

## 2.2. Observer Pattern

### 1. 2.2. Observer Pattern

Subject olarak adlandırılan, durumu (state) olan bir nesne olsun.

Sistem içerisinde bu subject'in durumuna bağlı olan birkaç ya da daha fazla nesne olabilir.

Bu nedenle, bağımlı nesneler, ilgili subject'in gözlemcisi (observer) olarak bu durumdan fayda sağlarlar: Subject'in durumuna bağlı oldukları için subject değişikliklerinin hepsinden haberdar olmaları yerinde bir çözüm olacaktır.

#### 1.1. Önerilen Yapı

1. Gözlemcilerin subject'e kaydet (abone yap).
2. Subject içerisinde bir değişiklik yaşandığında, **tüm aboneleri** bilgilendir.

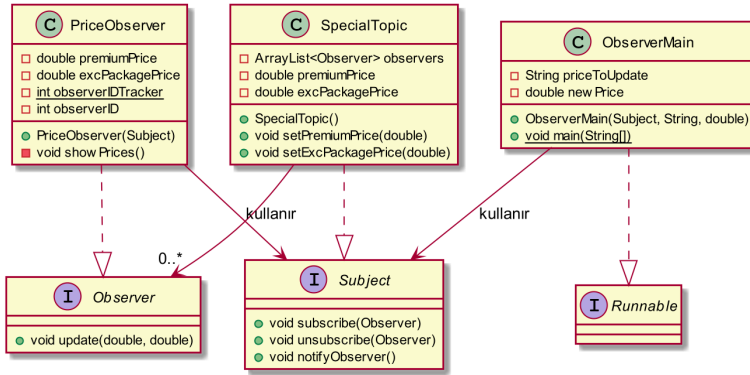


Figure 1. Observer Pattern UML Diyagramı

- **Subject:** Abone **eklemek** ve **çıkarmak** için kullanılan interfacedir.
  - **Subject::notifyObserver()** metodu ile **Observer::update()** metodu çağırılarak tüm abonelere durum değişikliği bildirilir.
- **Observer:** Subject'ten bildirim alan tüm gözlemciler için bir arayüzdür. Buradaki **update** metodu ile fiyatlar güncellenir.
- Fiyat ayarlama işlemi **ConcreateSubject** olan **SpecialTopic sınıfı** içerisinde yapılır.

## **1.2. Avantajı**

- Loose coupling uygulanmış olur.
  - Subject'in, observerların yapısı hakkında herhangi bir bilgiye sahip olması gerekmemekte.