***LENGUAJE DE PROGRAMACION***

1***.-*Gambas**

Es un lenguaje de programación [libre](https://es.wikipedia.org/wiki/Software_libre) derivado de BASIC (de ahí que Gambas quiere decir Gambas Almost Means Basic). Se distribuye con licencia [GNU GPL](https://es.wikipedia.org/wiki/Licencia_p%C3%BAblica_general_de_GNU). Cabe destacar que presenta ciertas similitudes con Java ya que en la ejecución de cualquier aplicación, se requiere un conjunto de librerías intérprete previamente instaladas (Gambas Runtime) que entiendan el bytecode de las aplicaciones desarrolladas y lo conviertan en código ejecutable por el computador.

Permite crear formularios con botones de comandos, cuadros de texto y muchos otros controles y enlazarlos a [bases de datos](https://es.wikipedia.org/wiki/Base_de_datos) como [MySQL](https://es.wikipedia.org/wiki/MySQL), [PostgreSQL](https://es.wikipedia.org/wiki/PostgreSQL) o [SQLite](https://es.wikipedia.org/wiki/SQLite) además de facilitar la creación de aplicaciones muy diversas como videojuegos (utilizando OpenGL), aplicaciones para dispositivos móviles, aplicaciones de red (con manejo avanzado de protocolos HTTP, FTP, SMTP, DNS), entre otras .

Año de lanzamiento 1999 desarrollado por Benoít Minisini. Ultima versión estable es la 3.14.2 lanzada el 19 de noviembre de 2019  
  
***Ejemplo de Hola Mundo***

Public Sub Form\_Open()

Message("Hello World!")

End

2**.-Lua 5.0.3**

Lua es un lenguaje de programación [multiparadigma](https://es.wikipedia.org/w/index.php?title=Lenguaje_de_programaci%C3%B3n_multiparadigma&action=edit&redlink=1), imperativo, estructurado y bastante ligero, que fue diseñado como un lenguaje interpretado con una [semántica](https://es.wikipedia.org/wiki/Sem%C3%A1ntica) extendible. El nombre significa «luna» en [portugués](https://es.wikipedia.org/wiki/Idioma_portugu%C3%A9s).

Lua fue creado en 1993 por Roberto Ierusalimschy, Luiz Henrique de Figueiredo y Waldemar Celes, miembros del Grupo de Tecnología en Computación Gráfica (Tecgraf) en la Pontificia Universidad Católica de Río de Janeiro, basándose en una estructura similar a [C](https://es.wikipedia.org/wiki/C_(lenguaje_de_programaci%C3%B3n)) y [Perl](https://es.wikipedia.org/wiki/Perl).

***Ejemplo de Hola Mundo***

print("Hello, World!")

3***.-Microsoft SQL Server 2005 y posteriores.***

El lenguaje de desarrollo utilizado (por línea de comandos o mediante la interfaz gráfica de Management Studio) es Transact-SQL (TSQL), una implementación del estándar ANSI del lenguaje SQL, utilizado para manipular y recuperar datos (DML), crear tablas y definir relaciones entre ellas (DDL).

***Ejemplo de Hola mundo***

select 'Hello, World!';

4.-**Microsoft Visual Basic 6**

Visual Basic (VB) es un lenguaje de programación [dirigido por eventos](https://es.wikipedia.org/wiki/Programaci%C3%B3n_dirigida_por_eventos), desarrollado por Alan Cooper para Microsoft. Este lenguaje de programación es un dialecto de BASIC, con importantes agregados. Su primera versión fue presentada en 1991, con la intención de simplificar la programación utilizando un ambiente de desarrollo.

La última versión fue la 6, liberada en 1998, para la que Microsoft extendió el soporte hasta marzo de 2008.

