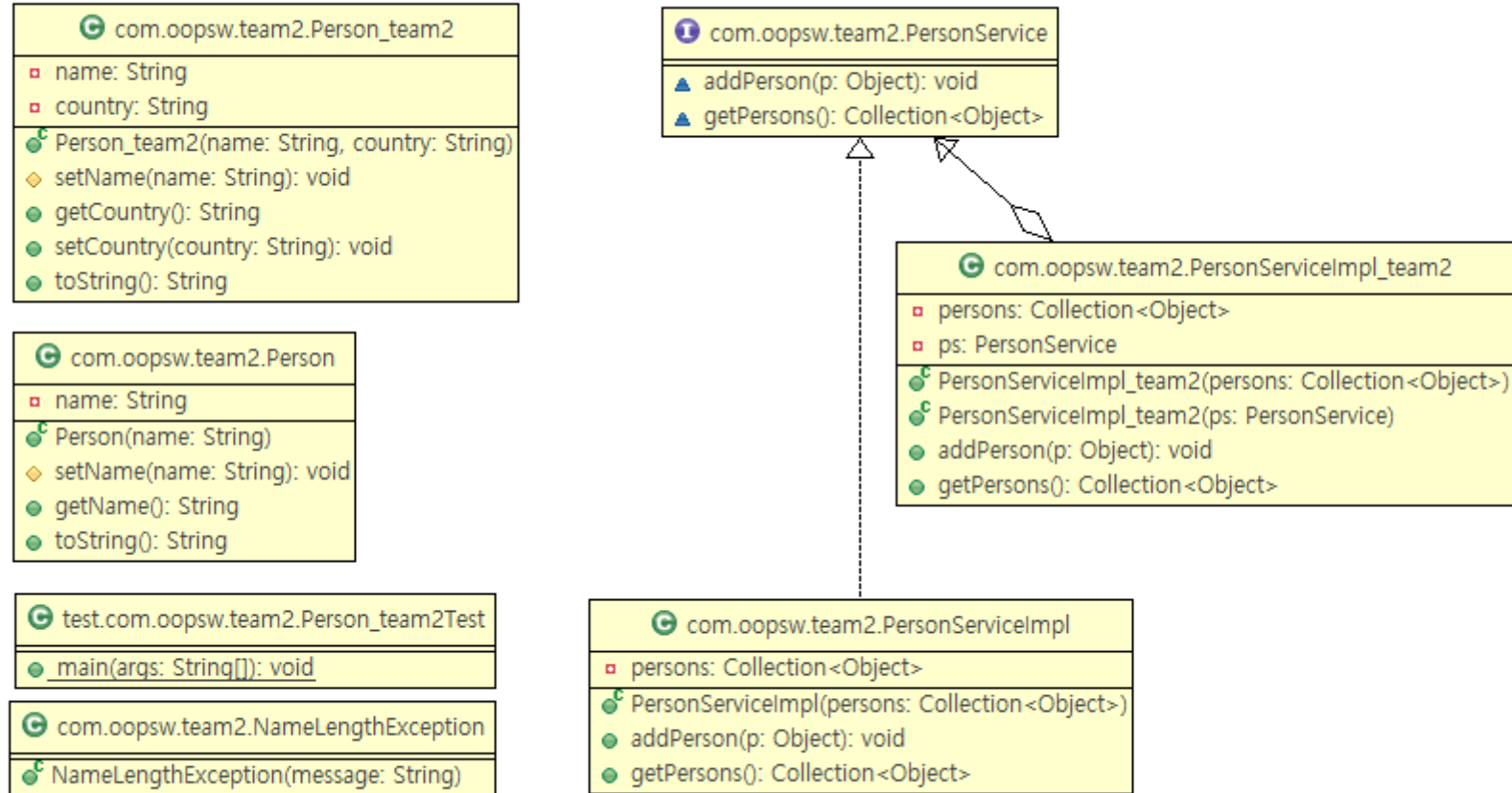


# OOP 디자인패턴 #2

A014손은빈

구조패턴 사용하기

# ClassDiagram



# 프로그램 소스

Person\_team2.java --> 리팩터링  
을 위해 별도의 데이터 클래스를  
작성함

```
1 package com.oopsw.team2;
2
3 public class Person_team2 {
4
5     private String name;
6     private String country;
7
8     public Person_team2(String name, String country) {
9         setName(name);
10        setCountry(country);
11    }
12
13    protected void setName(String name) {
14        if(name.length() > 50)
15            throw new RuntimeException("이름은 50자 이하만 가능");
16        this.name = name;
17    }
18    public String getCountry() {
19        return country;
20    }
21    public void setCountry(String country) {
22        this.country = country;
23    }
24    @Override
25    public String toString() {
26        return "이름=" + name + ", 국적=" + country;
27    }
28
29 }
30
```

