**C# Proxy**

Шаблон **Proxy** надає***сурогат(surrogate)*** або ***заповнювач(placeholder)*** для іншого об'єкта, щоб контролювати доступ до нього.

До класів та об'єктів, що беруть участь у цьому паттерні, відносяться

* **Proxy (MathProxy)**
* Зберігає посилання, яке дозволяє проксі отримати доступ до реального об'єкта.
* Проксі може посилатися на об'єкт, якщо інтерфейси ***RealSubject і Subject*** однакові.
* Надає інтерфейс, ідентичний інтерфейсу суб'єкта, щоб проксі можна було замінити на реальний суб'єкт.
* Контролює доступ до реального суб'єкта і може відповідати за його створення та видалення.
* Інші обов'язки залежать від типу проксі:
* Віддалені проксі відповідають за кодування запиту та його аргументів, а також за надсилання закодованого запиту реальному суб'єкту в іншому адресному просторі.
* Віртуальні проксі можуть кешувати додаткову інформацію про реальний об'єкт, щоб відкласти доступ до нього. Наприклад, ***ImageProxy*** з мотивації кешує розмір реального зображення.
* Захисні проксі перевіряють, чи має абонент права доступу, необхідні для виконання запиту.
* **Subject (IMath) -** визначає спільний інтерфейс для ***RealSubject*** і проксі, щоб проксі можна було використовувати скрізь, де очікується ***RealSubject.***
* **RealSubject (Math) -** визначає реальний об'єкт, який представляє проксі.

**Structural code in C#**

using System;

namespace Proxy.RealWorld

{

public class Program

{

public static void Main(string[] args)

{

// Create math proxy

MathProxy proxy = new MathProxy();

// Do the math

Console.WriteLine("4 + 2 = " + proxy.Add(4, 2));

Console.WriteLine("4 - 2 = " + proxy.Sub(4, 2));

Console.WriteLine("4 \* 2 = " + proxy.Mul(4, 2));

Console.WriteLine("4 / 2 = " + proxy.Div(4, 2));

// Wait for user

Console.ReadKey();

}

}

public interface IMath

{

double Add(double x, double y);

double Sub(double x, double y);

double Mul(double x, double y);

double Div(double x, double y);

}

public class Math : IMath

{

public double Add(double x, double y) { return x + y; }

public double Sub(double x, double y) { return x - y; }

public double Mul(double x, double y) { return x \* y; }

public double Div(double x, double y) { return x / y; }

}

public class MathProxy : IMath

{

private Math math = new Math();

public double Add(double x, double y)

{

return math.Add(x, y);

}

public double Sub(double x, double y)

{

return math.Sub(x, y);

}

public double Mul(double x, double y)

{

return math.Mul(x, y);

}

public double Div(double x, double y)

{

return math.Div(x, y);

}

}

}

**Output**

4 + 2 = 6

4 - 2 = 2

4 \* 2 = 8

4 / 2 = 2