**Un traductor** es un programa que tiene como entrada un texto escrito en un lenguaje (lenguaje fuente) y como salida produce un texto escrito en un lenguaje (lenguaje objeto) que preserva el significado de origen.

Veremos dos tipos de traductores, los compiladores e intérpretes, que representan dos aproximaciones muy distintas a la tarea de permitir el funcionamiento de los programas escritos en un determinado lenguaje de programación de alto nivel.

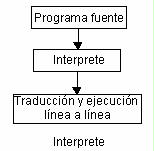
Los traductores se dividen en:

**Intérpretes**

Un intérprete es un traductor que toma un programa fuente, lo traduce y a continuación lo ejecuta (dicho programa por medio de la computadora desarrolla una tarea específica).

Un lenguaje que soporte un traductor de tipo intérprete se denomina lenguaje interpretado. BASIC es el modelo por excelencia interpretado.

Los programas fuente en BASIC se escriben con ayuda de un programa denominado editor que suele venir incorporado al programa intérprete.



**COMPILADORES**

Un compilador es un programa que traduce los programas fuente escritos en lenguajes de alto nivel a lenguaje máquina.

Los programas escritos en lenguajes de alto nivel (en el editor del lenguaje) se llaman programas fuente y el programa traducido programa objeto o código objeto. El compilador traduce (sentencia a sentencia) el programa fuente.

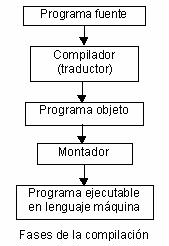
Lenguajes compiladores típicos son: PASCAL, COBOL, C..

Fases de la compilación

La compilación es el proceso de la traducción de programas fuente a programas objeto.

El programa objeto obtenido de la compilación no ha sido traducido normalmente a código máquina sino a ensamblador. Para conseguir el programa máquina real se debe utilizar un programa llamado montador o enlazador (linker). El proceso de montaje conduce a un programa en lenguaje máquina directamente ejecutable:

 La traducción por un compilador (la compilación) consta de dos etapas fundamentales, que a veces no están claramente diferenciadas a lo largo del proceso: la etapa de análisis del programa fuente y la etapa de síntesis del programa objeto. Cada una de estas etapas conlleva la realización de varias fases. El análisis del texto fuente implica la realización de un análisis del léxico, de la sintaxis y de la semántica. La síntesis del programa objeto conduce a la generación de código y su optimización.



RESUMEN

De acuerdo al tema de los traductores como se mencionan anteriormente los traductores son programas que tienen como entrada un texto escrito en un lenguaje los cuales los traductores tienen varios tipos de lenguaje con los cuales podemos interactuar a la hora de implementarlos ya sea que escojamos el que es un intérprete o un compilador.

El intérprete es un lenguaje que por sí solo arranca en el proceso y se ejecuta autónomamente

Por otro lado el intérprete tiene sus ventajas y desventajas de utilizarlo al igual que el compilador el cual es un lenguaje que traduce de manera formal los programas es más tardado pero hace casi igual la misma función que el intérprete.

