

Feuille de TD N° 1

Introduction à C# et au Framework .NET

I. Répondre aux questions suivantes

1. Dans la génération des technologies Visual Studio 2017,
 - Le numéro de la version du .NET Framework est :
 - Le numéro de version du langage C# est :
2. Dans quel dossier du système de fichier le framework .NET est-il installé ?
3. Comparer les deux déclarations suivantes en C# : (i) `int i = 22` ; (ii) `var i = 22` ;
4. Comparer les deux déclarations suivantes en C# : (i) `var i = 33` ; (ii) `dynamic i = 33` ;
5. Quelle est la différence entre les « value types » et les « reference types » ? Citer trois exemples de value types. Citer trois exemples de reference types.
6. Comment peut-on éviter le problème connu sous le nom de l'enfer des DLL (DLL hell) en .NET ?
7. Qu'est-ce que le GAC (Global Assembly Cache) ?
8. Dans quel dossier du système de fichier le framework le GAC est-il localisé ?
9. Quelle est la classe du .NET Framework dont toutes les classes héritent ?
10. Quels sont les mécanismes syntaxiques à mettre en place afin de redéfinir une méthode héritée ?
11. Quel est le mécanisme syntaxique qui permet d'interdire qu'une classe soit héritée ?
12. Quelle est la différence entre la classe `String` et la classe `StringBuilder` ?
13. Quels sont les namespaces du .NET Framework qui sont réservés aux Collections. Citez deux exemples de collections linéaires. Citez deux exemples de collections indexées.
14. Comment ça se fait qu'il soit possible que le littéral `string` (s en lowercase) soit interchangeable avec `String` ?
15. Comment déclarer une chaîne de caractères en C# contenant des anti-slash (\).
16. Transformer l'expression Lambda suivante en une expression LINQ : `abonnés.Where(a=>a.Numero.StartsWith("98"))`.
17. Qu'est-ce qu'un delegate en C# ?
18. Quel est l'artifice syntaxique qui permet de souscrire un delegate à une classe ?
19. Qu'est-ce qu'un event en C# ?
20. Quel est l'artifice syntaxique qui permet de souscrire un event à une classe ?
21. Quelle est la différence entre un delegate et un event ?
22. Donner l'effet des touches suivantes dans Visual Studio :

F4 :	F6 :
F5:	CTRL F5 :
F9:	F12 :
CTRL K C:	CTRL K D :

II. Expliquer les concepts suivants du .NET framework et les illustrer si possible à travers des exemples simples développés en C#

- Le langage LINQ
- Structures des assemblies .NET : méta-données des assemblies, Manifestes , Strong Names
 - a. Private Assembly and Shared Assembly
 - b. Global Assembly Cache (GAC)
- .Net Threads and Tasks
- Reflection API
- .NET Application Configuration : `machine.config` et `web.config/app.config`

III Feuille de TD N° 1/ Compléments

Questions de cours – .NET Framework et C# (DS GL3 – Avril 2014)

1. Quelle est la différence entre **un namespace** et **un assembly** en .NET Framework ?
Dans quel(s) assembly(ies) la Base Class Library du .Net Framework est elle packagée ?
2. Quelle est la différence entre les classes **String** et **StringBuilder** ?
Comment ça se fait qu'il soit possible que le littéral **string** (s en lowercase) soit interchangeable avec **String** ?...
Quelle instruction ou configuration doit on faire pour qu'il soit possible que littéral **thread** (t en lowercase) soit interchangeable avec **Thread** ?.....
3. Donner à chaque fois l'année d'apparition des facilités syntaxiques C# suivantes

Generics :

LINQ :

Co and contra variance :

Asyn features :

Extension Methods :

Nullable types :

Donner l'effet des touches suivantes dans Visual Studio :

F4 :

F5 :

F6 :

F12 :

4. Terminer le code ci-dessous de manière à ce que le résultat sera celui présenté dans la colonne de droite.

```
public void LinqQuestion4()
{
    int[] numbers = { 0, 4, 1, 3, 9 };
    var numsInPlace = .....
    .....
    Console.WriteLine("Number: In-place?");
    foreach (var n in numsInPlace)
    { Console.WriteLine("{0}: {1}", n.Num, n.InPlace); }
}
```

```
Number: In-place?
0: True
4: False
1: False
3: True
9: False
```

5. (2 points) Terminer le code ci-dessous de manière à ce que le résultat sera celui présenté dans la colonne de droite.

```
public void LinqQuestion5()
{
    int[] numbersA = { 0, 2, 8, 9 };
    int[] numbersB = { 1, 3, 5 };
    var pairs = .....
    .....
    Console.WriteLine("Pairs where a < b:");
    foreach (var pair in pairs)
    { Console.WriteLine("{0} is less than {1}", pair.a, pair.b); }
}
```

```
Pairs where a < b:
0 is less than 1
0 is less than 3
0 is less than 5
2 is less than 3
2 is less than 5
```

6. Mettre dans la colonne de droite le résultat d'exécution du Main() de la colonne de gauche

```
static List<int> MyList = new List<int>{1,2,3,4,5};
public static void Main(string[] args)
{
    foreach (int i in RunningTotal())
    { Console.WriteLine(i); }
}
public static IEnumerable<int> RunningTotal()
{
    int runningtotal=0;
    foreach(int i in MyList)
    {
        runningtotal += i;
        yield return (runningtotal);
    }
}
```

```
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
```

II. Questions de cours – .NET Framework et C# (DS GL3 – Avril 2015)

Question

Enumérez trois utilisations du mot clé **using** en C#?

