**شرح البوليمورفيزم (Polymorphism) في C#**

البوليمورفيزم (التعددية الشكلية) هو واحد من أهم مبادئ **البرمجة الكائنية التوجه (OOP)**، وهو يسمح باستخدام **نفس الدالة** بطرق مختلفة حسب السياق، مما يجعل الكود أكثر مرونة وقابلية لإعادة الاستخدام.

**أنواع البوليمورفيزم في C#**

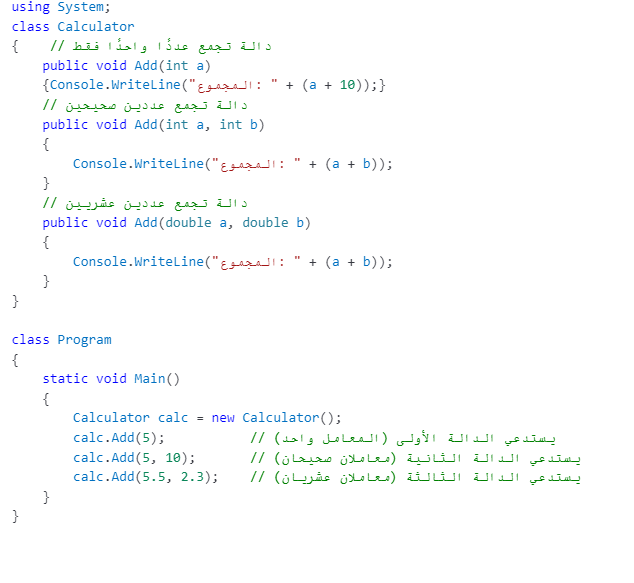
هناك نوعان أساسيان من **Polymorphism** في C#:

1. **التعددية الشكلية في وقت الترجمة (Compile-time Polymorphism)**
   * تُعرف أيضًا باسم **تحميل الدوال (Method Overloading)**.
   * تسمح بتعريف أكثر من دالة بنفس الاسم ولكن مع اختلاف في **عدد أو نوع المعاملات**.
2. **التعددية الشكلية في وقت التنفيذ (Run-time Polymorphism)**
   * تُعرف أيضًا باسم **تجاوز الدوال (Method Overriding)**.
   * تحدث عندما تُعيد **الوراثة (Inheritance)** تعريف دالة موجودة في **الكلاس الأب** داخل **الكلاس الابن**.

**1. التعددية الشكلية في وقت الترجمة (Method Overloading)**

يتم تنفيذها عن طريق **تحميل الدوال**، أي تعريف أكثر من دالة بنفس الاسم ولكن مع اختلاف في **عدد** أو **نوع** المعاملات

مثال على Method Overloading.



A white background with a black border

AI-generated content may be incorrect.

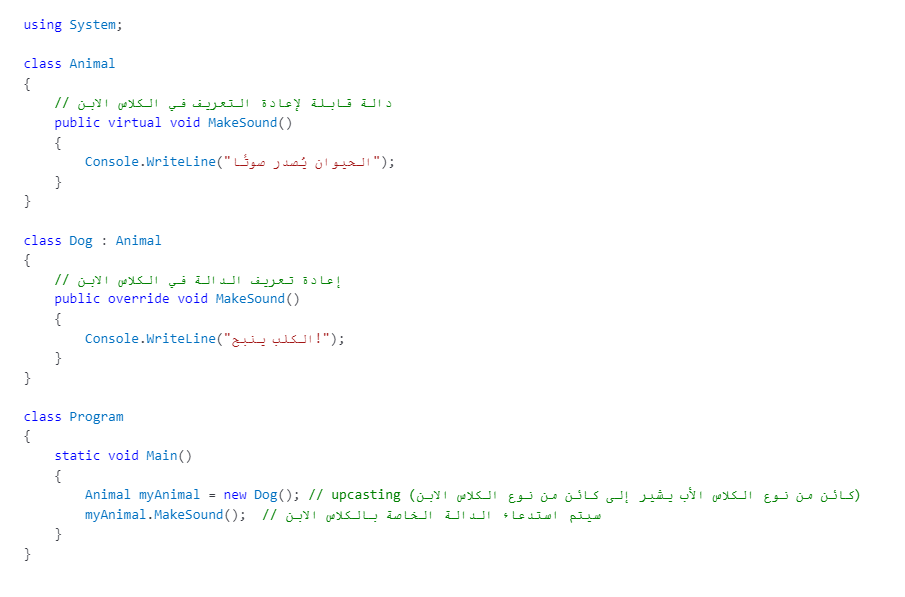
**ملحوظات**:

* كل الدوال لها نفس الاسم (Add).
* يتم استدعاء الدالة المناسبة بناءً على **عدد ونوع المعاملات**.

**2. التعددية الشكلية في وقت التنفيذ (Method Overriding)**

يتم تنفيذها باستخدام **الوراثة (Inheritance)** حيث يمكن **للكلاس الابن** إعادة تعريف دالة موجودة في **الكلاس الأب**.

**مثال على Method Overriding**

* **الكلاس Animal يحتوي على دالة MakeSound() وتم تمييزها بـ virtual حتى تكون قابلة لإعادة التعريف.**
* **الكلاس Dog أعاد تعريف نفس الدالة MakeSound() باستخدام override.**
* **عند استدعاء myAnimal.MakeSound(), يتم تنفيذ الدالة الخاصة بالكلاس الابن Dog بدلاً من Animal.**

**3. التعددية الشكلية باستخدام الواجهات (Interfaces)**

تسمح **الواجهات (Interfaces)** بتنفيذ **polymorphism** حيث يمكن لكلاسات مختلفة تنفيذ نفس الدالة بشكل مختلف.

**مثال على Polymorphism باستخدام الواجهات**

