

12주차 실습 보고서



강의명	객체지향프로그래밍및실습
담당교수	류기열
학과 학년	소프트웨어학과 2학년
학번	202220209
작성자	이육준

2025.11.20.

□ 1

(가) Pair<Employee> minmax(Employee[])

```
22  class ArrayAlg
38      public static Pair<Employee> minmax(Employee[] emp){
39          if(emp == null || emp.length == 0) return null;
40          Employee min = emp[0];
41          Employee max = emp[0];
42          for(int i = 0;i<emp.length;i++){
43              if(min.compareTo(emp[i]) > 0) min = emp[i];
44              if(max.compareTo(emp[i]) < 0) max = emp[i];
45          }
46          return new Pair<>(min, max);
47      }
48  }
```

(나) PairTest1

```
● lwj@lwj-code:~/workspace/JAVA/12$ javac PairTest1.java
● lwj@lwj-code:~/workspace/JAVA/12$ java PairTest1
min = Mary
max = little
min = [Hong,80000.0,2017-05-01]
max = [Lee,200000.0,2005-09-01]
```

(다) Generic

"T cannot be resolved to a type"이라는 error가 발생한다. Generic이 먼저 선언된 후에 method가 작성되어야 하는데 Pair<T>로는 Generic 선언이 이루어지지 않기 때문이다. 또한 Generic 선언을 하는 과정에서 Method내에서 사용하는 T.compareTo()에 대한 처리도 필요하다.

(라)

```
48     public static <T extends Comparable<T>> Pair<T> minmax(T[] a){  
49         if (a == null || a.length == 0) return null;  
50         T min = a[0];  
51         T max = a[0];  
52         for (int i = 1; i < a.length; i++){  
53             if (min.compareTo(a[i]) > 0) min = a[i]; |  
54             if (max.compareTo(a[i]) < 0) max = a[i];  
55         }  
56         return new Pair<>(min, max);  
57     }  
  
PROBLEMS 1 TERMINAL OUTPUT DEBUG CONSOLE PORTS  
  
PS C:\Users\corn7\OneDrive\바탕 화면\객체지향프로그래밍\12> & 'C:\Program Files\Java\jdk-11.0.1\bin\java' 'PairTest1'  
min = Mary  
max = little  
min = [Hong,80000.0,2017-05-01]  
max = [Lee,200000.0,2005-09-01]
```

□ 2

(나) Compile Error

The method `changePartner(Pair<Employee>, Employee)` in the type `PairTest2` is not applicable for the arguments (`Pair<Manager>, Manager`), m3는 Manager class로 선언되어 있는데, `changePartner`의 경우 두번째 parameter로 `Employee` 객체를 넘겨받는다. Type이 맞지 않아 error가 발생한다. `Manager`는 `Employee`를 상속받기 때문에 이를 Generic 선언에 추가해주면 해결할 수 있다.

(다) Bounded Type Parameter

Main 수정 전

```
2 public class PairTest2
3 {
4     Run | Debug
5     public static void main(String[] args)
6     {
7         Employee e1 = new Employee(name: "Carl Cracker", salary: 75000, year: 2007, month: 12, day: 15);
8         Employee e2 = new Employee(name: "Gus Greedy", salary: 80000, year: 2000, month: 1, day: 15);
9         Pair<Employee> p = new Pair<>(e1,e2);
10        System.out.println("First = "+p.getFirst());
11        System.out.println("Second = "+p.getSecond());
12
13        Employee e3 = new Employee(name: "Sid Sneaky", salary: 100000, year: 1989, month: 1, day: 1);
14
15        changePartner(p, e3);
16        System.out.println("First = "+p.getFirst());
17        System.out.println("Second = "+p.getSecond());
18        // Manager m1 = new Manager("Sarah Smith", 105000, 2007, 12, 15);
19        // Manager m2 = new Manager("Bobby Brown", 1280000, 2000, 1, 15);
20        // Pair<Manager> p2 = new Pair<>(m1,m2);
21        // System.out.println("First = "+p2.getFirst());
22        // System.out.println("Second = "+p2.getSecond());
23        // Manager m3 = new Manager("Mike Miller", 200000, 1989, 1,1);
24        // changePartner(p2, m3);
25        // System.out.println("First = "+p2.getFirst());
26        // System.out.println("Second = "+p2.getSecond());
27    }
28
29    public static <T extends Employee> void changePartner(Pair<T> p, T e)
30    {
31        p.setSecond(e);
32    }
}
PROBLEMS 1 TERMINAL OUTPUT DEBUG CONSOLE PORTS
PS C:\Users\corn7\OneDrive\바탕 화면\ 객체지향프로그래밍\12> cd 'c:\Users\corn7\OneDrive\바탕 화면\ 객체지향프로그래밍\12'
First = [Carl Cracker,75000.0,2007-12-15]
Second = [Gus Greedy,80000.0,2000-01-15]
First = [Carl Cracker,75000.0,2007-12-15]
Second = [Sid Sneaky,100000.0,1989-01-01]
```

