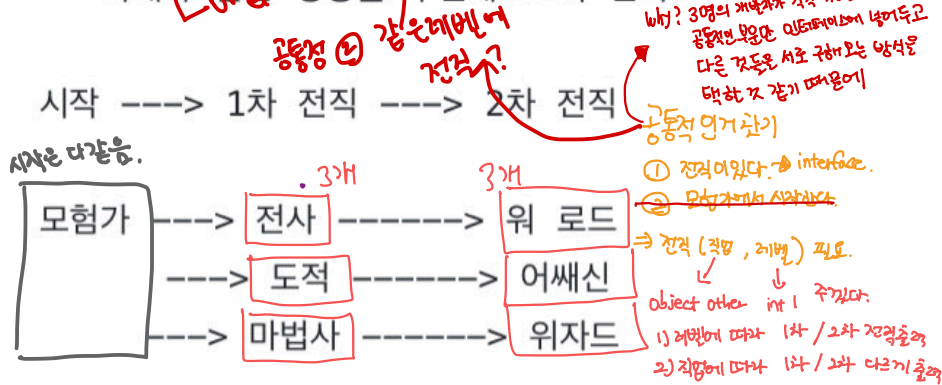


1) 질문 노트 작성하기

전체적인 저수 해석 후 확인 부탁드립니다.
(아직까지 완벽히 숙지 못했습니다 공부가 더 필요)

2) interface를 사용하여 가장 좋은 예는 게임이다.

아래와 같은 상황을 구현해보도록 한다.



☆ 각각의 캐릭터들은 공격 스킬을 두 개씩 가지고 있다.

전직을 할 때마다 추가적인 전직 스킬 두 개를 얻게 된다.

⇒ 모험가: 스킬1 스킬2
도적: 스킬1 스킬2 스킬3 스킬4
2차전직: 스킬1 스킬2 스킬3 스킬4 스킬5 스킬6 스킬7 스킬8

스킬을 사용할 때 모두가 일관되게

skill1(), skill2(), skill3() 형식으로

데미지가 계산되도록 만들어보자!

```
1 package Twelveth;
2
3 // 인터페이스 작성법
4 // 1. public 을 적고 interface를 적는다.
5 // 2. 인터페이스로 만들 이름을 적는다.
6 // 3. 내부에 다형성을 가지게 만들 매서드 프로토타입을 작성한다.
7
8 // 다형성 ?
9 // 누가 이것을 구현하냐에 따라 결과가 달라진다.
10 public interface Comparable { interface 작성하는 방법,
11     // 현재 이부분은 단순히 프로토타이핑만 한 것이다.
12     // 프로토타이핑이 무엇이나 ?
13     // 매서드의 리턴타입, 매서드이름, 매서드 인자의 입력만 정의한 형태
14     public int compareTo(Object other, int order);
15 }
```

다형성 → 여러 형태의 매서드를 만들 수 있다

프로토타이핑 →

```
1 package Twelveth;
2
3 import static Twelveth.Utility.MYSTRCMP;
4 import static Twelveth.Utility.STRCMPTEST;
5
6 // 실질적으로 interface 에 작성한 프로토타이핑한 매서드는
7 // 이것을 implements 하는 클래스에서 작성해줘야 한다.
8 public class MyStrcmp implements Comparable {
9     private double area;
10
11     public MyStrcmp(double a) {
12         area = a;
13     }
14
15     public int processCompare(MyStrcmp other) {
16         if (this.area < other.area) {
17             return -1;
18         } else if (this.area > other.area) {
19             return 1;
20         } else {
21             return 0;
22         }
23     }
24
25     public int processCompare(StrcmpTest other) {
26         if (this.area < other.getArea()) {
27             return -1;
28         } else if (this.area > other.getArea()) {
29             return 1;
30         } else {
31             return 0;
32         }
33     }
34
35     public int processMyStrcmp(Object otherObj) {
36         MyStrcmp other = (MyStrcmp) otherObj;
37
38         return processCompare(other);
39     }
40 }
```

만약 105.3 이면
area a = a = 105.3

ms2

105.3
ms2 a = 105.3

105.3 = 105.3

return -> 0;

7

getter 이용해서
이거는 맞다?

5

6

```
53     public int compareTo(Object otherObj, int ORDER) {
54         int res = 0;
55         switch (ORDER) {
56             case MYSTRCMP:
57                 res = processMyStrcmp(otherObj);
58                 break;
59             case STRCMPTEST:
60                 res = processStrcmpTest(otherObj);
61                 break;
62             default:
63                 break;
64         }
65
66         return res;
67     }
68 }
69 }
```

이도비로
해석하는 것이
안녕?

① 만약 ms 2 , MYSTRCMP 이면

②

③

④ 선피어러구 이동

```
1 package Twelveth;
2
3 public class StrcmpTest {
4     private double area;
5
6     public StrcmpTest(double a) {
7         area = a;
8     }
9
10    public double getArea() {
11        return area;
12    }
13 }
```

여기는 STR 값을 해당
변수에 할당
받는 코드?

```
1  package Twelveth;  
2  
3  public class Utility { // Switch Case & lot final  
4      public static final int MYSTRCMP = 1;  
5      public static final int STRCMPTEST = 2;  
6  }
```

```

1 package Twelveth;
2
3 import static Twelveth.Utility.MYSTRCMP;
4 import static Twelveth.Utility.STRCMPTEST;
5
6 // String compare with Interface
7 public class StrcmpInterfaceTest {
8     public static void main(String[] args) {
9         MyStrcmp ms1 = new MyStrcmp(105.3);
10        MyStrcmp ms2 = new MyStrcmp(105.3);
11
12        StrcmpTest st = new StrcmpTest(105.3);
13
14        // 뭐야 ? 왜 MyStrcmp 를 구지 Object 로 바꾸고
15        // 다시 이것을 형변환해서 MyStrcmp 로 만들었지 ?
16        if(ms1.compareTo(ms2, MYSTRCMP) > 0) {
17            System.out.println("ms1이 ms2 보다 더 크다.");
18        } else if(ms1.compareTo(ms2, MYSTRCMP) < 0) {
19            System.out.println("ms1은 ms2 보다 더 작다.");
20        } else {
21            System.out.println("ms1과 ms2는 같다.");
22        }
23
24        if(ms1.compareTo(st, STRCMPTEST) == 0) {
25            System.out.println("이게 되네 ?");
26        }
27    }
28 }

```

Handwritten annotations:

- 105.3.* (next to line 9)
- MyStrcmp Class* (with an arrow pointing to line 16)
- Objective order* (with an arrow pointing to `MYSTRCMP` in line 16)
- int order.* (with an arrow pointing to `STRCMPTEST` in line 24)