

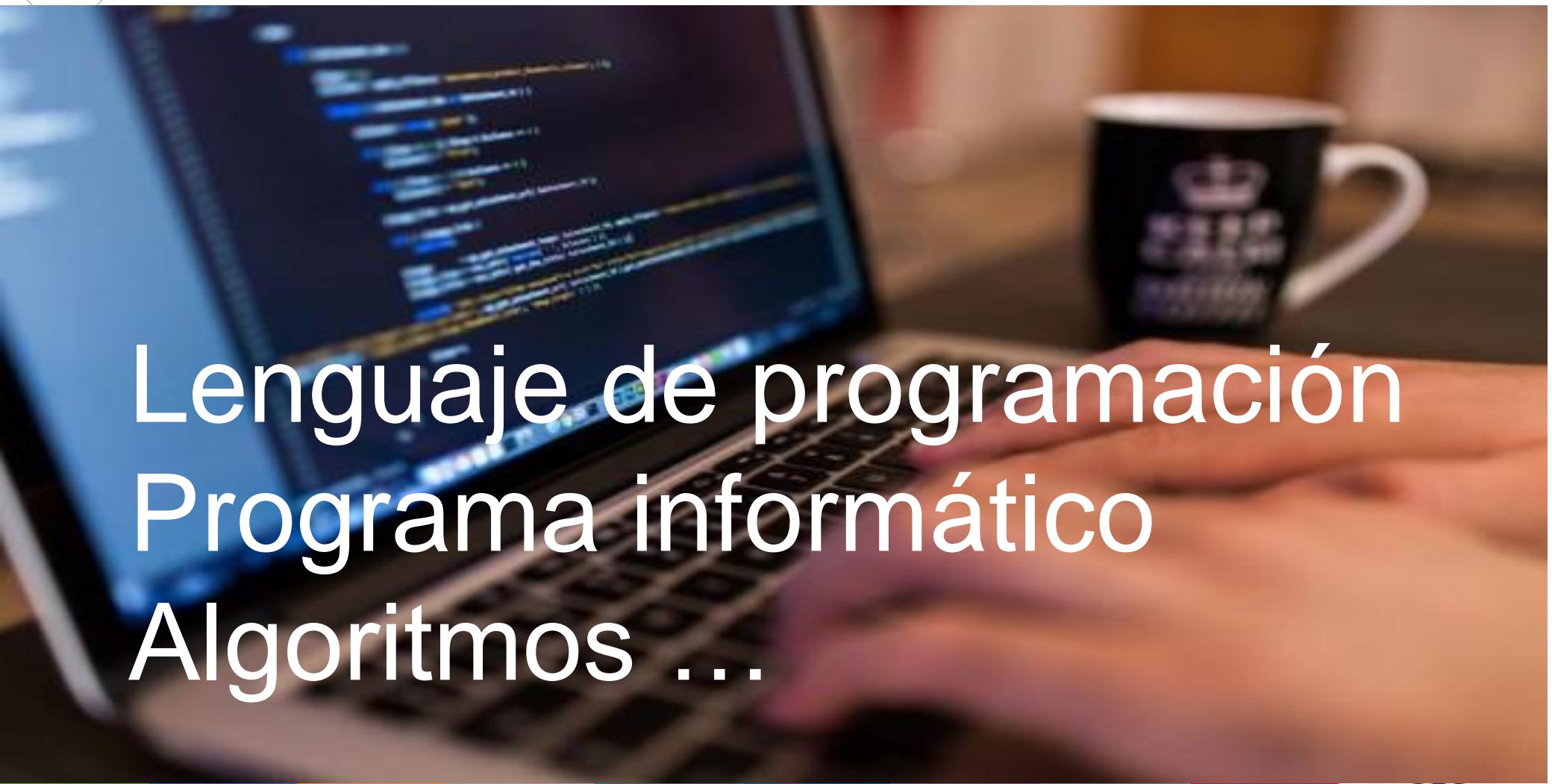


Clase presencial



Lenguaje de programación

Instructor: David Paúl Porras Córdova
@iscodem



Lenguaje de programación
Programa informático
Algoritmos ...

¿Qué es un Lenguaje de Programación?

Lenguaje formal diseñado para realizar procesos que pueden ser llevados a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina



