

Présentation de la plateforme dotNet

SOMMAIRE

L'ARCHITECTURE .NET	1
Présentation.....	2
Schéma d'architecture	3
SDK (Software Development Kit)	3
NFmk = « .Net Framework »	4
Fonctionnement d'une application .Net	7
ENVIRONNEMENT DE DEVELOPPEMENT	9
Environnements de développement.....	9
Installation de Visual Studio .Net 2010 Ultimate	10
Exécution en ligne de commande	11
Visual Studio : création d'un nouveau projet	12
Visual Studio : visualisation du nouveau projet	13
Visual Studio : paramétrage	15
Visual Studio : compilation / exécution	16
Aide en ligne du développeur	18
ANNEXE : TYPES DE FICHIERS	20
Projet local	20
Projet Web	22

L'architecture .Net

Présentation

« **.Net** » (Dot Net) désigne le nouveau modèle d'architecture logicielle de **Microsoft**, le nouvel environnement de développement et d'exécution associé, ainsi qu'un ensemble de spécifications (C#, CIL, ...). Le socle d'exécution « **CLR** » et le nouveau langage de Microsoft « **C#** » ont été normalisés par l'Ecma (European Computer Manufacturer's Systems).

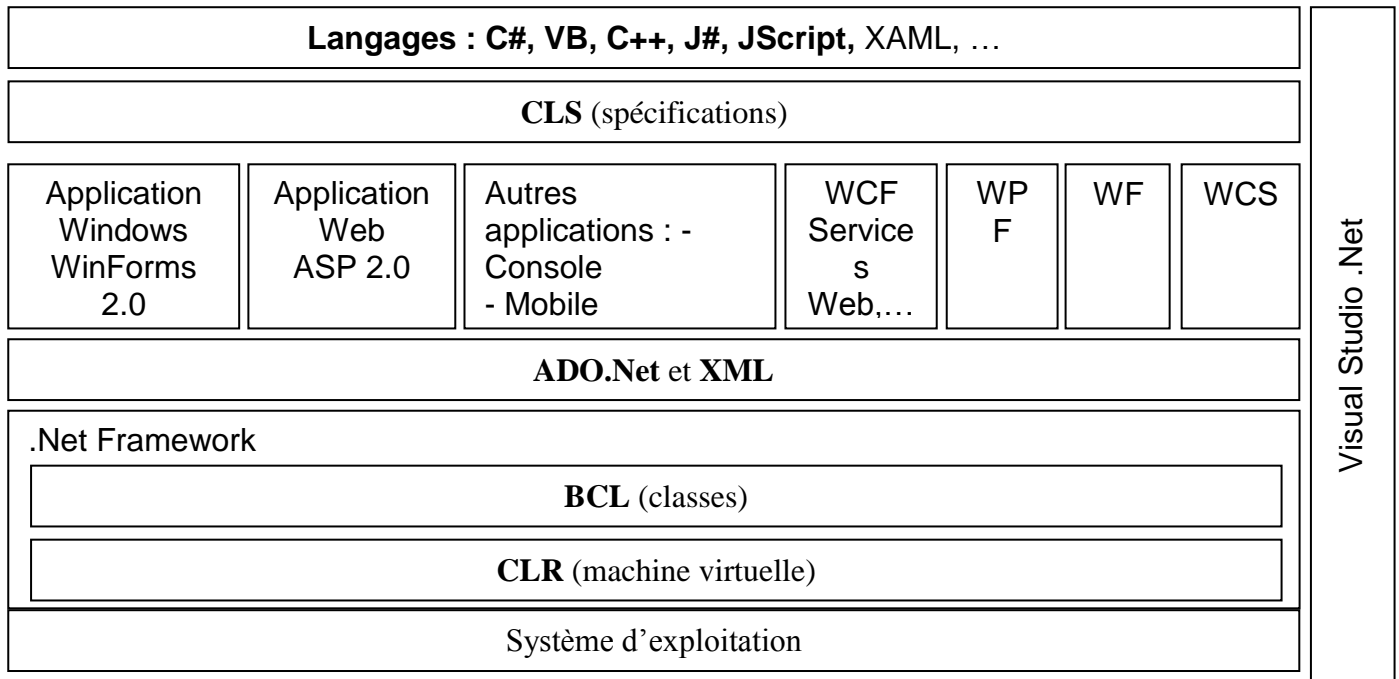
Historique

Date	Framework	EDI	Langage
2002	Fmk 1.0	Visual Studio 2002	C# 1.0
2003	Fmk 1.1	Visual Studio 2003	
2005	Fmk 2.0	Visual Studio 2005	C# 2.0
2006	Fmk 3.0 (ex WinFX) = Fmk 2.0 ... + WPF (Windows Presentation Foundation, ex Avalon) = moteur d'affichage basé sur XAML (pour Vista, mais fonctionne sous XP et 2003) + WCF (Windows Communication Foundation, ex Indigo) = serveur d'application regroupant services Web , .Net Remoting, MSMQ et BizTalk + WF (Windows Workflow Foundation) = moteur de workflows + WCS (Windows Card Space, ex Infocard) = technique de carte d'identité virtuelle		
Fin 2007	Fmk 3.5 = Fmk 3.0 ... + Implémentation pour Ajax ASP.Net + Langage de requête Linq + etc.		
Fin 2007		Visual Studio 2008 « Orcas » : Nouveaux outils de développement ... Web en Ajax et Silverlight (plugin WPF, ex WPF/E = WPF Everywhere) Plate-forme Office System 2007 - Vista (mise en œuvre simplifiée de WPF/XAML, WCF, WF)	C# 3.0
2012	Fmk 4.0 « Hawaï » : code modulaire permettant ... - un téléchargement à travers Internet uniquement des modules dont l'application à besoin - une installation de l'essentiel du framework dans Windows Server Core		

Plates-formes cibles

- Plate-forme propriétaire : Windows, Pocket PC (ex Win CE).
- Projet de portage du Framework (voir définition ci-dessous) :
 - projet **Mono 2.2** compatible DotNet 2.0 : Linux, Unix, Free BSD, Solaris et Mac OS
 - projet **Rotor** : Free BSD
 - projet **DotGnu** : Portable.DotNet.
- Annonce de portage sous Stinger (téléphones mobiles), Xbox (console de jeux) et TV Platform.

Schéma d'architecture



SDK (Software Development Kit)

Le **SDK** est un ensemble de compilateurs et d'outils de développement.

☞ Environnement de développement : **Visual Studio .Net** (voir ce chapitre).

☞ **Une vingtaine de langages** ont été complétés pour être compatibles avec la technologie « .Net ». Ils se partagent un système unique de types de données et une même bibliothèque de classes (**BCL**, voir ci-dessous).

- Langages fournis par Microsoft : Visual Basic (incompatible avec VB6), Visual C++, Visual C#, Visual J# (Java), JScript (devient compilé).

C# (C Sharp) est un nouveau langage réalisant la synthèse de C, C++ et Java : ramasse-miettes, typage fort, code managé ou non (avec pointeurs).

