

C# Professional

Версионность.

C# Professional

После урока обязательно



Повторите этот урок в видео формате на
[ITVDN.com](http://itvdn.com)



Проверьте как Вы усвоили данный материал на
[TestProvider.com](http://testprovider.com)

Версионность

C# Professional

Nonvirtual interface

Основная задача этого паттерна — разделение представления интерфейса и его реализации. Виртуальные функции объявляются как **protected**, а их вызов происходит внутри обычных функций, которые и предоставляются пользователю.

Во-первых, внутри не виртуальной функции вы можете осуществлять различные проверки, как до вызова виртуальной функции, так и после него.

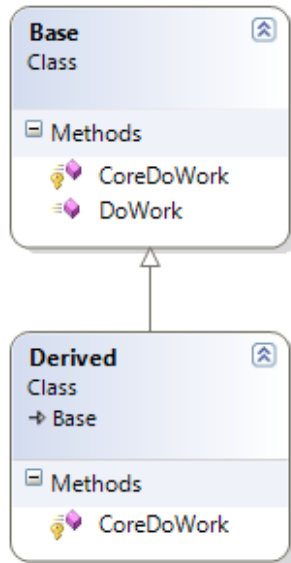
Во-вторых, интерфейсы виртуальной и не виртуальной функции не обязаны совпадать.

Открытая функция может иметь интерфейс, который будет наиболее удобен пользователю, а виртуальная — который будет наиболее удобен для замещения функциональности. Более того, набор и интерфейс виртуальных функций может быть совершенно иным, нежели набор и интерфейс открытых функций. Никто не мешает внутри открытой функции вызвать пять виртуальных.

Паттерн позволяет четко разделить обязанности — в этом его достоинство и основное назначение. Открытые функции предоставляют интерфейс, который в первую очередь удобен пользователю. Виртуальные функции предоставляют интерфейс, который в первую очередь удобен для замещения функциональности.

C# Professional

Не виртуальный интерфейс



Base – базовый класс.

DoWork() – метод, представляющий собой общий интерфейс (не может быть переопределен).

CoreDoWork() – метод, представляющий собой переопределяемое поведение.

Derived – производный класс.

C# Professional

new методы

Явно скрывает члены, унаследованные от базового класса.

При сокрытии унаследованного члена его производная версия заменяет версию базового класса.

new документирует тот факт, что производная версия предназначена для замены.

Q&A

Информационный видеосервис для разработчиков программного обеспечения

