

Le langage C# (.NET)

Code : DC03

Durée : 5 jours

Classe : Présentiel / à distance

Public

- Ce stage s'adresse aux développeurs débutants. Développeurs maîtrisant la programmation mais ayant acquis leurs connaissances à l'université en utilisant un langage non graphique. Développeurs d'applications d'entreprise.

Prérequis

Pour suivre ce cours c#, les participants doivent disposer des connaissances suivantes :

- Être familiarisés avec les fonctions de base de système d'exploitation et les maîtriser, notamment la manipulation de fichiers.
- Maîtriser les notions de base de programmation structurée, notamment des concepts (contrôle de flux, variables et paramètres et appels de fonction)
- Un minimum d'expérience en concepts de programmation orientée objet est préférable. La connaissance d'un langage à la syntaxe proche du C (C, C++, java) est indispensable pour être à l'aise dans les exercices. Sans elle, vous pourrez comprendre le stage mais serez grandement gênés dans la réalisation des travaux pratiques.

Objectifs

Plus concrètement à l'issue de la formation vous serez capable de :

- Comprendre les principes du développement objet (classe, méthode, héritage, etc.)
- Comprendre le rôle du framework .NET et savoir utiliser ses classes
- Configurer et utiliser Visual Studio
- Maîtriser la syntaxe et les concepts du langage C#.
- Utiliser ADO.NET et LINQ pour accéder aux données d'une base quelque soit leur provenance et les manipuler
- Savoir appliquer les grands principes de C# à la réalisation d'applications riches (Windows Forms ou WPF)
- Utiliser des services Web XML dans une application C#.
- Tester des fonctionnalités avancées de C#.
- Créer une première application Web en ASP.NET

Programme détaillé

1-Introduction à .Net

- Qu'entend-on par plateforme (framework) .net ?
- Pourquoi parle-t-on parfois indifféremment de C# ou VB .net ?
- La place de C# dans l'architecture globale de .NET
- Les nouveautés du Framework.Net 4.5, du langage C# 6, et de Visual Studio 2015
- Comparaisons de C# avec d'autres langages (Java, C++, etc.)

2-La conception Objet

- Qu'est-ce qui caractérise un programme « objet » ?
- Vocabulaire et concepts de base (classes, instances)
- Pourquoi l'encapsulation est-elle fondamentale ?
- Le cycle de vie d'un objet (constructeur, destructeur)
- Savoir réutiliser les traitements et les données (héritage)
- Donner le même nom à des traitements différents (polymorphisme)
- Exemples de mauvaises conceptions



Le langage C# (.NET)

Code : DC03

Durée : 5 jours

Classe : Présentiel / à distance

3-Les bases de C#

- Espace de nom, types énumérés.
- Gestion des références, typage.
- Différents modes de passages de paramètres
- Méthodes à nombre d'arguments variable.
- Surcharge d'opérateurs.
- Notion d'attribut, et importance des méta-données.

4-La programmation en C#

- Le modèle Objet avec le langage C#.
- Classes et objets : notion de méthode.
- Code géré et gestion de la mémoire.
- Héritage, surcharge et polymorphisme.
- Propriétés, indexeurs et énumérateurs.
- Les nouveautés de C# .NET : expression d'initialisation, types anonymes, propriétés automatiques, méthodes d'extension, méthodes partielles et lambda expressions
- Utilisation des "Generics"

5-La programmation avancée en C#

- Exceptions : gestion centralisée des erreurs, classes d'exceptions personnalisées, exceptions non- .Net.
- Délégués : délégués et événements, les délégués comme alternative au polymorphisme.
- Structure de l'architecture .Net : assembly, processus et domaine, support multi-langage en .Net, sécurité.

6-Les classes incontournables du framework

- Classe mathématiques, structures usuelles, gestion des dates.
- Gestion du formatage des chaînes.
- Filtrer l'information grâce aux expressions régulières.
- Conteneurs classiques : listes, listes triées, piles et tables de hashage.

7-Introduction à l'accès aux données en .NET

- Les principes d'ADO .Net
- Notion de classe client / classe fournisseur en C#
- Choisir d'accéder aux données avec DataReader ou DataSet ?
- La nouvelle philosophie de LINQ (Entity Framework, LINQ to SQL, Linq To Objects, Linq to XML, ...).

8-Introduction à WPF et aux Windows Forms

- Le modèle objet des Windows Forms : la simplicité de mise en œuvre
- Les possibilités de WPF (Windows Presentation Foundation) : la souplesse des ressources, (style, trigger, ...) et du data binding
- Comparaison entre les frameworks concurrents Windows Forms / WPF

9-ASP.Net/ C# et Webform

- L'architecture des Web Forms
- Séparation Métier / Code et Présentation
- Webservice (webmethode) en C#