**CUADRO COMPARATIVO**

|  |  |  |  |
| --- | --- | --- | --- |
| TRADUCTORES | CARACTERISTICAS | VENTAJAS | DESVENTAJAS |
| INTERPRETE | El **intérprete** ejecuta el programa directamente, traduciendo cada sentencia en una secuencia de una o más subrutinas ya compiladas en código máquina. Los términos lenguaje interpretado y lenguaje compilado​ no están bien definidos porque, en teoría, cualquier lenguaje de **programación** puede ser interpretado o compilado.  >usando un interprete, un solo arhivo fuente puede producir resultados iguales incluso en sistemas sumamente diferentes. | >Es mucho más cómodo para depurar y corregir errores, debido a su ejecución línea a línea.  >Se requiere el código fuente para ejecutar un programa.  >Si el programa es muy largo la ejecución es mucho más rápida, ya que no suele ejecutar todo el programa, y al analizar línea a línea, al finalizar el programa se cierra sin utilizar la parte del código que no se utilizó durante el ejercicio del programa. | >Cuando los programas son pequeños, la velocidad de ejecución es bastante lenta con respecto a los programas compilados.  >Es necesario tener el intérprete instalado en el equipo para que se ejecute el programa hecho en lenguaje de alto nivel.  >Por lo general, hay que configurar manualmente el PATH del intérprete en el sistema para que se ejecute desde cualquier localidad del usuario. |
| COMPILADORES | Un **compilador** genera un programa “stand-alone” es decir que tiene sentido y se puede ejecutar por sí solo, mientras que un programa interpretado siempre necesita su **intérprete** correspondiente para poder ejecutarse.  >usando un compilador, un solo archivo fuente puede producir resultados iguales solo si es compilado a distintos ejecutables específicos sistema. | **Autónomo y eficiente:** Una gran ventaja de los programas que se compilan es que son unidades autónomas listas para ser ejecutadas.  **Optimización del hardware:** Si bien estar restringido a un paquete de hardware específico tiene sus desventajas, compilar un programa también puede incrementar el desempeño de este último. Los usuarios pueden enviar opciones específicas a los compiladores, relacionadas con los detalles del hardware en el que se ejecutará el programa. | **Especifico a un hardware:** Dado que un compilador traduce el código fuente a un lenguaje máquina específico, los programas deben ser compilados específicamente para OS X, Windows o Linux, así como para arquitecturas de 32 o 64 bits.  **Tiempos de compilación:** Una de las desventajas de tener un compilador es que verdaderamente debe compilar código fuente. Mientras que los pequeños programas que muchos desarrolladores novatos codifican requieren cantidades triviales de tiempo para ser compilados, las grandes suites de aplicaciones pueden tardar cantidades significativas de tiempo en compilar |

**BIBLIOGRAFIAS**

<http://ing-radjchv-pmz-estl.blogspot.com/2009/09/14-traductor-y-su-estructura.html>

<http://proyectoinformaticaaadrar.blogspot.com/2012/05/2-traductores-compiladores-e.html>

