* **Initialiseurs d'objets**

Les initialiseurs d'objet vous permettent d'attribuer des valeurs aux champs ou propriétés accessibles d'un objet lors de la création sans avoir à appeler un constructeur suivi de lignes d'instructions d'affectation. La syntaxe d'initialisation de l'objet vous permet de spécifier des arguments pour un constructeur ou de les omettre (ainsi que la syntaxe entre parenthèses). L'exemple suivant montre comment utiliser un initialiseur d'objet avec un type nommé Catet comment appeler le constructeur sans paramètre. Notez l'utilisation de propriétés implémentées automatiquement dans la Catclasse. Pour plus d'informations, voir [Propriétés implémentées automatiquement](https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/classes-and-structs/auto-implemented-properties) .

# **L’inférence de type var**

* Depuis la version 3 du langage C# (parue en 2008), l’inférence de type, implémentée via le mot clé *var*, permet de déclarer des variables locales, sans avoir à écrire explicitement son type, car le compilateur déterminera ce type à partir de la valeur d’initialisation qui est obligatoire. Voici quelques exemples :

Var Code = "125ER78";   *// Variable est de type string*

var Identifiant = 1;    *// Variable est de type int*

* **Les Types anonymes :**
* Vous pouvez déclarer une variable locale de type anonyme de la façon suivante :

var variableTypeAnonyme = new { FirstName = "Flavien", Age = 23, DateOfBirth = new DateTime(1984, 11, 15) };

Le compilateur va donc créer une classe générique qui aura les propriétés correspondantes, et dont les types sont inférés grâce à l'expression d'initialisation (ici, FirstName de type string, Age de type int et DateOfBirth de type DateTime)