

Use instance fields instead of ordinals

배경

enums는 single int value에 대응됩니다. 그리고 enums constant가 몇 번째 위치하는지 `ordinal()`가 말해주죠.

문제

그래서 enums constant와 연관된 정수 값을 연결하려는 실수를 저지를 수 있습니다.

코드

```
// Abuse of ordinal to derive an associated value - DON'T DO THIS
public enum Ensemble {
    SOLO,    DUET,    TRIO,  QUARTET,  QUINTET,
    SEXTET,  SEPTET,  OCTET,  NONET,   DECTET;
    public int numberOfMusicians() { return ordinal() + 1; }
}
```

상수 명과 선언된 순서를 바탕으로 이 예제는 몇 명에서 연주하는지 출력해주는 method가 선언된 예제입니다.

무엇이 문제인가?

1. 순서

전적으로 선언된 순서에 의존하기 때문에 순서가 바뀌면 `ordinal()` 메서드도 잘못됩니다... 끔찍하죠.

2. 연속되지 않은 상수

11명으로 연주하는 경우는 없으나, 12명에서 연주하는 경우를 일컫는 상수가 있다고 합시다. 12명이 있다는 것을 나타내기 위해 어쩔 수 없이 11번째를 나타내는 더미 constant 작성해야 합니다.

해결

문제는 간단합니다. `ordinal`을 쓰지 말고 `field`를 선언하고 필드의 값을 출력하는 constant의 method를 선언하면 됩니다.

```
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; }  
}
```

그럼 언제 ordinal()을 사용할까요?

Most programmers will have no use for this method. It is designed for use by general-purpose enum-based data structures such as EnumSet and EnumMap

대부분 프로그래머는 사용할 일이 없습니다. Enum으로 만드는 자료구조에 사용하는 것이죠.