



JAVA

제네릭 (Generic)

한다은

(<https://github.com/RayJUNRein/JavaProgramming.git>)

◆ 제네릭 (Generic)

- : 변수의 선언이나 매서드의 매개변수를 하나의 참조 자료형이 아닌 여러 자료형으로 변환 될 수 있도록 함
데이터 타입을 컴파일 시에 미리 지정
- 컬렉션 프레임워크에서 많이 사용

✓ Object VS Generic

1. Object

- JDK 1.5 이전) 여러 타입 사용하는 대부분의 클래스나 매서드에서 인수나 반환값으로 사용
- 단점 : 반환된 Object 객체 다시 원하는 타입으로 변환해야 함 (런타임 에러 발생 가능)

2. Generic

- JDK 1.5 이후) 컴파일 시에 미리 타입 검사 수행
- 내부적으로 Object로 동작함
- 장점 : 1) 타입 변환 및 검사에 들어가는 노력 줄일 수 있음 (중복 제거)
2) 객체의 타입 안정성 높일 수 있음 (컴파일 단계에서 에러 검출 가능)

◆ 타입 변수 (Type Variable)

- 임의의 참조형 타입, 여러 데이터 타입 필요한 곳에 **동일한 문자로 사용** (ex) T, E, V
- **생성 시 실제 데이터 타입 명시해야 함**
- 중복 사용 가능 (심표로 구분)

✓ 선언 및 생성

```
class MyClass<T, E> {  
    T type;  
    E element;  
    MyClass(T type, E element) {  
        this.type = type;  
        this.element = element;  
    }  
}
```

```
public class GenericTest {  
    public static void main(String[] args) {  
        MyClass<TYPE, Integer> mc = new MyClass<TYPE, Integer>();  
    }  
}
```