<https://techlandia.com/ventajas-desventajas-compiladores-info_103356/>

\*Elabore una lista de al menos 5 lenguajes (cualquiera) de programación, identifique ambiente de desarrollo (¿qué es?, características más importantes), liste al menos 5 utilerías (bibliotecas, librerías), si las maneja ( ¿qué son?, ¿para qué sirven?) asociadas al lenguaje de programación , elabore cuadro con la información.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lenguaje De Programación | ¿Qué es? | Características | Librerías | ¿Para qué sirven? (librerías) |
| PHP | Lenguaje de programación interpretado, ya que puede utilizarse con scripts de forma estructurada o programación de objetos | 1.Lenguaje Multiplataforma  2.Orientado al desarrollo de aplicaciones web  3.Tiene manejo de excepciones  4.Biblioteca nativa de funciones sumamente amplia e incluida | Pchart | Genera gráficos de diversos tipos a partir de ficheros CSV, queries SQL o datos introducidos manualmente. |
| Upload | Ahorra subir y validar ficheros al servidor. Tan sólo unas cuantas |
| Gaufrette | facilita una capa de abstracción para trabajar con sistemas de ficheros de  distintos tipos |
| PHPMailer | enviar emails basada en el componente active server ASPMail |
| Underscore | Sirve para realizar cosas con estructuras de datos como pueden ser arrays arrays,funciones, objetos. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lenguaje De Programación | ¿Qué es? | Características | Librerías | Librerias |
| C++ | Lenguaje de programación orientada a objetos creado por Bjarne Stroustrup | Es potente en cuanto a lo que se refiere a creación de sistemas complejos un lenguaje muy robusto  Tiene un conjunto completo de instrucciones de control.  Permite la agrupación de instrucciones.  Permite la separación de un programa en módulos que admiten compilación independiente. | fstream: | Flujos hacia/desde ficheros. Permite la manipulación de archivos desde el programar, tanto leer como escribir en ellos. |
| iosfwd | Contiene declaraciones adelantadas de todas las plantillas de flujos y sus typedefs estándar. |
| iostream | Parte del a STL que contiene los algoritmos estándar, es quizá la más usada e importante |
| La biblioteca list | Parte de la STL relativa a contenedores tipo list; listas doblemente enlazadas |
| Biblioteca new | Manejo de memoria dinámica |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lenguaje | ¿Qué es? | Características | Librerías | ¿Para que sirven las librerías? |
| javascript | JavaScript es un lenguaje de programación que puede utilizarse para añadir interactividad a las páginas Web. | Imperativo y estructurado: compatible con gran parte de la estructura de programación de C  Dinámico  Evaluación en tiempo de ejecución: incluye la función eval que permite evaluar expresiones como expresadas como cadenas en tiempo de ejecución. | JQUERY | Es un archivo que contiene varias instrucciones, que permiten que el navegador ejecute muchas funcionalidades adicionales(cambiar tamaño, forma, estilo de cualquier elemento de página) |
| MODERNIZR | Modernizr es una librería que detecta funcionalidades HTML5 y CSS3 en los navegadores weLibrería |
| ASP.NET AJAX | JavaScript basada en ASP.NET, desarrollada por Microsoft. La usan todos los sitios de esta empresa y varios sitios de internet.b. |
| PROTOTYPE | Es un JavaScript framework, enfocado principalmente en creación de contenido dinámico y actualizaciones asíncronas en páginas y aplicaciones web. |
| SCRIPT.ACULO.US | Agrega efectos visuales ,controles, y utilidades |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lenguaje | ¿Qué es? | Características | Librerías | ¿Para que sirven las librerías? |
| java | Es un lenguaje orientado a objetos, de una plataforma independiente, fue desarrollado por la compañía SUN Microsystems ahora es propietario ORACLE. | * Orientado a objetos * Distribuido y dinámico * Robusto * Seguro * Multitarea * Portable | * java.lang | Aquí están las declaraciones de los objetos, clases, threads, excepciones, wrappers de los tipos de datos primitivos y otras clases fundamentales. |
| java.io | Soporta flujos de entrada y salida java. |
| * Java.net | Soporta facilidades de red (URL, sockets TCP, sockets UDP, direcciones IP, conversiones binarias a texto). |
| * Java.awt | proporciona un Abstract Window Toolkit para programación gráfica de (usuario) |
| * Java.math | Proporciona cálculos en entero grande y real grande. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lenguaje | ¿Qué es? | Características | Librerías | ¿Para que sirven las librerías? |
| Phyton | Es un lenguaje de programación de alto nivel , que permite procesar fácilmente todo tipo de estructuras de datos , tanto numéricos como de texto | Es un lenguaje interpretado, no compilado  Es multiplataforma, Lo cual es ventajoso para hacer ejecutable en cualquier sistema operativo  Es un lenguaje de programación multiparadigma (orientada a objetos, estructurada, imperativa y funcional)  El formato del código es estructural | PYGAME | Es una librería de código abierto la cual nos permite crear aplicaciones multimedia |
| COLLECTIONS | Esta librería nos permite trabajar con listas, tuplas, diccionarios, nos permite ordenar objetos y concatenar |
| RANDOM | Esta librería genera números pseudoalatorios |
| PYGTK | Esta librería nos permite crear programas de interfaz grafica |
| PILLOW | Una herramienta para manipular imágenes. Es un fork de PIL y es más fácil de usar, bastante útil si trabajas con imágenes frecuentemente. |

\*Identifique en su equipo de cómputo los lenguajes de programación instalados (inclúyalo en el reporte) y su tipo de traductor correspondiente.

|  |  |
| --- | --- |
| LENGUAJE DE PROGRAMACION | TRADUCTOR |
| * BASIC * VISUAL STUDIO | Interprete  Interprete |

