

Formation C# Développer en .NET Visual Studio

Durée :	5.0 jour(s)
Objectifs :	<ul style="list-style-type: none"> • Vous disposer d'une connaissance de la syntaxe C#. • Savoir maîtriser la programmation orientée objet en C#. • Apprendre à utiliser les fonctionnalités. • Savoir différencier les technologies d'accès aux données.
Prépare à la certification :	<ul style="list-style-type: none"> • Certification TOSA
Public :	<ul style="list-style-type: none"> • Développeurs désirant acquérir les techniques pour développer tous types d'applications .NET avec Visual Studio.
Prérequis :	<ul style="list-style-type: none"> • Disposer d'une expérience d'un langage de programmation (C, C++, VB...). • Connaître les principes de la programmation orientée objet. • Connaissance des concepts de bases de données relationnelles et du langage SQL.
Modalités et moyens pédagogiques	<p>Démonstrations visuelles et pratiques à travers des exercices d'application et/ou des cas concrets des stagiaires.</p> <p>Salle de formation équipée d'un poste PC par personne et de dispositif vidéo Grand Ecran.</p> <p>Portail web: maformation.vaelia.fr</p>
Modalités d'évaluation	Auto évaluation des acquis, exercices pratiques et/ou échanges avec le formateur.
Moyens d'encadrement	Un formateur expert spécialisé en Développement dont les compétences ont été validées par des diplômes et/ou testées et approuvées par l'éditeur et/ou Vaelia.
Satisfaction globale :	<p>4.80/5</p> <p><i>Calculée à partir des évaluations stagiaires sur les 12 derniers mois.</i></p>

Jour 1

Introduction à C# et au .NET framework

- Introduction au .Net Framework
- Création de projets avec Visual Studio
- Écrire une application C#
- Construire une application graphique
- Documenter une application
- Exécuter et déboguer des applications avec Visual Studio

Structure de programmation du C#

- Déclaration de variables et affectation de valeurs
- Utilisation d'expressions et d'opérateurs
- Création et utilisation des tableaux
- Instructions de décisions
- Instructions d'itérations

Déclaration et appel de méthodes

- Définir et appeler des méthodes
- Spécification de paramètres optionnels et de paramètres de sortie

Jour 2

Gestion d'exceptions

- Gestion des exceptions
- Soulever des exceptions

Lire et écrire dans des fichiers

- Accéder au système de fichiers
- Lecture et écriture dans des fichiers en utilisant les flux

Créer de nouveaux types de données

- Création et utilisation d'énumérations
- Création et utilisation de classes
- Création et utilisation de structures
- Comparaison des types références et types valeurs

Jour 3

Encapsulation de données et de méthodes

- Contrôler la visibilité des membres
- Partager méthodes et données

Héritage de classes et implémentation d'interfaces

- Utiliser l'héritage pour définir de nouveaux types références
- Définir et implémenter des interfaces
- Définir des classes abstraites

Gestion de la durée de vie des objets et contrôler les ressources

- Introduction au Garbage Collection
- Gestion des ressources

Jour 4

Encapsuler des données et définir des surcharges d'opérateurs

- Création et utilisation des propriétés
- Création et utilisation des indexeurs
- Surcharge d'opérateurs

Découpage de méthodes et gestion d'événements

- Déclaration et utilisation de délégué

- Utilisation des Lambda Expressions
- Gestion d'évènements

Utilisation des collections et construction de types génériques

- Utilisation des collections
- Création et utilisation des types génériques
- Définir des interfaces génériques et comprendre la variance
- Utilisation de méthodes génériques et des délégués

Jour 5

Construction et énumération de classes de collection personnalisées

- Implémenter une classe de collection personnalisée
- Ajouter un énumérateur à une classe de collection personnalisée

Utilisation de LINQ pour interroger des données

- Utilisation des méthodes d'extension LINQ et des opérateurs de requête
- Construction de requêtes et d'expressions LINQ dynamiques

Développement dirigé par les tests

- La place des tests dans le développement
- Modèles de conception d'application : MVC, MVVM
- Tests Unitaires dans Visual Studio