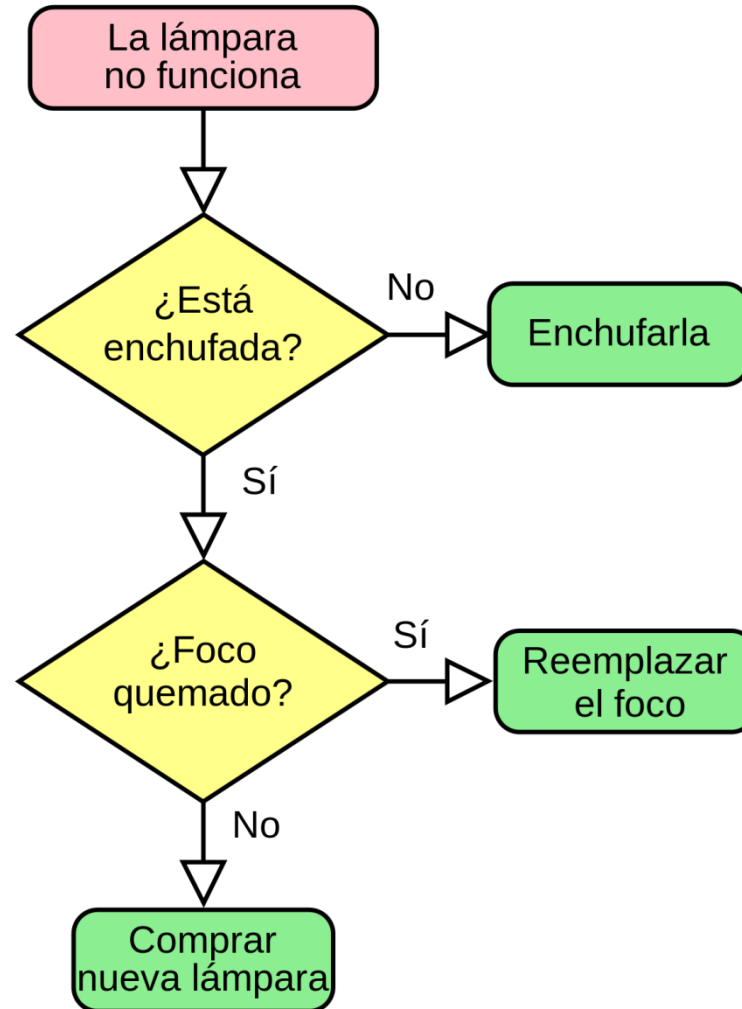
¿Qué es un programa informático?

Secuencia de instrucciones escritas para realizar una tarea específica en una PC. Este dispositivo requiere **programas** para funcionar, por lo general, ejecutando las instrucciones del **programa** en un CPU.

```
*  
* @var boolean  
*/  
define('PSI_INTERNAL_XML', false);  
  
if (version_compare("5.2", PHP_VERSION, ">")) {  
    die("PHP 5.2 or greater is required!!!");  
}  
if (!extension_loaded("pcre")) {  
    die("phpSysInfo requires the pcre extension to php in order to work  
    properly.");  
}  
  
require_once APP_ROOT.'/includes/autoloader.inc.php';  
  
// Load configuration  
require_once APP_ROOT.'/config.php';  
  
if (!defined('PSI_CONFIG_FILE') || !defined('PSI_DEBUG')) {  
    $tpl = new Template("/templates/html/error_config.html");  
    echo $tpl->fetch();  
    die();  
}
```

¿Qué es un algoritmo?

Los **algoritmos** pueden ser expresados de muchas maneras, incluyendo al lenguaje natural, pseudocódigo, diagramas de flujo y lenguajes de **programación** entre otros. El usar pseudocódigo y diagramas de flujo evita muchas ambigüedades.



Paradigmas de Programación

Propuesta tecnológica adoptada por una comunidad de programadores y desarrolladores cuyo núcleo central es incuestionable en cuanto que únicamente trata de resolver uno o varios problemas claramente delimitados.



Imperativo

¿**Cómo** resolver el problema?

- Programación modular
- Programación estructurada
- Orientada a eventos

VS.

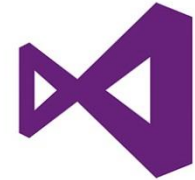
Declarativo

¿**Qué** hacer para resolver el problema?

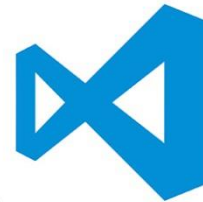
- Funcional
- **Lógica**
- Programación reactiva
- Lenguajes descriptivos

Paradigmas de Programación

Propuesta tecnológica adoptada por una comunidad de programadores y desarrolladores cuyo núcleo central es incuestionable en cuanto que únicamente trata de resolver uno o varios problemas claramente delimitados.