- Autres langages disponibles : C, Ada, Cobol, Fortran (Salford), F#, RPG, Perl, Eiffel, Python, Pascal, Delphi, Smalltalk, Mercury, Mondrian, Oberon, Standard ML, Dialog APL, Haskell, ...

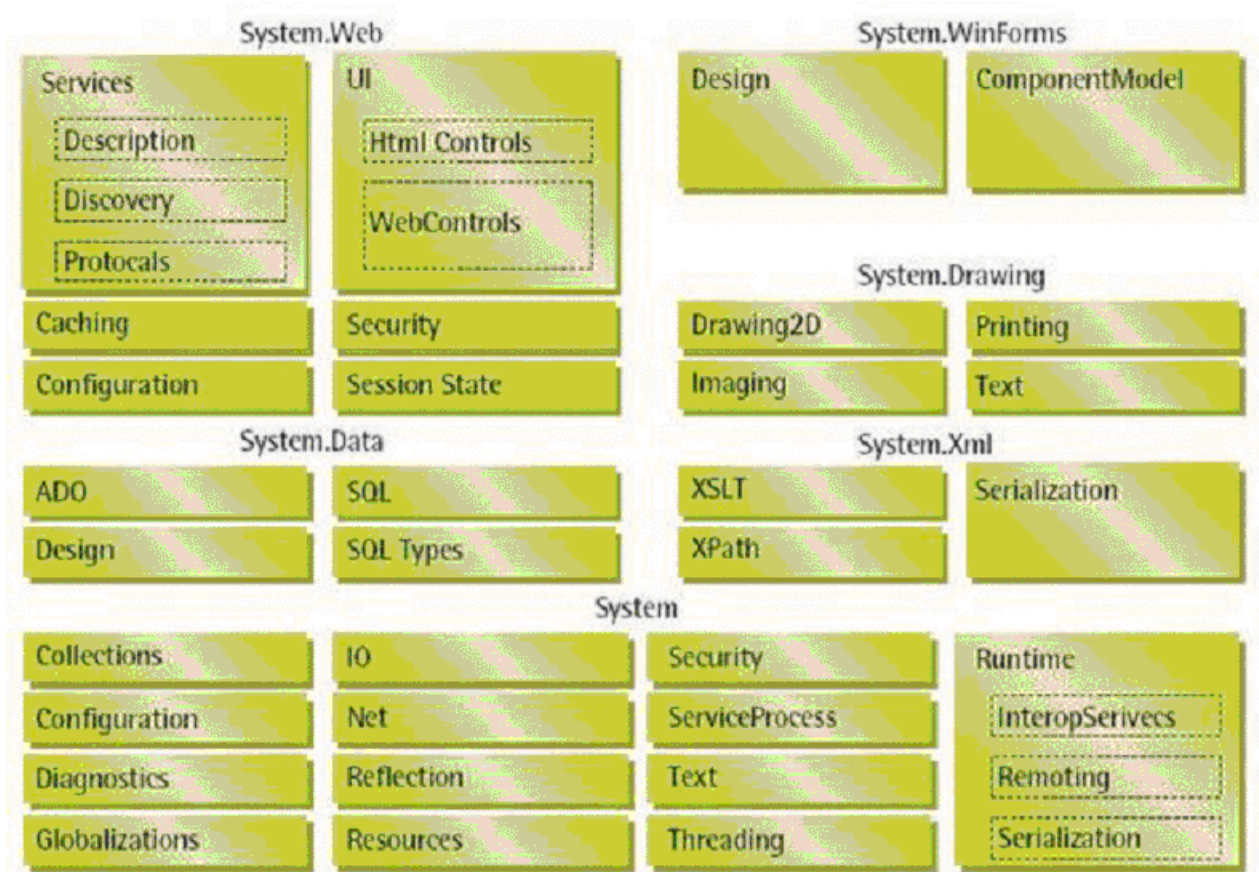
Pour qu'un langage soit portable sur « .Net », il faut qu'il réponde aux spécifications du **CLR** (voir ci-dessous) et qu'il existe un compilateur pour .Net. Le **CLS** (Common Language Specification) définit l'interface que doit respecter le langage pour attaquer les bibliothèques .Net.

☞ **Interopérabilité des langages** : par exemple, une classe écrite en C# peut être instanciée en VB.Net.

☞ **Interopérabilité** avec COM (ActiveX et anciennes DLL par COM Interop) et Win32 (API Windows).

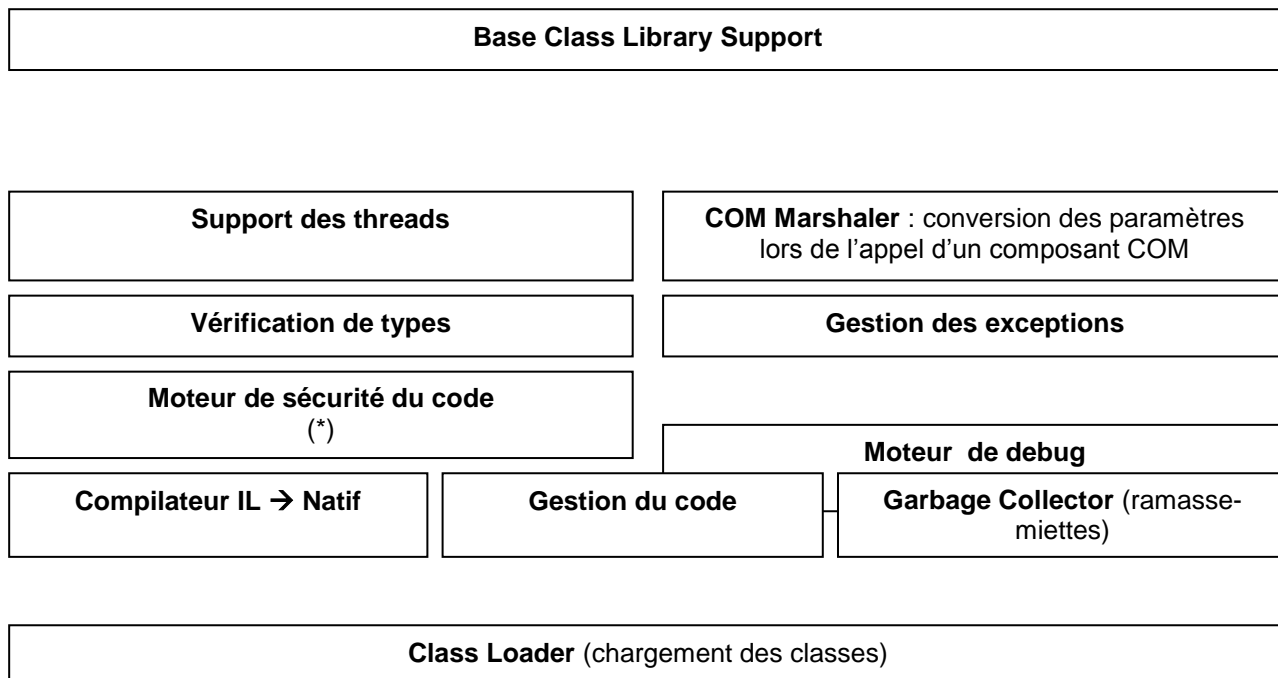
NFmk = « .Net Framework »

☞ **BCL** (Base Class Library) = **FCL** (Framework Class Library) = mscorlib.dll : **bibliothèque standard de classes** (ainsi que de structures et d'énumérations) de base commune (8 500 classes, chacune composée de dizaines de propriétés et méthodes) fournissant des services sous forme de **NamesSpaces** (espace de noms) = ensembles de classes = thèmes : System, Data, XML, Drawing, Web,...



