Codecat15

Observer pattern in swift

What is the observer pattern

Codecat15

The **observer pattern** is a software design pattern in which an object, called the **subject**, maintains a list of its dependents, called **observers**, and notifies them automatically of any state changes, usually by calling one of their methods.

Source: https://en.wikipedia.org/wiki/Observer_pattern

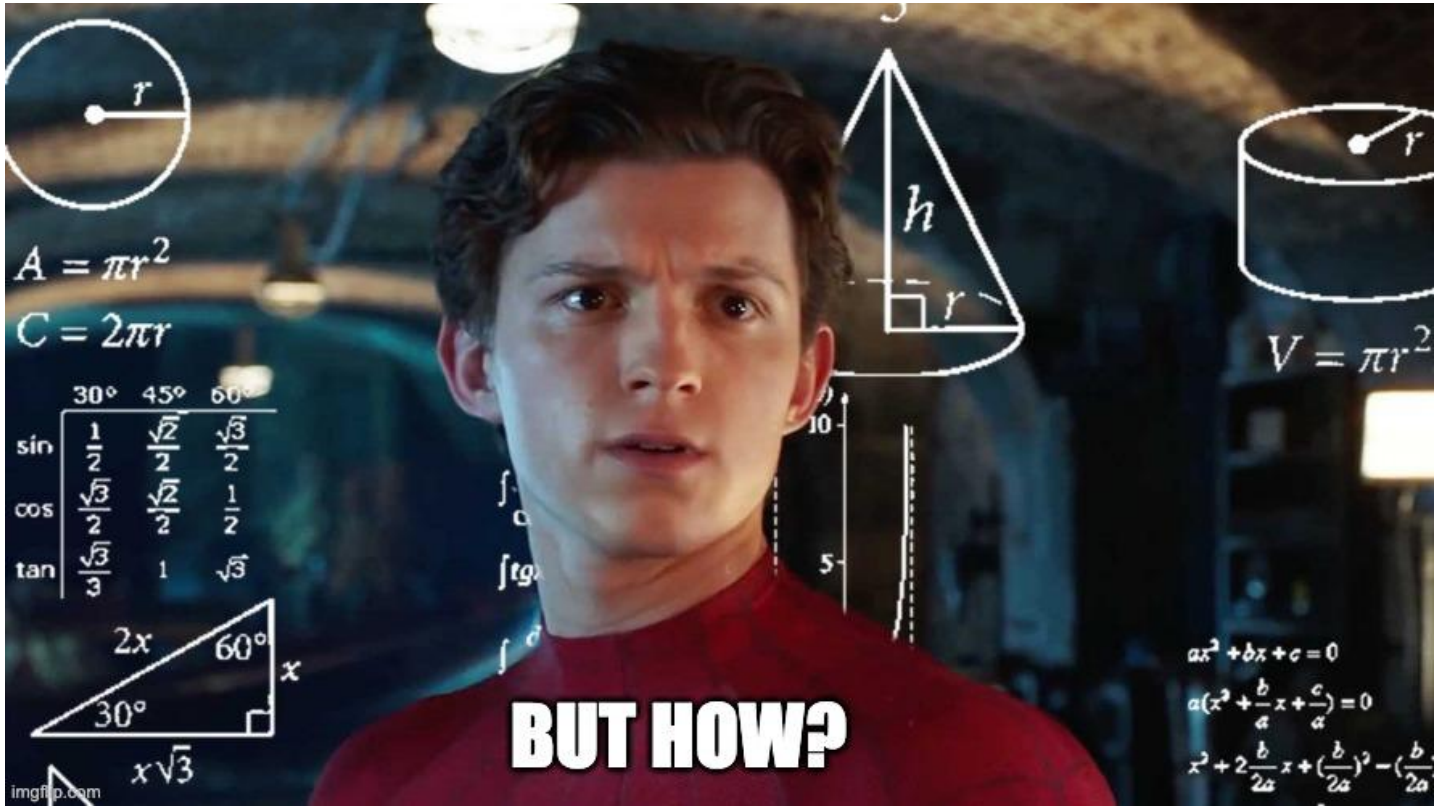
Codecat15

Codecat15 example of observer pattern

Codecat15

If you drive any vehicle then you have implemented the observer pattern :)

Wait how did I implement this by driving a vehicle?



Let’s analyze this

1. While we are in traffic we all **“OBSERVE”** the traffic lights.

2. Once the lights are green, we all start our vehicle and move away.

3. One object was observing the state of other object and acted accordingly **when the state changed.**

Codecat15

Let’s analyze this

1. You have lot of people observing the same light

2. People who cross the road, vendors, traffic police etc

3. For them each color has a meaning.

4. **Example:** When the lights are red, the vendors will try to sell us products.

