

UT1_4.- DESARROLLO DE SOFTWARE – LENGUAJES DE PROGRAMACIÓN

RESULTADOS DE APRENDIZAJE ASOCIADOS
1.- Reconoce los elementos y herramientas que intervienen en el desarrollo de un programa informático, analizando sus características y las fases en las que actúan hasta llegar a su puesta en funcionamiento.
CRITERIOS DE EVALUACIÓN
a) Se ha reconocido la relación de los programas con los componentes del sistema informático: memoria, procesador, periféricos, entre otros.
b) Se han identificado las fases de desarrollo de una aplicación informática.
c) Se han diferenciado los conceptos de código fuente, objeto y ejecutable.
d) Se han reconocido las características de la generación de código intermedio para su ejecución en máquinas virtuales.
e) Se han clasificado los lenguajes de programación.
f) Se ha evaluado la funcionalidad ofrecida por las herramientas utilizadas en programación.

UT1_4.- DESARROLLO DE SOFTWARE – LENGUAJES DE PROGRAMACIÓN

Índice de contenido

1.- Lenguajes de programación	3
1.1.- Tipos de lenguajes de programación.....	3
1.1.1- Clasificación según cercanía al hardware o al usuario.....	4
1.1.2.- Clasificación dependiendo de la forma en la que son ejecutados	6
1.2.- Características de los lenguajes más usados actualmente	8
2.- Referencias bibliográficas	10

1.- Lenguajes de programación

Un lenguaje de programación es un conjunto de reglas y símbolos que permiten a los programadores comunicarse con una computadora para escribir software y programas.



Estos lenguajes se utilizan para dar instrucciones precisas a una computadora sobre cómo realizar tareas específicas.

Un lenguaje de programación proporciona una forma estructurada y legible para que los humanos codifiquen las acciones que desean que la computadora realice.

En esencia, un lenguaje de programación **actúa como un intermediario entre el pensamiento humano y la ejecución de tareas por parte de la máquina.**

Los programadores escriben código utilizando un lenguaje de programación, y luego este código se traduce o **compila** en instrucciones que la computadora puede entender y ejecutar.

Todos los programas se desarrollan en algún lenguaje de programación.



Investiga...

¿Qué es ReactJS y AngularJs y para qué sirven?

1.1.- Tipos de lenguajes de programación

Los lenguajes de programación han evolucionado a lo largo del tiempo para aumentar su rendimiento y facilitar el trabajo a los programadores. Cada vez, existen lenguajes de programación que son más *user fiendly* y, por tanto, más fáciles de programar.



Pregunta

¿Qué significa la expresión *user friendly*?

1.1.1- Clasificación según cercanía al hardware o al usuario

A continuación, se muestran las características de los lenguajes de programación existentes en la actualidad:

– 1.- Lenguaje Máquina:

- Sus **instrucciones** son **complejas** y resultan **ininteligibles**, compuestas principalmente de combinaciones de unos y ceros.
- **No requiere traducción**, siendo el único lenguaje directamente comprensible por la computadora.
- Fue el primer lenguaje utilizado en la programación, requiriendo un profundo conocimiento del hardware por parte de los expertos de la época.
- Varía según el procesador, lo que significa que **las instrucciones no son transferibles entre diferentes equipos**.
- A excepción de casos particulares, en la actualidad, **prácticamente nadie programa en este lenguaje**.

2.- Lenguaje medio nivel o ensamblador

- Ante la complejidad y falta de portabilidad del lenguaje máquina, surgió el ensamblador como un reemplazo que **simplificó la tarea de programación**.
- Aunque sigue siendo cercano al hardware, en lugar de utilizar unos y ceros, se emplean **mnemotécnicos**, que son instrucciones más comprensibles para el programador y facilitan la comprensión de las acciones del programa.

- Este lenguaje **requiere ser compilado y traducido al lenguaje máquina** antes de poder ejecutarse.
- Se trabaja con registros del procesador y direcciones físicas.
- A pesar de ser más legible que el lenguaje máquina, todavía puede resultar **complicado de entender y programar**.
- En respuesta a la dificultad y falta de portabilidad del lenguaje máquina, el ensamblador se convirtió en una alternativa que simplificó significativamente la labor de programación.

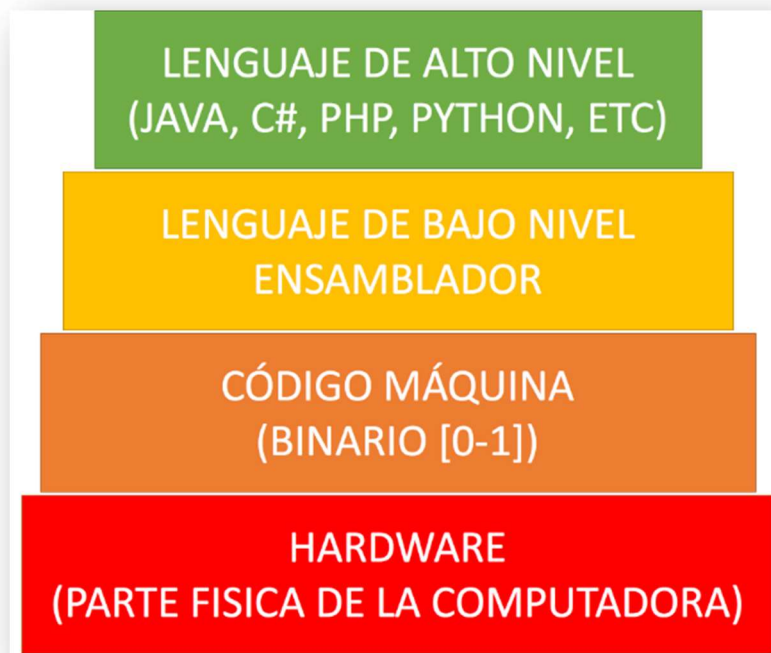
3.- Lenguaje de alto nivel

- La mayoría de los lenguajes de programación modernos pertenecen a esta categoría.
- Estos lenguajes ofrecen una forma de programar más intuitiva y sencilla, acercándose más al lenguaje humano que al lenguaje de máquina.
- Además, suelen incluir librerías y funciones predefinidas que ayudan a los programadores a resolver algunos de los problemas comunes que pueden surgir.
- En ocasiones, proporcionan frameworks que facilitan una programación eficiente y ágil.
- Estos lenguajes tienden a trabajar con un alto nivel de abstracción y están orientados a objetos, lo que simplifica la reutilización de componentes.



