**Tecnológico Nacional de México**

Campus Veracruz

**Carrera:**

Ingeniería en sistemas computacionales

**Materia:**

Lenguajes y autómatas I

**Semestre:** VI

**Alumno:** Acosta Ruiz Honorio

**Número de control:** 20021188

**Docente:**

Ofelia Gutiérrez Giraldi

**Tabla de contenido**

[ELL (EasyLearningLenguage) 2](#_Toc134187571)

[Propósito 2](#_Toc134187572)

[Origen del nombre ELL: EasyLearningLenguage 2](#_Toc134187573)

[Manual de Usuario 3](#_Toc134187574)

[Conceptos Básicos 3](#_Toc134187575)

[Tipo de Datos 3](#_Toc134187576)

[Palabras Clave 4](#_Toc134187577)

[Operadores 5](#_Toc134187578)

[Primeros pasos 5](#_Toc134187579)

[Hola mundo en ELL 5](#_Toc134187580)

[Variables en ELL 6](#_Toc134187581)

[Manual del Sistema 7](#_Toc134187582)

[Requisitos del sistema 7](#_Toc134187583)

[Descripción del sistema 7](#_Toc134187584)

[Tabla de valores 7](#_Toc134187585)

ELL (EasyLearningLenguage)

# Propósito

EasyLearningLenguage (ELL) se propone a ser un lenguaje de programación tipado y de propósito general que será diseñado para permitir el desarrollo de aplicaciones básicas.

Se centrará en la lógica de programación básica, lo que lo hará adecuado para la enseñanza de conceptos fundamentales como variables, operaciones, condicionales y ciclos. Aunque ELL no tendrá una amplia gama de características avanzadas, será diseñado para ser fácilmente escalable a medida que los usuarios adquieren más habilidades en programación.

ELL será creado con el propósito de ser un lenguaje de programación en español de nivel básico.

# Origen del nombre ELL: EasyLearningLenguage

El nombre de nuestro lenguaje es ELL, debido a que es un lenguaje de programación orientado a facilitar el aprendizaje de la lógica básica de programación para programadores novatos.



Figura 1 - Icono del lenguaje

El nombre ELL proviene de las siglas EasyLearningLanguage, lo que en español significa Lenguaje Fácil de Aprender.

Manual de Usuario

El respectivo manual busca proporcionar una guía clara y completa para nuestro lenguaje que permita a los programadores utilizar el lenguaje de manera efectiva, donde como objetivos tiene los siguientes:

* Facilitar el aprendizaje del lenguaje:

El manual proporciona una introducción clara al lenguaje de programación, explicando los conceptos básicos, sintaxis y características únicas del lenguaje.

* Ayudar a los programadores a detectar y corregir errores:

Incluye información sobre los errores que pueden ocurrir durante la programación con el lenguaje, incluyendo como corregir estos errores.

* Proporcionar información de referencia:

Es una fuente completa de información de referencia para el lenguaje, que incluye una lista detallada de las palabras clave, operadores y tipos de datos; permitiendo a los programadores buscar rápidamente información específica cuando la necesita.

Conceptos Básicos

# Tipo de Datos

El lenguaje ELL tiene los siguientes tipos de datos básicos:

* **Entero**: Tipo de dato que representa un numero entero.
* **Flotante**: Tipo de dato que representa un numero en decimal.
* **Cadena**: Tipo de dato que representa un texto.
* **Carácter**: Tipo de datos que representa un símbolo.
* **Booleano**: Tipo de dato que representa aquellos que tienen un valor de verdadero o falso.

# Palabras Clave

Las palabras clave del lenguaje ELL son:

|  |  |
| --- | --- |
| **Palaba clave** | **Descripción** |
| Inicio | Indica el arranque del programa. |
| Fin | Indica el cierre del programa. |
| Establecer | Indica la definición de una variable. |
| Interpretar | Indica la lectura de una variable mediante el tipo de dato. |
| Escribir | Indica la escritura de una variable o literal en terminal. |
| Falso | Indica un valor lógico falso. |
| Verdadero | Indica un valor lógico verdadero. |
| Para | Indica el inicio de un ciclo Para. |
| Hasta que | Indica la condición a evaluar del ciclo Para. |
| Con incremento | Indica el incremento del ciclo Para (el incremento por defecto es 1). |
| FinPara | Indica el final del ciclo Para. |
| Mientras | Indica el inicio del ciclo Mientras. |
| FinMientras | Indica el final del ciclo Mientras. |
| Repetir | Indica el inicio del ciclo Repetir. |
| Finaliza cuando | Indica la condición a evaluar del ciclo Repetir. |
| Si | Indica el inicio del condicional Si. |
| Entonces | Indica el cuerpo del condicional Si. |
| Sino | Indica el caso contrario del condicional Si. |
| FinSi | Indica el final del condicional Si. |
| Conforme | Indica el inicio del condicional Conforme. |
| Hacer | Indica el cuerpo del condicional Conforme. |
| Caso | Indica un caso del condicional Conforme. |
| En otro caso | Indica el caso por defecto del condicional Conforme. |
| FinConforme | Indica el fin del condicional Conforme. |

