



Patrón de diseño Interpreter

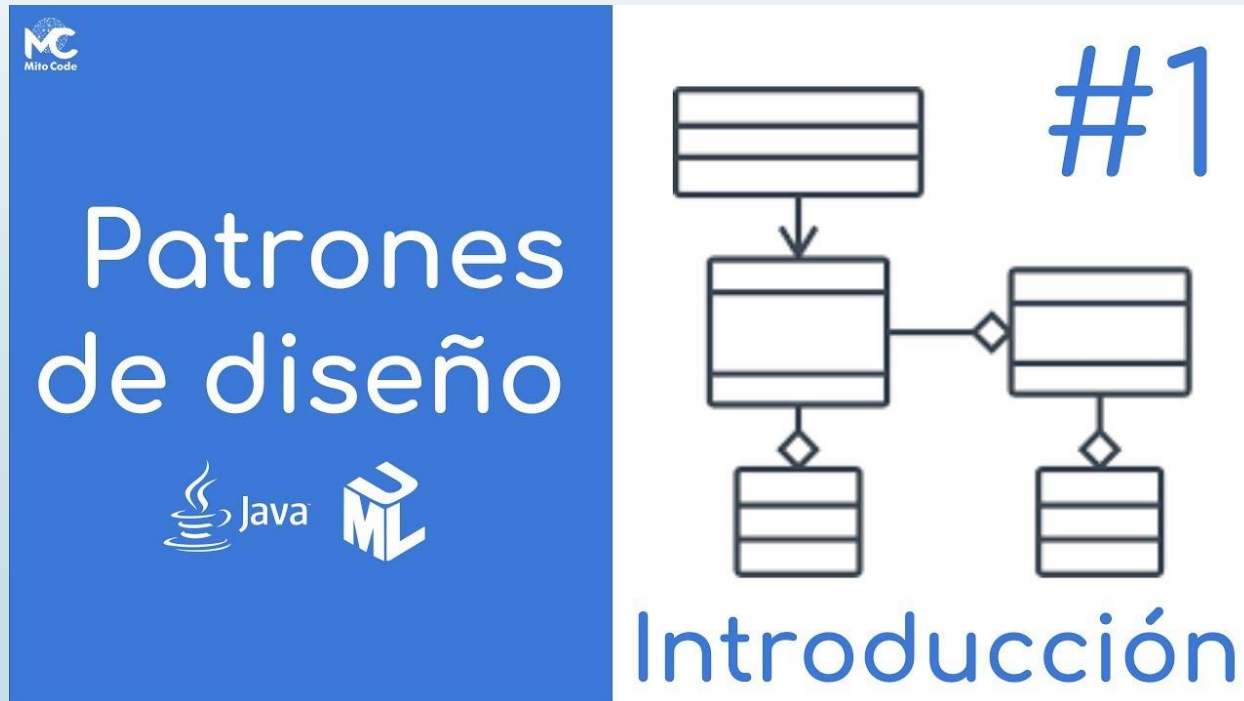
Paolo Induni

Keylor Ortega

Leonardo Mora

¿Qué es un patrón de diseño?

- Técnicas para resolver problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces.





¿Para qué nos sirven?

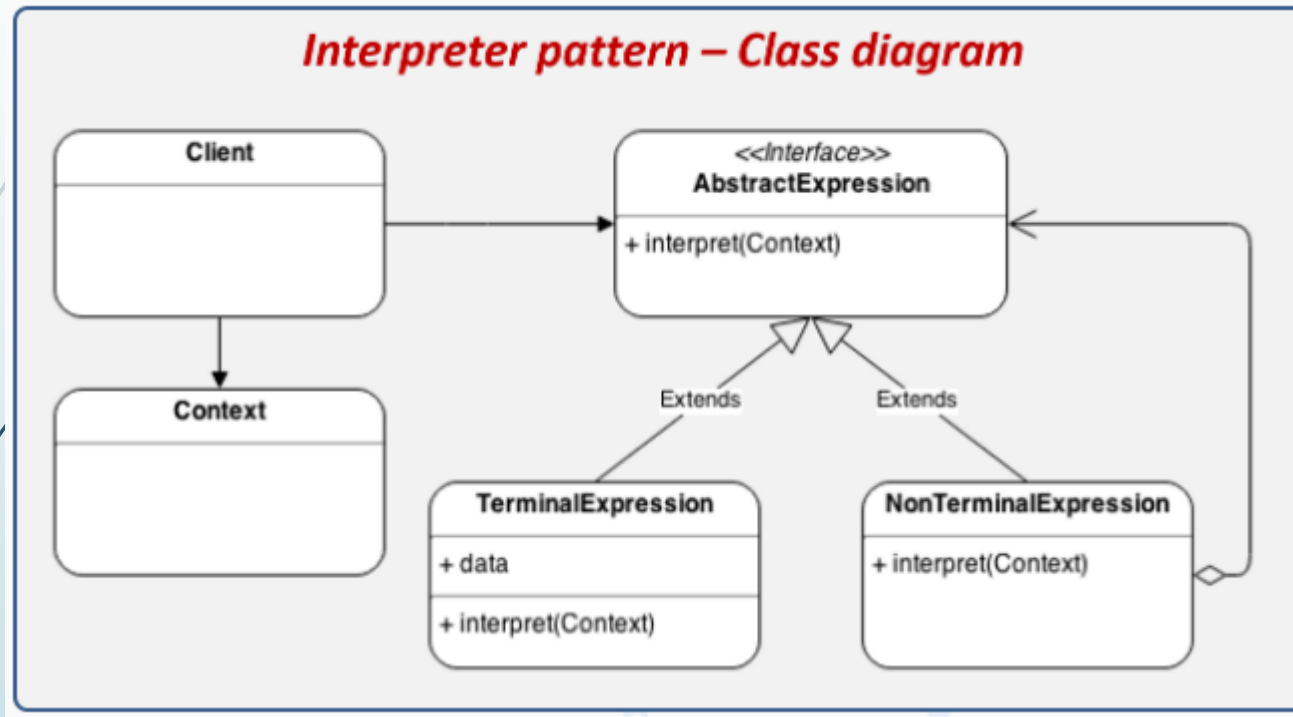
- Aumentan la flexibilidad de un código.
- Proporcionan distintos usos a objetos.
- Alimenta las posibilidades de reutilización de código, permitiendo así tener un proyecto más eficiente y ligero.