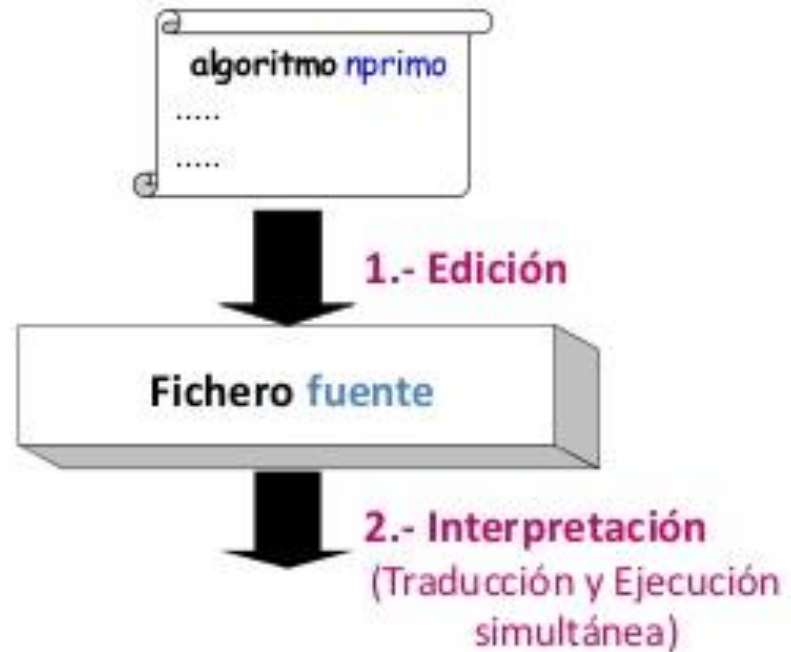
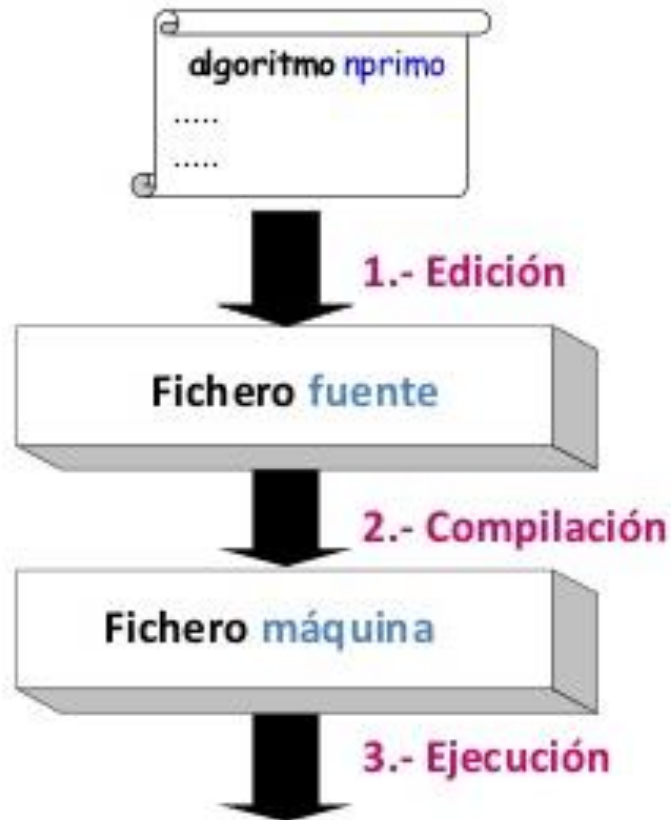


free pascal



Compiladores e Interpretes

En la interpretación se traduce a lenguaje máquina cada instrucción del lenguaje de alto nivel, una a una, en tiempo de ejecución.
En la compilación se traducen (compilan) mediante un programa (compilador) todas las instrucciones del lenguaje a lenguaje máquina, previamente a su ejecución.



Editores de código/ IDEs

Un **IDE** es un entorno de **programación** que ha sido empaquetado como un programa de aplicación, o sea, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica.



ATOM



Los Lenguajes de Programación más usados en el mundo

Enero 2018. Fuente: TIOBE – “The Importance of Being Earnest” (<https://www.tiobe.com/tiobe-index/>)

Jan 2018	Jan 2017	Change	Programming Language	Ratings	Change
1	1		Java	14.215%	-3.06%
2	2		C	11.037%	+1.69%
3	3		C++	5.603%	-0.70%
4	5	▲	Python	4.678%	+1.21%
5	4	▼	C#	3.754%	-0.29%
6	7	▲	JavaScript	3.465%	+0.62%
7	6	▼	Visual Basic .NET	3.261%	+0.30%
8	16	▲▲	R	2.549%	+0.76%
9	10	▲	PHP	2.532%	-0.03%
10	8	▼	Perl	2.419%	-0.33%
11	12	▲	Ruby	2.406%	-0.14%
12	14	▲	Swift	2.377%	+0.45%
13	11	▼	Delphi/Object Pascal	2.377%	-0.18%
14	15	▲	Visual Basic	2.314%	+0.40%

¿Qué Lenguaje de programación estudiaremos?

JavaScript



JavaScript



Algunos conceptos previos

HTML



CSS



JS



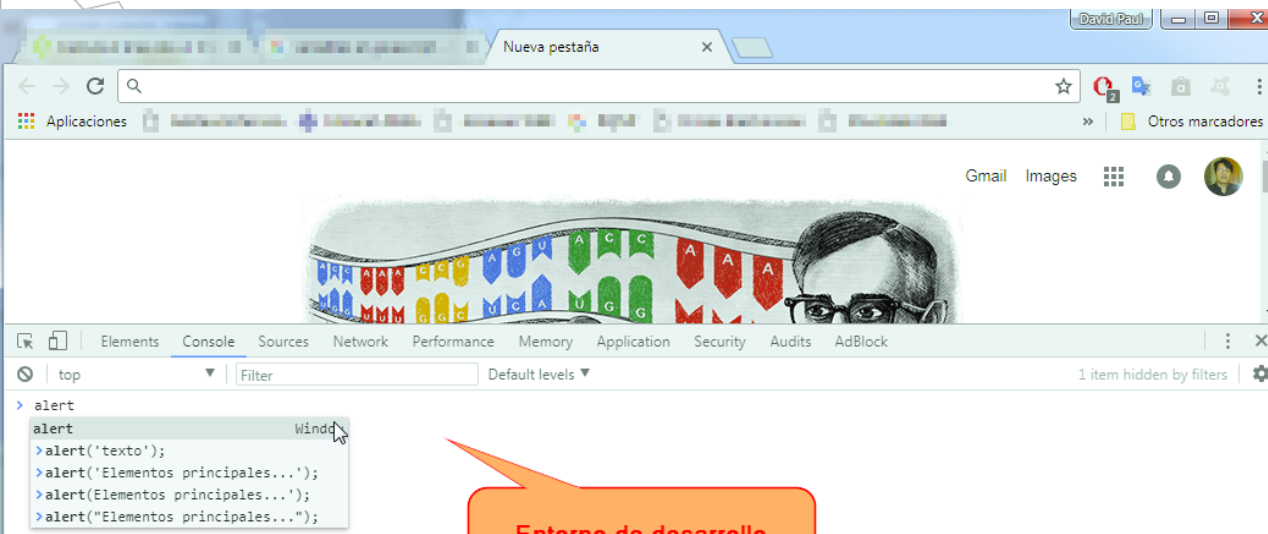
¿Qué necesitamos para iniciar a programar?



