

# Patrones de Diseño

1

Vamos a sobrevolar este concepto.

De la Real Academia Española.

**Patrón:** Modelo que sirve de muestra para sacar otra cosa igual.

En informática, con el paso del tiempo, se fueron encontrando un set de respuestas, que responden muy bien a cierto set de problemas.

# Patrones de Diseño

2

De esto que encontraron, se generalizaron ciertos **Patrones de Diseño**.

Cuando identificamos un **problema típico**, podemos aplicar con tranquilidad el **Patrón de Diseño** que lo resuelve.

Es un camino largo y tiene un grado de dificultad. Pero podemos conocer algunos patrones que usaremos mucho.

# Reactor Pattern

3

## Patrón Reactor

Muy coloquialmente, imaginemos:

Pido una pizza por teléfono en la oficina, y le aviso a mesa de entradas que por favor me avisen cuando llega el pedido.

Mesa de entrada es quien ve cuando llega el pedido, y es quien reaccionará avisándome.

# Reactor Pattern

4

Un poco menos coloquial:

Supongamos que desde nuestro código necesitamos hacer un **pedido a Internet** y necesitamos **ejecutar algo cuando este pedido sea respondido**.

Este pedido no es instantáneo, demora cierto tiempo.

Entonces estableceremos un manejador (handler) o una promesa (promise), que definimos de antemano, y que se ejecutará (reaccionará) cuando vuelva el pedido.

# Observer Pattern

5

## Patrón Observer

También muy coloquialmente, imaginemos:

1. Tenemos una **Editorial** que emite números de una **revista digital** que deben ser descargados de su web.
2. Tenemos muchos **interesados** en recibirlos.

# Observer Pattern

6

Imagino dos opciones:

- a. que los interesados estén consultándole todo el tiempo a la Editorial preguntándole: “¿Ya hay un nuevo número disponible?”
- b. que la Editorial les avise a los interesados.



# Observer Pattern

7

Este patrón propone **establecer una comunicación** tal que:

- la **Editorial** pueda decirle a la **interesados**:  
“Hay un nuevo número disponible!”
- **cada interesado reaccionará a su manera**:  
Algunos lo bajarán, otros lo leerán online, otros lo imprimirán...

La editorial se conoce como **Observable** y los interesados como **Observers**.

# Patrones de Diseño

8

## Patrón Reactor $\Leftrightarrow$ Patrón Observer

Estos patrones resuelven problemas concretos y son muy usados. Se parecen entre ellos, pero se diferencian por ejemplo en que:

- el **Reactor Pattern** establece una **comunicación/reacción de uno a uno**
- el **Observer Pattern** establece una **relación de muchos a uno**. Muchos Observers conectados a un Observable.



# Patrones de Diseño

9

## “Bibliografía”

Les recomendamos el libro “**Head First Design Patterns - O'Reilly Media**”, muy claro y con ejemplos desarrollados paso a paso...

...y a **Christopher Okhravi** que explica en YouTube todos los patrones de diseño, siguiendo el libro que mencionamos.

<https://www.youtube.com/playlist?list=PLrhzvIcii6GNjpARdnO4ueTUAVR9eMBpc>