



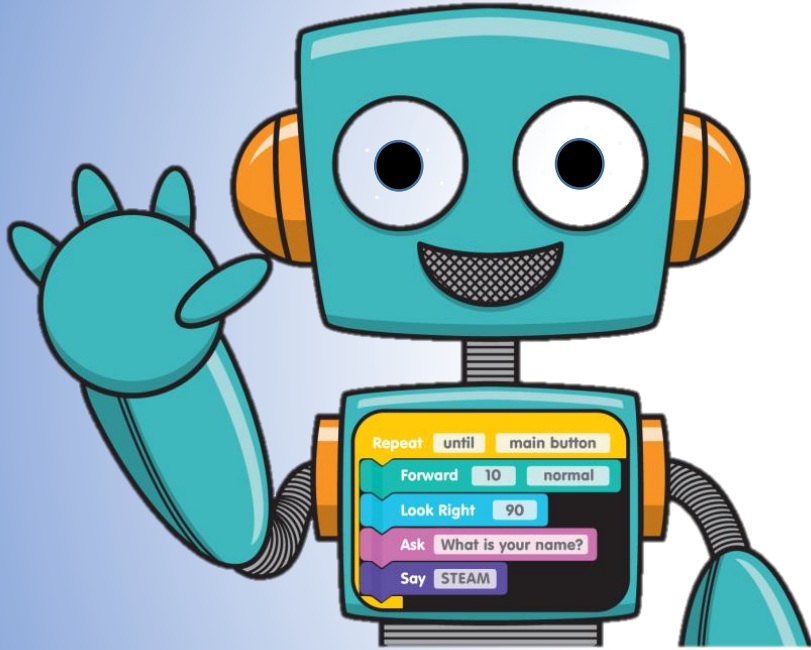
درس برنامه نویسی پیشرفته ۱

مبانی کامپیوتر و برنامه نویسی مقدماتی

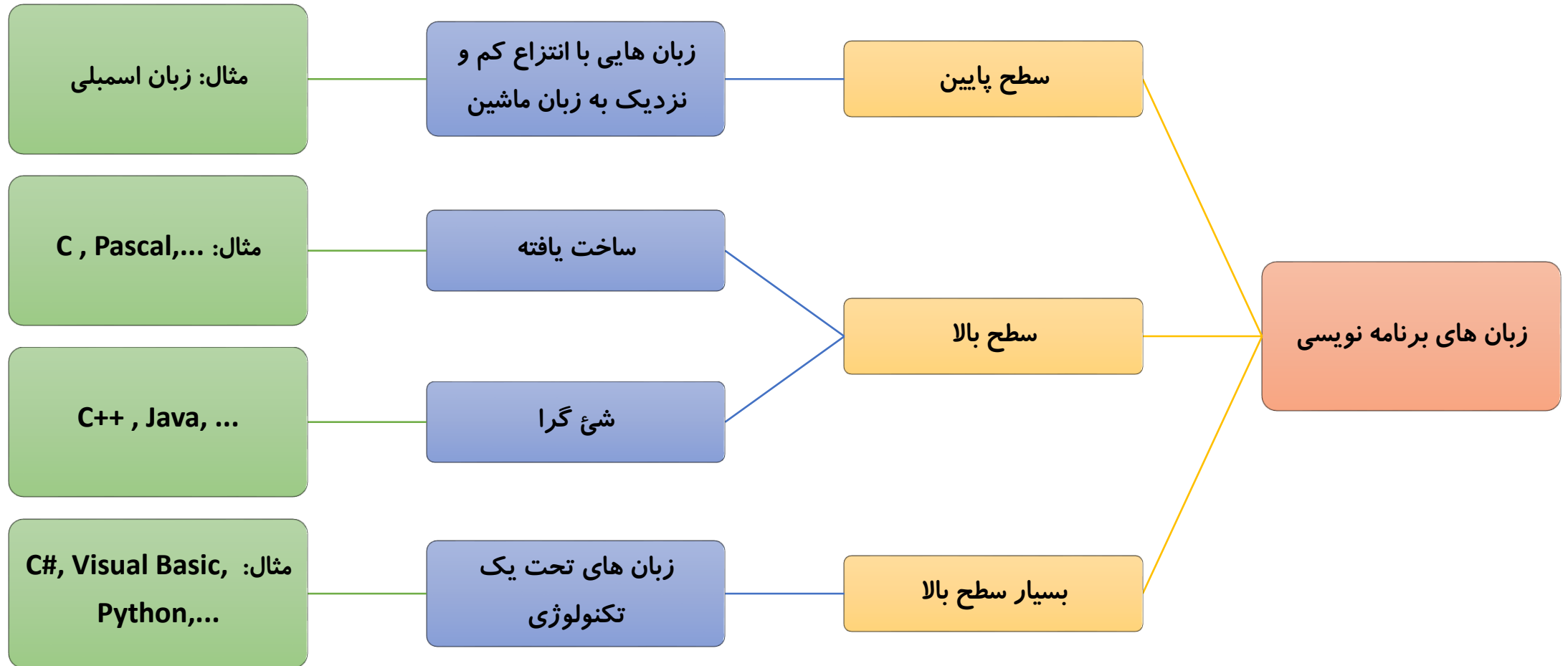
مدرس: مسعود بایمانی

فصل چہارم

برنامہ نویسی بہ زبان سی شارپ



دسته بندی زبان های برنامه نویسی



زبان های برنامه نویسی سطح پایین



- زبان های برنامه نویسی که به مقدار کم و یا هیچ انتزاعی نسبت به ریزپردازنده (Processor) فراهم نمی نمایند.
- به عبارت دیگر این زبان ها بسیار شبیه به زبان کار با سخت افزار بوده و برنامه نویسی تحت آنها بسیار دشوار است و نیاز به دانش فنی بالایی دارد.
- سطح پایین بودن این زبان های به معنی ارزش کم آنها نمی باشد؛ بلکه برعکس این زبان ها بسیار قدرتمند بوده و قابلیت پیاده سازی و اجرای هر دستور در سطح هسته سیستم عامل را دارند.
- برای مثال زبان اسمبلی یک زبان سطح پایین محسوب می شود. از این زبان بیشتر برای نوشتن درایورها و انجین ها استفاده می گردد. همچنین در برنامه نویسی برای شبکه نیز مورد استفاده قرار می گیرد.
- در دهه ۱۹۶۰ میلادی مشکلات زبان های برنامه نویسی سطح پایین نظیر تأخیر در زمان بندی تولید نرم افزار، هزینه بالای ایجاد نرم افزار، عدم قابل اعتماد بودن برنامه های تولیدی، عدم خوانایی سریع برنامه و تست و اشکال زدایی دشوار برنامه ها موجب توسعه و ایجاد زبان های برنامه نویسی سطح بالا شد.

زبان های برنامه نویسی سطح بالا



- زبان برنامه نویسی سطح بالا یا زبان برنامه نویسی بلندتر از (High-level programming language) برخلاف زبان برنامه نویسی سطح پایین، زبانی است که نسبت به زبان برنامه نویسی سطح پایین دارای انتزاع بیشتری هستند و برنامه نویسی با آن ها راحت تر است. همچنین این زبان ها قابل حمل بین سیستم های متفاوت می باشند.
- زبان های سطح بالا عموماً از عملیات CPU مانند مدل های دسترسی به حافظه و مدیریت حوزه، دور هستند.
- در حالی که زبان برنامه نویسی سطح بالا برنامه نویسی را آسان تر می نماید اما زبان برنامه نویسی سطح پایین برنامه های بهینه تری تولید می کند.
- در یک دسته بندی کلی زبان های برنامه نویسی سطح بالا به دو دسته ساخت یافته و شیء گرا تقسیم می گردند.

برنامه نویسی غیر ساخت یافته



- برنامه نویسی غیر ساخت یافته یک سبک برنامه نویسی رویه ای است که در آن دستورات به ترتیبی که توسط برنامه نویس نوشته شده اند، اجرا می شوند.
- در این سبک از برنامه نویسی، از دستور Goto به منظور انتقال کنترل اجرای برنامه از دستوری به دستور دیگر استفاده می گردد.

```
10 Input A
15 Input B
20 B = A + 10
30 IF B > 12 GOTO 60
40 C = B / 3
50 IF C < 24 GOTO 10
60 Write C
70 IF Write Failed GOTO 15
80 Input D
```

- به عبارت دیگر هنگامی که یک دستور Goto اجرا می شود کنترل اجرای برنامه به خط مشخص شده در جلوی دستور Goto منتقل شده و اجرای برنامه از آن خط به بعد ادامه می یابد. بنابراین تنها راه درک نحوه عملکرد یک برنامه در این سبک از برنامه نویسی، وانمود کردن اجرای برنامه توسط کامپیوتر (Trace) می باشد. این بدان معنی است که درک منطق چنین برنامه ای اغلب دشوار است.