**BIBLIOGRAFIAS**

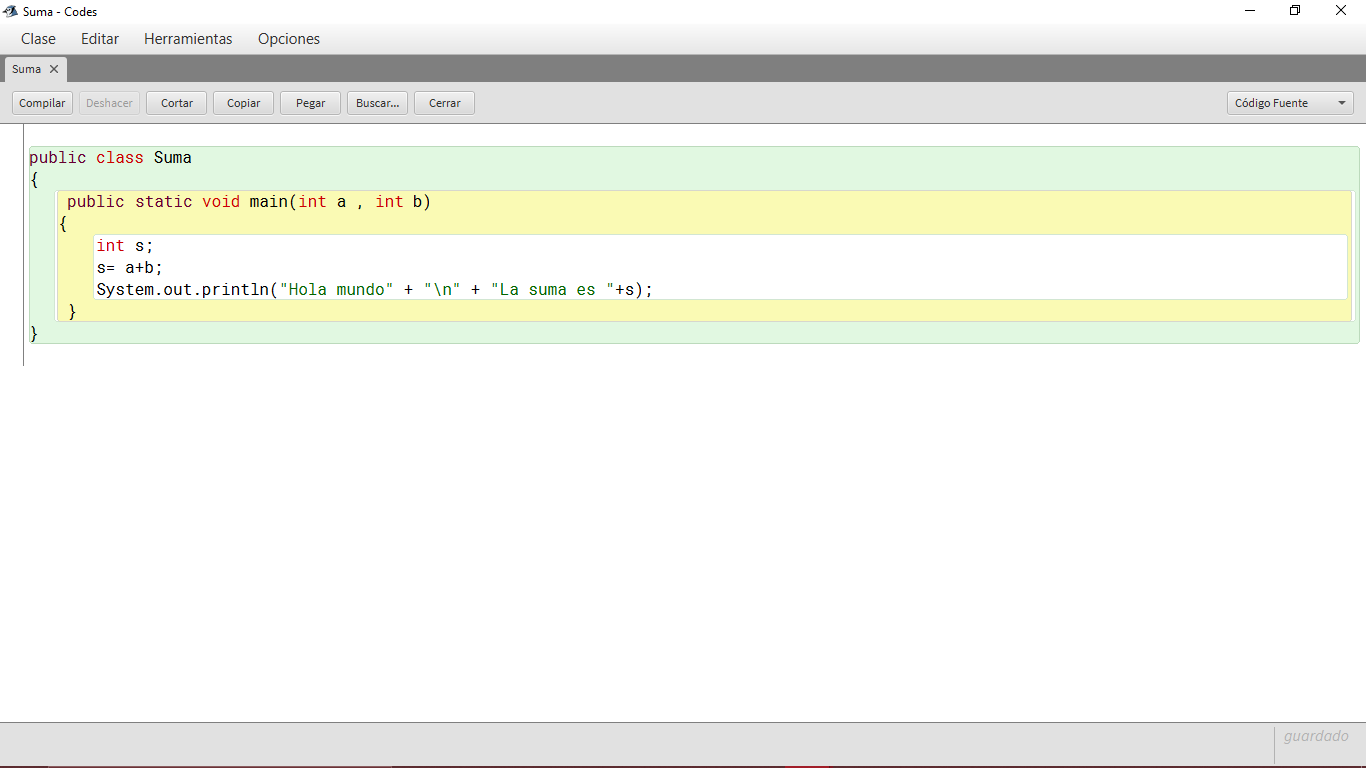
<https://www.mundojava.net/caracteristicas-del-lenguaje.html?Pg=java_inicial_4_1.html>

