Java Programming II Lab 3

교 과 명 : 자바프로그래밍 2
담당교수명 : 박경신 교수님
학 과 : 컴퓨터공학과
학 번 : 32185010
성 명 : 홍찬희
제 출 일 : 2021. 09. 30

DKU
단국대학교
DANKOOK UNIVERSITY

Lab 3

옵저버 패턴이란, 한 객체의 상태 변화에 따라 다른 객체의 상태도 연동되는 일대다 객체 의존 관계를 구현하는 패턴이다.
데이터의 변경이 발생하게되면, 상대 클래스나 객체에 의존하지 않으면서 데이터 변경을 통보한다.

Lab3의 MainTest에서는 우선 각각의 동물들을 객체를 선언해주고, AnimalKingdom Subject를 생성해준다.

```
// 각각의 동물종류별 객체들을 Animal 클래스로 upcasting
Animal dog = new Mammal( name: "Dog", move: "walk", breath: "lungs", reproduce: "live birth", numberOfLegs: 4, AnimalType.MAMMAL);
Animal bird = new Bird( name: "Ostrich", move: "fly", breath: "lungs", reproduce: "lay eggs", numberOfLegs: 2, AnimalType.BIRD);
Animal turtle = new Reptile( name: "Turtle", move: "walk", breath: "lungs", reproduce: "lay eggs", numberOfLegs: 4, AnimalType.REPTILE);
Animal frog = new Amphibian( name: "Frog", move: "walk & swim", breath: "lung & gills", reproduce: "lay eggs", numberOfLegs: 4, AnimalType.AMPHIBIAN);
Animal salmon = new Fish( name: "Salman", move: "swim", breath: "gills", reproduce: "lay eggs", numberOfLegs: 0, AnimalType.FISH);

AnimalKingdom kingdom = new AnimalKingdom();
```

그 다음 사용할 Observer들을 kingdom의 addObserver method를 통해 객체를 생성한뒤, 매개변수로 넣어준다.

```
// Observer 추가
kingdom.addObserver(new ListDisplay());
kingdom.addObserver(new PopulationDisplay());
kingdom.addObserver(new SurveyDisplay());
```

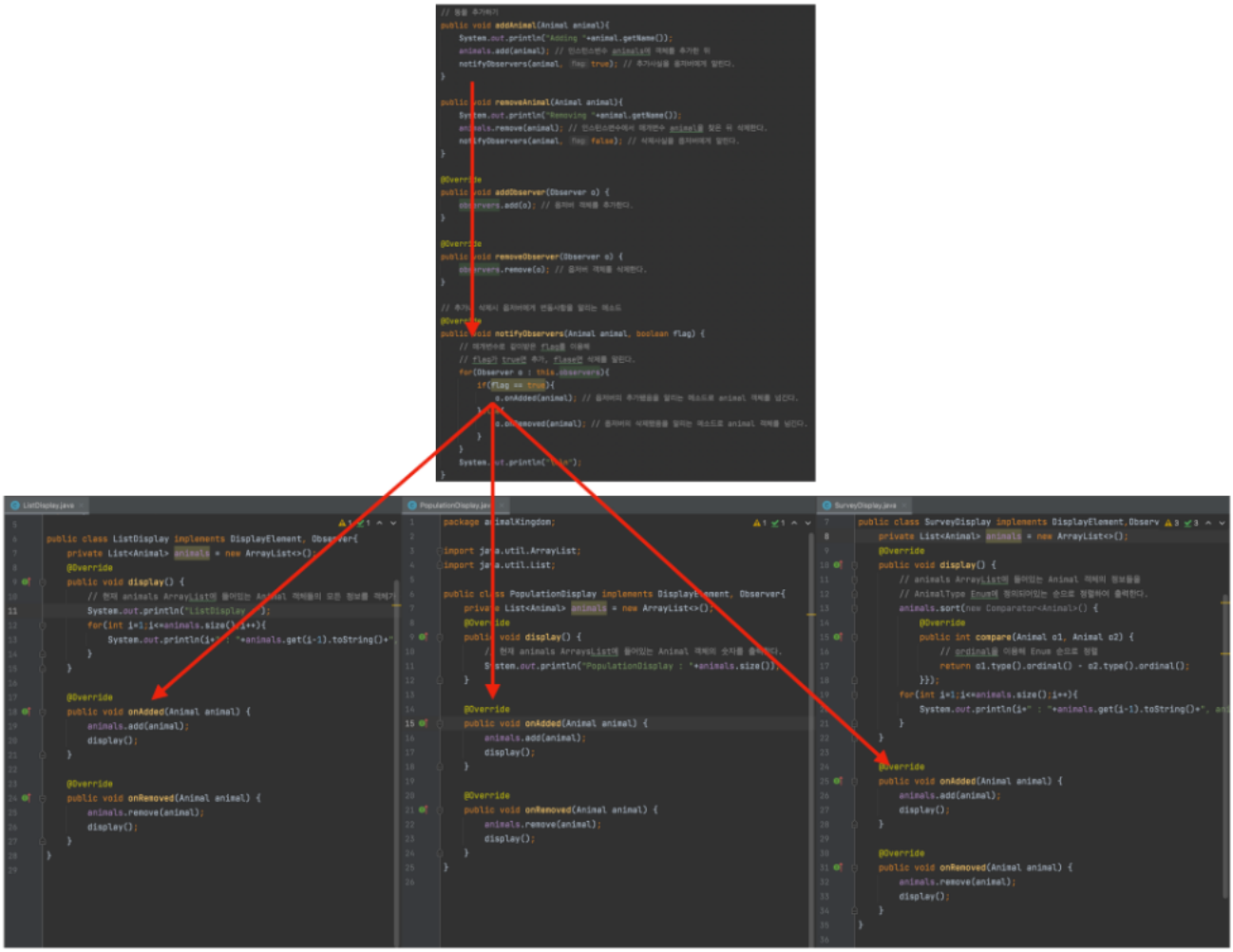
다음과같이 AnimalKingdom 객체의 addAnimal method로 선언해둔 객체를 매개변수로 넣어주게 되면, 각각의 addAnimal method를 실행시킬때마다, 추가해둔 List, Population, SurveyDisplay에게 알리고, 각각의 Observer들은 각 객체별로 구현되어있는 알고리즘별로 해당내용들을 출력한다.

```
// AnimalKingdom에 동물을 추가
kingdom.addAnimal(turtle);
Thread.sleep((int)(Math.random()*2000)); // 0~2초 대기
kingdom.addAnimal(salmon);
Thread.sleep((int)(Math.random()*2000));
kingdom.addAnimal(dog);
Thread.sleep((int)(Math.random()*2000));
kingdom.addAnimal(bird);
Thread.sleep((int)(Math.random()*2000));
kingdom.addAnimal(frog);
```