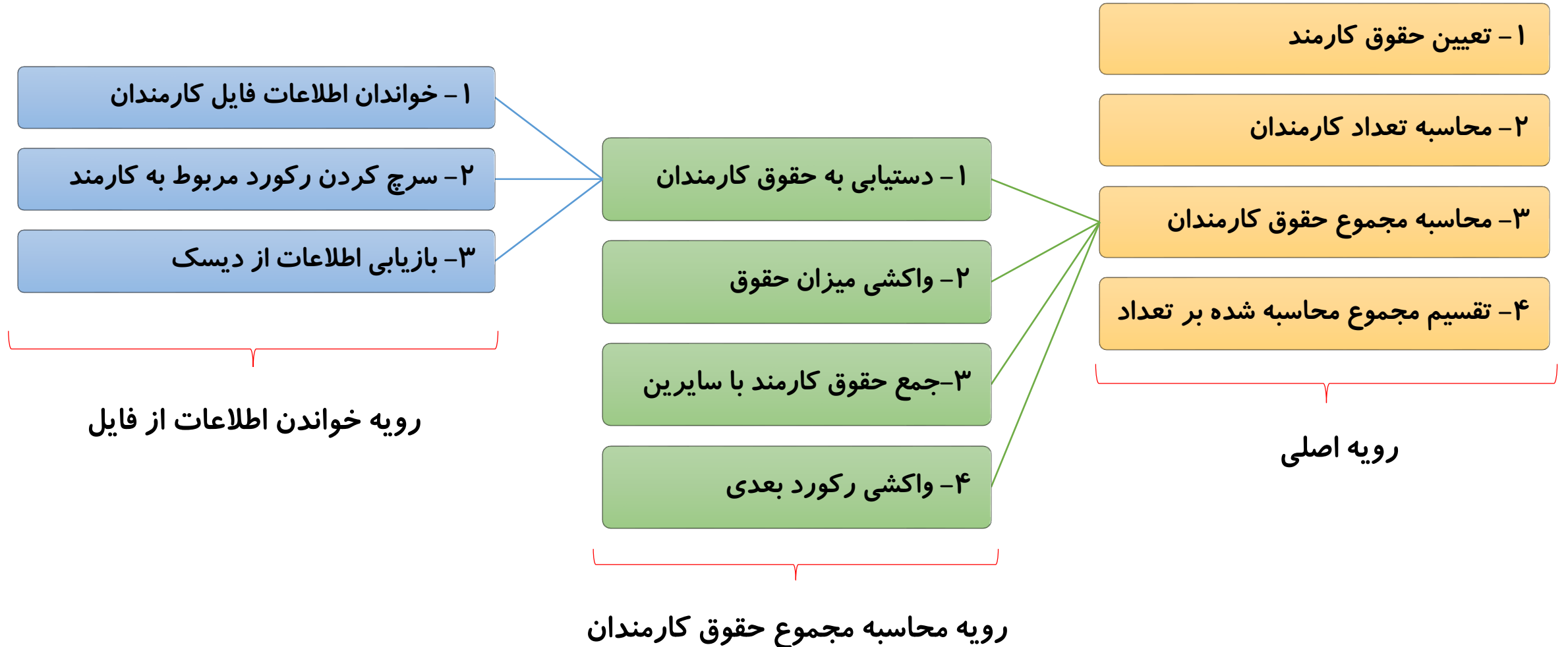
برنامه نویسی ساخت یافته



- برنامه نویسی ساخت یافته روشی منظم برای نوشتن برنامه هاست که منجر به ایجاد برنامه‌هایی با قابلیت خوانایی بالا، تست و اشکال زدایی آسان و با قابلیت اعتماد بالا می‌گردد.
- در این سبک از برنامه نویسی، برنامه به صورت مجموعه‌ای از فعالیت‌های معین تصور می‌گردد که می‌بایست بر روی داده‌های اعمال شوند تا نتیجه مطلوب حاصل گردد. در این روش هر برنامه پیچیده به مجموعه‌ای از مسائل کوچکتر تجزیه می‌شود تا اینکه قابل درک باشند.
- در واقع برنامه‌نویس رویه‌هایی می‌نویسد که نیازمندی‌های سیستم را برآورده می‌سازد.
- مثال: تجزیه برنامه محاسبه میانگین حقوق کارمندان یک شرکت



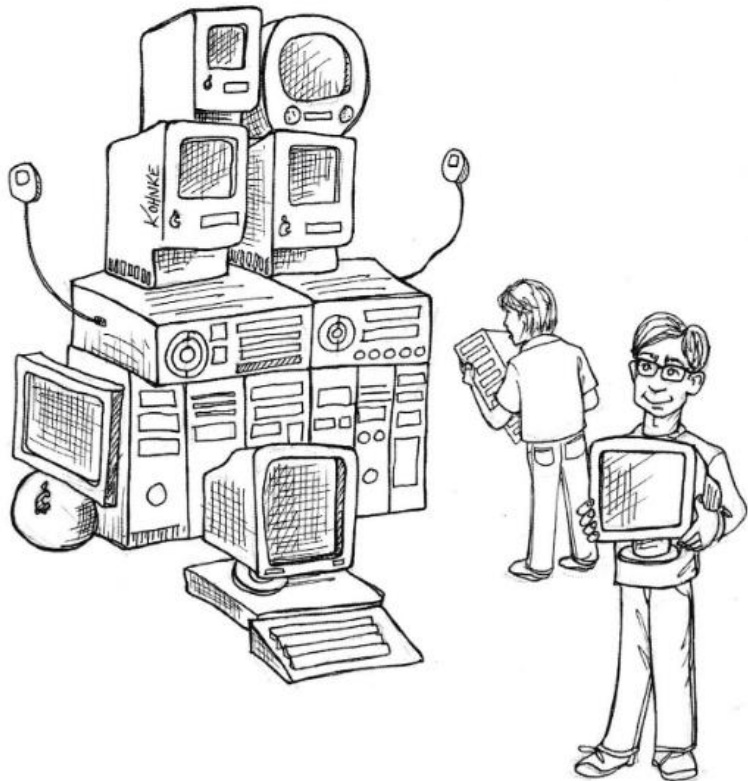
برنامه محاسبه میانگین حقوق کارمندان یک شرکت



مشکلات برنامه نویسی ساخت یافته



- در سبک برنامه نویسی ساخت یافته داده ها از فعالیت ها یا رویه هایی که آن ها را پردازش می کنند جدا می باشند به همین دلیل وقتی حجم داده ها زیاد باشد نگهداری آن ها مشکل شده و هر چه بیشتر با داده ها کار شود مشکلات بیشتر و عمیق تری نمایان می گردد.



برنامه نویسی شیء گرا

- در این سبک از برنامه نویسی، هر نرم افزار همانند یک سیستم الکترونیکی که از مجموعه ای از قطعات سخت افزاری ساخته شده است، **متشکل از مجموعه ای قطعات نرم افزاری** است که توانایی پردازش داده ها در کنار یکدیگر را فراهم می نمایند و بر مفهوم تفکر انتزاعی تأکید دارد.
- سرعت تولید برنامه در این روش بسیار بالا بوده و تست، عیب یابی و اصلاح برنامه آسان می باشد.

نکته مهم:

- اصل اساسی در برنامه نویسی شیء گرا، تمرکز بر مفهوم انتزاع (Abstraction) و پیاده سازی آن با ساختاری به نام کلاس (Class) است.
- انتزاع یا تجرید به معنای جداسازی و پنهان سازی جزئیات از کلیات می باشد.

سی شارپ چیست؟



سی شارپ (C#) یک زبان برنامه نویسی شیء گرا است که توسط شرکت مایکروسافت در سال ۲۰۰۰ میلادی ارائه شده است و ترکیبی از قابلیت‌های خوب زبان‌های C++، Java و Visual Basic را دارا می‌باشد.

از سی شارپ می‌توان برای ساخت برنامه‌های تحت ویندوز، تحت وب، وب سرویس‌ها، برنامه‌های موبایل و بازی‌ها استفاده کرد.

این زبان برنامه‌نویسی تنها زبانی است که به طور خاص برای دات نت فریم‌ورک (Net Framework) طراحی شده است.