5. So we can say different objects will react differently based on the color of the traffic light.

Codecat15

As per the definition

1. We are the observers who observe the traffic light.

2. The **subject** here is the traffic light which has a group of people and vehicles who **act as per it's changed state.**

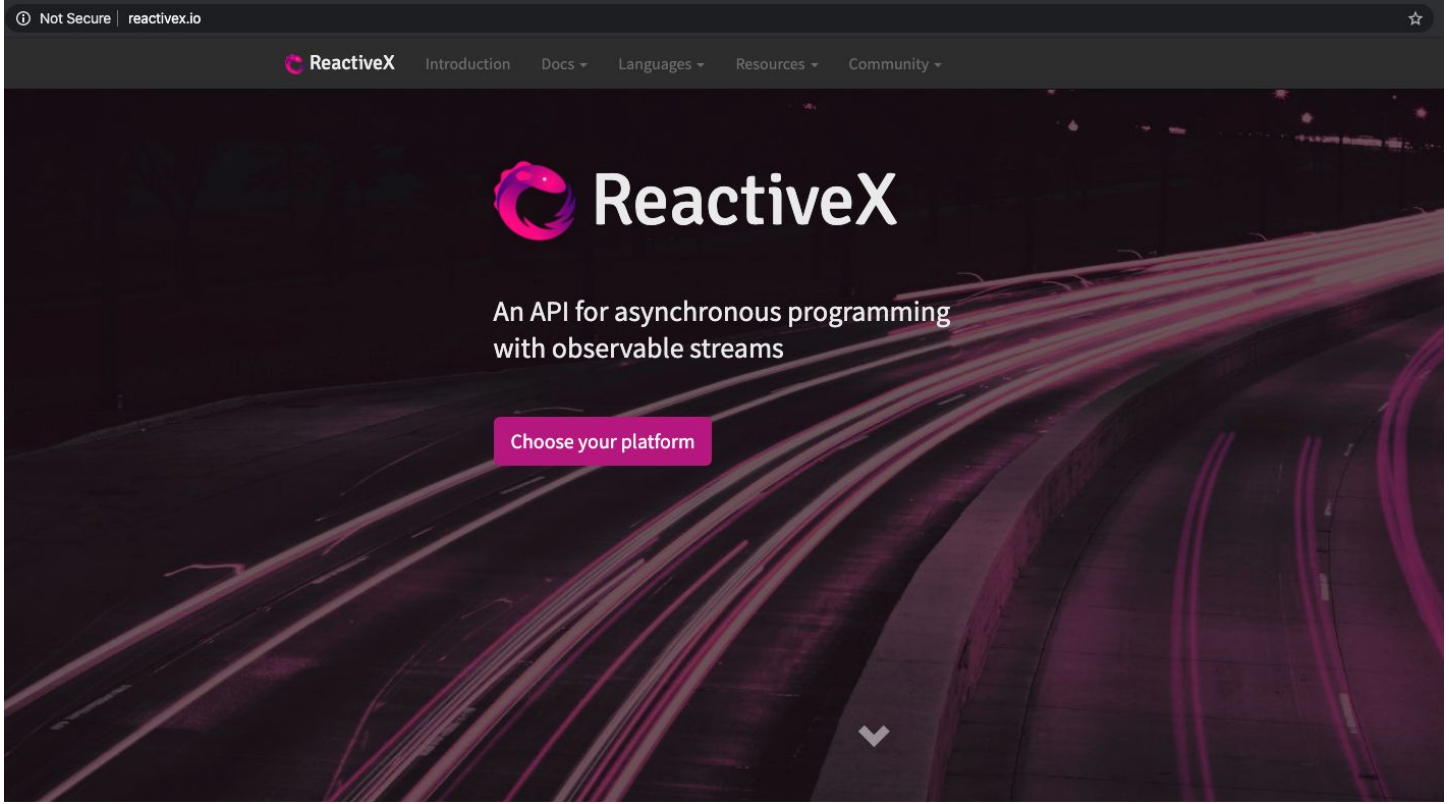
Codecat15

Where is this concept used in iOS programming

1. KVO or Key value observer. [Tutorial on this coming soon]

2. Reactive programming libraries like RxSwift use the observer pattern.

Codecat15



The Observer pattern done right

ReactiveX is a combination of the best ideas from
the Observer pattern, the Iterator pattern, and functional programming

Source: http://reactivex.io/

Codecat15

Let’s code the traffic light example using observer pattern