

Objectifs

Connaître l'architecture des Frameworks 2.0, 3.0, 3.5 et 4.0. Exploiter les classes basiques communes aux différents types d'applications .NET (Windows, Web, DLL, etc.) avec le langage C#. Créer des classes métier et tirer parti des mécanismes (encapsulation, héritage et polymorphisme) inhérents à la programmation orientée objet (POO) en .NET. Aborder le principe des accès aux données avec ADO.NET et les différentes mises en œuvre de LINQ ainsi que des applications et services Web : ASP.NET.

Prérequis

Pratique de la programmation structurée.

Pédagogie

La pédagogie est basée sur le principe de la dynamique de groupe avec alternance d'apports théoriques, de phases de réflexion collectives et individuelles, d'exercices, d'études de cas et de mises en situations observées. Formation / Action participative et interactive : les participants sont acteurs de leur formation notamment lors des mises en situation car ils s'appuient sur leurs connaissances, les expériences et mettront en œuvre les nouveaux outils présentés au cours de la session.

Profil de l'intervenant

Consultant-formateur expert. Suivi des compétences techniques et pédagogiques assuré par nos services.

Moyens techniques

Encadrement complet des stagiaires durant la formation. Espace d'accueil, configuration technique des salles et matériel pédagogique dédié pour les formations en centre. Remise d'une documentation pédagogique papier ou numérique à échéance de la formation.

Méthodes d'évaluation des acquis

Exercices individuels et collectifs durant la formation. La feuille d'émargement signée par demi-journée ainsi que l'évaluation des acquis de fin de stage sont adressées avec la facture.

Programme

La plateforme .NET

Principe et architecture de la plateforme .NET.

Architecture du Framework .NET : CLR, BCL, et CLS (support multi langage).

Aperçu des différents types d'applications dans un environnement multi cible.

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. Les nouveautés du GAC 4.0.

Syntaxe de base : données, expressions et instructions

Variables et expressions : déclaration, constantes, opérateurs, types anonymes (C# 3.0) et dynamiques (C# 4.0).

Types de données : Common Type System, rôle de la classe de base System.Object et transtypage.

Types valeur et types référence.

Instructions de contrôle de flux : boucles et tests.

Syntaxe de base : manipulation des tableaux, instructions de contrôle de flux, ...

Gestion des exceptions

Principe de la gestion des exceptions.

Déclenchement d'erreurs personnalisées avec throw.

Utilisation des exceptions pour le traitement centralisé des erreurs.

Programmation Orientée Objet

Classes et objets.

Modélisation du monde réel par objet.

Notions de champ, méthode et propriété.

Héritage.

Polymorphisme.

Interfaces.

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).

Manipulation des attributs : principe des métadonnées. 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 (C# 4.0).

Visual Studio - Programmation C#



5 jours - 35 heures

Code formation : LGDV210

www.adhara.fr

Classes métier semblables à celles du framework .NET : réalisation de propriétés, d'indexeurs et d'énumérateurs.

Génération de documentation.

Les nouveautés de C# 3.0 et 4.0 : expression d'initialisation, types anonymes, types dynamiques, paramètres optionnels, propriétés automatiques, méthodes d'extension, méthodes partielles et lambda expressions.

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.

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 et LINQ.

Qu'est-ce que LINQ : exemple avec LINQ To Entities et LINQ to SQL.