

# Java Programming II

## Lab3

---

514770-1

Fall 2025

9/25/2025

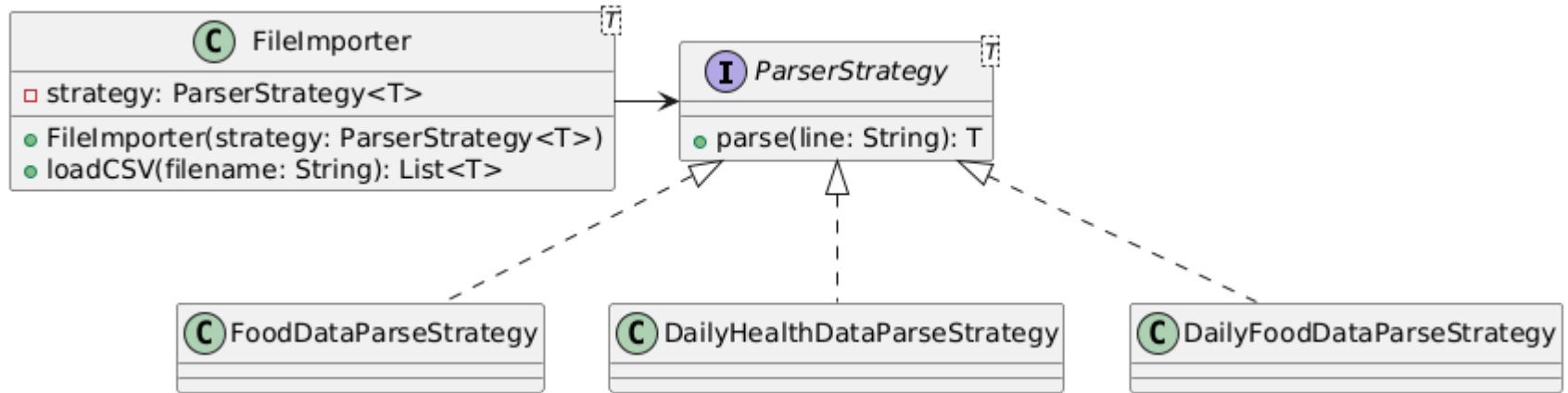
Kyoung Shin Park  
Computer Engineering  
Dankook University

# Lab3

- Observer 패턴을 사용하여 혈중당수치, 탄수화물섭취량, 인슐린수치를 관찰 중인 객체에 각 수치가 넘치는 상태이면 Alert을 통지하는 프로그램을 작성하라.
  - DailyHealthData - Date, BloodSugarLevel, InsulinDose, CarbsIntake
  - FoodData - Food, Carbohydrates (g), Calories (kcal)
  - DailyFoodData - Date, List of Food
    - 2024-09-01, "Apple, Banana" 데이터는 `date`와 `Apple`과 `Banana`를 가진 `List<FoodData>`를 의미한다
  - DiabetesManager는 `BloodSugarObserver`, `CarbsIntakeObserver`, `InsulinObserver`를 자유롭게 추가 또는 삭제 가능하다
    - DiabetesManager에 새 `DailyHealthData`가 추가됐을때, 등록되어있는 모든 `Observers`들에게 통지한다

# Lab3

- Strategy 패턴을 사용하여 3개 파일을 읽어 List로 만든다



- `List<FoodData> foodDataList = new FileImporter<>(new FoodDataParseStrategy()).loadCSV("fooddata.csv");`
- `List<DailyFoodData> dailyFoodDataList = new FileImporter<>(new DailyFoodDataParseStrategy(foodDataList)).loadCSV("dailyfooddata.csv");`
- `List<DailyHealthData> dailyHealthDataList = new FileImporter<>(new DailyHealthDataParseStrategy()).loadCSV("dailyhealthdata.csv");`

# Lab3

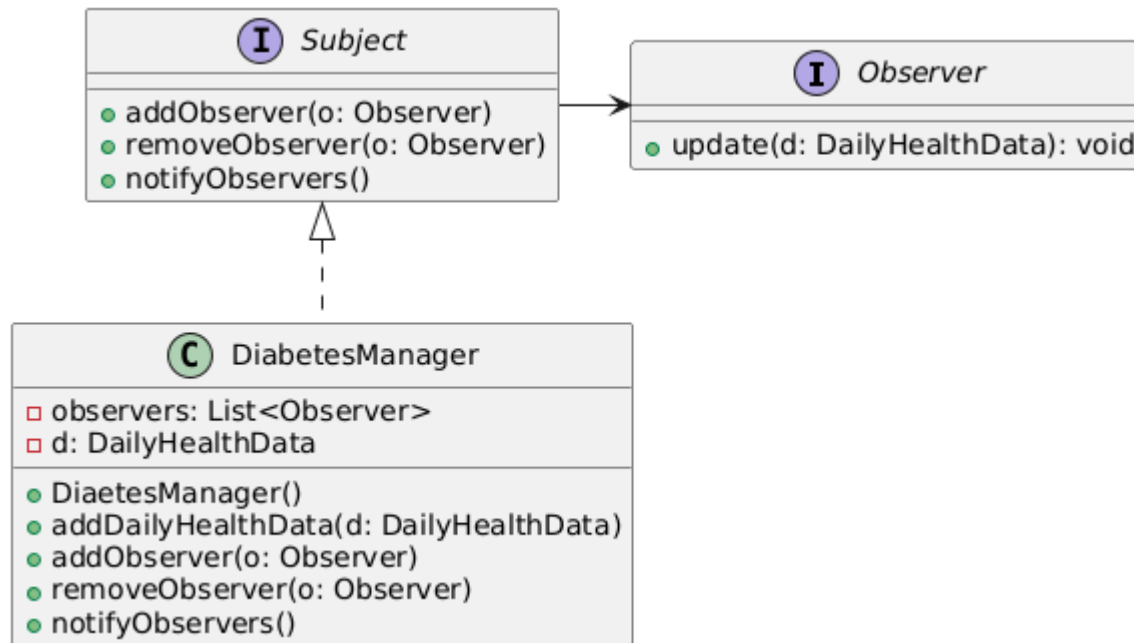
```
public class DailyHealthData {
    private String date;
    private double bloodSugarLevel;
    private double insulinDose;
    private double carbsIntake;
...}

public class FoodData {
    private String name;
    private double carbs;
    private double calories;
... }

public class DailyFoodData {
    private String date;
    private List<FoodData> foods;
... }
```