BCL repose sur Win32, mais propose une vision objet des API. BCL évité le problème des nombreuses API complexes (typées C) et redondantes.

☞ **CLR** (Common Language Runtime) : **Noyau d'exécution** (=run-time équivalent à la JVM = machine virtuelle Java) : chargement des assemblages dans l'espace mémoire du processus, compilation JIT du code CIL.



(*) **Sécurité du code** : un composants managé se voit attribuer un niveau de confiance en fonction de son origine (Internet, réseau local ou poste local) définit les droits d'accès de l'application aux ressources (fichiers, base de registre, ...).

Le CLR est normalisé par l'**Ecma** et l'**ISO**.

Le CLR peut être hébergé ...

- par le système d'exploitation (Windows, Pocket PC, Linux, ...),
- par ASP.Net,
- par le SGBD (SQL Server 5, DB2-UDB 8.2 « Stinger »),
- par un composant non managé qui le charge dans son processus, ...

Internet Explorer est un exemple d'application non managée qui héberge le CLR (sous la forme d'une extension MIME). ceci permet d'incorporer des contrôles WinForms dans des documents HTML

Plusieurs versions du CLR peuvent cohabiter sur la même machine (mscorlib.dll / mscorsvr.dll).

🔑 Framework = BCL + CLR

Version de Fmk	Prix	Description
NFmk SDK	gratuit	Le Kit de développement Microsoft .NET Framework SDK 2.0 comprend des outils, de la documentation et des exemples utiles aux développeurs pour écrire, générer, tester et déployer des applications .NET Framework sur des plates-formes x86
NFmk Redistribuable	gratuit	Pour le déploiement client Windows ou serveur Web. Contient les classes de bases (BCL) et l'environnement d'exécution (CLR)
.Net Compact Fmk	gratuit	Pour périphérique mobile (PDA,...)
Micro Fmk.Net	gratuit	Pour application embarquée « contrainte » (c'est-à-dire disposant de peu de ressources) telles que les télécommandes, les appareils électroménagers,...

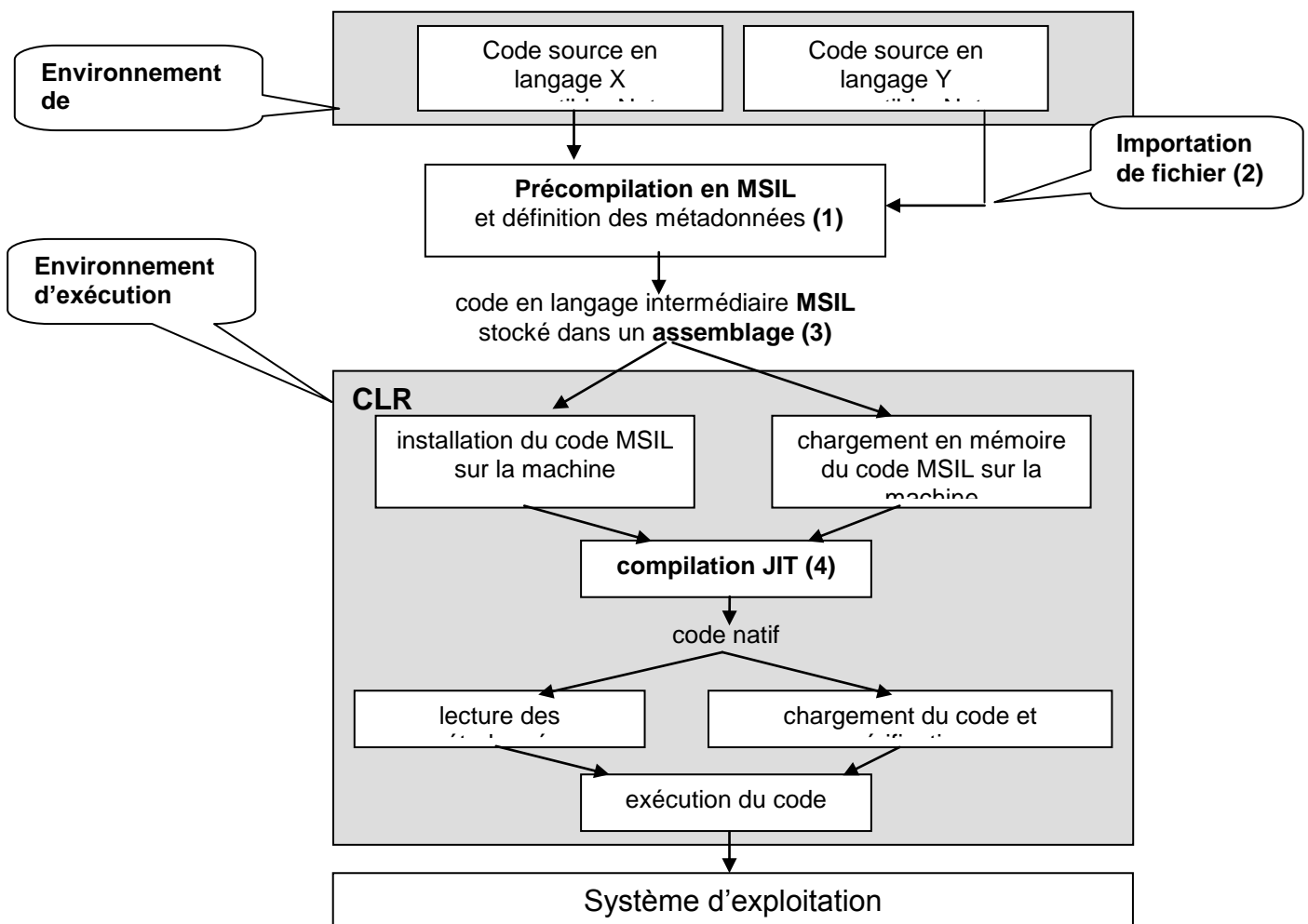
Le Framework est intégré aux versions récentes de Windows.

Fonctionnement d'une application .Net

Pour du **code non managé** (possible uniquement en C++ et C#), il y a exécution directe du code natif issu de la compilation.

Pour du **code managé**, la traduction du programme s'effectue en deux temps :

Etape	Outil	Format d'origine	Format produit
1- Précompilation	Précompilateur = compilateur spécifique au langage	Code source (C#, VB,...)	Code MSIL (équivalent du bytecode java)
2- Exécution	CLR = machine virtuelle = compilateur JIT + moteur d'exécution	Code MSIL	Code exécutable (natif / plate-forme)



(1) Le compilateur du langage concerné traduit le code source en instructions **MSIL** (Microsoft Intermediate Language).

Le MSIL est l'implémentation Microsoft du **CIL** (Common Intermediate language) qui est l'équivalent du bytecode Java. C'est un langage semi-compilé (proche de l'assembleur) qui est commun à tous les langages.