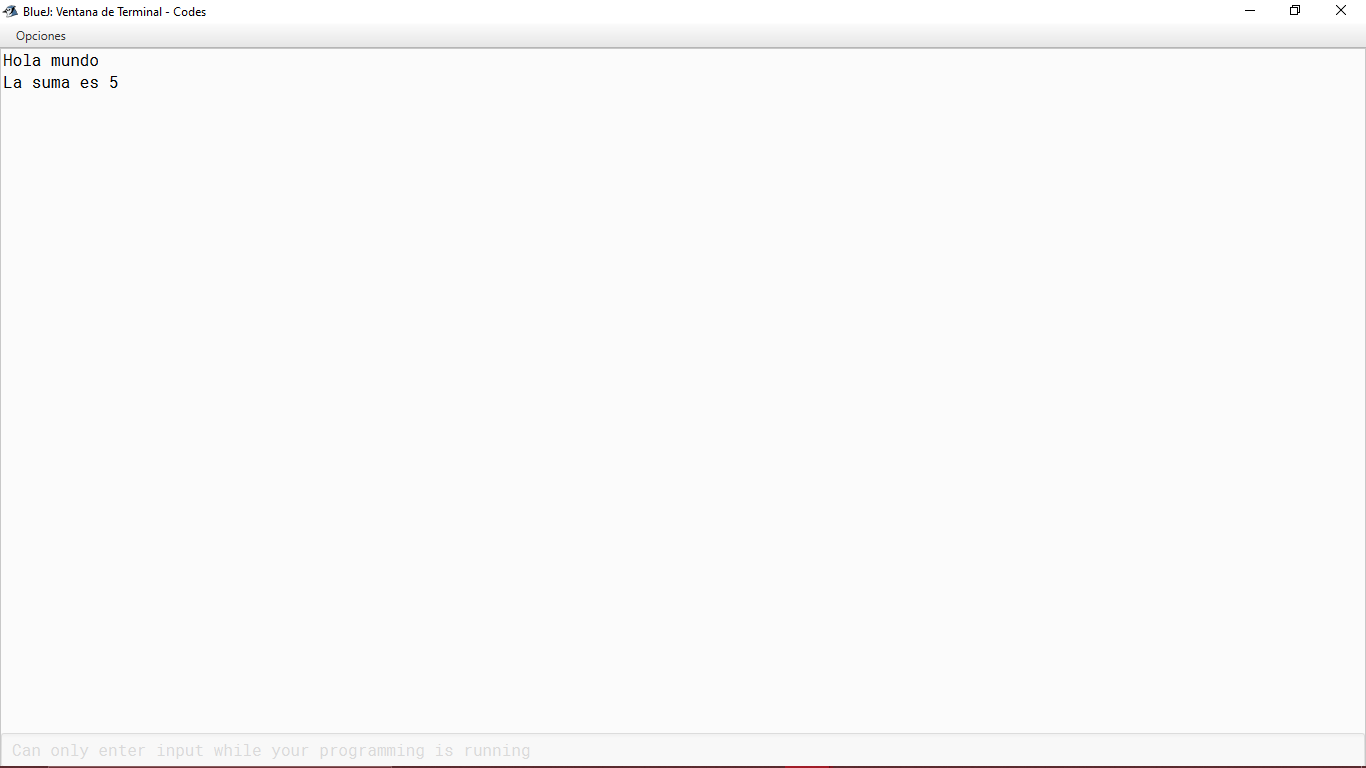
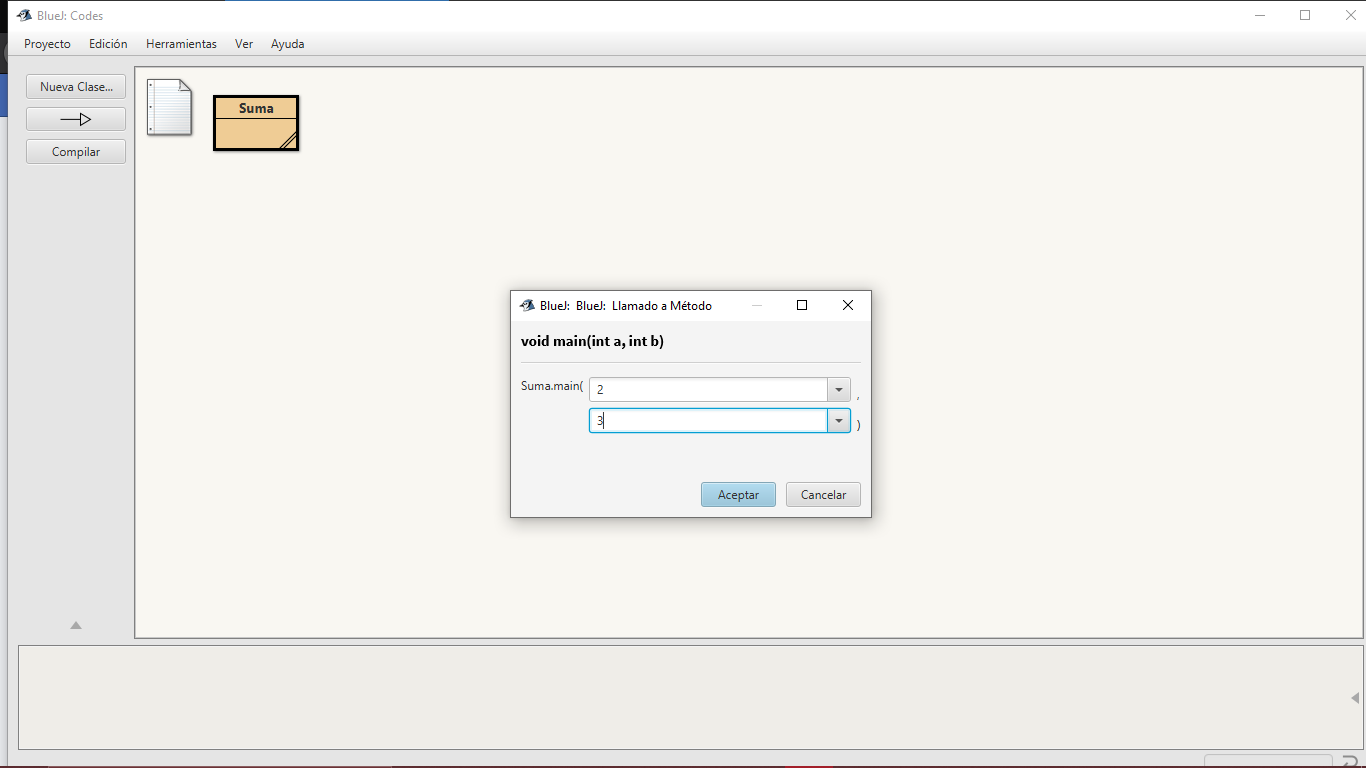
<https://blog.buhoos.com/lenguajes-de-programacion-cuadro-comparativo/>

