Formation C#, développer en .NET avec Visual Studio 2015/2013

N° ACTIVITÉ : 11 92 18558 92 | TÉLÉPHONE : 01 85 77 07 07 | E-MAIL : inscription@hubformation.com

Cette formation vous apprendra à connaître l'architecture des frameworks 2.0 à 4.7 et .Net Core, puis à exploiter les classes basiques communes aux différents types d'applications .NET (Windows, Web, DLL, etc.) avec le langage C#. Vous apprendrez à créer des classes métier et à tirer parti des mécanismes (encapsulation, héritage et polymorphisme) inhérents à la Programmation Orientée Objet (POO) en .NET. Vous verrez également le principe des accès aux données avec ADO.NET et les différentes mises en oeuvre de LINQ ainsi que des applications et services Web ASP.NET.

Référence	PSH
Durée	5 jours (35h)
Tarif	3 030 €HT
Repas	repas inclus

Objectifs

| Découvrir les principales technologies du framework .NET
| Maîtriser la syntaxe du langage C#
| Mettre en oeuvre la Programmation Orientée Objet avec C#
| Utiliser l'environnement de développement intégré Visual Studio

Public

| 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.

Programme de la formation

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

La plateforme .NET
| Principe et architecture de la plateforme .NET.
| Architecture du framework .NET : CLR, BCL, et CLS (support multilangage).
| Aperçu des différents types d'applications dans un environnement multicible.
| 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.

SESSIONS PROGRAMMÉES

A DISTANCE (FRA)
- du 4 au 8 août 2025
- du 1er au 5 septembre 2025

PARIS
- du 21 au 25 juillet 2025
- du 25 au 29 août 2025

AIX-EN-PROVENCE
- du 22 au 26 septembre 2025
- du 29 déc. au 2 janvier 2027

BORDEAUX
- du 22 au 26 septembre 2025
- du 29 déc. au 2 janvier 2027

GRENOBLE
- du 4 au 8 août 2025
- du 17 au 21 novembre 2025

LILLE
- du 22 au 26 septembre 2025
- du 29 déc. au 2 janvier 2027

LYON
- du 4 au 8 août 2025
- du 6 au 10 octobre 2025

NANTES
- du 22 au 26 septembre 2025
- du 29 déc. au 2 janvier 2027

VOIR TOUTES LES DATES

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.

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.

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.

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 oeuvre du polymorphisme par héritage ou avec une interface.

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.

Les différents types d'applications développées en .NET
| Principe et conception d'une bibliothèque de classes réutilisables.
| Principe et exemple des applications Web ASP.NET.
| 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 d'une page ASPX simple, avec affichage de données. Démonstration d'un service Web simple.

Méthode pédagogique

Chaque participant travaille sur un poste informatique qui lui est dédié. Un support de cours lui est remis soit en début soit en fin de cours. La théorie est complétée par des cas pratiques ou exercices corrigés et discutés avec le formateur. Le formateur projette une présentation pour animer la formation et reste disponible pour répondre à toutes les questions.

Méthode d'évaluation

Tout au long de la formation, les exercices et mises en situation permettent de valider et contrôler les acquis du stagiaire. En fin de formation, le stagiaire complète un QCM d'auto-évaluation.

Suivre cette formation à distance

Voici les prérequis techniques pour pouvoir suivre le cours à distance :

| Un ordinateur avec webcam, micro, haut-parleur et un navigateur (de préférence Chrome ou Firefox). Un casque n'est pas nécessaire suivant l'environnement.
| Une connexion Internet de type ADSL ou supérieure. Attention, une connexion Internet ne permettant pas, par exemple, de recevoir la télévision par Internet, ne sera pas suffisante, cela engendrera des déconnexions intempestives du stagiaire et dérangera toute la classe.
| Privilégier une connexion filaire plutôt que le Wifi.
| Avoir accès au poste depuis lequel vous suivrez le cours à distance au moins 2 jours avant la formation pour effectuer les tests de connexion préalables.
| Votre numéro de téléphone portable (pour l'envoi du mot de passe d'accès aux supports de cours et pour une messagerie instantanée autre que celle intégrée à la classe virtuelle).
| Selon la formation, une configuration spécifique de votre machine peut être attendue, merci de nous contacter.
| Pour les formations incluant le passage d'une certification la dernière journée, un voucher vous est fourni pour passer l'examen en ligne.
| Pour les formations logiciel (Adobe, Microsoft Office...), il est nécessaire d'avoir le logiciel installé sur votre machine, nous ne fournissons pas de licence ou de version test.
| Horaires identiques au présentiel.

Accessibilité

Les sessions de formation se déroulent sur des sites différents selon les villes ou les dates, merci de nous contacter pour vérifier l'accessibilité aux personnes à mobilité réduite.
Pour tout besoin spécifique (vue, audition...), veuillez nous contacter au 01 85 77 07 07.