¿Qué es un *framework*?

Un framework es un conjunto esencial de conceptos, estructuras, funciones y componentes que proporciona una base sólida para desarrollar aplicaciones. Estos elementos son fundamentales y se pueden personalizar y adaptar según las necesidades específicas de la aplicación en desarrollo.



1.1.2.- Clasificación dependiendo de la forma en la que son ejecutados

Dependiendo de la forma en la que son ejecutados, puede haber otra clasificación de los lenguajes de programación:

— a) Lenguajes Compilados:

Descripción: Los lenguajes compilados **requieren un programa traductor llamado compilador para convertir el código fuente en código máquina**, específico para la arquitectura de la computadora. Este proceso se llama "compilación". Los programas compilados tienden a ejecutarse más rápido que los interpretados o virtuales debido a que el código ya está traducido a lenguaje máquina.

Ventajas: Mayor velocidad de ejecución y rendimiento optimizado para la máquina objetivo.

Ejemplos de lenguajes: C, C++, Rust.

— b) Lenguajes Interpretados:

Descripción: Los lenguajes interpretados **no generan código objeto**. En su lugar, utilizan un programa llamado intérprete que se carga en memoria y

se encarga de leer, interpretar y ejecutar cada instrucción del código fuente en tiempo real.

Ventajas: Portabilidad y facilidad de depuración, ya que los errores se detectan durante la ejecución.

Ejemplos de lenguajes: Python, JavaScript, Ruby.

c) Lenguajes Virtuales:

Descripción: Los lenguajes virtuales generan un código intermedio durante la compilación, que no es código máquina directamente ejecutable. Este código intermedio es luego interpretado por una máquina virtual en tiempo de ejecución. Esto hace que estos lenguajes sean altamente portables, ya que el mismo código intermedio puede ser interpretado en diferentes sistemas.

Ventajas: Portabilidad extrema y flexibilidad, ya que el mismo código puede ejecutarse en diversas plataformas.

Ejemplos de lenguajes: Java (que utiliza la máquina virtual Java - JVM), C# (con el uso del Common Language Runtime - CLR).

ACTIVIDAD 4.1

- a) Busca 5 ejemplos de *frameworks* y los lenguajes de programación con los que trabaja.
- b) Además, intenta dar al menos 1 ejemplo de aplicación real/famos realizadas con cada uno de dichos *frameworks*.

1.2.- Características de los lenguajes más usados actualmente

A continuación, se presentan una serie de lenguajes de programación, los más usados actualmente. Existen muchos más que no están en esta lista.

- **Python:**

Sintaxis legible: Python se destaca por su sintaxis clara y legible.

Ejemplo de "Hola Mundo" en Python:

```
print("Hola, mundo!")
```

- **JavaScript:**

Lenguaje de scripting en el navegador: JavaScript permite la interactividad en las páginas web y se ejecuta en navegadores.

Ejemplo de "Hola Mundo" en JavaScript:

```
console.log("Hola, mundo!");
```

- **Java:**

Portabilidad: Java es altamente portátil y se ejecuta en máquinas virtuales (JVM).

Ejemplo de "Hola Mundo" en Java:

```
public class HolaMundo {  
  
    public static void main(String[] args)  
    {  
        System.out.println("Hola, mundo!");  
    }  
}
```


- **C#:**

Desarrollo en Windows: C# es ampliamente utilizado para crear aplicaciones de escritorio y juegos en Windows.

Ejemplo de "Hola Mundo" en C#:

```
using System;

class Program {

    static void Main()
    {
        Console.WriteLine("Hola, mundo!");
    }
}
```

- **Ruby:**

Productividad: Ruby se enfoca en la productividad del desarrollador y tiene una sintaxis elegante.

Ejemplo de "Hola Mundo" en Ruby:

```
puts "Hola, mundo!"
```

- **PHP:**

Desarrollo web: PHP es ampliamente utilizado en el desarrollo web y se ejecuta en el lado del servidor.

Ejemplo de "Hola Mundo" en PHP:

```
<?php echo "Hola, mundo!"; ?>
```



¡Probemos los códigos anteriores!

Accede a la siguiente URL → <https://www.onlinegdb.com/>

ACTIVIDAD 4.2

- Añade características nuevas, que no se hayan visto en clase, para cada lenguaje de programación de la lista anterior.
- Busca más lenguajes de programación (que no aparezcan en la lista anterior) y cita sus características más importantes.
- Busca algún gráfico (lo más actual posible) que muestre el porcentaje de los lenguajes de programación más usados en el mundo. ¿cuál es el lenguaje más usado en la actualidad?

2.- Referencias bibliográficas

- ❖ Moreno Pérez, J.C. *Entornos de desarrollo*. Editorial Síntesis.
- ❖ Ramos Martín, A. & Ramos Martín, M.J. *Entornos de desarrollo*. Grupo editorial Garceta.