C#; cours 2

14/01/2019

Syntax:

* “C” DNA
  + Bloc {}
  + Separateur ;
  + Control : if, …
  + Operateurs : + - \* / % & && | || ^ >> <<
* Identifiants contraintes
  + Alpahnumérique + “\_”
  + Ne peut commencer par un nombre
  + Ne pas commencer par un ‘\_’
* Readibility > Brevity
* Transcient (variable) elements (var, field, param) -> CamelCasing
* Strucutal elements (Ckass, const, type) -> PascalCasing
* Standard
  + Class
  + Interface
  + Enum
* Less Standard
  + Struct
  + Delegate
* Uncommon (associated with unsafe)
  + \* (pointer to)
* Static = no instanciation required
  + Stateless class / helper methods
  + No memory

**Classes**

* Encapsulation required
* Static: break encapsulation, no instanciation

Usage of “Static”:

* Stateless libraries
* Helper methods

**Encapsulation**

Access levels : Public, Protected, Internal (limited to assembly of declaration), Private

Variable privé par défaut, donc la déclarer public quand doit y accéder en dehors des {}

**Getter & setter**

Just write simplified get & set in property declaration, will be called upon invocation with “.”

No accessor code is required

**Heritage**

**Class Aeroplane : FlyingObject**

All classes inherit from System.object

Can only inherit from one class, but has many interfaces

“abstract” class is not instanciable; must be derived

“sealed” cannot be derived or abstracted

Must be an abstract or virtual function to be overriden

Base class behavior can be called with **base()**