***Ejemplo de Hola Mundo***

Option Explicit

Declare Function AllocConsole Lib "kernel32" () As Long

Declare Function FreeConsole Lib "kernel32" () As Long

Declare Function CloseHandle Lib "kernel32" (ByVal hObject As Long) As Long

Declare Function GetStdHandle Lib "kernel32" (ByVal nStdHandle As Long) As Long

Declare Function WriteConsole Lib "kernel32" Alias "WriteConsoleA" \_

(ByVal hConsoleOutput As Long, lpBuffer As Any, ByVal \_

nNumberOfCharsToWrite As Long, lpNumberOfCharsWritten As Long, \_

lpReserved As Any) As Long

Declare Function Sleep Lib "kernel32" (ByVal dwMilliseconds As Long) As Long

Private Sub Main()

'create a console instance

AllocConsole

'get handle of console output

Dim hOut As Long

hOut = GetStdHandle(-11&)

'output string to console output

Dim s As String

s = "Hello, World!" & vbCrLf

WriteConsole hOut, ByVal s, Len(s), vbNull, vbNull

'make a pause to look at the output

Sleep 2000

'close the handle and destroy the console

CloseHandle hOut

FreeConsole

End Sub

5.- Visual Prolog 7.2

Turbo Prolog es un completo sistema de desarrollo de software que incluye un compilador y un entorno de desarrollo integrado (IDE) para el lenguaje de programación PROLOG, desarrollado por Borland, dentro de la familia de productos Turbo (Turbo C, Turbo Pascal, Turbo Assembler, Turbo Basic, etc).

***Ejemplo de hola Mundo***

implement main

open core

constants

className = "main".

classVersion = "".

clauses

classInfo(className, classVersion).

clauses

run():-

console::init(),

stdio::write("Hello, World!"),

programControl::sleep(1000),

succeed().

end implement main

goal

mainExe::run(main::run).

6**.- Free Pascal, Turbo Pascal y posteriores**

Free Pascal es un compilador tanto de 32 como de 64 bits para múltiples arquitecturas de CPU y múltiples sistemas operativos. El compilador implementa los dialectos de Pascal de Borland Turbo Pascal y Delphi, además de algunas construcciones de MacPascal. Está disponible para la mayoría de los sistemas operativos populares.

Fecha de lanzamiento 1997, ultima versión estable 3.0.4 del 28 de noviembre de 2017.

***Ejemplo de Hola mundo***

program helloworld;

begin

writeln('Hello, World!');

end.

7.- **Perl 5.8.8 y rakudo-2010-8**

Es un lenguaje de programación diseñado por Larry Wall en 1987. Perl toma características del lenguaje C, del lenguaje interpretado bourne shell (sh), AWK, sed, Lisp y, en un grado inferior, de muchos otros lenguajes de programación.

Lanzado el 18 de diciembre de 1987 y diseñado por Larry Wall.

***Ejemplo de Hola mundo***

print "Hello, world!\n";

8**.-Borland C++**

C++Builder es un entorno de desarrollo rápido de aplicaciones en lenguaje C++ para Windows inicialmente propiedad de la empresa Borland, y actualmente de la empresa Embarcadero Technologies, quien compró de Borland la división Codegear encargada del producto. Codegear inicialmente se denominaba DTG='Developer Tools Group' hasta que adquirió el nombre oficial CodeGear previo a la venta.

C++Builder combina la biblioteca Visual Component Library y el IDE escrito en Delphi con un compilador de C++. El ciclo de lanzamiento es anual. Incluye herramientas que permiten desarrollo visual de arrastrar-y-soltar componentes sobre la aplicación e incorpora constructor de interfaz gráfica WYSIWYG en su IDE.

Desarrollado por Embarcadero Technologies, primera versión lanzada el año 1997 (versión 1.0)

***Ejemplo de Hola mundo***

Module Module1

Sub Main()

Console.WriteLine("Hello, World!")

End Sub

End Module

9**.- Oracle 10g SQL, Oracle 11g SQL**

Oracle surge en 1977 bajo el nombre de SDL (*Software Development Laboratories*).