Patrón Interpreter

- Es utilizado para evaluar expresiones de un lenguaje, asignarle un dominio y una gramática, con esto se crea un patrón que permite interpretar lenguajes como Java, C#, SQL o alguno inventado mientras tenga un significado
- Esto quiere decir que, mientras el código tenga algún significado o sentido, el patrón nos permitirá adaptarlo.

Diagrama de Clases



Client: Activa ejecución.

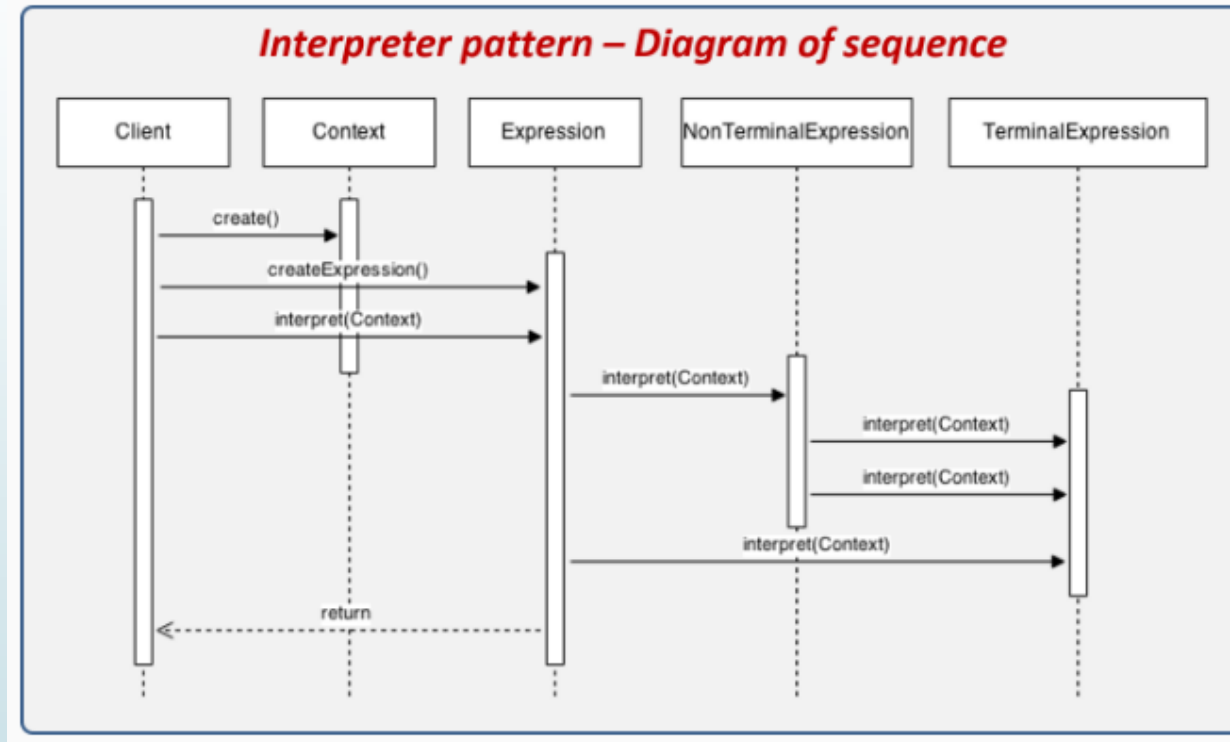
Context: Objeto para leer y almacenar Información global.

AbstractExpression: Estructura mínima de una expresión.

TerminalExpression: Marcan el final de una secuencia.

NonTerminalExpression: Expresiones compuestas, conforman la secuencia.

Diagrama de Secuencia



1. El cliente crea el contexto para la ejecución del interpreter.
2. El cliente obtiene la expresión a evaluar.
3. El cliente solicita la interpretación
4. Expression llama a NonTerminalExpression.
5. NonTerminalExpression a TerminalExpression
6. Se crea la solicitud de la interpretación
7. Se termina de evaluar la expresión y se da como resultado la interpretación

Veamos un ejemplo...