سی شارپ از کتابخانه کلاس دات نت که شامل مجموعه بزرگی از اجزاء از قبل ساخته شده است، استفاده می‌کند. این اجزا به ساخت هر چه سریع‌تر برنامه‌ها کمک می‌کنند و با استفاده از آن‌ها می‌توان برنامه‌هایی با قابلیت مدیریت بیشتر و درک آسان ایجاد کرد. از این رو برای اجرای یک برنامه سی شارپ می‌بایست ابتدا دات نت فریم‌ورک نصب شود.

دلیل پیدایش زبان C#



در سال ۱۹۹۹، شرکت Sun Microsystems اجازه استفاده از زبان برنامه‌نویسی JAVA را در اختیار Microsoft قرار داد تا در سیستم عامل خود از آن استفاده کند. جاوا در اصل به هیچ پلت فرم یا سیستم عاملی وابسته نبود، ولی مایکروسافت برخی از مفاد قرار داد را زیر پا گذاشت و قابلیت مستقل از سیستم عامل بودن جاوا را از آن برداشت. شرکت Sun Microsystems پرونده‌ای علیه مایکروسافت درست کرد و مایکروسافت مجبور شد تا زبان شیء‌گرای جدیدی با کامپایلر جدید که به ++C شبیه بود را درست کند. آندرس هلزبرگ (Anders Hejlsberg) سرپرستی و مدیریت این پروژه را بر عهده گرفت و گروهی را برای طراحی زبانی جدید تشکیل داد و نام آن را Cool گذاشت. مایکروسافت در نظر داشت اسم این زبان را تا آخر Cool قرار دهد، ولی به دلیل مناسب نبودن برای اهداف تجاری این کار را نکرد. در ارائه و معرفی رسمی چارچوب دات‌نت در سال ۲۰۰۰ این زبان به سی‌شارپ تغییر نام یافت.

نسخه های مختلف زبان C#



تاریخ ارائه	نسخه Visual Studio	نسخه .NET Framework	نسخه سی شارپ
January 2002	Visual Studio .NET 2002	.NET Framework 1.0	C# 1.0
April 2003	Visual Studio .NET 2003	.NET Framework 1.1	C# 1.1
November 2005	Visual Studio 2005	.NET Framework 2.0	C# 2.0
November 2007	Visual Studio 2008	.NET Framework 3.0\3.5	C# 3.0
April 2010	Visual Studio 2010	.NET Framework 4.0	C# 4.0
August 2012	Visual Studio 2012/2013	.NET Framework 4.5	C# 5.0
July 2015	Visual Studio 2015	.NET Framework 4.6	C# 6.0
March 2017	Visual Studio 2017	.NET Framework 4.6.2	C# 7.0



دات نت فریم ورک (.NET Framework) چیست؟

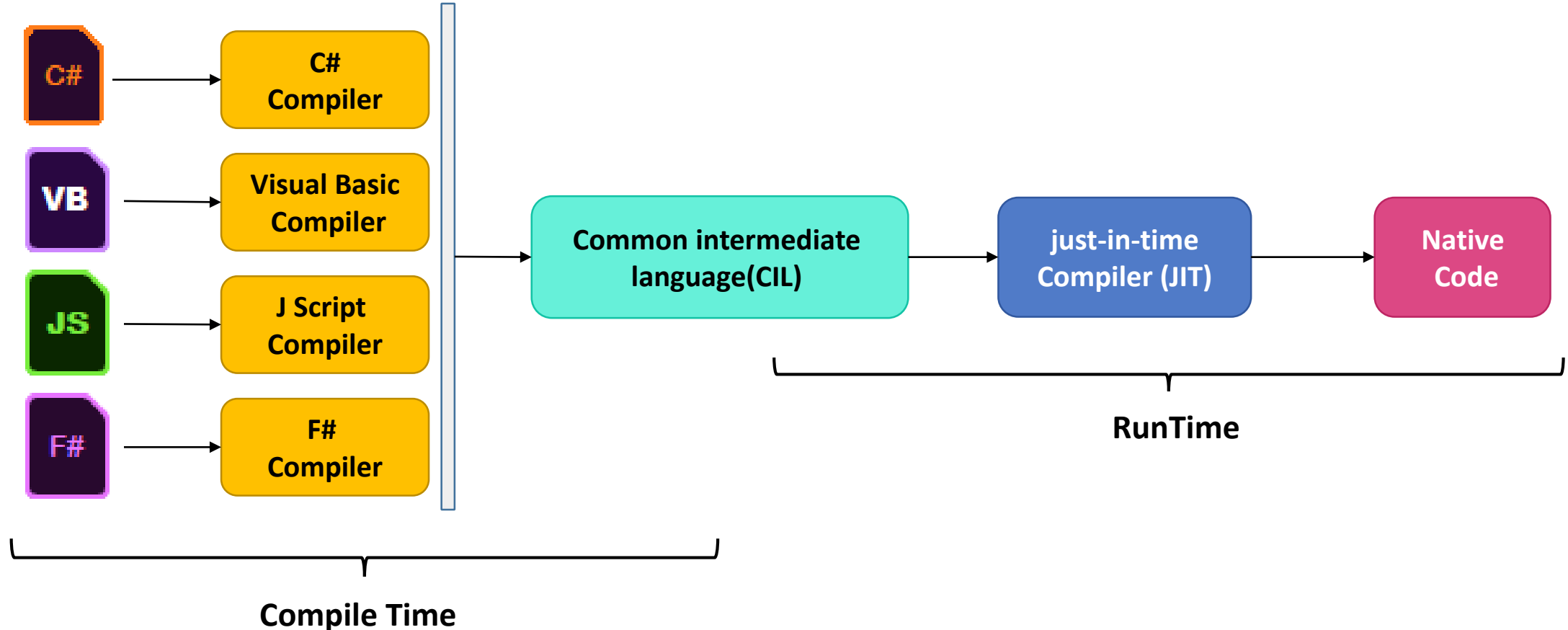
- .NET Framework یک چارچوب نرم افزاری است که توسط شرکت مایکروسافت برای توسعه انواع نرم افزارها، علی الخصوص نرم افزارهای تحت ویندوز طراحی شده است. همچنین می تواند برای توسعه نرم افزارهای تحت وب و تحت موبایل نیز مورد استفاده قرار بگیرد.
- این چارچوب شامل کتابخانه کلاس محیط کاری^۱ (FCL) است که در بر گیرنده کلاس ها، ساختارها، داده های شمارشی و... که امکان پیاده سازی برنامه های مختلف را فراهم می آورد. مهمترین قسمت .NET Framework زبان مشترک زمان اجرا^۲ (CLR) آن است که محیطی را فراهم می آورد که برنامه ها در آن اجرا شوند.
- زبان مشترک زمان اجرا (CLR) در حقیقت یک برنامه است که بر روی کامپیوتر اجرا می شود و اجرای کدهای مشترک میانی^۳ (IL) را مدیریت می نماید. از این رو کدهایی که تحت CLR و .Net اجرا می شوند، کدهای مدیریت شده نامیده می شوند، چون CLR جنبه های مختلف نرم افزار را در زمان اجرا مدیریت می کند.