Sintaxis y Nomenclatura

Taller 1: Calcular el área de un triángulo

Usando Google Chrome



Usando SublimeText



Sintaxis y Nomenclatura

Taller 1: Calcular el área de un triángulo



Conociendo Visual Studio .NET

Porqué C#

Un solo lenguaje, múltiples tipos de aplicaciones, C# es de propósito general:

- Web
- Consola
- Backend
- Desktop
- Servicios
- Mobile Apps
- IoT
- User mode drivers



Conociendo Visual Studio .NET

Porqué C#

No esta atado a una sola plataforma

- Microsoft
 - Windows Desktop
 - Windows Mobile
 - Windows Embedded
 - Windows IoT
- Apple
 - OSX
 - iOS
- Linux



Conociendo Visual Studio .NET

En que mercado se usa

Video Juegos
Agencias Digitales
Aplicaciones corporativas
Banca
Gobierno
Seguros
Fábricas de software

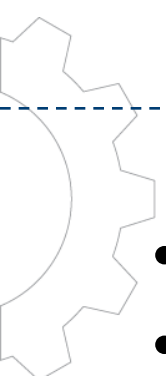


Evolución de C#

Pasado, presente y Futuro

2002	2005	2008	2010	2012	2015	2017
C# 1.0	C# 2.0	C# 3.0	C# 4.0	C# 5.0	C# 6.0	C# 7.0
Managed Code	Generics	Implicitly typed local variables	Dynamic binding	Asynchronous methods	Compiler-as-a-service (Roslyn)	Local functions
Pointers	Partial types	Object and collection initializers	Named and optional arguments	Caller info attributes	Import of static type members into namespace	Tuples
Type Safe	Anonymous methods	Auto-Implemented properties	Generic co- and contravariance		Exception filters	Pattern matching
Strong Typed	Iterators	Anonymous types	Embedded interop types ("NoPIA")		Await in catch/finally blocks	Records / algebraic data types
POO	Nullable types	Extension methods			Auto property initializers	Nullability tracking
	Getter/setter separate accessibility	Query expressions			Default values for getter-only properties	Async streams and disposal
	Method group conversions (delegates)	Lambda expressions			Expression-bodied members	Strongly typed access to wire formats
	Co- and Contra-variance for delegates	Expression trees			Null propagator (succinct null checking)	
	Static classes	Partial methods			String Interpolation	
	Delegate inference				nameof operator	
					Dictionary initializer	

Herramientas para crear software con C#

- 
- Bloc de Notas
 - Visual Studio Code
 - Visual Studio Community
 - Visual Studio Professional / Enterprise
-
- Sharp Develop
 - Mono Develop
 - Sublime
 - Etc.

Herramientas que usaremos en el curso, pre-requisitos

Conociendo Visual Studio .NET

<https://code.visualstudio.com/Download>

Visual Studio Code

Download Visual Studio Code

Free and open source. Integrated Git, debugging and extensions.

Stable

Insiders



↓ Windows

Windows 7, 8, 10



↓ Linux x64

Debian, Ubuntu, Fedora, CentOS

32bit version



↓ OS X

OS X Yosemite, El Capitan

Visual Studio Community

<https://www.visualstudio.com/products/visual-studio-community-vs>


Visual Studio Community

Un completo IDE extensible y gratuito para crear aplicaciones modernas para Windows, Android e iOS, además de aplicaciones web y servicios en la nube.


Descargar Community 2015



Paradigmas de programación

- 
1. Orientado a Objetos
 2. Imperativo
 3. Funcional
 4. Dinámico
 5. Meta programación (orientado a atributos)