Le compilateur produit également des **métadonnées** : elles ont pour rôle de décrire les types de données en provenance du code source (stockées dans des fichiers au format binaire).

(2) Grâce à l'enregistrement des types dans des métadonnées, il est possible de récupérer un type de données dans un langage autre que celui dans lequel il a été défini.

(3) L'assemblage (« assembly ») est l'unité de déploiement (EXE ou DLL). Attention, un exécutable « .Net » a besoin du CLR pour s'exécuter.

(4) Compilation JIT (Just In Time) des instructions MSIL en code natif (dépendant de la plateforme d'exécution) permettant d'exécuter le même assemblage sur plusieurs types de machines.

Environnement de développement

Environnements de développement

Environnement	Origine	Date	Fmk	Prix	Langages	Remarques
Compilateur en ligne Exemple : csc.exe	Microsoft	2002		gratuit (fourni avec le Framework)	Un compilateur par langage (exemple : C#)	Compilateur en ligne de commande
Visual Studio 2005 : - Versions Express	Microsoft	2005	2.0	gratuites	- C#, C++, VB.Net, J# - Web Developer (ASP) - SQL Server allégé	Ces versions font entre 35 et 70 Mo
Visual Studio 2005 - Version Standard - Version Professional	Microsoft	2005	2.0	320 € 855 €	C#, C++, VB.Net, J#, ASP.Net	- version française : début 2006 - SP 1 : juin 2006 avec Fmk 2.1
Visual Studio .Net 2005 - Version Team System VS Team Architect VS Team Developer VS Team Test Team System Suite	Microsoft	2006	2.0	- 5 899 € - 5 899 € - 5 899 € - 11 711 €		Offre des outils de gestion de versions, de tests, d'analyse de performance et de gestion de projet
SharpDevelop 2	open source	2006	2.0	gratuit	C#	http://www.icsharpcode.net/opensource/sd/Default.aspx
C# Builder 2006	Borland	2006	1.1	gratuit pour la version personnelle	C#	
Delphi 2006	Borland	2006	1.1	payant	Pascal, C#	
Dreamweaver CS3	Macromedia	2005	1.1	payant	ASP.Net	
Plug-in C# pour Eclipse	Open source				C#	

☞ **Visual Studio** est un IDE (Integrated Development Environment), ou EDI (Environnement de Développement Intégré), commun à tous les langages .Net. Il assure toutes les fonctions liées au développement d'application : édition/compilation de code, debug, maquettage de l'interface utilisateur (fenêtre Windows, page Web, états d'imprimante, ...), lien avec le SGBD, ...

Installation de Visual Studio .Net 2010 Ultimate

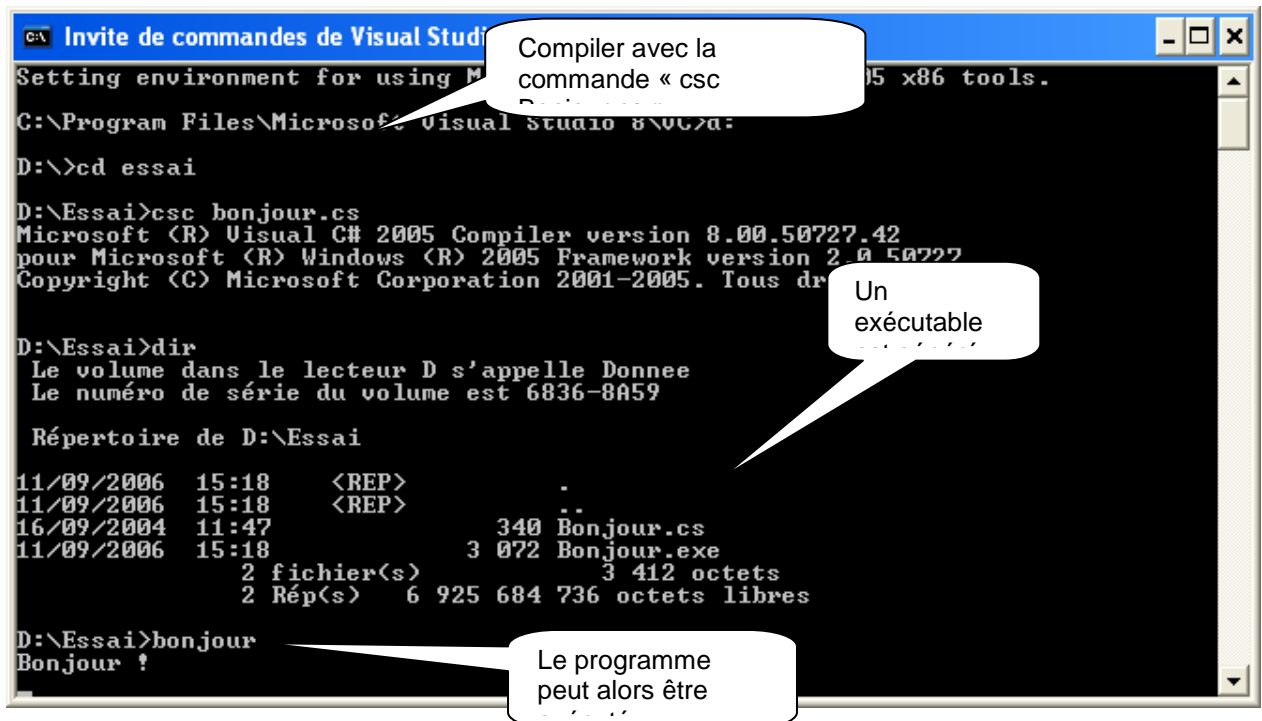
Le programme d'installation se trouve dans
`Z:\Developpement\vs2010ultimate`

Lancez le fichier setup.exe, suivez l'assistant d'installation.

Exécution en ligne de commande

- ☞ Créer un simple fichier source avec le Bloc-notes.
- ☞ Compiler à partir de la ligne de commande : **csc.exe** (voir aussi « al.exe »)

Utiliser le fichier de commande fourni avec Visual Studio « Visual Studio Tools – Invite de commande de Visual Studio 2010 ». Celui-ci complète la variable d'environnement PATH pour accéder au compilateur.



The screenshot shows a Windows command prompt window titled "Invite de commandes de Visual Studio". The window contains the following text:

```
Setting environment for using Microsoft Visual Studio 8.00 x86 tools.  
C:\Program Files\Microsoft Visual Studio 8\VC>cd essai  
D:\>cd essai  
D:\Essai>csc bonjour.cs  
Microsoft (R) Visual C# 2005 Compiler version 8.00.50727.42  
for Microsoft (R) Windows (R) 2005 Framework version 2.0.50727  
Copyright (C) Microsoft Corporation 2001-2005. Tous droits réservés.  
D:\Essai>dir  
Le volume dans le lecteur D s'appelle Donnee  
Le numéro de série du volume est 6836-8A59  
  
Répertoire de D:\Essai  
11/09/2006 15:18 <REP> .  
11/09/2006 15:18 <REP> ..  
16/09/2004 11:47 340 Bonjour.cs  
11/09/2006 15:18 3 072 Bonjour.exe  
2 fichier(s) 3 412 octets  
2 Rép(s) 6 925 684 736 octets libres  
  
D:\Essai>bonjour  
Bonjour !
```

Three callout boxes provide additional context:

- "Compiler avec la commande « csc »" points to the `csc bonjour.cs` command.
- "Un exécutable" points to the `Bonjour.exe` file in the directory listing.
- "Le programme peut alors être" points to the `bonjour` command, which runs the executable.