En 1979, SDL cambia su nombre por Relational Software, Inc. (RSI).

La fundación de SDL fue motivada principalmente a partir de un estudio sobre los SGBD (Sistemas Gestores de Base de Datos) de George Koch. Computer World definió este estudio como uno de los más completos jamás escritos sobre bases de datos. Este artículo incluía una comparativa de productos que dirigía a Relational Software como el más completo desde el punto de vista técnico. Esto se debía a que usaba la filosofía de las bases de datos relacionales, algo que por aquella época era todavía desconocido.

***Ejemplo de Hola Mundo***

begin

dbms\_output.put\_line('Hello, World!');

end;

10**.-REXX**

REXX (REstructured eXtended eXecutor) es un lenguaje de programación desarrollado en IBM por Mike Cowlishaw del que existen numerosas implementaciones disponibles con código abierto. Es un lenguaje de programación estructurado de alto nivel diseñado para ser al mismo tiempo fácil de entender y fácil de leer. Hay intérpretes para REXX tanto comerciales como de código abierto para un amplio rango de plataformas y hay compiladores para los mainframes de IBM.

El nombre también se aplica a la familia de lenguajes relacionados con el lenguaje de programación REXX: el propio REXX (llamado en ese contexto REXX clásico o tradicional), Object REXX y NetREXX.

***Ejemplo de Hola mundo***

say 'Hello, World!'

11.-**Rhino**

Rhino comenzó a ser desarrollado en 1997 por Norris Boyd para Netscape Communications Corporation. En esos momentos la compañía necesitaba implementar soporte para JavaScript en su proyecto de suite de Internet Javagator.

En [1998](https://es.wikipedia.org/wiki/1998), cuando el proyecto Javagator fue finalmente cancelado, Rhino fue liberado bajo Licencia Pública de Netscape y entregado a la Fundación Mozilla para que continuara su desarrollo.

***Ejemplo de Hola Mundo***

if (typeof console === 'object') {

console.log('Hello, World!');

} else if (typeof document === 'object') {

document.write('Hello, World!');

} else {

print('Hello, World!');

}

12**.- Ruby**

es un lenguaje de programación interpretado, reflexivo y orientado a objetos, creado por el programador japonés Yukihiro "Matz" Matsumoto, quien comenzó a trabajar en Ruby en 1993, y lo presentó públicamente en 1995. Combina una sintaxis inspirada en Python y Perl con características de programación orientada a objetos similares a Smalltalk. Comparte también funcionalidad con otros lenguajes de programación como Lisp, Lua, Dylan y CLU. Ruby es un lenguaje de programación interpretado en una sola pasada y su implementación oficial es distribuida bajo una licencia de software libre.

***Ejemplo de Hola mundo***

puts "Hello, World!"

13**.-Groovy**

Es un lenguaje de programación orientado a objetos implementado sobre la plataforma Java. Tiene características similares a Python, Ruby, Perl y Smalltalk. La especificación JSR 241 se encarga de su estandarización para una futura inclusión como componente oficial de la plataforma Java.

Groovy usa una sintaxis muy parecida a Java, comparte el mismo modelo de objetos, de hilos y de seguridad. Desde Groovy se puede acceder directamente a todas las API existentes en Java. El bytecode generado en el proceso de compilación es totalmente compatible con el generado por el lenguaje Java para la Java Virtual Machine (JVM), por tanto puede usarse directamente en cualquier aplicación Java. Todo lo anterior unido a que la mayor parte de código escrito en Java es totalmente válido en Groovy hacen que este lenguaje sea de muy fácil adopción para programadores Java; la curva de aprendizaje se reduce mucho en comparación con otros lenguajes que generan bytecode para la JVM, tales como Jython o JRuby. Groovy puede usarse también de manera dinámica como un lenguaje de scripting.

