**C# Adapter**

Паттерн проектування ***Адаптер(Adapter)*** перетворює інтерфейс класу на інший інтерфейс, який очікують клієнти. Цей паттерн дозволяє класам працювати разом, які інакше не могли б працювати через несумісні інтерфейси.

Класи та об'єкти, що беруть участь у цьому паттерні, включають

* **Target (ChemicalCompound)** - визначає специфічний для домену інтерфейс, який використовує клієнт.
* **Adapter (Compound)** - адаптує інтерфейс ***Adaptee*** до інтерфейсу ***Target***.
* **Adaptee (ChemicalDatabank)** - визначає існуючий інтерфейс, який потрібно адаптувати.
* **Client (AdapterApp)** - взаємодіє з об'єктами, що відповідають інтерфейсу ***Target***.

**Structural code in C#**

using System;

namespace Adapter.RealWorld

{

public class Program

{

public static void Main(string[] args)

{

// Non-adapted chemical compound

Compound unknown = new Compound();

unknown.Display();

// Adapted chemical compounds

Compound water = new RichCompound("Water");

water.Display();

Compound benzene = new RichCompound("Benzene");

benzene.Display();

Compound ethanol = new RichCompound("Ethanol");

ethanol.Display();

// Wait for user

Console.ReadKey();

}

}

public class Compound

{

protected float boilingPoint;

protected float meltingPoint;

protected double molecularWeight;

protected string molecularFormula;

public virtual void Display()

{

Console.WriteLine("\nCompound: Unknown ------ ");

}

}

public class RichCompound : Compound

{

private string chemical;

private ChemicalDatabank bank;

// Constructor

public RichCompound(string chemical)

{

this.chemical = chemical;

}

public override void Display()

{

// The Adaptee

bank = new ChemicalDatabank();

boilingPoint = bank.GetCriticalPoint(chemical, "B");

meltingPoint = bank.GetCriticalPoint(chemical, "M");

molecularWeight = bank.GetMolecularWeight(chemical);

molecularFormula = bank.GetMolecularStructure(chemical);

Console.WriteLine("\nCompound: {0} ------ ", chemical);

Console.WriteLine(" Formula: {0}", molecularFormula);

Console.WriteLine(" Weight : {0}", molecularWeight);

Console.WriteLine(" Melting Pt: {0}", meltingPoint);

Console.WriteLine(" Boiling Pt: {0}", boilingPoint);

}

}

public class ChemicalDatabank

{

// The databank 'legacy API'

public float GetCriticalPoint(string compound, string point)

{

// Melting Point

if (point == "M")

{

switch (compound.ToLower())

{

case "water": return 0.0f;

case "benzene": return 5.5f;

case "ethanol": return -114.1f;

default: return 0f;

}

}

// Boiling Point

else

{

switch (compound.ToLower())

{

case "water": return 100.0f;

case "benzene": return 80.1f;

case "ethanol": return 78.3f;

default: return 0f;

}

}

}

public string GetMolecularStructure(string compound)

{

switch (compound.ToLower())

{

case "water": return "H20";

case "benzene": return "C6H6";

case "ethanol": return "C2H5OH";

default: return "";

}

}

public double GetMolecularWeight(string compound)

{

switch (compound.ToLower())

{

case "water": return 18.015;

case "benzene": return 78.1134;

case "ethanol": return 46.0688;

default: return 0d;

}

}

}

}

**Output**

Compound: Unknown ------

Compound: Water ------

Formula: H20

Weight : 18.015

Melting Pt: 0

Boiling Pt: 100

Compound: Benzene ------

Formula: C6H6

Weight : 78.1134

Melting Pt: 5.5

Boiling Pt: 80.1

Compound: Alcohol ------

Formula: C2H6O2

Weight : 46.0688

Melting Pt: -114.1

Boiling Pt: 78.3