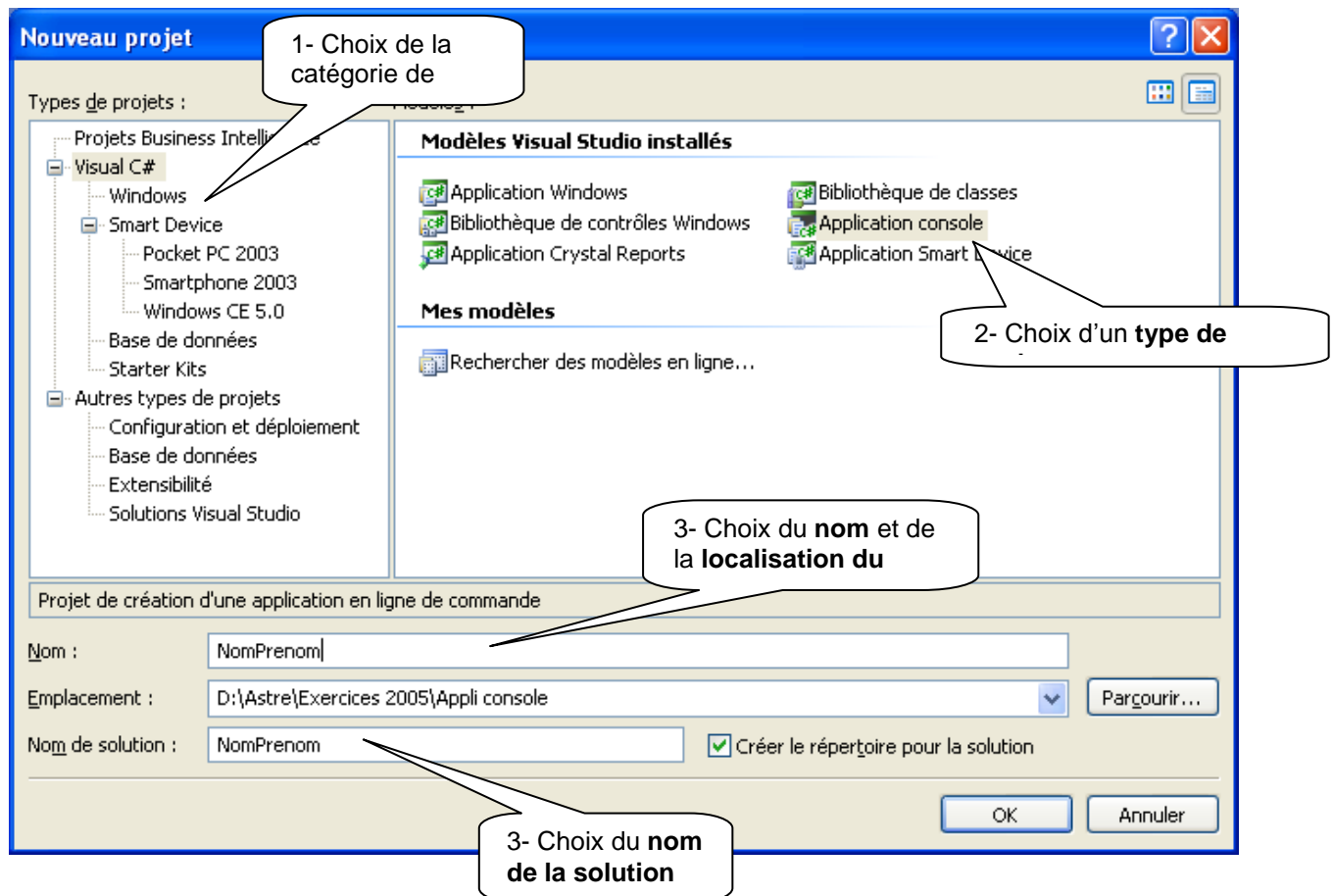
Exemple de compilation de « File.cs » pour produire « File.exe » : `csc File.cs`

Exemple de compilation de « File.cs » pour créer « My.exe » : `csc /out:My.exe File.cs`

Visual Studio : création d'un nouveau projet

☞ Le menu « **Fichier – Nouveau ...** » offre le choix de créer un nouveau projet, site web ou fichier.