1.
2.
3.

Question

Un shared .NET Assembly doit obligatoirement posséder les caractéristiques suivantes :

-
-
-

Question 4

You are developing an application by using C#. The application includes the following code segment. (Line numbers are included for reference only.)

```
01 public interface IDataContainer
02 {
03     string Data { get; set; }
04 }
05 void DoWork(object obj)
06 {
07     // This is Line 07
08     if (dataContainer != null)
09     {
10         Console.WriteLine(dataContainer.Data);
11     }
12 }
```

The `DoWork()` method must throw an `InvalidCastException` exception if the `obj` object is not of type `IDataContainer` when accessing the `Data` property. You need to meet the requirements. Which code segment should you insert at **line 07**?

1. `var dataContainer = (IDataContainer)obj;`
2. `dataContainer = obj as IDataContainer;`
3. `var dataContainer = obj is IDataContainer;`
4. `dynamic dataContainer = obj;`

Question

You use the `Task.Run()` method to launch a long-running data processing operation. The data processing operation often fails in times of heavy network congestion. If the data processing operation fails, a second operation must clean up any results of the first operation. You need to ensure that the second operation is invoked only if the data processing operation throws an unhandled exception. What should you do?

1. Create a task by calling the `Task.ContinueWith()` method
2. Use the `TaskScheduler` class to create a task and call the `TryExecuteTask()` method on the class.
3. Create a `TaskFactory` object and call the `ContinueWhenAll()` method of the object.
4. Create a task within the operation, and set the `Task.StartOnError` property to true.

Question 6

You are creating a console application by using C#. You need to access the application assembly. Which code segment should you use?

1. `Assembly.GetAssembly(this);`
2. `This.GetType();`
3. `Assembly.Load();`
4. `Assembly.GetExecutingAssembly();`

Question 7

You are developing an application that uses structured exception handling. The application includes a class named `ExceptionLogger`. The `ExceptionLogger` class implements a method named `LogException` by using the following code segment: `public static void LogException(Exception ex)`

You have the following requirements:

- log all exceptions by using the `LogException()` method of the `ExceptionLogger` class.
- rethrow the original exception, including the entire exception stack.

You need to meet the requirements. Which code segment should you use?

1. `catch (Exception ex){ ExceptionLogger.LogException(ex); throw; }`
2. `catch (Exception ex){ ExceptionLogger.LogException(ex); throw ex; }`
3. `catch { ExceptionLogger.LogException(new Exception()); throw; }`
4. `catch { var ex = new Exception(); throw ex; }`

Question 8

Proposer une implémentation C# du design pattern Singleton. Un singleton est une classe qui n'a qu'une seule instance et qui fournit un point d'accès global à cette instance. Il faut prendre garde d'assurer dans votre implémentation la propriété de « Thread safety » de manière garantir la cohérence de l'instance unique lors d'un accès concurrent par plusieurs threads.

III. .NET Framework et C# (Examen GL3 – Juin 2015)**Exercice 1.**

- Clarifier les différences entre les deux mots clés **is** et **as** en C#.
- Clarifier les différences entre les concepts de **hiding** et d'**overriding** en .NET.
- Clarifier les différences entre les concepts de **hiding** et d'**overloading** en .NET.

Exercice 2.

On dispose d'une liste d'objets en mémoire représentant des commandes passées par un client. Chaque commande est composée d'une liste d'articles où l'on indique pour chaque article la date de la commande et le montant total d'achat.

```
commande1
  article1 -> 18/05/2015, 100.570
  article2 -> 18/05/2015, 200.430
commande2
  article3 -> 19/05/2015, 300.000
```

(i) Ecrire une requête LINQ qui retourne l'article ayant le montant le plus élevé. Pour l'exemple ci-dessus, on afficherait article3.

(ii) Ecrire une seule requête LINQ qui retourne la liste des dates distinctes et la somme des achats pour cette date tous articles confondus. Un exemple de résultat est illustré ci après.

```
18/05/2015  301.000
19/05/2015  300.000
```

Exercice 3.

Transformer la méthode C# suivante afin de la rendre asynchrone

```
public bool Create (Customer obj)
{ try
  { using (CustomersContext context =
        new CustomersContext ())
    { context.Customers.Add(obj);
      context.SaveChanges ();
      return true;
    }
  }
  catch (Exception ex)
  { Console.WriteLine(ex.Message);
  }
  return false;
}
```

```
.....
.....
.....
.....
.....
.....
.....
.....
.....
```