1- Framework Class Library (FCL) 2- Common Language Runtime (CLR) 3- Common intermediate language(CIL)

دات نت فریم ورک (.NET Framework) چیست؟



- تمامی زبان های تحت چارچوب دات نت می بایست در زمان کامپایل یک برنامه، کدهای آن برنامه را به زبان مشترک میانی (CIL) که بسیار نزدیک و تقریباً شبیه به زبان اسمبلی است ترجمه نمایند. چرا که CIL تنها زبانی است برای دات نت قابل فهم است. این موضوع دلیل اصلی ارتباط برنامه های تحت Net. با یکدیگر است.





برنامه‌های مختلف در داده‌نگه‌ها با زبان‌های متفاوتی نوشته شده‌اند به دلیل ترجمه شدن در زمان کامپایل به زبان مشترک میانی (CIL) می‌توانند به راحتی با هم ارتباط برقرار کرده و به تعامل داده بپردازند.



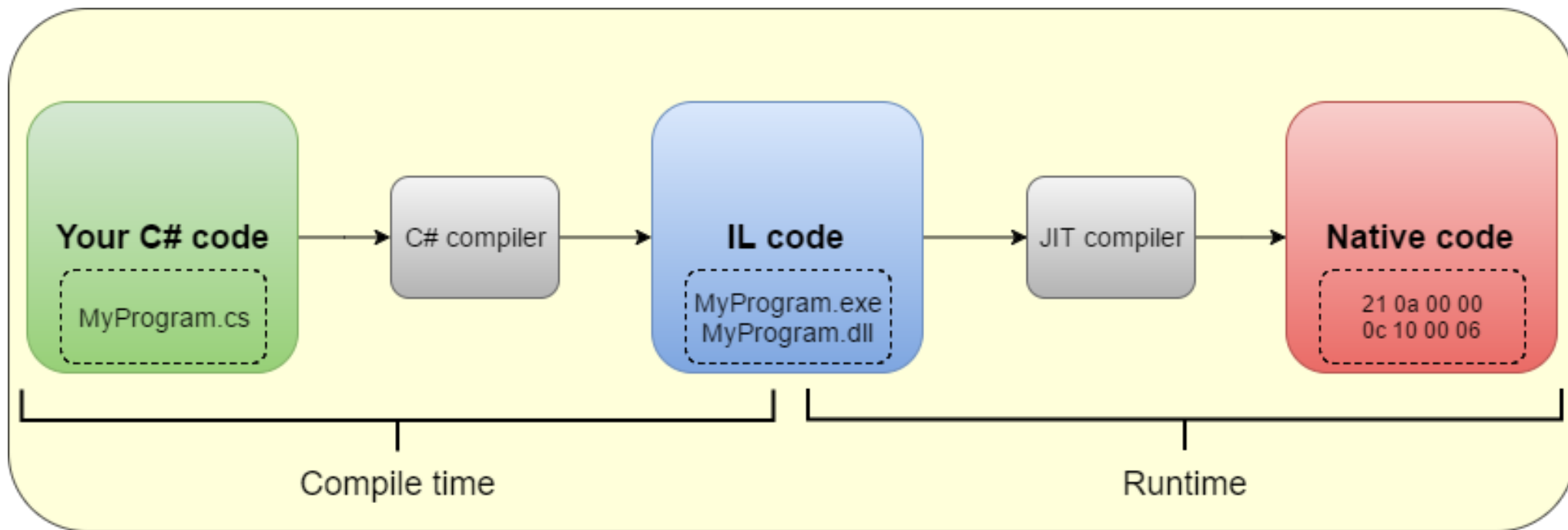
نحوه کامپایل برنامه در زبان های تحت .Net.

- ۱- ترجمه کدهای برنامه (برای مثال کدهای C# و Visual Basic.Net) به کدهای معادل آن در زبان مشترک میانی (CIL).
- ۲- ذخیره کدهای ترجمه شده توسط CIL در یک فایل اسمبلی مانند exe یا dll.
- ۳- انتقال کدهای ترجمه شده به زبان میانی به کامپایلر فقط در زمان (JIT)
- ۴- تبدیل کدهای زبان مشترک میانی به کدهایی که برای کامپیوتر قابل خواندن باشند توسط کامپایلر JIT در هر لحظه.

نکته:

کامپایلر JIT در لحظه فقط کدهایی را که برنامه در آن زمان نیاز دارد به زبان ماشین ترجمه میکند.

