

## به نام خدا

برنامه نویسی شی گرا

مرتضی صداقت یار

استاد: میثاق یاریان

شماره دانشجویی: ۰۱۲۰۰۳۳۷۲۰۰۲۳

### چند نمونه از عملیات Extension Method

Extension Method ها به شما این امکان را می‌دهند که بدون نیاز به تغییر کد کلاس‌های موجود، متدهای جدیدی به آنها اضافه کنید. این کار برای افزایش قابلیت‌ها و انعطاف‌پذیری کد بسیار مفید است.

#### ۱. اضافه کردن متدهای جدید به کلاس‌های پایه:

```
public static class StringExtensions
{
    public static string Reverse(this string str)
    {
        return new string(str.Reverse().ToArray());
    }
}

string text = "Hello, World!";
string reversedText = text.Reverse();
Console.WriteLine(reversedText); // Output: !dlroW ,olleH
```

#### ۲. گروه بندی متدهای مرتبط:

```
public static class StringExtensions
{
    public static string Reverse(this string str)
    {
        // ...
    }

    public static bool IsPalindrome(this string str)
    {
        // ...
    }

    public static int CountWords(this string str)
    {
        // ...
    }
}
```

### ۳. ساده‌سازی کد:

یک متد برای بررسی اینکه آیا یک عدد زوج است یا فرد

```
public static class IntExtensions
{
    public static bool IsEven(this int number)
    {
        return number % 2 == 0;
    }
}

int num = 10;
if (num.IsEven())
{
    Console.WriteLine("The number is even.");
}
else
{
    Console.WriteLine("The number is odd.");
}
```

### کار با کنترلرهای ASP.NET Core:

می‌توان از Extension Method ها برای اضافه کردن رفتار جدید به کنترلرهای ASP.NET Core خود استفاده کرد.

```
public static class HomeControllerExtensions
{
    public static string GetCurrentUserNames(this HomeController controller)
    {
        if (controller.User.Identity.IsAuthenticated)
        {
            return controller.User.Identity.Name;
        }
        else
        {
            return "Anonymous";
        }
    }
}

public class HomeController : Controller
{
    public IActionResult Index()
    {
        string userName = this.GetCurrentUserNames();
        ViewBag.UserName = userName;
        return View();
    }
}
```