

## 12-2. Enum

### 1. 열거형(Enum)이란?

- JDK 1.5부터 도입
- enumeration type 으로 앞글자를 따서 enum 이라 지칭
- 관련있는 상수를 편리하게 선언 및 사용하기 위한 클래스
- 내부적으로 static final 클래스로 생성됨

#### 장점

- 가독성이 좋으며 코드가 단순해짐
- 컴파일 시 타입을 체크하여 타입 안정성이 보장됨

### 2. 정의 및 사용

- enum 키워드로 정의
- 생성자를 갖을 수 있으며, private 과 default 만 허용
  - 내부적으로 Java 의 abstract Enum 클래스를 extends 하게 되는데, 이 Enum 클래스의 생성자의 접근 제어자가 protected 이기 때문
  - 참고: <https://docs.oracle.com/javase/8/docs/api/java/lang/Enum.html>

```
// 정의
enum 열거형이름 {
    상수명1, 상수명2, ...;
}

// 실제 정의
enum City {
    SEOUL, JEJU, BUSAN, DAEGU, GWANGJU;
}
```

```
// 사용예
class User {
    String name;
    int age;
    City city;

    public User(String name, int age, City city) {
        this.name = name;
        this.age = age;
        this.city = city;
    }
}
```

```

}

// 사용자 목록
class SomeSite {
    void someMethod() {
        // 공통된 값을 명시적으로 이용 가능
        List<User> users = new ArrayList<>();
        users.add(new User("김서울", 20, City.SEOUL));
        users.add(new User("이제주", 22, City.JEJU));
        users.add(new User("박부산", 33, City.BUSAN));
        users.add(new User("이서울", 25, City.SEOUL));
    }

    void checkCity(User user) {
        // 다른 타입 자체가 올 수 없음 == 타입 안정성
        if (user.getCity() == City.SEOUL) {
            // ...
        }
    }
}

```

### 3. Enum 생성자

- 내부적으로 name, ordinal 이란 이름의 변수를 사용하고 있으므로 변수 추가 시 주의가 필요
- name 은 enum 의 상수명이 값으로 들어감
- ordinal 은 내부에 정의된 상수값(zero base)이 들어감

#### Constructor Summary

##### Constructors

Modifier	Constructor and Description
protected	<b>Enum(String name, int ordinal)</b> Sole constructor.

### 3. Enum 기본 메서드

### Method Summary

All Methods	Static Methods	Instance Methods	Concrete Methods
Modifier and Type		Method and Description	
protected	Object	clone()	Throws CloneNotSupportedException.
int		compareTo(E o)	Compares this enum with the specified object for order.
boolean		equals(Object other)	Returns true if the specified object is equal to this enum constant.
protected void		finalize()	enum classes cannot have finalize methods.
Class<E>		getDeclaringClass()	Returns the Class object corresponding to this enum constant's enum type.
int		hashCode()	Returns a hash code for this enum constant.
String		name()	Returns the name of this enum constant, exactly as declared in its enum declaration.
int		ordinal()	Returns the ordinal of this enumeration constant (its position in its enum declaration, where the initial constant is assigned an ordinal of zero).
String		toString()	Returns the name of this enum constant, as contained in the declaration.
static <T extends Enum<T>> T		valueOf(Class<T> enumType, String name)	Returns the enum constant of the specified enum type with the specified name.

## 4. 추가 활용

- 생성자를 재정의하여 멤버를 추가할 수 있다
- 추상메서드를 추가할 수 있다

```
enum City {  
    SEOUL("서울"),  
    JEJU("제주"),  
    BUSAN("부산"),  
    DAEGU("대구"),  
    GWANGJU("광주");  
  
    private String cityName;  
  
    City(String cityName) {  
        this.cityName = cityName;  
    }  
  
    public String getCityName() {  
        return this.cityName;  
    }  
}
```