**PATRÓN OBSERVADOR (Observer)**

* Lisa decide comprar un vestido nuevo para su fiesta de 18 años. Ella ingresa a su página favorita de compras y le gusta un vestido rojo. El detalle es que ya no hay vestidos rojos disponibles en la tienda.
* Ella busca otro vestido pero no encuentra ninguno que le guste. Así que ingresa de nuevo a el apartado del vestido rojo, aquí está un campo de texto en donde ella ingresa su correo y después presiona el botón “Notificarme cuando esté disponible” para que la página le envié un correo en cuanto haya más vestidos rojos disponibles.
* La razón de toda esta historia es que el botón “Notificare cuando esté disponible” es un ejemplo del patrón Observador (Observer).
* Tan pronto como el vestido esté disponible, lisa será notificada con un aviso a su correo.
* Al igual que Lisa, puede haber otras chicas (usuarios) esperando que el vestido rojo esté disponible.
* Suponiendo que existen otras tres chicas que se suscriben al igual que Lisa, todas ellas serán notificadas en cuanto el vestido rojo esté disponible.
* Suponiendo que existe otra chica que vio el vestido rojo pero no se interesó en él, entonces esta chica no ingresa su correo y tampoco presiona el botón de “Notificarme cuando esté disponible”, así que ningún aviso se le enviará a su correo en cuanto el vestido rojo esté disponible.
* Si alguna chica que estaba interesada en el vestido rojo decide que ya no le gusta y prefiere otro, entonces, la chica cancelará o eliminará la opción de notificar a su correo. Por lo tanto no será informada en cuanto el vestido rojo esté disponible.

**OBSERVADORAS**

* Todas las chicas que se registran para recibir la notificación son observadoras.

**SUJETO OBSERVABLE**

* El sujeto que está siendo observado es el vestido rojo.

**Existe un vestido rojo y varias chicas (observadoras) están interesadas en el vestido**

* El patrón observador (observer) define una relación de uno a muchos entre un sujeto observable y los observadores.
* Cuando el estado del vestido rojo pase de “Vestido No disponible” a “Vestido Disponible” todas las chicas (observadores) serán notificadas.
* Cuando el sujeto observable cambia de estado, todos sus observadores son notificados.
* Los observadores se registran o suscriben a un sujeto observable.

**PASOS**

1ro: Creamos la interface

SujetoObservable.java

2do: Creamos la clase concreta que implementara la interface SujetoObservable

SujetoObservableConcreto.java

3ro: Crear la interface

Observadora.java

4to: Crear la clase concreta que implementara la interface Observadora

ObservadoraConcreta.java

5to: Crear el Main

Main.java

1ro: Creamos la interface

SujetoObservable.java

|  |
| --- |
|  |

|  |  |  |
| --- | --- | --- |
| 2do: Creamos la clase concreta que implementara la interface SujetoObservable  SujetoObservableConcreto.java | |  |
| 3ro: Crear la interface  Observadora.java |  | |

|  |  |
| --- | --- |
| 4to: Crear la clase concreta que implementara la interface Observadora  ObservadoraConcreta.java |  |

|  |  |
| --- | --- |
| 5to: Crear el Main  Main.java |  |