نحوه کامپایل برنامه در زبان های تحت .Net



سایر ویژگی های .Net Framework

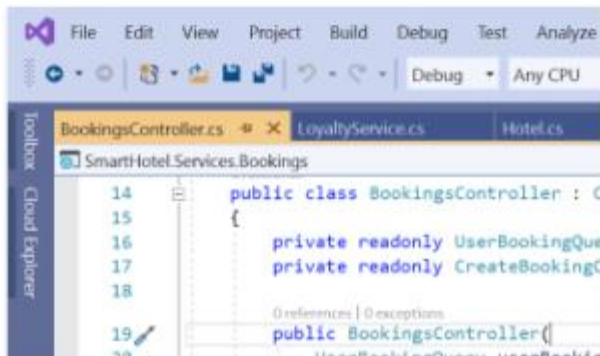


- دات نت ویژگی دیگری به نام سیستم نوع مشترک (CTS) نیز دارد که بخشی از CLR است و نقشه‌ای برای معادل سازی انواع داده ها در دات نت می‌باشد.
- با CTS نوع int در سی‌شارپ و نوع Integer در ویژوال بیسیک یکسان هستند، چون هر دو از نوع System.int32 مشتق می‌شوند.
- پاک کردن خانه‌های بلا استفاده حافظه در یک فایل (Garbage collection) یکی دیگر از ویژگی‌های دات‌نت فریم ورک است. هنگامی که از منابعی، زیاد استفاده نشود دات‌نت فریم ورک حافظه استفاده شده توسط برنامه را آزاد می‌کند.

ویژوال استودیو محیط توسعه یکپارچه ای (IDE) است، که دارای ابزارهایی برای کمک به برنامه نویسان جهت توسعه برنامه های سی شارپ و سایر زبان های تحت چارچوب دات نت می باشد.

همانطور که پیشتر نیز گفته شده می توان یک برنامه سی شارپ را با استفاده از برنامه Notepad یا هر برنامه ویرایشگر متن دیگر نوشت و با استفاده از کامپایلر سی شارپ آن را کامپایل کرده و از آن استفاده نمود. اما این کار بسیار سخت است چون اگر برنامه دارای خطا باشد خطایابی آن بسیار دشوار و زمان بر می باشد.

ویژوال استودیو، محیطی دارای ویژگی های زیاد برای کمک به برنامه نویسان جهت توسعه برنامه های سی شارپ است و تعداد زیادی از پردازش های لازم در زمان توسعه یک برنامه که موجب هدر رفتن وقت برنامه نویسان می شود را به صورت خودکار انجام می دهند.



- ویژگی اینتلی سنس (Intellisense):

این ویژگی به تایپ سریع کدها کمک می کند.

- ویژگی Break Point:

این ویژگی به برنامه نویس اجازه می دهد تا در طول اجرای برنامه مقادیر موجود در متغیرها را چک کند.

- خطایابی خودکار (Auto Debugging):

با این ویژگی ویژوال استودیو برنامه شما را خطایابی می کند و حتی خطاهای کوچک (مانند بزرگ یا کوچک نوشتن حروف) را برطرف می کند.

- امکان ساخت رابط گرافیکی:

ویژگی ویژوال استودیو دارای ابزارهای طراحی زیادی برای ساخت یک رابط گرافیکی است که بدون ویژوال استودیو برای ساخت چنین رابط های گرافیکی باید هزاران خط کد نوشت.

- افزایش بازدهی:

ویژوال استودیو محیطی قدرتمند برای طراحی برنامه ها می باشد و با ویژگی های بسیاری که فراهم می کند بازدهی برنامه نویس را افزایش داده و در مدت زمان لازم جهت طراحی یک برنامه صرفه جویی می کند.

محیط های توسعه مورد پشتیبان ویژوال استدیو

Solutions



Windows

Develop apps and games to reach every device running Windows



Mobile Apps

Create native or hybrid apps targeting Android, iOS, and Windows



Azure Apps

Build, manage, and deploy cloud scale apps to Azure with ease



Web Apps

Develop modern web apps with flexibility and powerful open tools



Office

Utilize powerful tools for all types of Office development



Games

Design, code, and debug games with cutting-edge graphics and scripting tools



Extensions

Write your own extensions for Visual Studio



Database

Develop and deploy SQL Server and Azure SQL databases with ease

نسخه های ویژوال استودیو

- در حال حاضر آخرین نسخه ویژوال استودیو، Visual Studio 2019 است. این نسخه به ۳ نسخه زیر تقسیم می گردد:

• Visual Studio Community

نسخه متن باز (Open Source) ویژوال استودیو می باشد و بطور رایگان در دسترس عموم است. این نسخه تمام ویژگی های نسخه Professional را دارا بوده و هیچگونه محدودیتی برای طراحی و توسعه برنامه های تحت چارچوب دات نت با اهداف شخصی، تحقیقات دانشگاهی و توسعه نرم افزار در تیم های کوچک نرم افزاری ندارد. ولی استفاده از این نسخه در همه شرایط جایز نیست.

• Visual Studio Professional

نسخه ارزان قیمت ویژوال استودیو می باشد که نسبت به نسخه Community کمی کاملتر و نسبت به نسخه Enterprise دارای امکانات کمتری است. این نسخه به منظور استفاده در سازمان های کوچک و تیم هایی با حداقل ۶ عضو طراحی شده است.

• Visual Studio Enterprise

این نسخه علاوه بر ویژگی های ارائه شده در نسخه Professional، مجموعه زیادی از ابزارهای لازم جهت توسعه نرم افزار، توسعه پایگاه داده، تست و گزارشگری، طراحی معماری، طراحی معیار و... را ارائه می دهد.

