بسمه تعالی

آموزش جامع C#

به صورت ساده و خلاصه

تهیه‌شده توسط:

حسین محمودی (iHMahmoodi)

ایمیل :

iHMahmoodi@gmail.com

سلام

حسین محمودی هستم و قصد دارم توی این PDF که در حال نوشتنش هستم **C#** نه به صورت کتابی و خشک بلکه همونطور که خودم درک کردم و فهمیدم با زبانی ساده و انشالله قابل‌فهم برای شما خلاصه‌شده توضیح بدم که یکجورایی باعث مشکلاتی که در درک بعضی مفاهیم برنامه‌نویسی من باهاش برخورد کردم و شما راحت تر درک کنید.

به هر حال درک اصول برنامه‌نویسی، یادگرفتن خود زبان برنامه‌نویسی رو خیلی راحت می کنه !! (جمله‌ای از خودم D:)

به هر حال امیدوارم این آموزش مفید واقع بشه و اینکه فقط یک صلوات برای ظهور آقا امام زمان و سلامتی ایشون و همچنین خانواده‌ی عزیزم بفرستید + کپی‌رایت هم رعایت کنید (دستتون درد نکنه D:)

دیگه چیزی نمونده بگم !

اهان یک چیز دیگه اونم اینه که انشالله اگه وقت بشه هر دفعه یک تیکش رو می‌نویسم و می‌زارم(چون کار رو زندگی و درس و فلان و بیسان و ... داریم D:)

بریم سراغ آموزش!

اول بیاید با تعریف **Framework** شروع کنیم

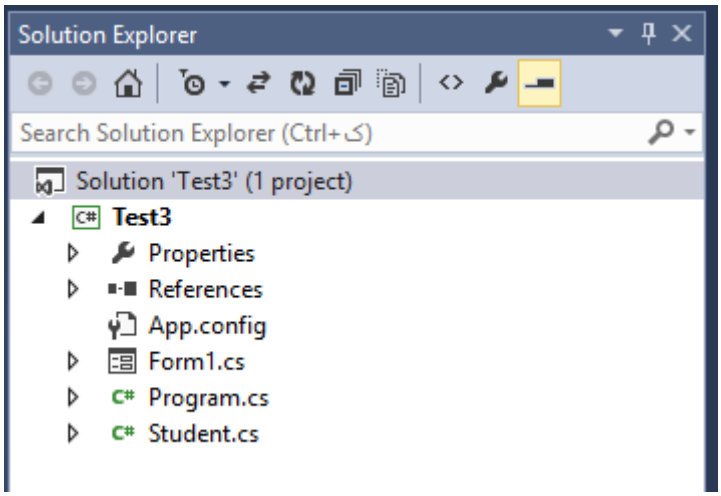
در واقع فریم ورک ، نیازمندی‌های برنامه برای اجرای برنامه (مثل Microsoft Visual C++ 2005) توسط یک بسته مثل Net Framework. بر روی ویندوز نصب می‌شود تا برنامه اجرا بشود که نتیجه‌اش چیه ؟! خوب معلومه حجم برنامه کمتر میشه و مشکلات توسط شرکت فریم ورک رفع میشه !

پس به همین خاطر هست که بعضی برنامه‌ها به Net framework 3.5. احتیاج دارن (مثل فتوشاپ ، 3d max، اتوکد)

مفهوم شی گرایی در C# :

اینجا فعلاً یک تعریف کوچیک از شی گرایی می‌کنیم و بعداً کاملش می‌کنیم

همه چیز در سی شارپ بر پایه‌ی شی گرایی هست و میشه گفت هر چیزی در Solution Explorer باشد یک شی حساب میشه !



آناتومی یک برنامه در سی شارپ :

به طور خلاصه میخواد بگه که یک برنامه که میخواید در سی شارپ بسازید شامل قسمت های ثابتی میشه

مثل یک بلاک Main وجود داره،دستورات using برای صدا زدن قسمت کتابخانه ای، name space ها و...

خلاصه نگران نباشید چیز مهمی نیس :D

```
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Test3
{
    0 references
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        0 references
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
```

منطق‌های اولیه در سی شارپ :

خوب در مورد بعضی از قوانین و اسم‌های که می‌شنوید هم اینجا یکم صحبت کنیم

- تمامی دستورات (Command) در سی شارپ با **;** تموم میشه به غیر از بلاکی‌ها (دستوراتی که بین {} قرار میگیرن)
- سی شارپ Case Sensitive هست یعنی به حروف کوچک و بزرگ حساسه در نتیجه a و A با هم فرق دارن!!
- IDE چیه ؟ یعنی محیط یکپارچه ویژوال استدیو که شامل محیط کد،طراحی،کامپایلر و ... میشه!

متغیر و انواع مختلف داده در C# :

متغیر یا variable یعنی بخضی از حافظه برای ذخیره محاسبات و مقادیر دریافتی از کابر و ...

حالا چطوری متغیر تعریف کنیم ؟

کاری نداره ، قالب زیر رو در نظر بگیرید

نام متغیر + **نوع متغیر** **int** a **, bool** answer

```
private void Form1_Load(object sender, EventArgs e)
{
    int a, b, c, d;
    bool answer = true;
    object AnyThingYouDontKnow;

    if (answer == false)
    {
        AnyThingYouDontKnow = "Test";
        a = 1; b = 2;
        c = a + b;
        d = b + c;

    }
```

چون من دارم خلاصه میگم توی گوگل یک جست‌وجو بزنید در مورد " انواع مختلف متغیر و داده در سی شارپ " و میتونید انواع مختلفش رو با اندازه‌ای که پشتیبانی میکنن ببینید!

فقط یک نکته بگم که اگه در حال برنامه‌نویسی نمی دونستید از چه نوع متغیری استفاده کنید از نوع object استفاده کنید که انواع داده‌ها رو می پذیره !

همونطور که در عکس می‌بینید میشه چند متغیر از یک نوع رو فقط با ویرگول "،" ایجاد کرد (int a,b,c,d)

برای اضافه کردن کامنت یا توضیح یا مستندسازی یا به عبارت ساده‌تر واسه اینکه بعداً بفهمید یک تیکه از برنامه چیکار می کنه و بیاید توضیح براش بنویسید از "//" استفاده می‌کنید و #cدیگه اونو به عنوان کد نمیخونه و هیچی در نظرش نمی گیره!

مقدار پیش‌فرض بعضی از داده‌ها هم اینجا بگم دیگه تموم

bool = false

متغیرهای عددی = 0

Null= هیچی یا خالیه :D

مفهوم تابع (Function) :

به طور ساده یک بسته حاوی کد که میتونه ورودی داشته باشه یا نداشته باشه (یک کامپیوتر در نظر بگیر ،اگه شما ورودی (برنامه)بهش بدید میتونه خروجی بهتون بده (میتونه هم نده اگه زورش برسه) و گرنه کامپیوتر رو روشن کن بشینید پشتش ، خبری شد !؟)

فعلا در اینجا میگم که تابع باید حتما مقدار بازگشتی داشته باشه (return)

```
int MySum (int x , int y)
{
return x+y;
}
```

خوب توی پرانتز کد بالا بهش میگن ورودی یعنی همون x ،y

مقدار بازگشتی هم که همون x+y هست!

فعلاً تا همین جا داشته باشید!

اگر خدا بخواهد ادامه دارد.

دوستان عزیز نظری هم داشتید میتونید ایمیل بزنید!

iHMahmoodi@gmail.com

iHMahmoodi.blogfa.com

iHMahmoodi.blog.ir

با احترام

حسین محمودی