☞ **Nouveau projet** : les choix possibles dépendent de ce qui a été installé

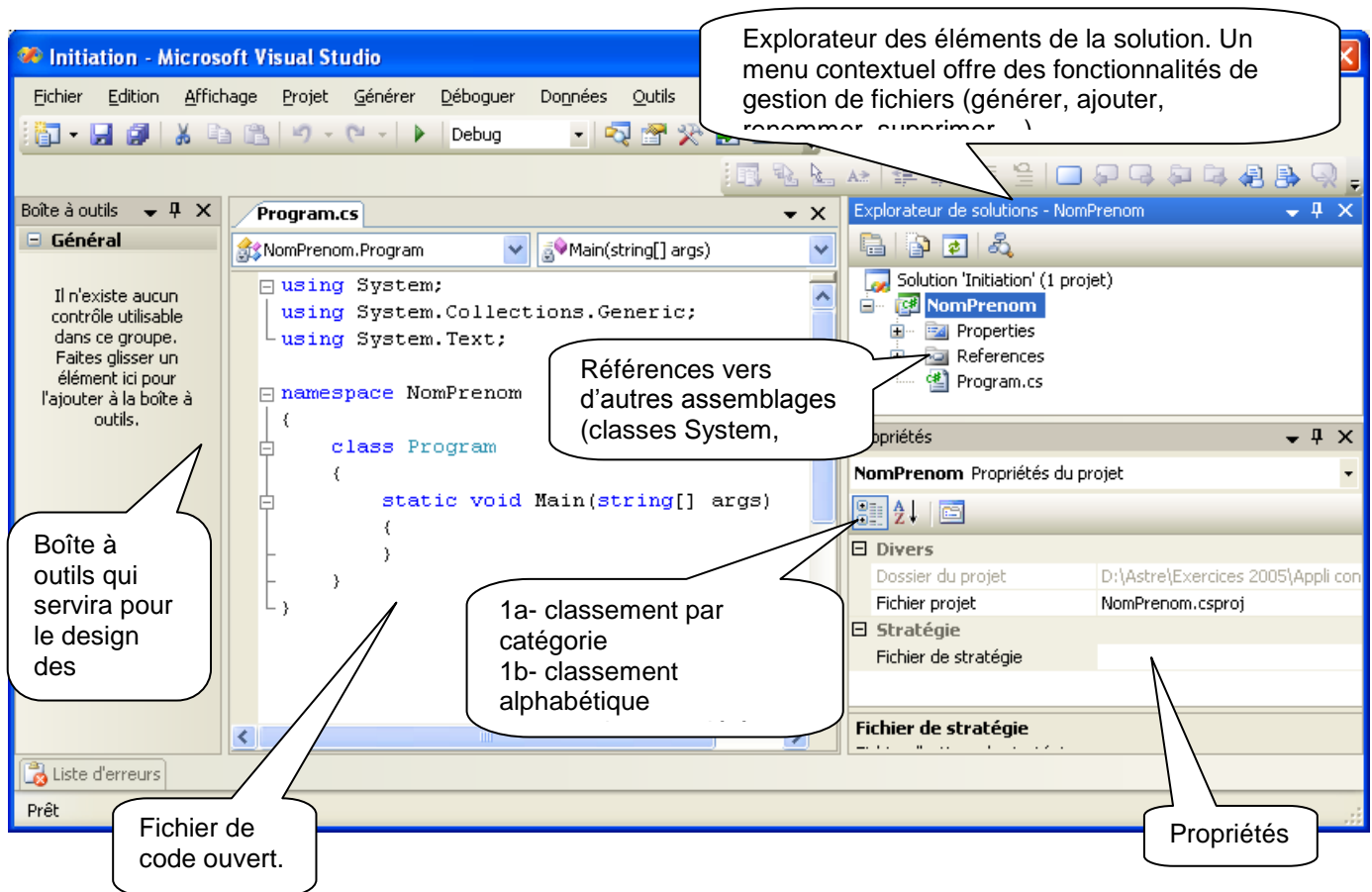


Visual Studio : visualisation du nouveau projet

☞ Sur le disque.

Dossiers	Description
<i>solution</i> <i>projet</i> bin obj properties	Les répertoires bin et objet contiendront les versions exécutables.
Fichiers	Description
<i>solution.sln</i>	Fichier de solution = Organise les projets, les éléments de projet et les éléments de solution au sein de la solution en fournissant à l'environnement les références relatives à leur emplacement sur le disque.
<i>projet.csproj</i>	Fichier de projet C#.
<i>projet.csproj.user</i>	Fichier d' options de projet C#.
AssemblyInfo.cs	Fichier de description de l'assemblage (auteur, version, ...).
Program.cs	Fichier de code qui contient une déclaration de classe avec la méthode Main().

☞ Sous Visual Studio.



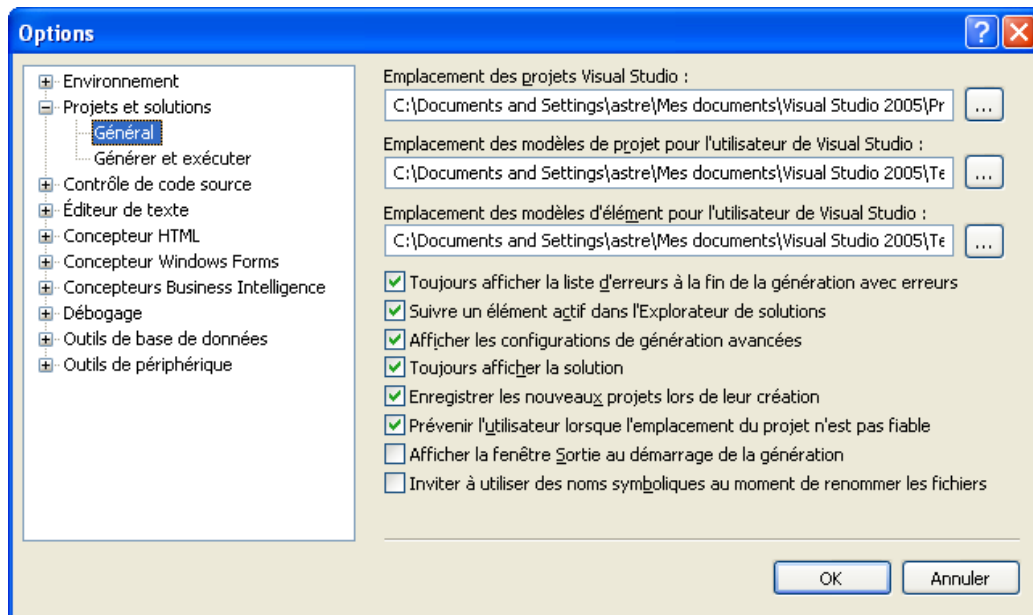
Un **projet** peut contenir **plusieurs répertoires** qui correspondent en général à des **namespaces** distincts

Une **solution** peut contenir **plusieurs projets** : par exemple, un projet d'application Windows, un projet de bibliothèque de classes et un diagramme UML réalisé sous Visio. Si aucune solution n'est ouverte, la création d'un nouveau projet crée automatiquement une nouvelle solution. Dans l'écran ci-dessus, la solution « Initiation » contient un projet qui a été renommé en « NomPrenom ».

💡 Par défaut, le nom d'une solution mono-projet ne s'affiche pas. Cela peut se modifier avec le menu « Outils – Options – Projets et solutions – Général » - case « **Toujours afficher la solution** ».

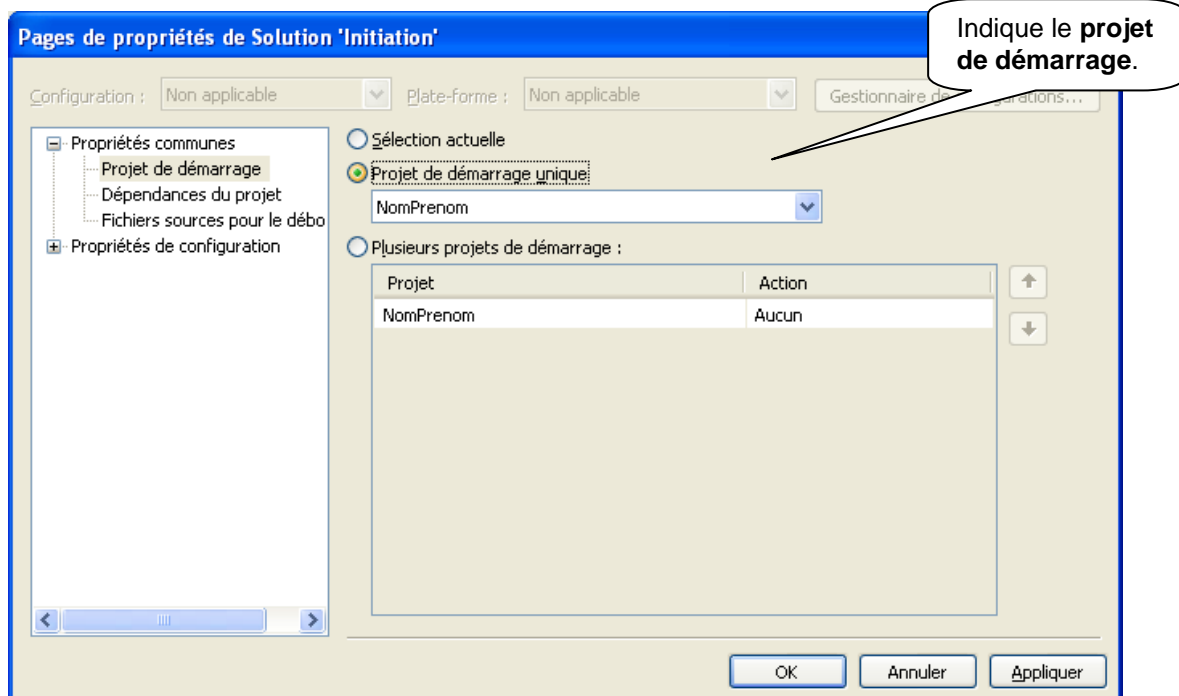
Visual Studio : paramétrage

☞ **Paramétrage de l'environnement de développement** : menu « Outils – Options »



