

# C#, développer en .NET avec Visual Studio

Cours Pratique de 5 jours - 35h

Réf : PSH - Prix 2024 : 2 970CHF HT

Cette formation présente le framework .NET et .Net Core. Des travaux pratiques indiquent comment programmer en mode orienté objet, avec le Visual Studio, en langage C#, pour exploiter les classes proposées par Microsoft. Sont aussi abordés les accès aux données avec ADO.NET et LINQ ainsi que les services Web ASP.NET.

## OBJECTIFS PÉDAGOGIQUES

À l'issue de la formation l'apprenant sera en mesure de :

Découvrir les principales technologies du framework .NET

Maîtriser la syntaxe du langage C#

Mettre en œuvre la Programmation Orientée Objet avec C#

Utiliser l'environnement de développement intégré Visual Studio

## TRAVAUX PRATIQUES

Au tout début du stage, les aspects syntaxiques du langage C# et les classes de base sont illustrés avec des exemples simples.

## LE PROGRAMME

dernière mise à jour : 06/2021

### 1) La plateforme .NET

- Principe et architecture de la plateforme .NET.
- Architecture du framework .NET : CLR, BCL, et CLS (support multilingage).
- Aperçu des différents types d'applications dans un environnement multible.
- Structure d'une application .NET : notion d'espace de noms.
- Outils et environnement de développement.
- Langage de CIL : principe du langage intermédiaire, principe de la compilation JIT (Just In Time Compiler).
- Notion d'assembly, de métadonnées et de déploiement.
- Assembly privé et assembly partagé : signature, rôle du GAC (Global Assembly Cache), mise en place dans le GAC.
- Évolution .NET Core, l'Open Source et le multiplateforme.

*Travaux pratiques : Exemple de programme C# minimum. Exécution en mode géré.*

*Utilisation de l'environnement de développement Visual Studio.NET pour l'écriture du premier programme.*

### 2) Syntaxe de base : données, expressions et instructions

- Variables et expressions : déclaration, constantes, opérateurs, types anonymes et dynamiques.
- Types de données : Common Type System, rôle de la classe de base System.Object et transtypage.
- Quelles différences entre les types valeur et types référence ?
- Syntaxe de base : manipulation des tableaux, instructions de contrôle de flux...
- Nouveautés de C# 6 (nul-conditionnel, amélioration des propriétés automatiques, fonctions "Expression Bodied"...).
- Nouveautés de C# 7 (lisibilité des constantes, variables "out", tuples, pattern matching, retour de référence...).

*Travaux pratiques : Ecriture de programmes en C# mettant en oeuvre des algorithmes classiques.*

## PARTICIPANTS

Développeurs désirant acquérir les techniques pour développer tous types d'applications .NET avec Visual Studio.

## PRÉREQUIS

Bonnes connaissances en programmation. Connaissances de base des concepts Objet. Expérience requise en développement logiciel avec un langage de type C/C++ ou Java.

## COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

## MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques... Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

## MOYENS PÉDAGOGIQUES ET TECHNIQUES

- Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
- À l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
- Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le stagiaire a bien assisté à la totalité de la session.

## MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

## ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Vous avez un besoin spécifique d'accessibilité ? Contactez Mme FOSSE, référente handicap, à l'adresse suivante psh-accueil@orsys.fr pour étudier au mieux votre demande et sa faisabilité.

### 3) Gestion des exceptions

- Comment la philosophie des exceptions doit changer votre façon de programmer.
- Déclenchement d'erreurs personnalisées avec "throw".
- Utilisation des exceptions pour le traitement centralisé des erreurs.

*Travaux pratiques : Gestion des erreurs de saisie en utilisant les exceptions.*

### 4) Programmation Orientée Objet

- Classes et objets.
- Modélisation du monde réel par objet.
- Notions de champ, méthode et propriété.
- Héritage.
- Polymorphisme.
- Interfaces pour l'héritage multiple.

### 5) Classes et objets en C#

- Définition des classes. Définition des objets.
- Définition du contenu de la classe : méthodes et propriétés.
- Visibilité des membres d'une classe : propriétés et méthodes. Utilisation des espaces de noms.
- Cycle de vie des objets : constructeur, destructeur. Gestion de la mémoire avec le Garbage Collector.
- Le mécanisme de surcharge (constructeurs, méthodes et opérateurs).
- Dérivation et héritage des classes : principe de dérivation, contrôle d'accès lors de la dérivation.
- Principe du polymorphisme.
- Principe des interfaces (définition et implémentation, utilisation pour l'héritage multiple).
- Manipulation des attributs : principe des méta-données. Attributs de classe, de méthode, de champ.
- Régions de code et classes partielles. Classes génériques.
- Délégués, covariance, contravariance et événements. Les variances appliquées aux interfaces.
- Classes métiers semblables à celles du framework .NET : réalisation de propriétés, d'indexeurs et d'énumérateurs.
- Génération de documentation.

*Travaux pratiques : Ecriture de classes de base. Manipulation de méthodes et propriétés dans les classes. Dérivation de classe. Exemple d'implémentation d'une interface et de mise en œuvre du polymorphisme par héritage ou avec une interface.*

### 6) Objet et classes de base du framework .NET

- Qu'est-ce qu'un framework ? Principe. Hiérarchie des classes.
- Traitement des dates et des durées. Traitement des chaînes avec StringBuilder et les expressions régulières.
- Classes incontournables : manipulation du système de fichier, Math, Random, etc.
- Les différents types de collections, de dictionnaires et de tables de hachage.
- Les collections génériques et les bases de Linq avec LINQ To Object sur les collections standard.

*Travaux pratiques : Utilisation des expressions régulières et d'une table de hachage. Rendre une collection réalisée précédemment compatible avec LINQ et utiliser LINQ To Objects pour l'interroger.*

### 7) Les différents types d'applications développées en .NET

- Principe et conception d'une bibliothèque de classes réutilisables.
- Principe et exemple des applications WPF.
- Principe et exemple des applications Web Asp.Net MVC.
- Principe et exemple des services Web ASP.NET.

- Introduction aux accès aux données avec ADO.NET ou Entity Framework et LINQ.

*Travaux pratiques* : Réalisation d'un formulaire Windows interrogeant une base de données.

Démonstration de Asp.Net MVC avec un contrôleur et une page Razor, avec affichage de données.

Démonstration d'un service Web simple.

## LES DATES

---

CLASSE À DISTANCE

2024 : 03 juin, 16 sept., 16 déc.