

# به نام خدا

عرفان مالکی

درس سمت سرور

استاد میثاق یاریان

دانشکده شمسی پور

تمرین 2 :

تفاوت *bool float string int* و ... در حافظه

انواع حافظه در رم (*stack*) و (*heap*)

کدام متغیر ها *reference type* هستند و کدام *value type*

# انواع داده ها

## 1- داده عددی یا Integer

اولین نوع داده ، داده عدد صحیح که به انگلیسی Integer نامیده میشه، می تونن اعداد صحیح رو در خودشون ذخیره کنن  
این مدل داده معمولا در زبان های برنامه نویسی با کلمه int معرفی میشود.

## 2 - داده کاراکتر یا character

همونطور که از اسمش مشخصه می تونه کاراکتر های مختلف رو در خودش ذخیره کنه.  
این نوع داده معمولا در زبان های برنامه نویسی با کلمه ی char مشخص میشه .

## 3 - داده اعشاری یا float

این مدل داده می تونه اعداد اعشاری ممیز شناور رو در خودش ذخیره کنه.

## 4 - داده Double

اعداد اعشاری رو در خودش ذخیره می کنه اما با این تفاوت که می تونه اعداد با اعشار زیاد رو ذخیره کنه

## 5 - داده منطقی یا Boolean

معمولا با کلمه ی bool معرفی میشه. این نوع داده تنها می تونه دو مقدار true و false رو به خودش بگیره. از این نوع داده می تونیم در شرط ها استفاده کنیم.

## 6 - داده string یا رشته ای

این داده می تونه رشته ای از کاراکتر ها رو در خودش ذخیره کنه . برای مثال این مدل داده می تونه رشته های "rasoli" یا "sara" یا "12345" رو در خودش ذخیره کنه.

## حافظه Stack memory و Heap memory

حافظه در رم به دو دسته اصلی تقسیم می‌شود:

- حافظه استک (Stack memory): این نوع حافظه برای ذخیره داده‌های محلی (local variables) و پارامترهای توابع (function parameters) استفاده می‌شود. حافظه استک به صورت LIFO (Last In First Out) مدیریت می‌شود، یعنی آخرین داده‌ای که در حافظه استک قرار می‌گیرد، اولین داده‌ای است که از حافظه استک خارج می‌شود.
- حافظه هیپ (Heap memory): این نوع حافظه برای ذخیره داده‌هایی استفاده می‌شود که اندازه و عمر آنها مشخص نیست. حافظه هیپ به صورت خودکار توسط سیستم عامل مدیریت می‌شود.

### حافظه استک

حافظه استک در قسمت بالای حافظه رم قرار دارد و به صورت LIFO مدیریت می‌شود. یعنی آخرین داده‌ای که در حافظه استک قرار می‌گیرد، اولین داده‌ای است که از حافظه استک خارج می‌شود.

در زبان‌های برنامه‌نویسی، حافظه استک برای ذخیره داده‌های زیر استفاده می‌شود:

- متغیرهای محلی (local variables): متغیرهایی که در بدنه یک تابع یا روش تعریف می‌شوند، متغیرهای محلی نامیده می‌شوند. این متغیرها در حافظه استک ذخیره می‌شوند.
- پارامترهای توابع (function parameters): داده‌هایی که به یک تابع یا روش ارسال می‌شوند، پارامترهای آن تابع یا روش نامیده می‌شوند. این پارامترها در حافظه استک ذخیره می‌شوند.
- آدرس بازگشتی (return address): آدرس کدی که پس از بازگشت از یک تابع یا روش باید اجرا شود، آدرس بازگشتی نامیده می‌شود. این آدرس در حافظه استک ذخیره می‌شود.

## حافظه هیپ

حافظه هیپ در قسمت پایین حافظه رم قرار دارد و به صورت خودکار توسط سیستم عامل مدیریت می‌شود.

در زبان‌های برنامه‌نویسی، حافظه هیپ برای ذخیره داده‌های زیر استفاده می‌شود:

- اشیا (objects): در زبان‌های برنامه‌نویسی شی‌گرا، اشیا در حافظه هیپ ذخیره می‌شوند.
- آرایه‌های بزرگ (large arrays): آرایه‌هایی که اندازه آنها بیش از اندازه حافظه استک است، در حافظه هیپ ذخیره می‌شوند.
- داده‌هایی که اندازه و عمر آنها مشخص نیست: داده‌هایی که اندازه و عمر آنها مشخص نیست، در حافظه هیپ ذخیره می‌شوند.