Main 수정 후

```
2  public class PairTest2
3  {
4      Run | Debug
5      public static void main(String[] args)
6      {
7          // Employee e1 = new Employee("Carl Cracker", 75000, 2007, 12, 15);
8          // Employee e2 = new Employee("Gus Greedy", 80000, 2000, 1, 15);
9          // Pair<Employee> p = new Pair<>(e1,e2);
10         // System.out.println("First = "+p.getFirst());
11         // System.out.println("Second = "+p.getSecond());
12
13         // Employee e3 = new Employee("Sid Sneaky", 100000, 1989, 1,1);
14
15         // changePartner(p, e3);
16         // System.out.println("First = "+p.getFirst());
17         // System.out.println("Second = "+p.getSecond());
18         Manager m1 = new Manager(name: "Sarah Smith", salary: 105000, year: 2007, month: 12, day: 15);
19         Manager m2 = new Manager(name: "Bobby Brown", salary: 1280000, year: 2000, month: 1, day: 15);
20         Pair<Manager> p2 = new Pair<>(m1,m2);
21         System.out.println("First = "+p2.getFirst());
22         System.out.println("Second = "+p2.getSecond());
23         Manager m3 = new Manager(name: "Mike Miller", salary: 200000, year: 1989, month: 1,day: 1);
24         changePartner(p2, m3);
25         System.out.println("First = "+p2.getFirst());
26         System.out.println("Second = "+p2.getSecond());
27     }
28
29     public static <T extends Employee> void changePartner(Pair<T> p, T e)
30     {
31         p.setSecond(e);
32     }
}
```

PROBLEMS 1 TERMINAL OUTPUT DEBUG CONSOLE PORTS

```
PS C:\Users\corn7\OneDrive\바탕 화면\객체지향프로그래밍\12> c:; cd 'c:\Users\corn7\OneDrive\바탕 화면\객체
sers\corn7\AppData\Roaming\Code\User\workspaceStorage\62e35814a1959c24b191789fbe19541f\redhat.java\jdt_ws\1
First = [Sarah Smith,105000.0,2007-12-15][0.0]
Second = [Bobby Brown,1280000.0,2000-01-15][0.0]
First = [Sarah Smith,105000.0,2007-12-15][0.0]
Second = [Mike Miller,200000.0,1989-01-01][0.0]
```

□ 3

(가)

The method printEmployee(ArrayList<Employee>) in the type Test3 is not applicable for the arguments (ArrayList<Manager>), printEmployee Method는 parameter로 ArrayList<Employee>를 받고 있으나 Method를 호출할 때 parameter로 ArrayList<Manager>를 넘겨주어 type이 맞지 않아 Error가 발생함

(나)

```
1 import java.util.*;
2
3 public class Test3
4 {
5     Run | Debug
6     public static void main(String[] args)
7     {
8         ArrayList<Employee> le = new ArrayList<>();
9         le.add(new Employee(name: "Carl Cracker", salary: 75000, year: 2007, month: 12, day: 15));
10        le.add(new Employee(name: "Gus Greedy", salary: 80000, year: 2000, month: 1, day: 15));
11        le.add(new Employee(name: "Sid Sneaky", salary: 100000, year: 1989, month: 1, day: 1));
12
13        printEmployee(le);
14
15        ArrayList<Manager> lm = new ArrayList<>();
16        lm.add(new Manager(name: "Sarah Smith", salary: 105000, year: 2007, month: 12, day: 15));
17        lm.add(new Manager(name: "Bobby Brown", salary: 1280000, year: 2000, month: 1, day: 15));
18        lm.add(new Manager(name: "Mike Miller", salary: 200000, year: 1989, month: 1, day: 1));
19
20        printEmployee(lm);
21    }
22
23    public static void printEmployee(ArrayList<? extends Employee> le)
24    {
25        for(Employee e : le)
26            System.out.println(e);
27    }
28
29 }
```

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE PORTS

```
PS C:\Users\corn7\OneDrive\바탕 화면\객체지향프로그래밍\12> & 'C:\Program Files\Java\jdk-11.0.2\bin\java.9c24b191789fbe19541f\redhat.java\jdt_ws\12_a5fb28d3\bin' 'Test3'
[Carl Cracker,75000.0,2007-12-15]
[Gus Greedy,80000.0,2000-01-15]
[Sid Sneaky,100000.0,1989-01-01]
[Sarah Smith,105000.0,2007-12-15][0.0]
[Bobby Brown,1280000.0,2000-01-15][0.0]
[Mike Miller,200000.0,1989-01-01][0.0]
```