아래 그림의 출력처럼 AnimalKingdom이 가지고 있는 객체들의 정보들을 각각의 Observer들의 display method를 통해 출력하게 된다. 객체들을 추가할때마다 출력이 발생한다.

```
Adding Dog
ListDisplay :
1 : name='Turtle, move='walk, breath='lungs, reproduce='lay eggs, numberOfLegs=4, animalType=REPTILE
2 : name='Salman, move='swim, breath='gills, reproduce='lay eggs, numberOfLegs=0, animalType=FISH
3 : name='Dog, move='walk, breath='lungs, reproduce='live birth, numberOfLegs=4, animalType=MAMMAL
PopulationDisplay : 3
1 : name='Dog, move='walk, breath='lungs, reproduce='live birth, numberOfLegs=4, animalType=MAMMAL
2 : name='Turtle, move='walk, breath='lungs, reproduce='lay eggs, numberOfLegs=4, animalType=REPTILE
3 : name='Salman, move='swim, breath='gills, reproduce='lay eggs, numberOfLegs=0, animalType=FISH
```

<kingdom.addAnimal(dog); 가 실행되었을때의 출력값>



addAnimal method가 실행되게 되면, AnimalKingdom의 notifyObservers 메소드에 animal 객체와 boolean true 값을 넘긴다. (true = add, false = remove) AnimalKingdom의 notifyObservers 메소드에서는 인스턴스 변수로 저장되어있는 observers ArrayList들을 for-each 구문을 통해 모두 돌면서 매개변수로 받은 boolean 값이 true이므로, 객체가 추가되었다는 onAdded 메소드를 실행시키고 매개변수는 notifyObservers에서 받았던 매개변수 animal을 인자값으로 넘겨준다.

onAdded method로 animal을 매개변수로 받은 List, Population, SurveyDisplay 클래스에서는 각 클래스에 있는 인스턴스 변수인 animals의 ArrayList에 똑같이 Animal 객체를 추가해주고 display method를 통해 각 클래스에 정의되어 있는 형식대로 출력을 해준다.

그렇게 AnimalKingdom 객체의 addAnimal method를 실행시킬 때마다 List, Population, SurveyDisplay 클래스의 display method가 실행되게 되는 것이다.

AnimalKingdom 객체의 removeAnimal method를 실행시킬때도 마찬가지로, notifyObservers를 통해 모든 옵저버들에게 삭제사실을 알려 옵저버의 각각의 객체에서 해당 Animal 객체를 삭제하고, 클래스 각각의 display method를 실행시켜 Animal 객체들의 정보를 출력해준다.

```
Removing Ostrich
ListDisplay :
1 : name='Turtle, move='walk, breath='lungs, reproduce='lay eggs, numberOfLegs=4, animalType=REPTILE
2 : name='Salman, move='swim, breath='gills, reproduce='lay eggs, numberOfLegs=0, animalType=FISH
3 : name='Frog, move='walk & swim, breath='lung & gills, reproduce='lay eggs, numberOfLegs=4, animalType=AMPHIBIAN
PopulationDisplay : 3
1 : name='Turtle, move='walk, breath='lungs, reproduce='lay eggs, numberOfLegs=4, animalType=REPTILE
2 : name='Frog, move='walk & swim, breath='lung & gills, reproduce='lay eggs, numberOfLegs=4, animalType=AMPHIBIAN
3 : name='Salman, move='swim, breath='gills, reproduce='lay eggs, numberOfLegs=0, animalType=FISH
```