Cómo funciona C#

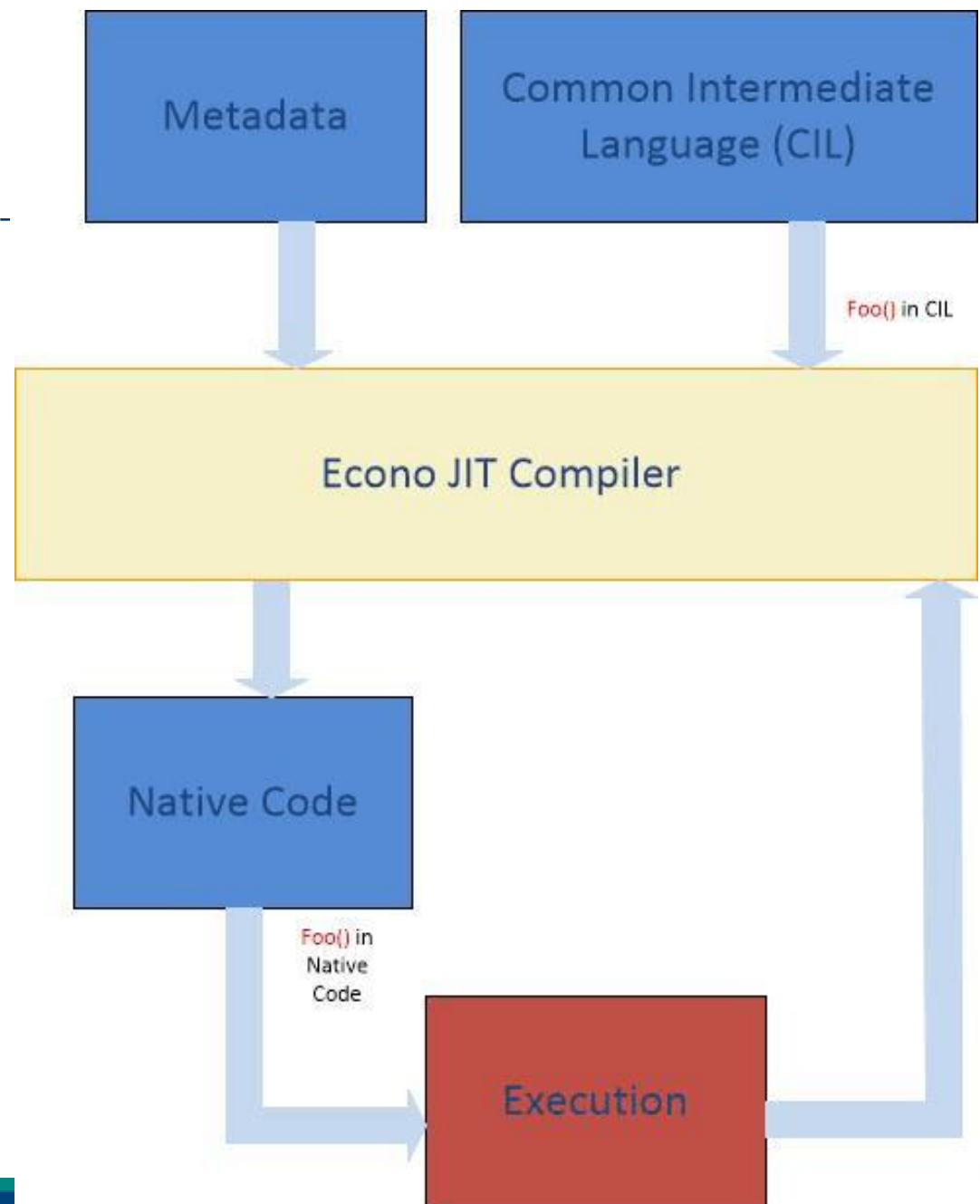


Interpretado
Compilado
Nativo
JIT



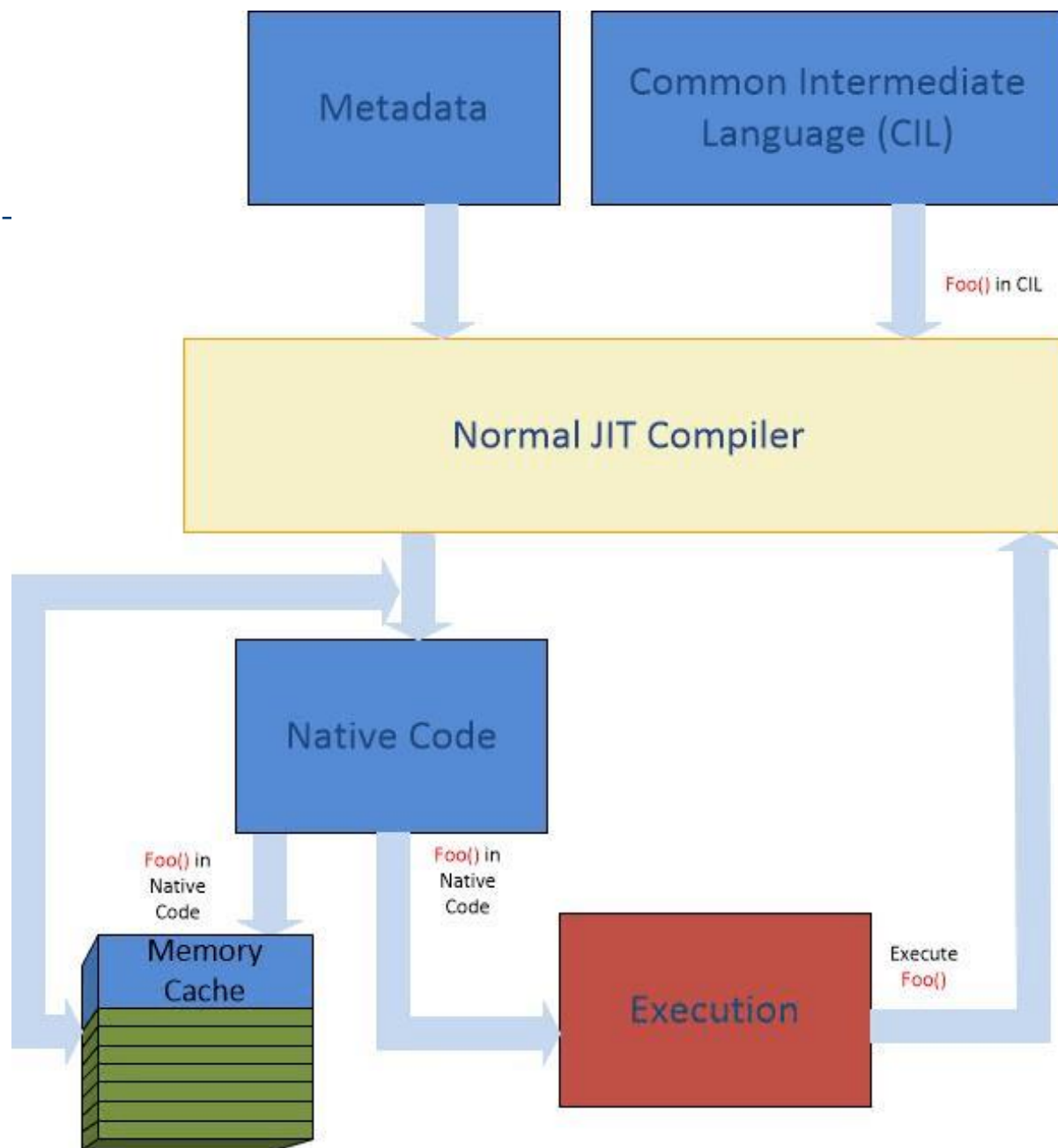
Compilación “Económica”

