

Item 35

ordinal 메서드 대신 인스턴스 필드를 사용하라

목차

1

ordinal 메서드

2

ordinal 메서드 문제점

3

해결책

목차

1

ordinal 메서드

2

ordinal 메서드 문제점

3

해결책

ordinal 메서드



```
public enum Number{  
    ONE,      → 0  
    TWO,      → 1  
    THREE,    → 2  
    FOUR      → 3  
}
```

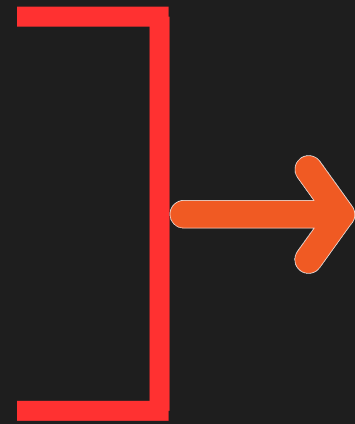
열거 타입 상수는 하나의 **정수 값에 대응**

정수 값은 **0부터 시작**

ordinal 메서드



```
public enum Number{  
    ONE, → 0  
    TWO, → 1  
    THREE, → 2  
    FOUR → 3  
}
```



index

열거 타입 상수 몇 번째 위치

= **index** 반환하는

ordinal 메서드

ordinal 메서드



목차

1

ordinal 메서드

2

ordinal 메서드 문제점

3

해결책

ordinal 메서드 문제점

```
public enum Ensemble{  
    SOLO, DUET, TRIO, QUARTET, QUINTET,  
    SEXTET, SEPTET, OCTET, NONET, DECTET;  
  
    public int numberOfMusicians(){ return ordinal()+1; }  
}
```

합주단의 종류를 연주자 명 수에 따라 정의한 **열거 타입 예시**

ordinal 메서드 문제점

1. 상수 선언 **순서를 바꾸는 순간** 오동작 한다.
2. 이미 사용 중인 정수와 **값이 같은 상수**는 추가할 수 없다.
3. 정수 값을 **중간에 비워둘 수 없다.**

ordinal 메서드 문제점

1. 상수 선언 순서를 바꾸는 순간 오동작 한다.

1

```
public enum Ensemble{  
    SOLO, DUET, TRIO, QUARTET, QUINTET,  
    SEXTET, SEPTET, OCTET, NONET, DECTET;  
  
    public int numberOfMusicians(){  
        return ordinal()+1;  
    }  
}
```

2

```
public enum Ensemble{  
    SOLO, TRIO, DUET, QUARTET, QUINTET,  
    SEXTET, SEPTET, OCTET, NONET, DECTET;  
  
    public int numberOfMusicians(){  
        return ordinal()+1;  
    }  
}
```

ordinal 메서드 문제점

1. 상수 선언 순서를 바꾸는 순간 오동작 한다.



```
public class Main {  
    public static void main(String[] args) {  
        // Enum 상수 사용  
        Ensemble ensemble = Ensemble.DUET;  
  
        // ordinal() 메서드를 이용하여 Enum 상수의 순서를 가져옴  
        int ordinalValue = ensemble.ordinal();  
        System.out.println(ordinalValue);  
    }  
}
```

결과

1

1

2

2

달라짐!!

ordinal 메서드 문제점

2. 사용 중인 정수와 값이 같은 상수는 추가 X

연주자가 8명인 합주단의 종류는 2가지.

- 8중주
- 복4중주

복4중주 = DOUBLE_QUARTET도 열거 타입에 넣고 싶다.

ordinal 메서드 문제점

2. 사용 중인 정수와 값이 같은 상수는 추가 X

```
public enum Ensemble{  
    SOLO, DUET, TRIO, QUARTET, QUINTET,  
    SEXTET, SEPTET, OCTET, DOUBLE_QUARTET, NONET, DECTET;  
  
    public int numberOfMusicians(){ return ordinal()+1; }  
}
```

OCTET = 7

DOUBLE_QUARTET=8

ordinal 메서드 문제점

3. 정수 값을 중간에 비워둘 수 없다.

연주자 12명인 3중 4중주 = TRIPLE_QUARTET

열거 타입에 추가 원함

그런데 연주자 11명인 합주단의 이름 X

ordinal 메서드 문제점

3. 정수 값을 중간에 비워둘 수 없다.

```
public enum Ensemble{
    SOLO, DUET, TRIO, QUARTET, QUINTET,
    SEXTET, SEPTET, OCTET, NONET, DECTET,
    DUMMY, TRIPLE_QUARTET;

    public int numberOfMusicians(){ return ordinal()+1; }
}
```

쓰이지 않는

dummy 상수 추가

목차

1

ordinal 메서드

2

ordinal 메서드 문제점

3

해결책

해결책

열거 타입 상수에 연결된 값은

ordinal 메서드로 얻지 말고

인스턴스 필드에 저장

해결책

```
public enum Ensemble {  
    SOLO(1), DUET(2), TRIO(3), QUARTET(4), QUINTET(5),  
    SEXTET(6), SEPTET(7), OCTET(8), DOUBLE_QUARTET(8),  
    NONET(9), DECTET(10), TRIPLE_QUARTET(12);  
  
    private final int numberOfMusicians;  
    Ensemble(int size) { this.numberOfMusicians = size; }  
    public int numberOfMusicians() { return numberOfMusicians; }  
}
```

순서 바뀌어도 문제 X

같은 값 저장 가능

중간에 빈 값 문제 X

결론!

설계 용도에 맞는 사용이 아니면

ordinal 메서드는 절대 사용하지 말자