# 프로그램 소스

PersonService.java --> 리팩터링 전  
인터페이스를 가정함

```
1 package com.oopsw.team2;
2
3 import java.util.Collection;
4
5 public interface PersonService {
6
7     /* 이름 50자까지 허용 */
8     void addPerson(Object p);
9
10    /* 국적 확인 */
11    Collection<Object> getPersons();
12
13 }
14
15
```

# 프로그램 소스

PersonServiceImpl\_team2 -->  
가정한 인터페이스의 추상 메서드를 구현함

```
1 package com.oopsw.team2;
2
3 import java.util.Collection;
4
5 public class PersonServiceImpl_team2 implements PersonService{
6
7     private Collection<Object> persons;
8     private PersonService ps;
9
10    public PersonServiceImpl_team2(Collection<Object> persons) {
11        this.persons = persons;
12    }
13
14    public PersonServiceImpl_team2(PersonService ps) {
15        this.ps = ps;
16    }
17
18    @Override
19    public void addPerson(Object p) {
20        ps.addPerson(p);
21    }
22
23    @Override
24    public Collection<Object> getPersons() {
25        return ps.getPersons();
26    }
27
28 }
```

# 테스트 실행 결과

사용자 정의  
예외를 사용  
한 경우

```
18      try {
19          p.addPerson(new Person_team2("김사람", "한국"));
20          p.addPerson(new Person_team2("박사람", "미국"));
21          p.addPerson(new Person_team2("Lorem Ipsum is simply dummy text of
22      }catch(RuntimeException e) {
23          e.getMessage();
24          e.printStackTrace();
25      }catch(Exception e) {
26          e.getMessage();
27          e.printStackTrace();
28      }
```

Console

<terminated> Person\_team2Test [Java Application] C:\Program Files\Java\jdk1.8.0\_202\bin\javaw.exe (2022. 3. 25. 오후 10:16:25)

com.oopsw.team2.NameLengthException: 이름은 최대 50자까지 가능합니다.

at com.oopsw.team2.Person\_team2.setName(Person\_team2.java:16)

at com.oopsw.team2.Person\_team2.<init>(Person\_team2.java:9)

at test.com.oopsw.team2.Person\_team2Test.main(Person\_team2Test.java:21)

# 테스트 실행 결과

이름 필드의  
길이가 50자  
이상일 때,  
RuntimeException이 발생함

```
18     try {
19         p.addPerson(new Person_team2("김사람", "한국"));
20         p.addPerson(new Person_team2("박사람", "미국"));
21         p.addPerson(new Person_team2("Lorem Ipsum is simply dummy text of
22     } catch (RuntimeException e) {
23         e.getMessage();
24         e.printStackTrace();
25     } catch (Exception e) {
26         e.getMessage();
27         e.printStackTrace();
28     }
29 }
```

Console

<terminated> Person\_team2Test [Java Application] C:\Program Files\Java\jdk1.8.0\_202\bin\javaw.exe (2022. 3. 25. 오후 9:20:09)

java.lang.RuntimeException: 이름은 50자 이하만 가능

at com.oopsw.team2.Person\_team2.setName(Person\_team2.java:15)

at com.oopsw.team2.Person\_team2.<init>(Person\_team2.java:9)

at test.com.oopsw.team2.Person\_team2Test.main(Person\_team2Test.java:21)

# 테스트 실행 결과

사용자 정의  
예외를 사용한  
경우

```
13 protected void setName(String name) {  
14     if(name.length() > 50)  
15 //         throw new RuntimeException("이름은 50자 이하만 가능");  
16         throw new NameLengthException("이름은 최대 50자까지 가능합니다.");  
17     this.name = name;  
18 }
```

Console

<terminated> Person\_team2Test [Java Application] C:\Program Files\Java\jdk1.8.0\_202\bin\javaw.exe (2022. 3. 25. 오후 10:16:25)

com.oopsw.team2.NameLengthException: 이름은 최대 50자까지 가능합니다.  
at com.oopsw.team2.Person\_team2.setName(Person\_team2.java:16)  
at com.oopsw.team2.Person\_team2.<init>(Person\_team2.java:9)  
at test.com.oopsw.team2.Person\_team2Test.main(Person\_team2Test.java:21)



# 테스트 실행 결과

객체 정보를 출력  
함 --> toString()

```
31      try {  
32          for (Object person : p.getPersons()) {  
33              System.out.println(person);  
34          }  
35      } catch (RuntimeException e) {  
36          e.getMessage();  
37          e.printStackTrace();  
38      } catch (Exception e) {  
39          e.getMessage();  
40          e.printStackTrace();  
41      }  
42  }
```

Console

<terminated> Person\_team2Test [Java Application] C:\Program Files\Java\jdk1.8.0\_202\bin\javaw.exe (2022. 3. 25. 오후 9:22:22)

이름=김사람, 국적=한국

이름=박사람, 국적=미국

