

# JavaScript Design Pattern

observer Pattern

# 패턴의 범주별 분류

## 생성 관련 패턴

객체 인스턴스 생성을 위한 패턴으로, 클라이언트와 그 클라이언트에서 생성해야 할 객체 인스턴스 사이의 연결을 끊어주는 패턴

싱글턴,  
추상 팩토리,  
팩토리 메소드,  
빌더,  
프로토타입

## 행동 관련 패턴

클래스와 객체들의 상호작용 하는 방법 및 역할을 분담하는 방법과 관련된 패턴.

템플릿 메소드,  
커맨드,  
이터레이터,  
**옵저버**,  
스테이트,  
스트래티지

## 구조 관련 패턴

클래스 및 객체들을 구성을 통째로 더 큰 구조로 만들 수 있게 해주는 것과 관련된 패턴.

데코레이터,  
컴포지트,  
프록시,  
퍼사드,  
어댑터

# Observer Pattern

## Observer



- a.k.a 발행/구독 패턴
- 객체의 상태 변화를 관찰하는 관찰자(옵저버)를 객체에 등록(구독)하여, 변화가 있을때마다 등록된 목록의 관찰자들에게 통지 하도록 하는 패턴
- Broadcast System을 구축할 수 있음  
각 모듈을 느슨한 결합으로 만들어 , 재사용성을 높임.