□ 코드 전문

PairTest1.java

```
public class PairTest1
{
    public static void main(String[] args)
    {
        String[] words = { "Mary", "had", "a", "little", "lamb" };
        Pair<String> mm = ArrayAlg.minmax(words);
        System.out.println("min = " + mm.getFirst());
        System.out.println("max = " + mm.getSecond());
        //

        Employee[] emps = new Employee[4];
        emps[0] = new Employee("Kim", 100000.0, 2015, 3, 1);
        emps[1] = new Employee("Lee", 200000.0, 2005, 9, 1);
        emps[2] = new Employee("Hong", 80000.0, 2017, 5, 1);
        emps[3] = new Employee("Park", 150000.0, 2008, 3, 1);
        Pair<Employee> ee = ArrayAlg.minmax(emps);
        System.out.println("min = " + ee.getFirst());
        System.out.println("max = " + ee.getSecond());
    }
}

class ArrayAlg
{
    // public static Pair<String> minmax(String[] a)
    // {
    //     if (a == null || a.length == 0) return null;
    //     String min = a[0];
    //     String max = a[0];
    //     for (int i = 1; i < a.length; i++)
    //     {
    //         if (min.compareTo(a[i]) > 0) min = a[i];
    //         if (max.compareTo(a[i]) < 0) max = a[i];
    //     }
    //     return new Pair<>(min, max);
    // }

    // public static Pair<Employee> minmax(Employee[] emp){
    //     if(emp == null || emp.Length == 0) return null;
    //     Employee min = emp[0];
    //     Employee max = emp[0];
    //     for(int i = 0;i<emp.Length;i++){
    //         if(min.compareTo(emp[i]) > 0) min = emp[i];
    //         if(max.compareTo(emp[i]) < 0) max = emp[i];
    //     }
    // }
```

```

        //
        }
        //    return new Pair<>(min, max);
    // }

    public static <T extends Comparable<T>> Pair<T> minmax(T[] a){
        if (a == null || a.length == 0) return null;
        T min = a[0];
        T max = a[0];
        for (int i = 1; i < a.length; i++){
            if (min.compareTo(a[i]) > 0) min = a[i];
            if (max.compareTo(a[i]) < 0) max = a[i];
        }
        return new Pair<>(min, max);
    }
}

```

PairTest2.java

```

public class PairTest2
{
    public static void main(String[] args)
    {
        Employee e1 = new Employee("Carl Cracker", 75000, 2007, 12, 15);
        Employee e2 = new Employee("Gus Greedy", 80000, 2000, 1, 15);
        Pair<Employee> p = new Pair<>(e1,e2);
        System.out.println("First = "+p.getFirst());
        System.out.println("Second = "+p.getSecond());

        Employee e3 = new Employee("Sid Sneaky", 100000, 1989, 1,1);

        changePartner(p, e3);
        System.out.println("First = "+p.getFirst());
        System.out.println("Second = "+p.getSecond());
        // Manager m1 = new Manager("Sarah Smith", 105000, 2007, 12, 15);
        // Manager m2 = new Manager("Bobby Brown", 1280000, 2000, 1, 15);
        // Pair<Manager> p2 = new Pair<>(m1,m2);
        // System.out.println("First = "+p2.getFirst());
        // System.out.println("Second = "+p2.getSecond());
        // Manager m3 = new Manager("Mike Miller", 200000, 1989, 1,1);
        // changePartner(p2, m3);
        // System.out.println("First = "+p2.getFirst());
        // System.out.println("Second = "+p2.getSecond());
    }
}

```

```
public static <T extends Employee> void changePartner(Pair<T> p, T e)
{
    p.setSecond(e);
}
```

Test3.java

```
import java.util.*;

public class Test3
{
    public static void main(String[] args)
    {
        ArrayList<Employee> le = new ArrayList<>();
        le.add(new Employee("Carl Cracker", 75000, 2007, 12, 15));
        le.add(new Employee("Gus Greedy", 80000, 2000, 1, 15));
        le.add(new Employee("Sid Sneaky", 100000, 1989, 1,1));

        printEmployee(le);

        ArrayList<Manager> lm = new ArrayList<>();
        lm.add(new Manager("Sarah Smith", 105000, 2007, 12, 15));
        lm.add(new Manager("Bobby Brown", 1280000, 2000, 1, 15));
        lm.add(new Manager("Mike Miller", 200000, 1989, 1,1));

        printEmployee(lm);
    }

    public static void printEmployee(ArrayList<? extends Employee> le)
    {
        for(Employee e : le)
            System.out.println(e);
    }
}
```