* Tomado de
<http://www.telerik.com/blogs/understanding-net-just-in-time-compilation>

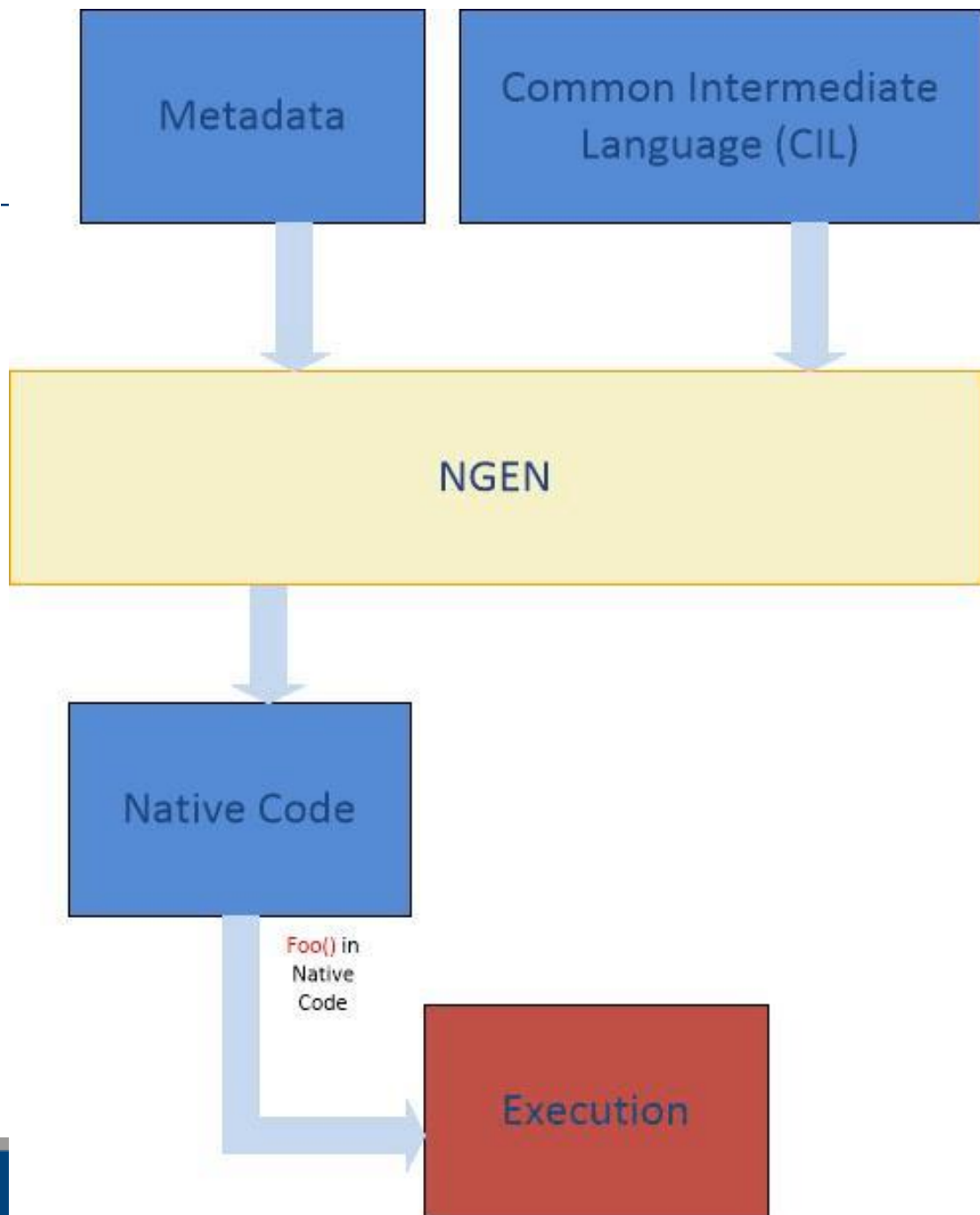


Compilación normal

* Tomado de
<http://www.telerik.com/blogs/understanding-net-just-in-time-compilation>

