Beuscart Benjamin

Lupa Valentin

Malbranque Louis

Verpoort Alexia

En deux mots :

C# : Microsoft 2002

Partie de Frameworks.net. Combinaison des frameworks et du langage permet la cr éation d’une application.

Par rapport au C : permet la compatibilité car transformation en langage CIL ou MSIL plutôt qu’en binaire.

Par rapport au C++ : plus simple

Par rapport au java : plus de features.

C# est un langage orienté objet et orienté composant. -> EXPLICATION

Type-safe, gestion des exceptions, garbage collection

Compilation C# : empaquement dans assemblys qui implémentent des bibliothèques (.dll) ou des applications (.exe)

Types :

Types unifiés donc partage d’opération commune.

Prise en charge des types références et valeurs.

L’utilisateur peut créer ses propres types à l’aide des constructions struct, classe, interface, enum sur lesquelles nous reviendrons plus tard.

Type valeur : une variable de type valeur contient directement la donnée. Une opération sur une variable n’influence pas une autre variable de même type.

Type référence : une variable de type référence contient la référence à leur donnée. Une opération sur un variable peut influencer une autre variable de même type.

Type-safe permet de vérifier que les opérations sur les variables sont bien utilisées.

La déclaration d’une variable s’effectue en déclarant le type de la variable directement (float, int…) ou en utilisant var qui permet au compilateur de déduire. Une variable déclarée ne peut pas être assigné à une variable d’un autre type sans utiliser le type-casting qui est une conversion explicite d’une type. Certains cas entrainent cependant une conversion implicite s’il n’y a pas de perte de donnée et cela est effectué par le compilateur.

Exemple de cast : int a=10 ;

a=(double)a ;

<https://docs.microsoft.com/fr-fr/dotnet/csharp/language-reference/language-specification/introduction>

https://openclassrooms.com/fr/courses/1526901-apprenez-a-developper-en-c/2866796-tableaux-listes-et-enumerations