Groovy 1.0 apareció el 2 de enero de 2007. Después de varias versiones beta y otras tantas candidatas a release, el 7 de diciembre de 2007 apareció la versión Groovy 1.1 que finalmente fue renombrada a Groovy 1.5 con el fin de notar la gran cantidad de cambios que ha sufrido con respecto a la versión 1.0. En diciembre de 2009 se publicó la versión 1.7.

***Ejemplo de Hola mundo***

println "Hello, World!"

14**.- Limbo**

Es un lenguaje de programación para escribir sistemas distribuidos. Además es el lenguaje usado para escribir aplicaciones para el sistema operativo [Inferno](https://es.wikipedia.org/wiki/Inferno_(sistema_operativo)). Fue diseñado en los Laboratorios Bell por Sean Dorward, [Phil Winterbottom](https://es.wikipedia.org/w/index.php?title=Phil_Winterbottom&action=edit&redlink=1) y [Rob Pike](https://es.wikipedia.org/wiki/Rob_Pike).

El compilador de Limbo genera código objeto independiente de la plataforma que luego es interpretado por la máquina virtual Dis o compilado antes de su ejecución para mejorar el rendimiento. Por ello, todas las aplicaciones hechas con Limbo son completamente portables en todas las plataformas soportadas por Inferno.

El enfoque de manejo de la concurrencia que tiene Limbo fue inspirada por el CSP de Hoare.

***Ejemplo de Hola Mundo***

implement Hello;

include "sys.m";

sys: Sys;

include "draw.m";

Hello: module {

init: fn(nil: ref Draw->Context, argv: list of string);

};

init(nil: ref Draw->Context, argv: list of string)

{

sys = load Sys Sys->PATH;

sys->print("Hello, World!\n");

}

15**.- Dart**

Dart (originalmente llamado Dash) es un lenguaje de programación de código abierto, desarrollado por Google. Fue revelado en la conferencia goto; en Aarhus, Dinamarca el 10 octubre de 2011.[3](https://es.wikipedia.org/wiki/Dart#cite_note-3)​ El objetivo de Dart no es reemplazar JavaScript como el principal lenguaje de programación web en los navegadores web, sino ofrecer una alternativa más moderna. El espíritu del lenguaje puede verse reflejado en las declaraciones de Lars Bak, ingeniero de software de Google, que define a Dart como un “lenguaje estructurado pero flexible para programación Web”.

***Ejemplo de Hola Mundo***

main() => print("Hello, World!");

**Ejemplo Final**

**Ejemplo de Codigo Ruby en el cual se declaran animales y sus características permitiendo asi identificar que animal es.**

class Mamifero

def respira

puts "inhala y exhala"

end

end

class Gato<Mamifero

def habla

puts "Meow"

end

end

misifus = Gato.new

misifus.respira

misifus.habla

class Ave

def acicala

puts "Estoy limpiando mis plumas."

end

def vuela

puts "Estoy volando."

end

end

class Pinguino<Ave

def vuela

puts "Lo siento, prefiero nadar."

end

end

class Aguila<Ave

end

puts "Pinguino"

p = Pinguino.new

p.acicala

p.vuela

puts "Aguila"

a = Aguila.new

a.acicala

a.vuela

begin

File.open('p014estructuras.rb', 'r') do |f1|

while linea = f1.gets

puts linea

end

end

# Crer un archivo y escribir en el

File.open('prueba.txt', 'w') do |f2|

f2.puts "Creado desde un programa Ruby!"

end

rescue Exception => msg

# mostar el mensaje de error generado por el sistema

puts msg

end

p014estructuras.rb

var = 5

if var > 4

puts "La variable es mayor que 4"

puts "Puedo tener muchas declaraciones aqui"

if var == 5

puts "Es posible tener if y else anidados"

else

puts "Too cool"

end

else

puts "La variable no es mayor que 4"

puts "Puedo tener muchas declaraciones aqui"

end

# Loops

var = 0

while var < 10

puts var.to\_s

var += 1

end