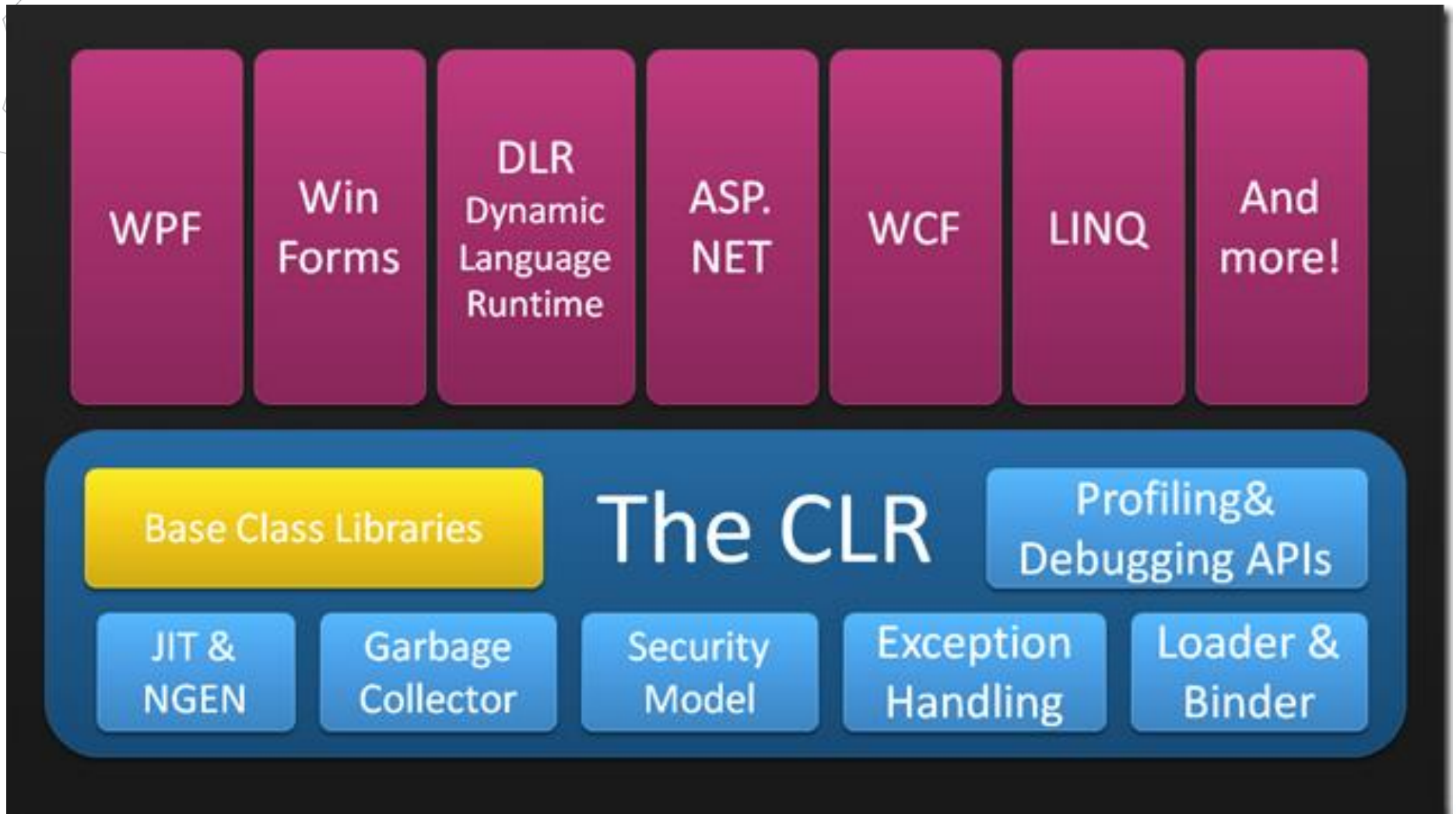


Compilación previa



* Tomado de
<http://www.telerik.com/blogs/understanding-net-just-in-time-compilation>

Cómo funciona C#



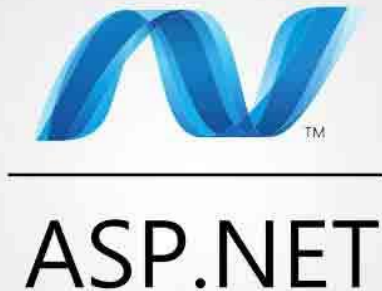
Aplicaciones en Consola

- Escribir y leer en la consola
- Que es using
- Descripción general
 - solución
 - proyecto
 - Namespace
 - Clase
 - Método
- Compilar
- En C# TODO es un objeto , TODO
- Extraer a método
- Donde se guarda el programa
- Recibir parámetros
- Sentencia If, preguntar si hay params
- Parámetros desde Visual Studio
- Sentencia Switch, cambiar el mensaje
- Explicar generalidades de la configuración del proyecto
- Sentencia while
- Incrementos ++
- Sentencia for
- Sentencia foreach



Web Stack

- HTML 5
- CSS 3
- JS
- frameworks



.net Stack

- C#
- ASP.Net
 - WebForms
 - Web Pages
 - MVC
- .net Framework
- Librerias

Backend

Demo ASP.Net Web API
Con Json ext de chrome

Conociendo Visual Studio .NET



Website

Demo ASP.Net MVC

Apps para Windows

- Precargar VS como Admin
- Precargar emulador
- App con botón
- Mostrar botón desde el diseñador
- Mostrar botón desde XAML
- Panel de propiedades
- Ejecutar en Win 10
- Ejecutar en Windows Mobile
- Explicar todo lo que se puede hacer en el emulador