تفاوت های نسخه های ویژوال استودیو از لحاظ فنی

Supported Features	Visual Studio Community Free download	Visual Studio Professional Buy	Visual Studio Enterprise Buy
⊕ Supported Usage Scenarios	● ● ● ○	● ● ● ●	● ● ● ●
Development Platform Support ²	● ● ● ●	● ● ● ●	● ● ● ●
⊕ Integrated Development Environment	● ● ● ○	● ● ● ○	● ● ● ●
⊕ Advanced Debugging and Diagnostics	● ● ○ ○	● ● ○ ○	● ● ● ●
⊕ Testing Tools	● ○ ○ ○	● ○ ○ ○	● ● ● ●
⊕ Cross-platform Development	● ● ○ ○	● ● ○ ○	● ● ● ●
⊕ Collaboration Tools and Features	● ● ● ●	● ● ● ●	● ● ● ●

نیازمندی های سخت افزار و نرم افزاری جهت نصب Visual Studio

سیستم عامل	سخت افزار
Windows 10	1.6 GHz or faster processor
Windows 8.1	1 GB of RAM (1.5 GB if running on a virtual machine)
Windows 8	4 GB of available hard disk space
Windows 7 Service Pack 1	5400 RPM hard disk drive
Windows Server 2012 R2	DirectX 9-capable video card that runs at 1024 x 768 or higher display resolution
Windows Server 2012	
Windows Server 2008 R2 SP1	

دانلود Visual Studio

<https://www.visualstudio.com/en-us/downloads/download-visual-studio-vs.aspx>



Visual Studio 2019

Version 16.5

[Release notes >](#)

Full-featured integrated development environment (IDE) for Android, iOS, Windows, web, and cloud

[Compare editions >](#)

[How to install offline >](#)