<https://sites.google.com/site/musitioswebprogramacionenc/caracteristicas-de-c>

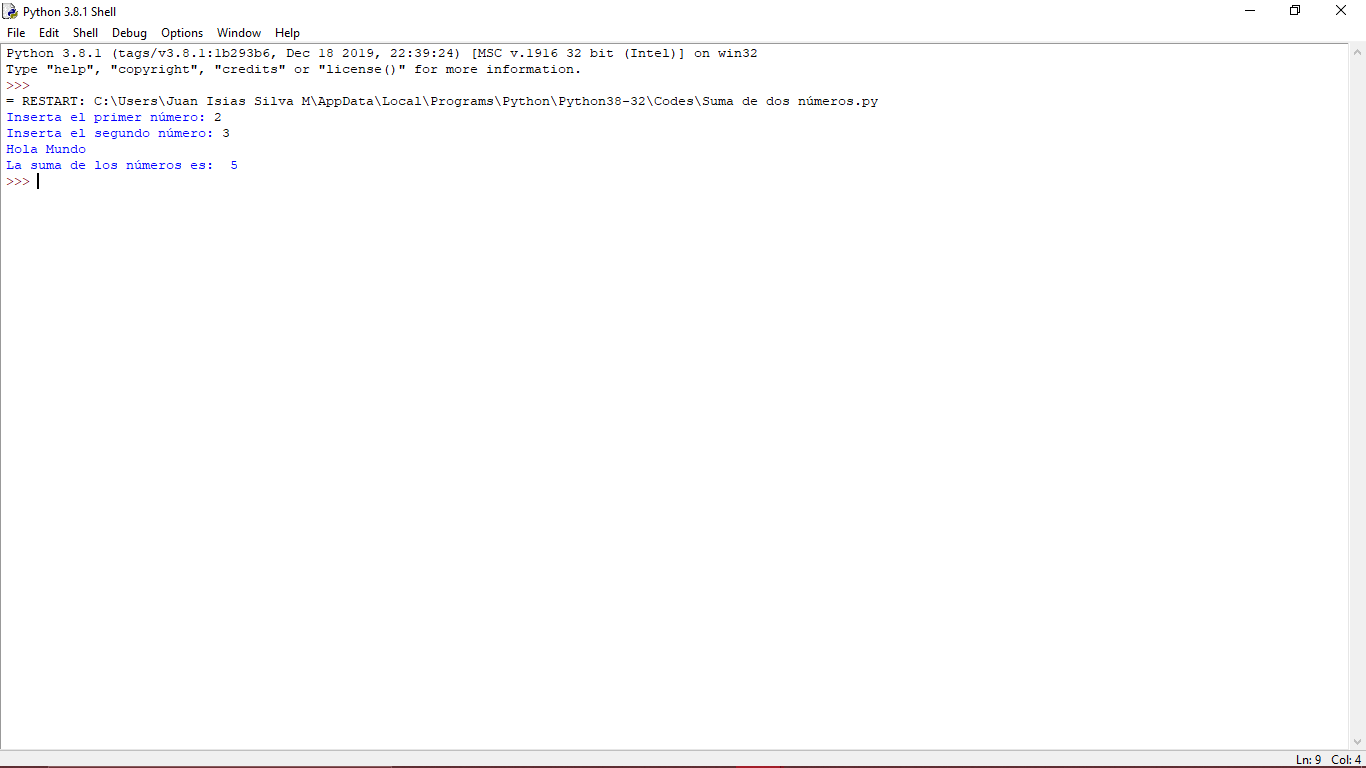
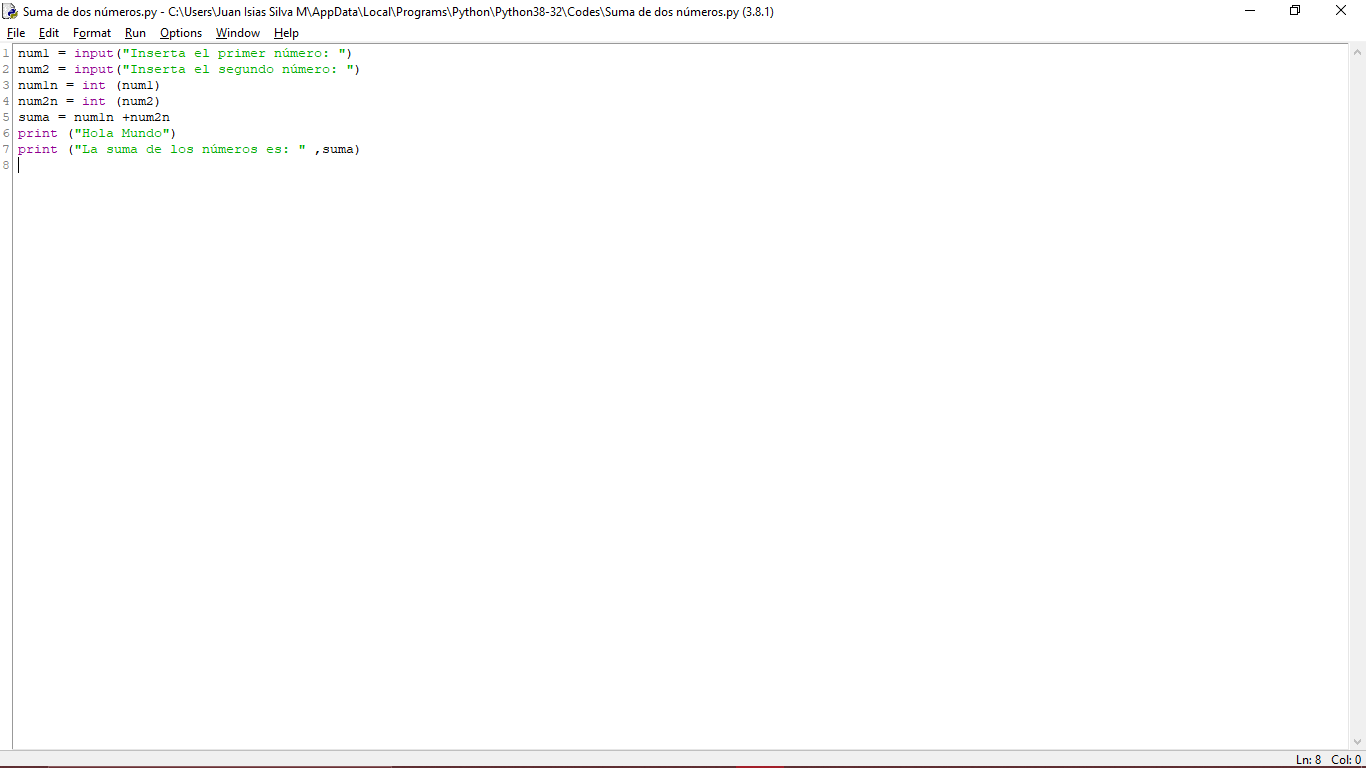
**¿QUÉ ES UN COMPONENTE LÉXICO?**

Es la secuencia lógica y coherente de caracteres relativo a una categoría: identificador, palabra reservada, literales (cadena/numérica), operador o carácter de puntuación, además de que un componente léxico puede tener uno o varios lexemas.

