼각형, 정사각형, 정육각형을 선택할 수 있도록 만듭니다.

각 버튼에 따라서 이를 올바르게 처리할 수 있도록 서포트하여

올바른 넓이값을 화면상에 출력할 수 있도록 만들어봅시다!

(이번 케이스는 구지 적분 방식으로 미소 면적을 모두 합산하지 않아도 됩니다)


```

16 @RequestMapping("/9th/area")
17 public class Bank9Dash3Controller {
18
19     @RequestMapping("/triangle")
20     public String triangle() {
21         log.info("triangle");
22
23         Triangle tri = new Triangle( WIDTH: 2, HEIGHT: 2);
24         tri.calcTriArea();
25
26         model.addAttribute("triangle", tri.getTriArea());
27         //왜 빨갈게 나오는 건지 모르겠음...
28         //alt+Enter 누르면 create local variable 'model' 이라는 문구가 뜬다.
29         // AttributedString model; <만들어주면
30         //"triangle"이 attribute: 라고 쓰지만 빨간 밑줄이 뜨면서
31         //'addAttribute(java.text.AttributedString.Attribute, java.lang.Object)' in 'java.text.AttributedString' cannot be applied to '(java.lang.String, int)'
32         //라는 문구가 뜬다. int형이 아니라 string형으로 입력하라는 뜻인지... triangle이라는 int 변수를 만든 적도 없는데 왜 뜨는지 모르겠음...
33
34         return "9th/triangle";
35     }
36
37     @RequestMapping("/rectangle")
38     public String rectangle(){
39         log.info("rectangle");
40
41         Rectangle rect = new Rectangle( WIDTH: 2, HEIGHT: 2);
42
43         model.addAttribute("rectangle", rect.getAreaBase());
44         return "9th/rectangle";
45     }
46
47     @RequestMapping("/hexagon")
48     public String hexagon(){
49         log.info("hexagon");
50
51         Hexagon hex = new Hexagon( WIDTH: 2, HEIGHT: 2);
52
53         model.addAttribute("hexagon", rect.getAreaBase());
54
55         return "9th/hexagon";
56     }
57 }

```

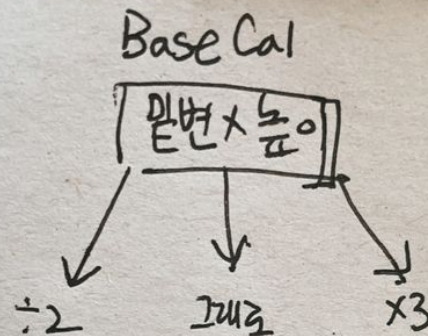
//정삼각형 : 밑변 x 높이 / 2

//정사각형 : 밑변 x 높이

//정육각형 : 밑변 x 높이 x 3 (정육각형은 작은 정삼각형 6개가 있는 형태)

//모두 밑변x높이의 과정은 공통적으로 들어간다.

//부가적인 계산은 변수 값을 받아서 각자 클래스에서 마저 진행해도 될 것 같다.



* 직육각형 x 4개 스텝이 없이 인출이로 3개씩네.

BaseCalc.java

```
1 package com.example.homework.utility.area;
2
3 public class BaseCalc {
4
5     protected int WIDTH;
6     protected int HEIGHT;
7     int areaBase;
8
9     public BaseCalc(int WIDTH, int HEIGHT) { int areaBase = WIDTH * HEIGHT; }
10
11 }
12
13
```

Rectangle.java

```
1 package com.example.homework.utility.area;
2
3 import lombok.Data;
4
5 @Data
6 public class Rectangle extends BaseCalc {
7
8     private int rectArea;
9
10    public Rectangle(int WIDTH, int HEIGHT){
11        super(WIDTH, HEIGHT);
12        rectArea = areaBase;
13        //정사각형 넓이는 areaBase의 계산과 다르지 않기 때문에
14        //값을 그대로 받기만 하면 된다.
15    }
16
17 }
```

```

1 package com.example.homework.utility.area;
2
3 import lombok.Data;
4
5 @Data
6 //빨간 밑줄과 함께 Lombok needs a default constructor in the base class 라는 글자가 뜬다.
7 //빨간 전구모양과 뜨는 화살표 클릭하면
8 //inspection 'Lombok annotations' options < 이런 게 뜨는데
9 //뭣 모르고 건드렸다가 큰일 날까봐 못 만지겠음...
10 public class Triangle extends BaseCalc {
11
12     private int triArea;
13
14     public Triangle(int WIDTH, int HEIGHT){
15         super(WIDTH, HEIGHT);
16         calcTriArea();
17     }
18
19     public void calcTriArea() { triArea = areaBase/2; }
20 }

```

← 이 문제는 Triangle, Rectangle, Hexagon 모든 클래스에서 발생하는 문제입니다 ^^

```

1 package com.example.homework.utility.area;
2
3 import lombok.Data;
4
5 @Data
6 public class Hexagon extends BaseCalc{
7
8     private int hexArea;
9
10    public Hexagon(int WIDTH, int HEIGHT) { super(WIDTH, HEIGHT); }
11
12
13
14    public void calcHexArea() { hexArea = areaBase*3; }
15
16
17
18 }

```

정리하면서 보니 상속만 할 게 아니라 AreaManager와 전역변수를 활용했어도 좋았을 것 같다. 세가지 도형만 있으면 되니까 생각을 못했는데 넓이 구하기 외의 다른 계산을 위한 코드를 추가하게 될 일이 생긴다면 이게 더 편할 것 같다.