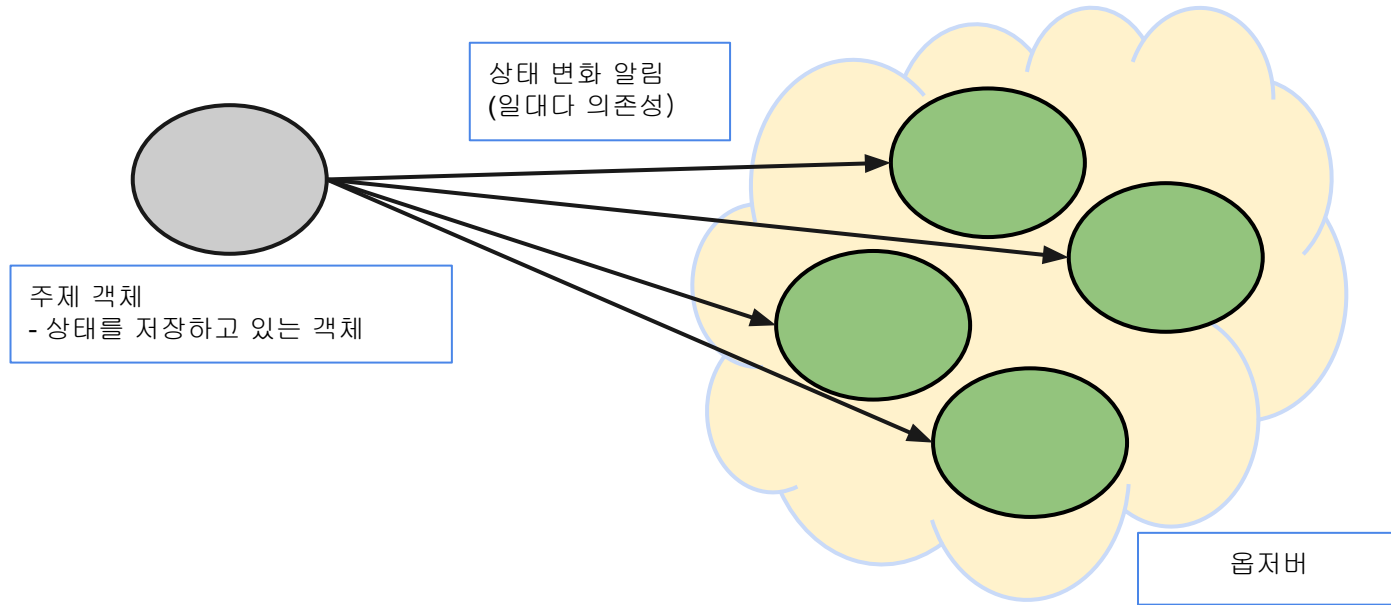
# Observer Pattern Example

-신문 구독 매커니즘

1. 신문사 -> 신문발행
2. 독자가 특정 신문사에 구독 신청
3. 신문 구독 해지 신청
4. 신문사가 영업하는 한 구독 및 해지 발생

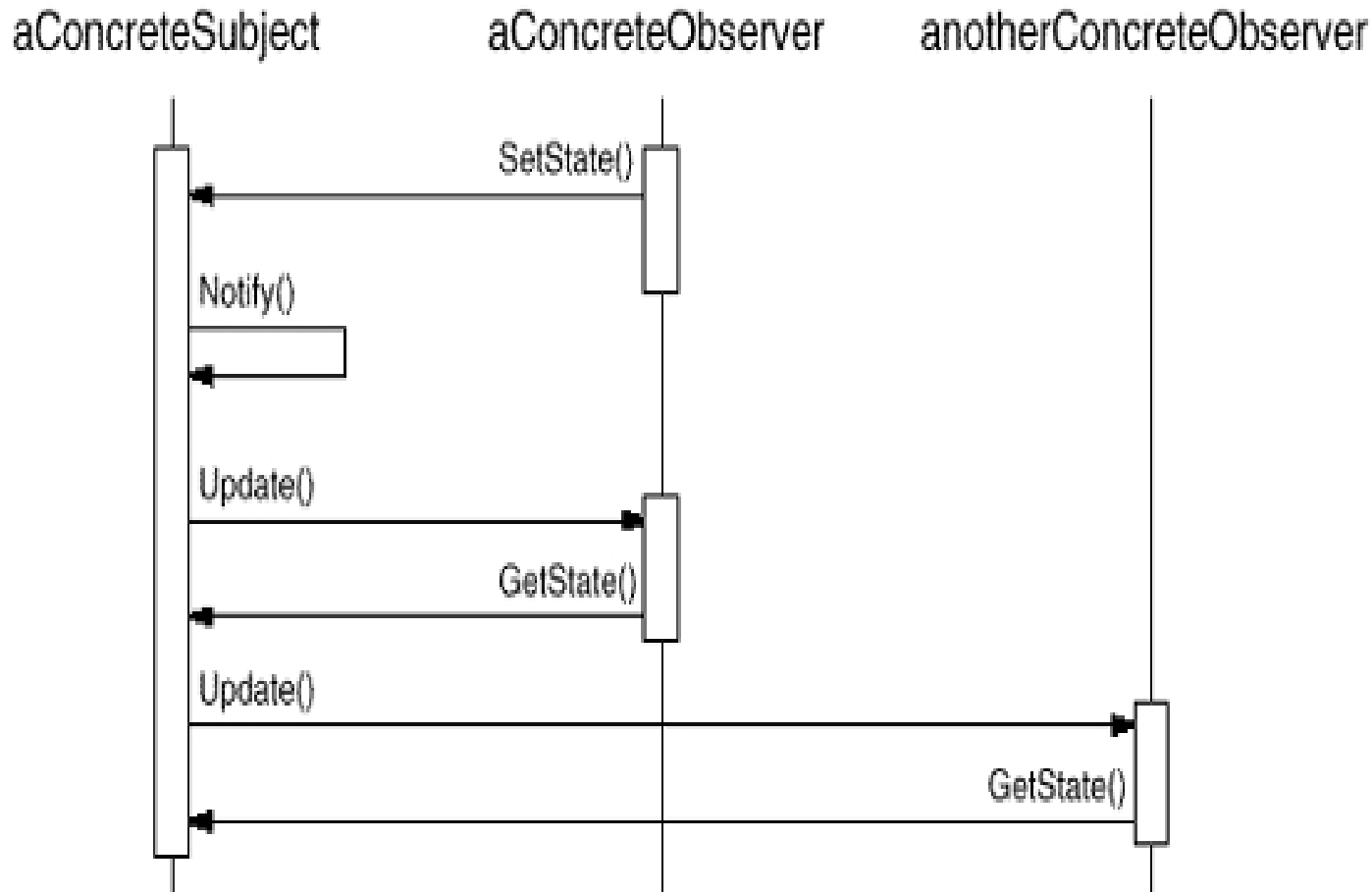
여기서 신문사 + 구독자 = 옵저버 패턴

# Structure Of Observer Pattern

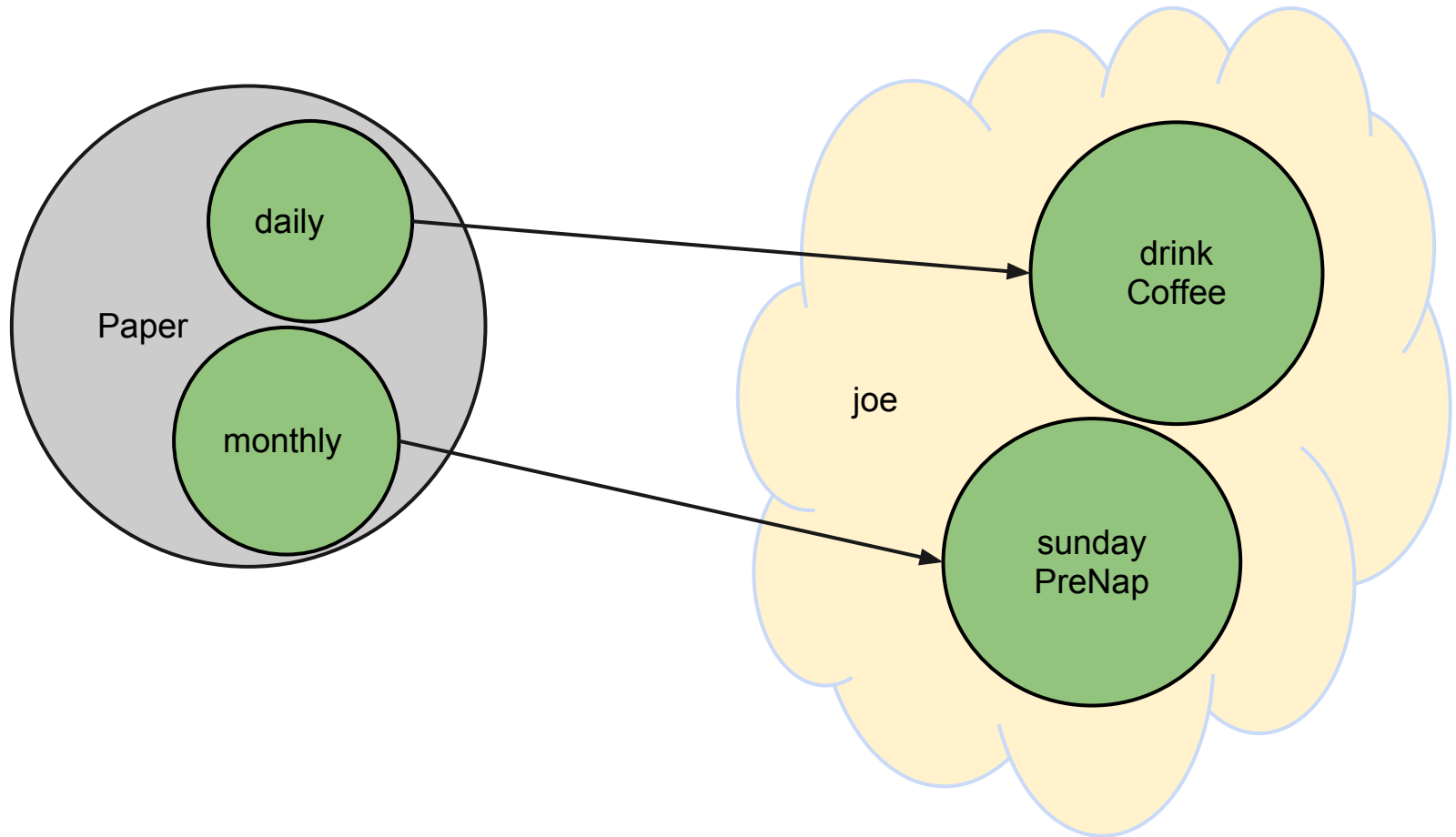


- 핵심은 옵저버 또는 리스너(listener)라 불리는 하나 이상의 객체를 등록하거나 자신을 등록시킨다.  
그리고 관찰되는 객체(또는 주제)에서 발생하는 이벤트를 전달하는 것이다.

# Structure Of Observer Pattern



# Observer Pattern



# Observer Pattern

- js. Fiddle

[Push방식의 observer Pattern](#)



# Observer Pattern

- 장점

- 1.복합 내의 객체는 매우 느슨하게 결합

- 2.코드 재사용을 향상

- 단점

1. 구독을 취소하는 동안 이벤트를 장담받을 수 없다.