



はじめの一步 | 基本編1日目

名前空間(namespace)

- **Sample101**参照
- プログラムを分類する「入れ物」の名前を決める
- 同一ネームスペース内に作成されるプログラムは互いに関係性が深い

名前空間の使用

```
namespace Sample101
```

usingステートメント

- **using**という記述は、**usingステートメント**と呼ばれる
- 利用する名前空間を指定
- 指定した名前空間のクラスを利用可能になる
- プログラムの先頭で指定する

usingステートメントの使用

```
using System;
```

クラス名

- **class**の後に記述されているものを**クラス名**と呼ぶ
- クラスとはプログラムのまとまった処理の単位のこと
- 同一名前空間内には同じクラスが存在してはいけない
- 名前空間が異なれば同じクラス名が存在してもかまわない

クラス名の指定

```
class (クラス名)
```

エントリーポイント

- **static void Main(string[] args)**は**エントリーポイント**と呼ばれる
- C#言語は、原則的にこの中に処理を書くことにより、プログラムが実行される
- 処理の中身は、“{”および“}”で囲まれている

Console.WriteLine

- ()内のデータを表示し、**改行**
- 文字列の場合は、"""（ダブルクォーテーション）で挟む
- 数値の場合はそのまま値を入力すればよい

Console.WriteLineの使用例

```
Console.WriteLine(123);
```

Console.Write

- ()内のデータを表示し、**改行しない**
- 文字列の場合は、"""（ダブルクォーテーション）で挟む
- 数値の場合はそのまま値を入力すればよい

Console.WriteLineの使用例

```
Console.Write ("HelloWorld.");  
Console.Write (123);
```

コメント

- プログラム中の//や、/* */という記号は**コメント**という
- プログラムの**注釈**を付けるためのもの
- 処理の意味などを記述する時に用いられる
- コメントを付けるとプログラムが非常にわかりやすくなる
- 実行結果には何ら影響を与えない

C#で用いられるコメント

記述方法	名前	特徴	使用例
<code>/* */</code>	ブロックコメント	<code>/*</code> と、 <code>*/</code> の間に囲まれた部分がコメントになる。 複数行にわたってコメントをつけることができる	<code>/*</code> このように複数行 コメント可能です。 <code>*/</code>
<code>//</code>	行コメント	一行にコメントをつけることができる	<code>//</code> コメント

セミコロン

- 処理の終わりには必ず;(セミコロン) を付ける
- セミコロンを付けることにより処理の終わりを意味する

Console.WriteLine("HelloWorld.");

処理の終わりには必ずセミコロンつける