

Introduction à la programmation en C# : Travaux Pratiques - 4 jours

formation 419

- Vous apprendrez à**
- Créer, compiler et exécuter des programmes C# orientés objet en utilisant Visual Studio
 - Écrire et utiliser les constructions, la syntaxe et la sémantique du langage C#
 - Développer des programmes pour les applications de bureau et les applications Web en utilisant des architectures multicouche robustes
 - Développer des composants .NET réutilisables via la réalisation d'interface et des design pattern standard
 - Exploiter les espaces de nommage et les classes du framework .NET principaux
 - Accéder aux bases de données en utilisant LINQ (Language Integrated Query) et Entity Framework (EF)
- Objectif** C# est un langage de programmation moderne et orienté objet permettant de créer des programmes à la fois simples et robustes. Spécifiquement conçu pour exploiter les spécifications CLI, C# est le principal langage du framework Microsoft .NET. Lors de cette formation, vous acquerez les compétences pour exploiter les fonctionnalités de C# et du framework .NET et développer des programmes pour un grand nombre d'applications Web et de bureau.
- À qui s'adresse cette formation** Toute personne intéressée par la programmation en C#. Une expérience d'un langage tel que VB, Java, Pascal ou C/C++ est nécessaire. Les personnes ne connaissant que le langage COBOL, RPG, SQL, HTML ou équivalent, devraient suivre la formation 502, "Introduction à la programmation avec .NET".
- Travaux pratiques** Vous acquerez de l'expérience en créant vos propres applications en C#:
- Écriture et compilation de programmes C# à l'aide de Visual Studio
 - Construction de classes C# avec ou sans héritage
 - Écriture d'applications de bureau et Web avec les Windows Forms et les Web Forms
 - Création et déploiement de composants.NET personnalisés
 - Mise en œuvre d'une logique de requêtes pour les bases de données en utilisant LINQ et EF
 - Accélération du développement avec l'infrastructure .NET

Introduction à la programmation en C# : Travaux Pratiques - 4 jours

formation 419

Introduction au langage C#

L'évolution de C#

- Comparaison des différentes versions de C#
- Expression des modèles C# avec UML

C# et l'infrastructure .NET

- CLI (Common Language Infrastructure)
- La philosophie du code managé
- Le Common Intermediate Language (CIL) et les méta-données

Les bases du langage

Types de données et constructions de contrôles

- Déclaration de variables implicites et explicites
- Types valeur et référence
- Caractères et chaînes unicodes

Définition et appel de méthodes

- Spécifications de la méthode Main
- Passage d'arguments et retour de valeurs
- Étendue et durée de vie des variables
- Méthodes nommées et symboliques
- Gestion des exceptions
- Récupération des ressources

Utilisation des classes de la bibliothèque .NET

- Éviter les collisions en utilisant les espaces de nommage
- Réalisation d'E/S à l'aide des classes `Flot` et de la sérialisation
- Collections génériques et standards

Développement de classes C#

Définition des classes

- Utiliser les méthodes et les propriétés pour encapsuler des attributs
- Fournir une initialisation cohérente grâce aux constructeurs
- Surcharge des méthodes et des constructeurs
- Réutilisation grâce à l'héritage et au polymorphisme

Création et utilisation d'objets

- Allocation de mémoire d'objet avec `new`
- Passage de valeurs initiales aux constructeurs
- Choix entre l'allocation de valeurs ou de références
- Compartimentation et non-compartimentation
- Appel de méthodes et accès aux propriétés

Interconnexion d'objets

Association de classes

- Manipulation des références
- Équivalence physique et logique
- Choix des classes de collection de la bibliothèque
- Augmentation de la fiabilité grâce à l'utilisation des Generics

Exposition d'interfaces

- Définition de la spécification d'une interface
- Implantation d'une interface dans une classe
- Polymorphisme d'interface
- Événements et délégués

Simplification du développement de composants

Caractéristiques des composants .NET

- Manifestes et assemblages
- Déploiement de composants et d'applications
- Introspection à l'exécution en utilisant la réflexion
- Métamodèle d'assemblage .NET

Écriture de composants C#

- Création et utilisation de composants personnalisés
- Extension de `System.ComponentModel.Component`

Interfaçage avec des composants hérités

- Accès à COM/DCOM
- Outils pour la compatibilité dans les deux sens
- Appel de composants existants

Intégration de C# avec d'autres langages

- Harmonisation des composants grâce au CLR
- Accès aux méta-données
- Gestion des différences d'un langage à l'autre

Mise en œuvre et amélioration des solutions C#

Construction d'applications multinationales

- Exploiter des patterns architecturaux robustes (MVC et EDM)
- Remplacer l'interface utilisateur
- Coder les design pattern standard en C#
- Distribuer une application C#

Travailler avec des bases de données relationnelles

- Accéder à des bases de données avec Entity Framework
- Intégrer les fonctionnalités avancées de C# avec LINQ

Normes et versions

- Mise aux normes avec ECMA/ISO
- Caractéristiques des différentes normes C#

Techniques avancées

- Automatisation de la documentation avec XML
- Mise en place de l'interface `IEnumerable<T>`
- Utilisation de méthodes d'extension
- Utiliser des événements, des délégués et des expressions lambda
- Spécification d'attributs de développement