☞

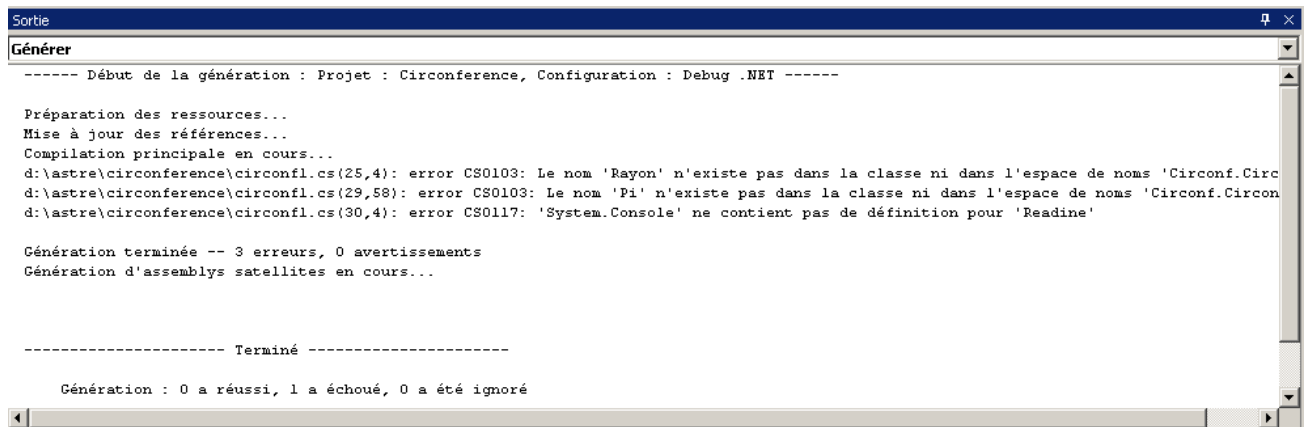
Paramétrage de la solution : « Explorateur de solutions » cliquer sur la solution - menu contextuel « Propriétés »



Visual Studio : compilation / exécution

☞ Compilation

Le **compilateur** effectue une **analyse syntaxique du code**, qui délivre les messages d'erreurs éventuels, avant de produire le code exécutable.



The screenshot shows the 'Sortie' (Output) window in Visual Studio, with the 'Générer' (Generate) tab selected. The window displays the output of a compilation process for a project named 'Circonference' in 'Debug .NET' configuration. The output shows the start of generation, resource preparation, reference updates, and the start of the main compilation. Three errors are listed: two CS0103 errors for 'Rayon' and 'Pi' not existing in the 'Circonf.Circ' namespace, and one CS0117 error for 'Readline' not being a member of 'System.Console'. The generation process then terminates, reporting 3 errors and 0 warnings. A summary line at the bottom indicates that 0 generations were successful, 1 failed, and 0 were ignored.

```
Sortie
Générer
----- Début de la génération : Projet : Circonference, Configuration : Debug .NET -----

Préparation des ressources...
Mise à jour des références...
Compilation principale en cours...
d:\astre\circonference\circonfl.cs(25,4): error CS0103: Le nom 'Rayon' n'existe pas dans la classe ni dans l'espace de noms 'Circonf.Circ
d:\astre\circonference\circonfl.cs(29,58): error CS0103: Le nom 'Pi' n'existe pas dans la classe ni dans l'espace de noms 'Circonf.Circon
d:\astre\circonference\circonfl.cs(30,4): error CS0117: 'System.Console' ne contient pas de définition pour 'Readline'

Génération terminée -- 3 erreurs, 0 avertissements
Génération d'assemblés satellites en cours...

----- Terminé -----

Génération : 0 a réussi, 1 a échoué, 0 a été ignoré
```


☞ Le **lancement de l'application** déclenche automatiquement la précompilation (si celle-ci n'a pas été faite).

- **Icône Démarrer ►**
- **Menus** « Déboguer – Démarrer le débogage » et « Déboguer – Exécuter sans débogage »

☞ Il est aussi possible de lancer la **précompilation sans exécution** : menus « Générer – Générer/Régénérer solution/projet ».

Aide en ligne du développeur

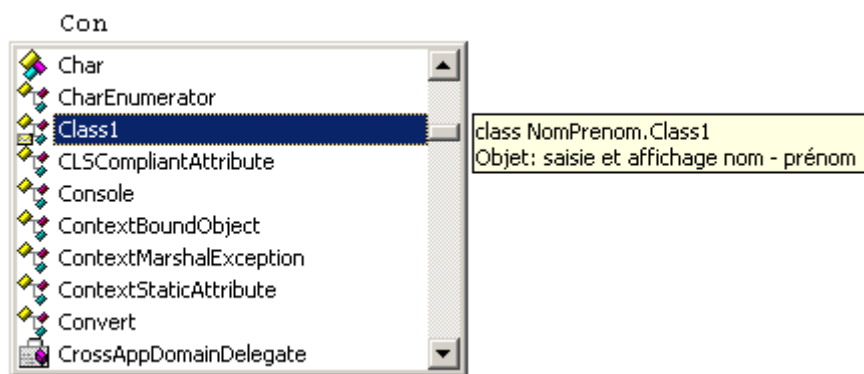
☞ **IntelliSense** (AutoCompletion) propose un ensemble d'options qui facilitent l'accès aux guides de référence du langage lors de la saisie du code.

- **Infos express** : bulle d'aide sur l'élément pointé par la souris

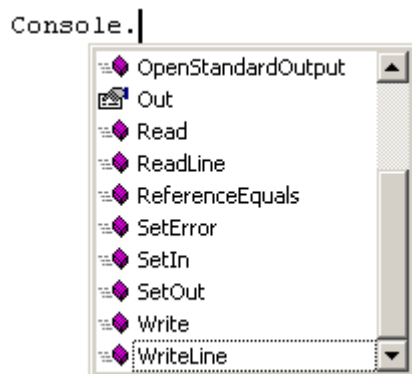
```
Console.Write("\n\n\tNom : ");  
strNom=Console.  
Console.Write  
strPrenom=Console.ReadLine();
```

void Console.Write (string value) (+ 17 surcharges)
Écrit la valeur de chaîne spécifiée dans le flux de sortie standard.

- Saisie semi-automatique lors de frappe du code : taper quelques lettre, puis « Ctrl+Espace ».



- Saisie semi-automatique des membres d'une classe : taper le nom de la classe, puis le point.



