**State pattern**

State is a behavioral design pattern that lets an object alter its behavior when its internal state changes. It appears as if the object changed its class.

classin obyketinn in veziiyeti deyisikide muxtelif reaksiyalar gostermesi ucundur.

Futbol numunesi - defense state, attack state

**Strategy Pattern**

Strategy is a behavioral design pattern that lets you define a family of algorithms, put each of them into a separate class, and make their objects interchangeable.

Navigator - via car, bus example

***State ve strategy patternlerni eynidir, laki state patternde Stateler deyisir, digerinde ise alqoritmler(strategyler) deyisir.***

**Iterator Pattern**

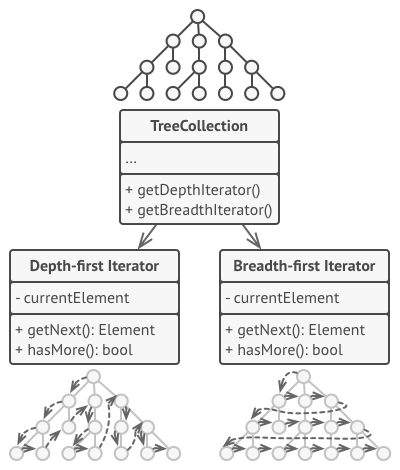
Iterator is a behavioral design pattern that lets you traverse elements of a collection without exposing its underlying representation (list, stack, tree, etc.).

Listlerin,toplularin uzerinde gezmeyin en dogru yolu ucun istifade olunur.

Depth-first search - en dibe gedib geri qayidir, ve yeniden eyni sey

Breadh-first search - en sagdan en sola sonra asagi ve tersi

(nextleri ferqlidir)



**Observer Pattern**

Observer is a behavioral design pattern that lets you define a subscription mechanism to notify multiple objects about any events that happen to the object they’re observing.

publisher ve subscriber, notification sistem

**Template Method Pattern**

Template Method is a behavioral design pattern that defines the skeleton of an algorithm in the superclass but lets subclasses override specific steps of the algorithm without changing its structure.

Bir metod var ve templatedir, bundan ferqli olanlar ondan torenir ve alqoritmi desyisir.

Funksiyalarin ardicillig eynidir lakin daxili deyisir