

継承と初期化

コンストラクタ初期化子

派生クラスは、どのように基底クラスのインスタンスを生成しているのでしょうか？
少なくとも、前回の内容からも基底クラスのインスタンス化は暗黙的に行われているはずだ

実は、基底クラスのインスタンスは、派生クラスのインスタンスが生成される前に作られます
派生クラスのコンストラクタが呼び出されると、コンストラクタはステートメントを実行する前に
まず基底クラスのコンストラクタを呼び出しインスタンス化します
このとき呼び出すコンストラクタは、常にデフォルトコンストラクタです

```
class Kitty {
    public Kitty() {
        System.Console.WriteLine("Kitty on your lap");
    }
}

class TokyoMM : Kitty {
    public TokyoMM() {
        System.Console.WriteLine("Tokyo mew mew");
    }
    static void Main() {
        TokyoMM obj = new TokyoMM();
    }
}
```

TokyoMM は Kitty クラスの派生クラスです
TokyoMM() コンストラクタを実行してインスタンス化しようとする
先に Kitty() コンストラクタを実行して、その後に TokyoMM() コンストラクタが実行されます

この関係は、クラスが何段階に継承されても同じで
常にもっとも上位の基底クラスのコンストラクタから実行され派生順に実行されます

しかし、暗黙のうちに実行されるコンストラクタは常にデフォルトコンストラクタです
派生クラスから明示的にコンストラクタに引数を渡したい場合、これでは実現できません
そこで **コンストラクタ初期化子** を使用します

constructor() : base(...) {}

constructor() は派生クラスのコンストラクタ宣言子です
base() は基底クラスの呼び出すコンストラクタを表しています
これによって、特定のシグネチャの基底コンストラクタを実行することができます

```
class Kitty {
    public Kitty() {
        System.Console.WriteLine("Kitty on your lap");
    }
    public Kitty(string str) {
        System.Console.WriteLine(str);
    }
}

class TokyoMM : Kitty {
    public TokyoMM(string str) : base(str) {
        System.Console.WriteLine("Tokyo mew mew");
    }
    static void Main() {
        TokyoMM obj = new TokyoMM("Di Gi Charat");
    }
}
```

TokyoMM() コンストラクタは base() 初期化子を使って
コンストラクタ実行前に Kitty(string) を明示的に呼び出します
このように、コンストラクタ初期化子にはコンストラクタの仮引数値をそのまま渡せます

この結果、コンソールは文字列 Di Gi Charat と Tokyo mew mew を出力するでしょう
このことから、Kitty(string) が実行されたことを証明できます

また、コンストラクタ初期化子は this() というものもあります
これは、クラス内の別のコンストラクタを明示的に呼び出します

constructor() : this(...) {}

this で指定した引数の型に合うシグネチャのコンストラクタが呼び出されます
全てのコンストラクタに共通の処理を行いたい場合に非常に便利です

```
class Kitty {
    public Kitty() : this("Kitty on your lap") {
        System.Console.WriteLine("End of default constructor");
    }
    public Kitty(string str) {
        System.Console.WriteLine(str);
    }
}

class Test {
    static void Main() {

```

```
Kitty obj = new Kitty();  
}  
}
```

Kitty クラスのデフォルトコンストラクタ Kitty() を呼び出すと
このコンストラクタは Kitty(string) にデフォルトの文字列として "Kitty on your lap" を渡します
Kitty(string) 型のコンストラクタの処理が終了すると、制御が呼び出し元に復帰し
さらに Kitty() のステートメントが実行されます。次のような結果になりました

Kitty on your lap
End of default constructor

最初にデフォルトコンストラクタが呼び出され、"Kitty on your lap" という文字列で
Kitty(string) コンストラクタを呼び出していることがわかりますね

[前のページへ](#)

[戻る](#)

[次のページへ](#)