- Saisie semi-automatique des paramètres d'une méthode : taper le nom de la méthode, puis une parenthèse ouvrante.

```
Console.Write(  
▲ 1 sur 18 void Console.Write (string value)  
value: Valeur à
```

Les flèches haut/bas
permettent de naviguer
entre les différentes
surcharges de la

☞ **MSDN** (Microsoft developer Network)

Le site <http://msdn.microsoft.com/fr-fr/library/> regorge d'informations sur C# et Visual Studio.

Cette documentation est accessible en appuyant sur le touche F1 dans visual studio.

Annexe : types de fichiers

Projet local

Élément de projet	Extension	Objet de l'élément de projet
Windows Form	.cs	Formulaire destiné à la création d'applications Windows.
Classe	.cs	Fichier de code qui contient une déclaration de classe.
Classe de composant	.cs	Classe pour la création de composants à l'aide du concepteur visuel.
Contrôle utilisateur	.cs	Classe destinée à la création d'un contrôle Windows Form à l'aide du concepteur visuel.
Assistant Formulaire de données	.cs (.aspx pour les projets Web)	Formulaire de données destiné à la création d'applications Windows.
DataSet	.xsd	Fichier destiné à la création d'un schéma XML à l'aide de classes DataSet .
Fichier XML	.xml	Fichier XML vide.
Schéma XML	.xsd	Fichier destiné à la création d'un schéma destiné aux documents XML.
Fichier de code	.cs	Fichier de code vide.
Contrôle Custom	.cs	Classe destinée à la création d'un contrôle Windows Form dessiné par l'utilisateur.
Page HTML	.htm	Page HTML pouvant inclure du code côté client.
Formulaire hérité	.cs	Formulaire reposant sur un Windows Form existant.
Contrôle Web personnalisé	.cs	Classe destinée à la création d'un contrôle serveur ASP.NET..
Contrôle utilisateur hérité	.cs	Nouveau contrôle reposant sur un contrôle Windows Form existant.
Service Windows	.cs	Classe destinée à la création de services Windows.
Classe COM	.cs	Classe pouvant être exposée au modèle COM..
Composant transactionnel	.cs	Classe destinée à être utilisée avec des composants transactionnels.
Fichier texte	.txt	Un fichier texte vide.
Jeu de frames	.htm	Fichier HTML qui contient plusieurs pages HTML.
Fichier XSLT	.xslt	Fichier utilisé pour transformer des documents XML.
Feuille de style	.css	Feuille de style en cascade utilisée pour les définitions de style HTML élaborées.
Classe Installer	.cs	Classe à appeler au moment de l'installation..
Crystal Report	.rpt	Fichier Crystal Report qui publie des données dans un formulaire Windows.
Fichier bitmap	.bmp	Un fichier image bitmap vide qui peut servir à créer des images simples.
Fichier curseur	.cur	Fichier qui permet de créer des curseurs personnalisés.
Fichier icône	.ico	Un fichier image qui permet de créer une icône personnalisée.
Fichier de ressources de l'assembly	.resx	Fichier de ressources d'application au format XML.
Fichier d'informations de l'assembly	.cs	Fichier contenant des informations d'assembly générales.
Fichier de configuration d'application	.config	Fichier utilisé pour définir des paramètres de configuration d'une application.
Fichier JScript	.js	Fichier de script contenant du code JScript.
Fichier VBScript	.vbs	Fichier de script contenant du code VBScript.
Windows Script Host	.wsf	Fichier contenant un script exécuté comme un programme

		Windows.
--	--	----------

Projet Web

Élément de projet	Extension	Objet de l'élément de projet
Web Form	.aspx et .cs	Formulaire destiné à la création d'applications Web.
Service Web	.asmx et .cs	Composant offrant la possibilité d'échanger des messages capables d'interopération dans un environnement faiblement couplé à l'aide de protocoles standard, tels que HTTP, XML, XSD, SOAP et WSDL.
Formulaire web mobile	.aspx	Formulaire destiné à la création d'applications Web mobiles.
Classe	.vb ou .cs	Fichier de code qui contient une déclaration de classe.
Module (Visual Basic uniquement)	.vb	Fichier permettant de stocker des groupes de fonctions.
Classe de composant	.vb ou .cs	Classe pour la création de composants à l'aide du concepteur visuel.
Assistant Formulaire de données	.aspx (.cs pour les projets locaux)	Formulaire de données destiné à la création d'applications Web. Pour plus d'informations, consultez Assistant Formulaire de données.
DataSet	.xsd	Fichier destiné à la création d'un schéma XML à l'aide de classes Data .
Contrôles utilisateur Web	.ascx	Contrôle serveur ASP.NET créé à l'aide du concepteur visuel.
Contrôle Utilisateur Web Mobile	.ascx	Contrôle serveur ASP.NET créé à l'aide du concepteur visuel et utilisé dans une application Web mobile.
Page HTML	.htm	Page HTML pouvant inclure du code côté client.
Jeu de frames	.htm	Fichier HTML qui contient plusieurs pages HTML.
Feuille de style	.css	Feuille de style en cascade utilisée pour les définitions de style HTML élaborées.
Fichier XML	.xml	Fichier XML vide.
Schéma XML	.xsd	Fichier destiné à la création d'un schéma destiné aux documents XML.
Fichier XSLT	.xslt	Fichier utilisé pour transformer des documents XML. Pour plus d'informations, consultez Guide du développeur XSLT.
Contrôle Web personnalisé	.cs	Classe destinée à la création d'un contrôle serveur ASP.NET. ASP.NET .
Fichier de code	.cs	Fichier de code vide.
Fichier de découverte statique	.disco	Fichier facultatif qui se comporte comme un mécanisme de découverte pour le service Web XML. Le fichier .disco n'est pas créé automatiquement pour un service Web XML.
Classe d'application globale	.asax	Parfois appelé fichier asax, ce fichier vous permet d'écrire du code pour gérer les événements d'application ASP.NET globaux comme Session_OnStart et Application_OnStart . Le fichier porte le nom global.asax, lequel ne peut être modifié.
Fichier de configuration Web	.config	Fichier utilisé par ASP.NET pour configurer les paramètres Web d'un projet Web. Le fichier porte le nom Web.config, lequel ne peut être modifié.
Fichier texte	.txt	Un fichier texte vide.
Classe Installer	.cs	Classe à appeler au moment de l'installation..
Crystal Report	.rpt	Fichier Crystal Report qui publie des données dans un Web Form.
Fichier bitmap	.bmp	Un fichier image bitmap vide qui peut servir à créer des images simples.
Fichier curseur	.cur	Fichier qui permet de créer des curseurs personnalisés.
Fichier icône	.ico	Un fichier image qui permet de créer une icône personnalisée.
Fichier de ressources de l'assembly	.resx	Fichier utilisé pour modifier et définir des ressources d'application.
Fichier d'informations de l'assembly	.cs	Fichier contenant des informations d'assembly générales.
Fichier JScript	.js	Fichier de script contenant du code JScript.
Fichier VBScript	.vbs	Fichier de script contenant du code VBScript..
Windows Script Host	.wsf	Un fichier de code vide utilisé pour les scripts Windows..