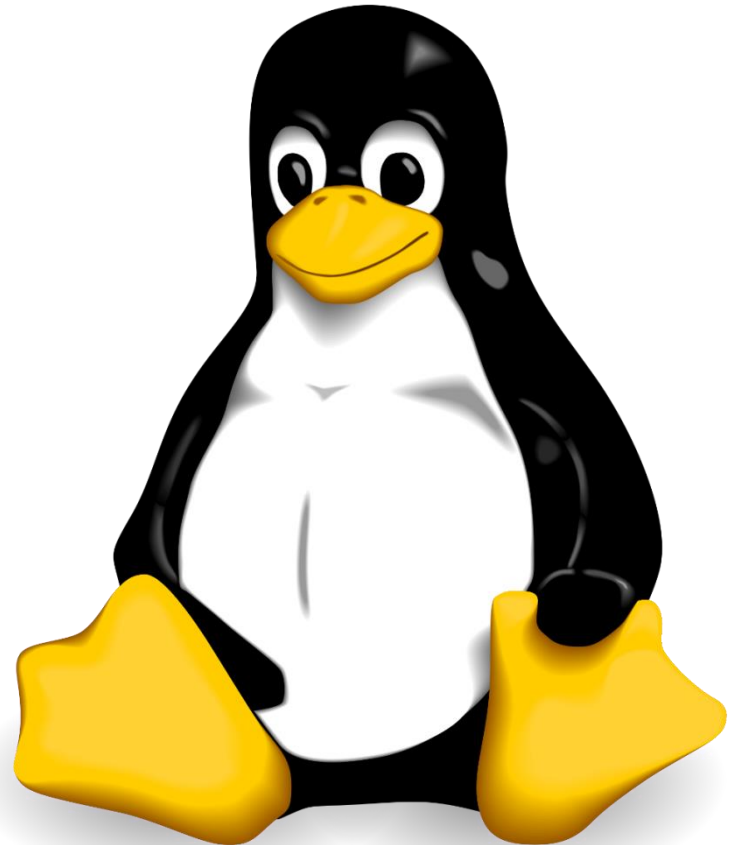
Apps para Android

- Precargar VS como Admin
- Precargar emulador Android
- App con botón
- Mostrar botón desde el diseñador
- Ejecutar emulador
- Pegar código HttpClient

C# Multiplataforma

Por qué ?
Para qué?

Estado Actual : RC 1



Instalar Visual Studio Code

```
sudo add-apt-repository ppa:ubuntu-desktop/ubuntu-make  
sudo apt-get update  
sudo apt-get install ubuntu-make  
  
sudo umake web visual-studio-code
```

Cómo instalar .Net Core

<http://dotnet.github.io/getting-started/>

Habilitar C# en vs code

- Abrir VS Code
- Presionar F1
- En la caja que se abre escribir ext install y presionar ENTER
- En la lista desplegada buscar C#.
- Una vez instalado se reinicia VS code
- Al abrir el primer proyecto toma un tiempo descargando algunos componentes.

Hola Mundo en Linux

- Una vez instalado .net Core abrir el folder desde VS Code

ASP.Net Core en Linux

<https://docs.asp.net/en/latest/getting-started/installing-on-linux.html>

```
sudo apt-get install make automake libtool curl
```

```
curl -sSL https://github.com/libuv/libuv/archive/v1.8.0.tar.gz | sudo tar zxfv - -C /usr/local/src  
cd /usr/local/src/libuv-1.8.0
```

```
sudo sh autogen.sh  
sudo ./configure
```

```
sudo make  
sudo make install
```

```
sudo rm -rf /usr/local/src/libuv-1.8.0 && cd ~/  
sudo ldconfig
```

ASP.Net Core en Linux

Clonar repositorio de ejemplos

```
git clone https://github.com/aspnet/Home.git
```

Ir al ultimo reléase estable (actualmente 1.0.0-rc1-update1)

```
cd samples/1.0.0-rc1-update1/HelloMvc
```

Restaurar los paquetes

```
dnx restore
```

Lanzar el sitio

```
dnx web
```




Tipos de Dato

Short Name	.NET Class	Type	Width	Range (bits)
byte	Byte	Unsigned integer	8	0 to 255
sbyte	SByte	Signed integer	8	-128 to 127
int	Int32	Signed integer	32	-2,147,483,648 to 2,147,483,647
uint	UInt32	Unsigned integer	32	0 to 4294967295
short	Int16	Signed integer	16	-32,768 to 32,767
ushort	UInt16	Unsigned integer	16	0 to 65535
long	Int64	Signed integer	64	-9223372036854775808 to 9223372036854775807
ulong	UInt64	Unsigned integer	64	0 to 18446744073709551615
float	Single	Single-precision floating point type	32	-3.402823e38 to 3.402823e38
double	Double	Double-precision floating point type	64	-1.79769313486232e308 to 1.79769313486232e308
char	Char	A single Unicode character	16	Unicode symbols used in text
bool	Boolean	Logical Boolean type	8	True or false
object	Object	Base type of all other types		
string	String	A sequence of characters		
decimal	Decimal	Precise fractional or integral type that can represent decimal numbers with 29 significant digits	128	$\pm 1.0 \times 10e-28$ to $\pm 7.9 \times 10e28$



Comentarios...