  
**JAVA**



|  |  |
| --- | --- |
| Componente léxico | JAVA |
| Palabra reservada | public static void main, int, System.out.println, \n |
| Carácter de puntuación | ; ( ) “ ” { } |
| Operador | = + |
| Identificador | s a b |
| Literal | Hola Mundo La suma es |

**Python**

|  |  |
| --- | --- |
| Componente léxico | Python |
| Carácter de puntuación | () “ ” |
| Palabra reservada | input, int, print |
| Operador | = + |
| Identificador | num1, num2, num1n, num2n, suma |
| Literal | Inserta el primer número, Inserta el segundo número, Hola Mundo, La suma de los números es: |

**BIBLIOGRAFIAS**

http://teleformacion.edu.aytolacoruna.es/PASCAL/document/intro.htm

https://sites.google.com/site/teoriadelenguajesformaless/1-5-herramientas-computacionales

http://acodigo.blogspot.com/2014/02/analizador-lexico.html

http://www.suigeneris.org/UCABTI/Definiciones%20Basicas.html

**CONCLUSIÓN PERSONAL**

Los lenguajes de programación no han dejado de evolucionar y nunca dejarán de hacerlo, con el avance del tiempo existirán más lenguajes que podrían resultar menos complejos y por lo tanto más traductores que serán de capaces de traducir lenguajes que utilizamos en la actualidad convirtiéndose en “obsoletos”, sin embargo es importante el analizar cómo han cambiado los componentes léxicos de diferentes lenguajes ya que así tendremos una idea de su parentesco entre sí.

**CONCLUSION PERSONAL (Paty)**

Podemos decir que los traductores en la identificación de los componentes léxicos en los lenguajes de programación, son muy útiles en la vida cotidiana, ya que sin un traductor no podríamos utilizar un lenguaje para programar alguna instrucción y desarrollarla, es muy importante entender estos conceptos ya que más adelante los utilizares conforme vayan avanzando las clases.

**DESCRIPCION**

**Patricia Fernanda Castellanos Ruiz (Traductores)**

Por mi parte me toco la primera parte de los traductores, que son, como se clasifican, sus características que tiene, su funcionamiento, al igual que hacer un cuadro comparativo de sus ventajas y desventajas y sus características en general, y por ultimo hacer un resumen respecto a la información de los traductores.

**Mayra Esmeralda Castillo Montenegro (Lenguajes de programación)**

En acuerdo con mis compañeros me toco realizar la lista de los cinco lenguajes de programación en forma de cuadro agregando su ambiente de desarrollo como: que es, características, librerías, para que sirven, así como también agregar los lenguajes de programación instalados e identificar el tipo de traductor

**Jesús Eduardo Silva Vázquez**

**Desarrollo de la práctica (Componentes léxicos)**

Para poder realizar la parte práctica se tuvo que descargar dos programas que manejen tipos de lenguajes de programación diferentes para así poder comparar los diferentes componentes léxicos que ofrece cada lenguaje, para eso se descargó un programa orientado a Java y otro en Python.

Empezando por la programación en Java utilizando BlueJ como entorno de programación se realizó el código para que mostrara el mensaje de “Hola Mundo” y a su vez una suma de dos números, la codificación fue muy básica porque solo era cuestión de imprimir en la pantalla el mensaje y realizar una operación de variables de números enteros con operadores lógicos. A diferencia de la implementada en Java, en el lenguaje de programación Python al no depender mucho de métodos predefinidos resultó mucho más sencillo de realizar.

Con la programación utilizando ambos lenguajes se pudieron realizar las operaciones requeridas para la práctica y por lo tanto capturar al menos 5 componentes léxicos de ambos lenguajes.