Délégués :

Les délégués, déclarés avec delegate, permettent la mise en place du principe d’ouvert/fermé, autrement dit, ils permettent de passer des méthodes comme argument lors des appels. Ils sont en quelque sorte un moyen de modifier le comportement d’une application sans modifier son code source.

Leur effet est lié à celui de la catégorie de classes abstract

Les délégués peuvent être déclarés dans la partie namespace ou alors au sein même d’une classe.

EXEMPLE PRATIQUE :

namespace UsingDelegate

{

delegate string ActionOnString(string text);

class Program

{

//Regular method that matches signature

static string normalPrint(string text)

{

return text;

}

static void Main(string[] args)

{

//Instantiate delegate with named method

ActionOnString test1 = normalPrint;

//Invoke delegate test1:

//Normal print

string result1 = test1("Première phrase de test.");

Console.WriteLine(result1 + "\n");

//Instantiate delegate with anonymous method

//All in uppercase

ActionOnString test2 = delegate (string text)

{

string tmp = text.ToUpper();

return tmp;

};

string result2 = test2("Deuxième phrase de test.");

Console.WriteLine(result2 + "\n");

//Instantiate delegate with lambda expression

//concatenate 2 times the same string

ActionOnString test3 = s => s + s;

string result3 = test3("Troisième phrase de test.");

Console.WriteLine(result3 + "\n");

// Keep the console window open in debug mode.

Console.WriteLine("Press any key to exit.");

Console.ReadKey();

}

}