Community

Powerful IDE, free for students, open-source contributors, and individuals

[Free download](#) ↓

[Download Preview](#) ↓

Professional

Professional IDE best suited to small teams

[Free trial](#) ↓

[Download Preview](#) ↓

Enterprise

Scalable, end-to-end solution for teams of any size

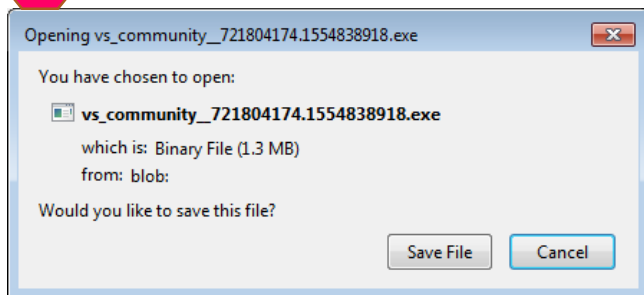
[Free trial](#) ↓

[Download Preview](#) ↓

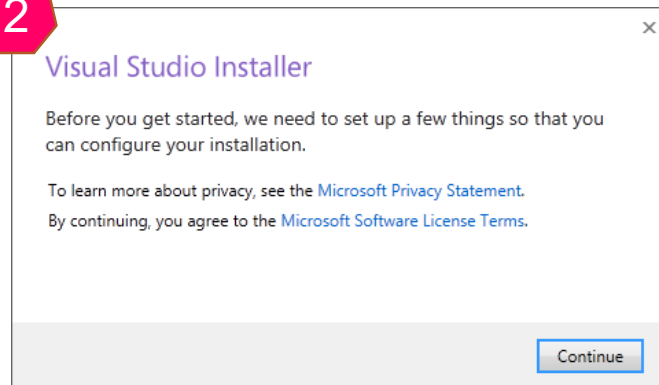
مراحل نصب Visual Studio 2019



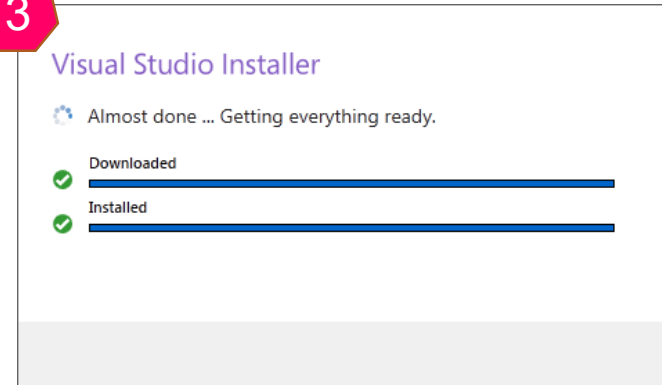
1



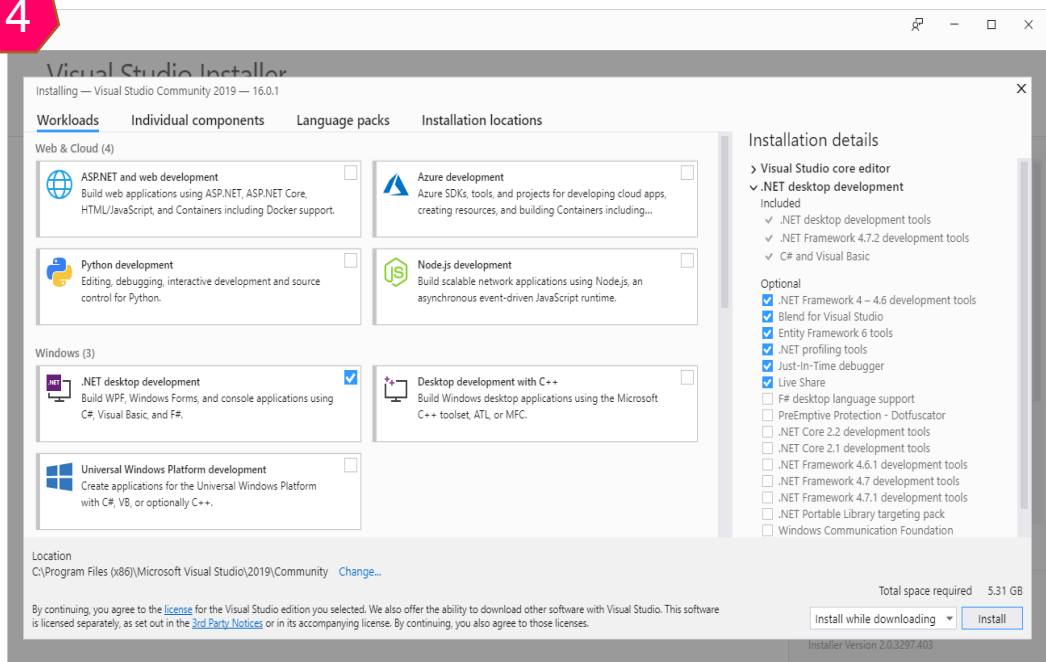
2



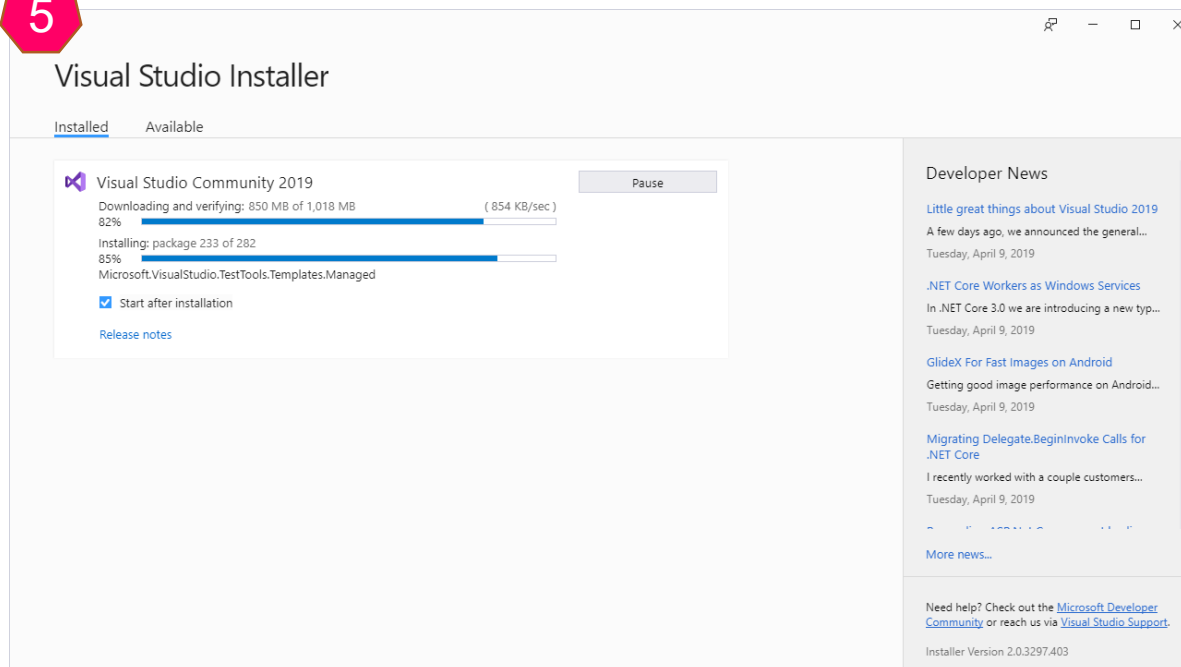
3



4



5



Visual Studio 2019 مراحل نصب



6

Reboot required

Success! One more step to go. Please restart your computer before you start Visual Studio Community 2019.

[Get troubleshooting tips](#)

7



<http://goo.gl/hMPYnE>

8

Visual Studio

Welcome!

Connect to all your developer services.

Sign in to start using your Azure credits, publish code to a private Git repository, sync your settings, and unlock the IDE.

[Learn more](#)

No account? [Create one!](#)

[Not now, maybe later.](#)

9

Visual Studio 2019 مراحل نصب



10

Visual Studio

Start with a familiar environment

Development Settings: **General**

Choose your color theme

Blue

Blue (Extra Contrast)

Dark

Light

You can always change these settings later.

Start Visual Studio

11

Visual Studio 2019

Open recent

As you use Visual Studio, any projects, folders, or files that you open will show up here for quick access.