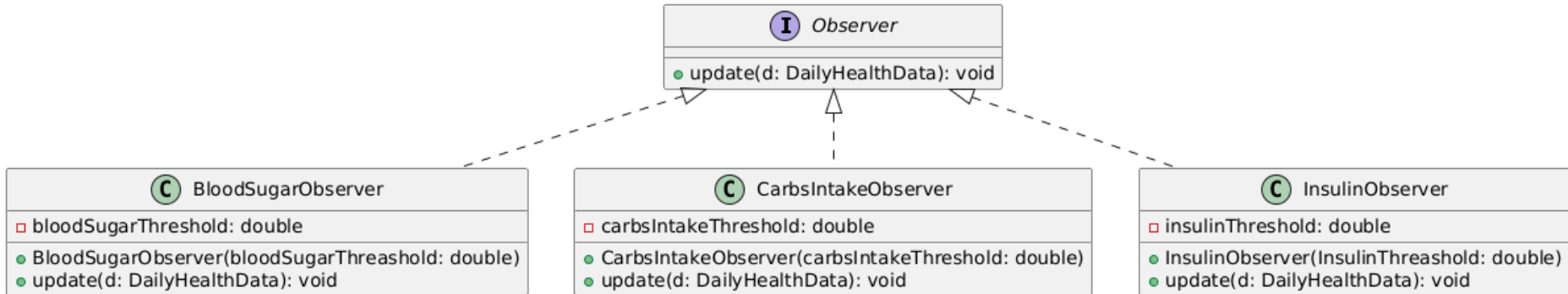
# Lab3

- **DiabetesManager**의 addDailyHealthData는 데이터를 추가하고 notifyObservers를 호출한다.
  - **notifyObservers()** updates **DailyHealthData** to all subscribed observers.



# Lab3

## □ Observers



- BloodSugarObserver – update 에서 DailyHealthData의 혈중당수치가 bloodSugarThreshold보다 넘치는 경우 Blood Sugar Alert 출력.
- CarbsIntakeObserver – update 에서 DailyHealthData의 탄수화물섭취량이 carbsIntakeThreshold보다 넘치는 경우 Carb Intake Alert 출력.
- InsulinObserver – update 에서 DailyHealthData에서 인슐린수치가 insulineThreshold보다 넘치는 경우 Insulin Dose Alert 출력

# Lab3

```
public class BloodSugarObserver
implements Observer {
    double bloodSugarThreshold;
    public BloodSugarObserver(double
bloodSugarThreshold) {
        this.bloodSugarThreshold = bloodSugarThreshold;
    }
    @Override
    public void update(DailyHealthData d){
        // if d.getBlookSugarLevel() >
bloodSugarThreshold, then print ALERT on high blood
sugar level
    }
}
```

# Lab3

```
public class MainTest {  
    public MainTest() {  
        // Load fooddata, dailyfooddata, dailyhealthdata  
        List<FoodData> foodDataList = ... // 이전 슬라이드  
        List<DailyFoodData> dailyFoodDataList = ...  
        List<DailyHealthData> dailyHealthDataList = ...  
        // Subject  
        DiabetesManager manager = new DiabetesManager();  
        // Observer  
        BloodSugarObserver bo = new BloodSugarObserver(150);  
        CarbIntakeObserver co = new CarbsIntakeObserver(100);  
        InsulinObserver io = new InsulinObserver(10);  
        manager.addObserver(bo);  
        manager.addObserver(co);  
        manager.addObserver(io);  
    }  
}
```



# Lab3

```
new Thread(() -> { // Simulate food affecting carbs & insulin
    for (DailyHealthData h : dailyHealthDataList) {
        DailyFoodData f = dailyFoodDataList.stream().filter(e ->
            e.getDate().equals(h.getDate())).findAny().orElse(null);
        double totalCarbs = f.getFoods().stream().mapToDouble(e ->
            e.getCarbs()).sum();
        totalCarbs += h.getCarbsIntake();
        h.setCarbsIntake(totalCarbs);
        double insulinDose = (h.getInsulinDose() + totalCarbs) / 10.0;
        h.setInsulinDose(insulinDose);
        manager.addDailyHealthData(h);
        try {
            Thread.sleep(1000); // Wait for 1 second before next update
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        System.out.println("\n\n\n");
    }
}).start();
}}
```

# Submit to e-learning

---

- 메인에 Observer 추가 삭제 루틴 포함할 것
- Lab3 과제에 yourcode (e.g.: 다른 Observer 예 또는 Food affecting health data)를 추가 (yourcode 없을시 10점에서 -1점 감점)
- **Java24-2-HW3-YourID-YourName.zip** 과제(보고서에 반드시 yourcode 설명 포함)를 e러닝에 제출 (**due by 10/1**).