تفاوت‌های حافظه استک و هیپ

جدول زیر تفاوت‌های حافظه استک و هیپ را خلاصه می‌کند:

ویژگی	حافظه استک	حافظه هیپ
محل قرارگیری	بالای حافظه رم	پایین حافظه رم
نحوه مدیریت	توسط CPU	توسط سیستم عامل
نحوه دسترسی	LIFO	FIFO
کاربرد	ذخیره داده‌های محلی، پارامترهای توابع، آدرس بازگشتی	ذخیره اشیا، آرایه‌های بزرگ، داده‌هایی با اندازه و عمر مشخص نیست

## نتیجه‌گیری

حافظه استک و هیپ دو نوع حافظه اصلی در رم هستند که برای ذخیره داده‌های مختلف استفاده می‌شوند. حافظه استک برای ذخیره داده‌های محلی، پارامترهای توابع و آدرس بازگشتی استفاده می‌شود. حافظه هیپ برای ذخیره اشیاء، آرایه‌های بزرگ و داده‌هایی با اندازه و عمر مشخص نیست استفاده می‌شود.

## Reference type Value type

در C#، متغیرها به دو نوع اصلی تقسیم می‌شوند: رفرنس تایپ و ولیو تایپ.

- رفرنس تایپ ها به متغیرهایی گفته می‌شود که اشاره گر به یک شیء را ذخیره می‌کنند. هنگامی که یک متغیر رفرنس تایپ را به یک شیء اختصاص می‌دهید، هر دو متغیر به همان شیء اشاره می‌کنند.

- ولیو تایپ ها به متغیرهایی گفته می‌شود که مقدار خود را ذخیره می‌کنند. هنگامی که یک متغیر ولیو تایپ را به یک مقدار اختصاص می‌دهید، هر دو متغیر همان مقدار را ذخیره می‌کنند.

در اینجا چند مثال از متغیرهای رفرنس تایپ و ولیو تایپ در C# آورده شده است:

```
// رفرنس تایپ
int[] a = new int[] { 1, 2, 3 };
int[] b = a;
a.Add(4);

Console.WriteLine(a); // [1, 2, 3, 4]
Console.WriteLine(b); // [1, 2, 3, 4]

// ولیو تایپ
int c = 10;
int d = c;
c++;

Console.WriteLine(c); // 11
Console.WriteLine(d); // 10

Console.WriteLine(a); // [1, 2, 3, 4]
Console.WriteLine(c); // 11
```

در اینجا یک جدول خلاصه از تفاوت های بین رفرنس تایپ ها و ولیو تایپ ها آورده شده است:

ویژگی	رفرنس تایپ	ولیو تایپ
ذخیره مقدار	اشاره گر به یک شیء	مقدار خود
تغییر	تغییر در یک متغیر رفرنس تایپ باعث تغییر در متغیرهای دیگر نیز می شود	تغییر در یک متغیر ولیو تایپ فقط باعث تغییر در خود متغیر می شود
کپی	کپی کردن یک متغیر رفرنس تایپ فقط یک کپی از اشاره گر را ایجاد می کند	کپی کردن یک متغیر ولیو تایپ یک کپی کامل از مقدار را ایجاد می کند

انواع رفرنس تایپ شامل آرایه ها، کلاس ها، توابع و استثناها هستند.

آرایه ها

آرایه ها در **#C** رفرنس تایپ هستند. هنگامی که یک آرایه را به یک متغیر اختصاص می دهید، متغیر اشاره گر به آرایه را ذخیره می کند.

کلاس ها

کلاس ها در **#C** رفرنس تایپ هستند. هنگامی که یک شیء از یک کلاس را به یک متغیر اختصاص می دهید، متغیر اشاره گر به شیء را ذخیره می کند.

توابع

توابع در **#C** رفرنس تایپ هستند. هنگامی که یک تابع را به یک متغیر اختصاص می دهید، متغیر اشاره گر به تابع را ذخیره می کند.

استثناها

استثناها در **#C** رفرنس تایپ هستند. هنگامی که یک استثناء را به یک متغیر اختصاص می دهید، متغیر اشاره گر به استثناء را ذخیره می کند.

انواع و لیو تایپ در #C عبارتند از:

- اعداد صحیح (byte ،short ،long ،int)
- اعداد اعشاری (double ،float)
- رشته ها (string)
- بولی (bool)
- مقدار null (null)