



## 제네릭 (Generic)

‘컴파일 단계’에서 ‘잘못된 타입 사용될 수 있는 문제’ 제거 가능.

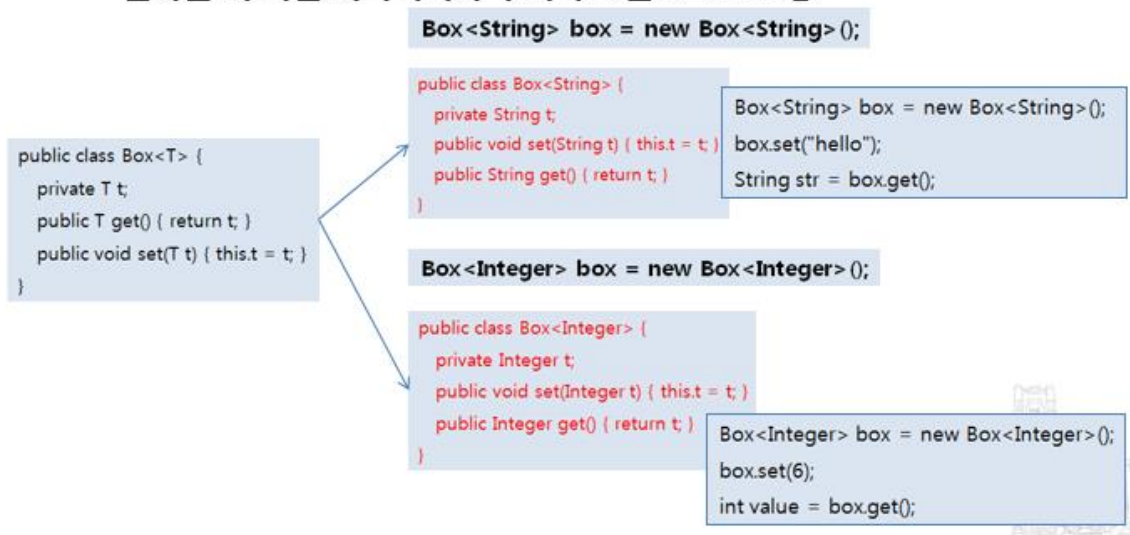
자바5부터 새로 추가됨. 제네릭을 모르면 자바 API 문서를 이해하기 어렵기 때문에 학습이 필요함.

선언 시 클래스 또는 인터페이스 이름 뒤에 “<>”를 붙이고 “<>” 사이에는 타입 파라미터가 위치.

### ❖ 제네릭 타입 사용 여부에 따른 비교

#### ■ 제네릭 타입 사용한 경우

- 클래스 선언할 때 타입 파라미터 사용
- 컴파일 시 타입 파라미터가 구체적인 클래스로 변경



### ArrayList에 객체타입이 저장되는 예제 코드

```
import java.util.ArrayList;  
  
public class GenericEx1 {  
    public static void main(String[] args) {  
        StudentDAO dao = new StudentDAO();  
  
        dao.add(new Student("S001", "홍길동", 18));  
        dao.add(new Student("S002", "김삿갓", 8));  
        dao.add(new Student("S003", "이순신", 10));  
  
        for(Student s : dao.getAll()) {  
            s.displayInfo();  
        }  
    }  
}
```

```

class StudentDAO {
    private ArrayList<Student> list = new ArrayList<Student>(); // ctrl + 1 눌러 import 한다

    public void add(Student s) {
        list.add(s);
    }

    public ArrayList<Student> getAll() {
        return list;
    }
}

class Student {
    private String studentNo;
    private String name;
    private int age;

    public Student(String studentNo, String name, int age) { // 메뉴에서 생성자 만드시오
        super();
        this.studentNo = studentNo;
        this.name = name;
        this.age = age;
    }

    public void displayInfo() {
        System.out.println("학번: " + studentNo);
        System.out.println("이름: " + name);
        System.out.println("나이: " + age);
        System.out.println("-----");
    }

    public String getStudentNo() { // 메뉴에서 getter, setter를 만드시오
        return studentNo;
    }

    public void setStudentNo(String studentNo) {
        this.studentNo = studentNo;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }
}

```

학번: S001

이름: 홍길동

나이: 18

-----

학번: S002

이름: 김삿갓

나이: 8

학번: S003

이름: 이순신

나이: 10

```
import java.util.ArrayList;

public class GenericEx1 {
    public static void main(String[] args) {
        DAO<Student> dao = new DAO<Student>();

        dao.add(new Student("S001", "홍길동", 18));
        dao.add(new Student("S002", "김삿갓", 8));
        dao.add(new Student("S003", "이순신", 10));

        dao.printList();

        DAO<Employee> empDAO = new DAO<Employee>();
        empDAO.add(new Employee("1001", "나사원"));
        empDAO.add(new Employee("1002", "장나라"));
        empDAO.add(new Employee("1003", "장동건"));

        empDAO.printList();
    }
}

class DAO<T extends Displayable> {
    private ArrayList<T> list = new ArrayList<T>(); // ctrl + 1 눌러 import 한다

    public void add(T s) {
        list.add(s);
    }

    public ArrayList<T> getAll() {
        return list;
    }

    public void printList() {
        for(T s : getAll()) {
            s.displayInfo();
        }
    }
}

interface Displayable {
    void displayInfo();
}

class Student implements Displayable {
    private String studentNo;
    private String name;
    private int age;

    public Student(String studentNo, String name, int age) {
```

```

        super();
        this.studentNo = studentNo;
        this.name = name;
        this.age = age;
    }

    public void displayInfo() {
        System.out.println("학번: " + studentNo);
        System.out.println("이름: " + name);
        System.out.println("나이: " + age);
        System.out.println("-----");
    }

    public String getStudentNo() {
        return studentNo;
    }

    public void setStudentNo(String studentNo) {
        this.studentNo = studentNo;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }
}

class Employee implements Displayable {
    private String empNo;
    private String name;

    public Employee(String empNo, String name) {
        super();
        this.empNo = empNo;
        this.name = name;
    }

    public void displayInfo() {
        System.out.println("사번: " + empNo);
        System.out.println("이름: " + name);
        System.out.println("-----");
    }

    public String getEmpNo() {
        return empNo;
    }

    public void setEmpNo(String empNo) {
        this.empNo = empNo;
    }

    public String getName() {

```

```
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
}
```

학번: S001

이름: 홍길동

나이: 18

-----

학번: S002

이름: 김삿갓

나이: 8

-----

학번: S003

이름: 이순신

나이: 10

-----

사번: 1001

이름: 나사원

-----

사번: 1002

이름: 장나라

-----

사번: 1003

이름: 장동건

-----

THINKING CODING 