You can pin anything that you open frequently so that it's always at the top of the list.

Get started

Clone or check out code
Get code from an online repository like GitHub or Azure DevOps

Open a project or solution
Open a local Visual Studio project or .sln file

Open a local folder
Navigate and edit code within any folder

Create a new project
Choose a project template with code scaffolding to get started

[Continue without code](#)



Start Page

Get Started

[Learn about new features in Enterprise 2017](#)

[See what's new in the .NET Framework](#)

[Explore what's new in Visual Studio Team Services](#)

Recent

C:\Users\Baeimani\documents\visual studio 2017\Projects\ConsoleCal...

This week

شهرستان کارون
C:\Users\Baeimani\Desktop\کارون شهرستان

AdvanceProgmming2-Class1.sln
C:\Users\Baeimani\documents\visual studio 2017\Projects\AdvancePr...

SmartSchool.sln
C:\Users\Baeimani\documents\visual studio 2017\Projects\SmartScho...

JahadFestival.sln

Open

Get code from a remote version control system or open something on your local drive.

Checkout from:

Visual Studio Team Services

Open Project / Solution

Open Folder

Open Website

New project

Search project templates

Recent project templates:

Console App (.NET Framework) C#

ASP.NET Web Application (.NET Fra... C#

Windows Forms App (.NET Framewor... C#

[Create new project...](#)



Developer News

Solution Explorer

Solution Explorer Team Explorer

Error List Output

Ready

