

A principal diferença entre os dois.

No primeiro exemplo usa uma interface e uma classe concreta para implementar o padrão Observer, enquanto o segundo usa a classe EventObject e a interface EventListener.

No primeiro exemplo, a interface Observer define o método update(), que é chamado quando o estado do objeto observável muda. A classe concreta ConcreteObserver implementa o método update() e define o comportamento específico que deve ser executado quando o estado do objeto observável mudar.

No segundo exemplo, a classe StateChangeEvent estende a classe EventObject e adiciona um novo atributo, novoEstado, que representa o novo estado do objeto observável. A interface StateChangeListener define o método stateChanged(), que é chamado quando um evento de mudança de estado é disparado.

A classe concreta StateChangeObserver implementa o método stateChanged() e define o comportamento específico que deve ser executado quando um evento de mudança de estado for disparado.

Em poucas palavras, o primeiro exemplo é mais simples e direto, enquanto o segundo exemplo é mais flexível e pode ser usado em uma variedade de contextos.

Exemplos:

Característica	Exemplo 1	Exemplo 2
Interface	Observer	EventListener
Classe Concreta	ConcreteObserver	StateChangeObserver
Classe de Evento	Não	StateChangeEvent
Atributo de Evento	Não	novoEstado

Considerando os projetos observer criados, discuta a diferença usando ou não a classe event:

A principal diferença entre usar ou não a classe Event é a flexibilidade que ela oferece. A classe Event fornece uma estrutura genérica para representar eventos, o que torna possível usar o padrão Observer em uma variedade de contextos.

No primeiro exemplo, o padrão Observer é implementado de forma muito simples. A interface Observer define apenas um método, update(), que é chamado quando o estado do objeto observável muda. A classe concreta ConcreteObserver implementa o método update() e define o comportamento específico que deve ser executado quando o estado do objeto observável mudar.

No segundo exemplo, a classe StateChangeEvent estende a classe EventObject e adiciona um novo atributo, novoEstado, que representa o novo estado do objeto observável. A interface StateChangeListener define o método stateChanged(), que é chamado quando um evento de mudança de estado é disparado. A classe concreta StateChangeObserver implementa o método stateChanged() e define o comportamento específico que deve ser executado quando um evento de mudança de estado for disparado.

Em poucas palavras, o primeiro exemplo é mais simples e direto, enquanto o segundo exemplo é mais flexível e pode ser usado em uma variedade de contextos.

Exemplos das principais diferenças entre os dois:

Característica	Usar (Event)	Não usar (Event)
Flexibilidade	Alta	Baixa
Complicação	Alta	Baixa
Especificidade	Geral	Específica