

TP No5

Exercice 1:

Partie1 :

Créez une solution 'MaSolution5', contenant une projet WCF de type 'Application de service WCF' nommée 'WcfService1'

Paramétrez votre projet 'WcfService1' comme suit :

Authentification anonyme	Activé
Authentification Windows	Désactivé
Mode pipeline géré	Intégré
SSL activé	False
Toujours commencer lors du débogage	True

Ce paramétrage sera à appliquer à tous les services WCF créés dans ce TP.

!!! N'oubliez pas de passer le fichier 'Service1.svc' comme page de démarrage

Puis démarrez votre projet (F5)

Obtenez l'URL de votre service.

Partie 2 :

Cette partie permet de tester que la partie 1 de l'exercice 1 est fonctionnelle.

Ajoutez une application console nommée 'WcfService1ClientConsole', ajoutez une référence de service nommée 'Service1' pointant sur le service 'WcfService1', puis appelez le service dans le main de l'application comme suit :

```
static void Main(string[] args)
{
    Service1.Service1Client service1Client = new Service1.Service1Client();
    String result = service1Client.GetData(10);
    Console.WriteLine("Résultat de GetData du service 'Service1': {0}", result);
    Console.Read();
}
```

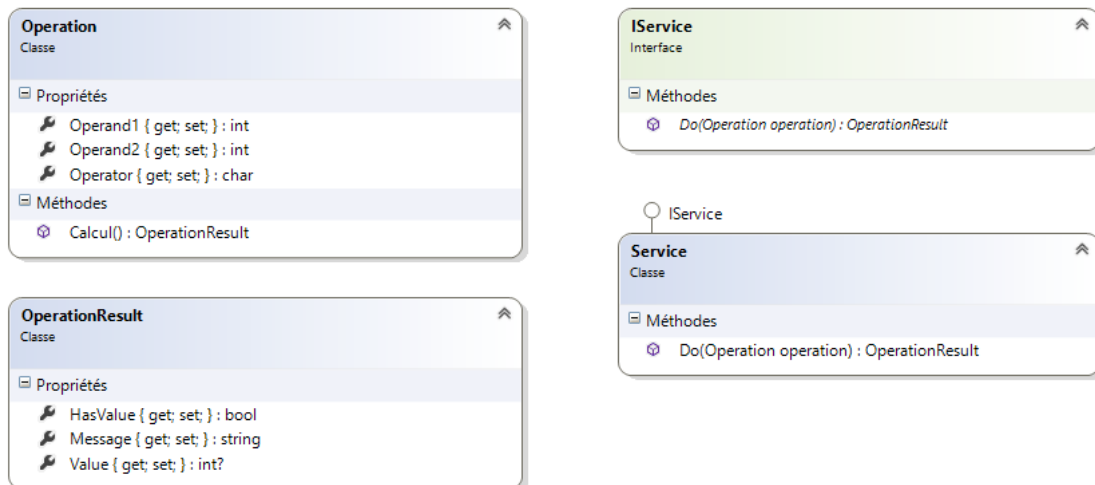
Exercice 2 :

Partie1 :

Ajoutez un projet 'MonService' de type 'WCF/Application de service WCF', votre point de terminaison sera nommé 'Service.svc'

Le but de cet exercice est de créer un service qui permet à des clients de réaliser une opération simple (+, -, *, /) entre 2 opérandes.

Pour cela, implémentez les classes et interfaces suivantes :



La classe 'Operation' contient :

2 opérandes et un opérateur, ainsi que méthode Calcul permettant de réaliser le calcul entre les 2 opérandes et de retourner le resultat sous forme d'un objet OperationResult

Ex : $10 + 2 = 12$

La classe 'OperationResult' contient :

HasValue : permet de savoir si le resultat contient une valeur

Message : Indique le message de retour en cas de problème

Value : renseigne la valeur de retour, attention notez qu'il s'agit d'un type `Nullable<Int32>` d'où le '?' sur le diagramme ci-dessous.

Le service est exposé par la classe Service implémentant l'interface IService contenant :

Do : Une méthode prenant en paramètre une instance de 'Operation' et retournant le result sous forme de 'OperationResult'

!!! N'oubliez pas de passer le fichier 'Service.svc' comme page de démarrage

Puis démarrez votre projet (F5)

Obtenez l'url de votre service.

Partie 2 :

Cette partie permet de tester que la partie 1 de l'exercice 2 est fonctionnelle.

Ajoutez une application console nommée 'MonServiceClientConsole', ajoutez un référence de service nommée 'MonService pointant sur le service 'Service', puis appelez le service dans le main de l'application comme suit :

```

static void Main(string[] args)
{
    MonService.ServiceClient monService = new MonService.ServiceClient();
    MonService.Operation operation = new MonService.Operation { Operand1 = 10, Operand2 = 2, Operator = '+' };

    MonService.OperationResult operationResult = monService.Do(operation);
    if (operationResult.HasValue)
    {
        Console.WriteLine("Resultat de l'opération {0} {1} {2} = {3}", operation.Operand1, operation.Operator,
operation.Operand2, operationResult.Value);
    }
    else
    {
        Console.WriteLine("L'opération n'a pas abouti pour la raison suivante: {0}", operationResult.Message);
    }
    Console.ReadLine();
}

```

Partie 3 :

Ajoutez dans votre projet 'MonService', une classe que l'on transformera en énumération (enum) et qui s'appellera 'OperatorType'

```

public enum OperatorType
{
}

```

Enumérez les opérateurs dans l'enum créée en les associants à un entier unique.

Utilisez cette enum dans la classe 'Operation' pour l'attribut 'Operator'.

Mettez à jour le code de votre service de sorte à ce que le service effectue le travail voulu.

Exercice 3 :

Partie 1:

Au préalable, réaliser une copie (.zip ?) de votre solution 4

Dans la solution du précédent TP (TP4), ajouter un projet de type 'WCF/Application de service WCF', votre point de terminaison sera nommé 'PersonneService.svc'

Créez un service permettant de lister, d'ajouter, de mettre à jour et de supprimer une personne.

!!! N'oubliez pas de passer le fichier 'PersonneService.svc' comme page de démarrage

Puis démarrez votre projet (F5)

Obtenez l'url de votre service.

Partie 2 :

Adaptez votre projet form pour utiliser le service 'PersonneService'.

Exercice 4 : (Bonus)

Ajouter à la partie 1 de l'exercice précédent la gestion des adresses de personnes.