Tabla 1 - Palabras clave

# Operadores

Los operadores con los que cuenta el lenguaje ELL son:

|  |  |
| --- | --- |
| **Operador** | **Descripción** |
| = | Operador que indica asignación. |
| + | Operador que indica una suma. |
| - | Operador que indica una resta. |
| \* | Operador que indica una multiplicación. |
| / | Operador que indica una división. |
| % | Operador que indica al módulo. |
| && | Operador lógico AND. |
| || | Operador lógico OR. |
| ! | Operador lógico NOT. |
| == | Operador de comparación Igual que. |
| != | Operador de comparación diferente. |
| > | Operador de comparación mayor que. |
| < | Operador de comparación menor que. |
| >= | Operador de comparación mayor o igual que. |
| <= | Operador de comparación menor o igual que. |

Tabla 2 - Operadores

Primeros pasos

# Hola mundo en ELL

Al igual que cualquier otro lenguaje de programación, la primera aplicación creada en ELL es el conocido programa “Hola Mundo”.

El código de esta primera aplicación en ELL es el siguiente:

Inicio

Escribir “Hola, mundo”;

Fin

Figura 2 - Ejemplo con: Hola, mundo.

# Variables en ELL

Las variables son un espacio de memoria para almacenar un dato. Para definir una variable seguimos esta estructura:

Inicio

Establecer [Tipo\_De\_Dato][Nombre\_Variable];

Fin

Figura 3 – Estructura de cómo crear variables.

Algunos ejemplos de variables serían:

Inicio

Establecer Entero edad; # Esta en una variable entera

Establecer Flotante estatura; # Esta en una variable flotante

Establecer Cadena nombre; # Esta es una variable cadena

Establecer Carácter sexo; # Esta es una variable tipo carácter

Establecer Booleano estaVivo; # Esta es una variable tipo booleana

Fin

Figura 4 – Ejemplo de creación de variables.

Manual del Sistema

# Requisitos del sistema

* Tener Java instalado.
* Tener JavaCC instalado.

# Descripción del sistema

EC (EasyCompiler) es un compilador de código abierto que admite el lenguaje de programación ELL (EasyLearningLenguage). El compilador se ejecuta en cualquier plataforma que tenga instalada Java.

# Tabla de valores

|  |  |  |
| --- | --- | --- |
| **Valor** | **Tipo** | **Expresión regular** |
| Inicio | PR\_Arranque\_Programa |  |
| Fin | PR\_Cierre\_Programa |  |
| # | Simbolo\_Comentario |  |
| edad | Variable |  |
| Establecer | PR\_Definicion\_Variable |  |
| Entero | PR\_Tipo\_De\_Dato |  |
| Flotante | PR\_Tipo\_De\_Dato |  |
| Cadena | PR\_Tipo\_De\_Dato |  |
| Carácter | PR\_Tipo\_De\_Dato |  |
| Booleano | PR\_Tipo\_De\_Dato |  |
| ; | Delimitador |  |
| Interpretar | PR\_Lectura |  |
| Escribir | PR\_Escritura |  |
| 123 | Numero\_Entero |  |
| 123.123 | Numero\_Flotante |  |
| "Cadena" | Cadena\_Texto |  |
| 'C' | Caracter\_Texto |  |
| = | Operador\_Asignacion |  |
| + | Operador\_Suma |  |
| - | Operador\_Resta |  |
| \* | Operador\_Mult |  |
| / | Operador\_Div |  |
| % | Operador\_Mod |  |
| Falso | PR\_Booleano\_Falso |  |
| Verdadero | PR\_Booleano\_Verdadero |  |
| && | Operador\_Logico\_AND |  |
| || | Operador\_Logico\_OR |  |
| ! | Operador\_Logico\_NOT |  |
| == | Operador\_igualQue |  |
| != | Operador\_diferente |  |
| > | Operador\_mayorQue |  |
| < | Operador\_menorQue |  |
| >= | Operador\_mayorIgualQue |  |
| <= | Operador\_menorIgualQue |  |
| Para | PR\_Ciclo\_Para |  |
| Hasta que | PR\_Condicion\_Ciclo\_Para |  |
| Con incremento | PR\_Incremento\_Ciclo\_Para |  |
| FinPara | PR\_Fin\_Ciclo\_Para |  |
| Mientras | PR\_Ciclo\_Mientras |  |
| FinMientras | PR\_Fin\_Ciclo\_Mientras |  |
| Repetir | PR\_Ciclo\_Repetir |  |
| Finaliza cuando | PR\_Condicion\_Ciclo\_Repetir |  |
| Si | PR\_Condicional\_Si |  |
| Entonces | PR\_Entonces |  |
| Sino | PR\_Conficional\_Sino |  |
| FinSi | PR\_Fin\_Condicional\_Si |  |
| Conforme | PR\_Condicional\_Conforme |  |
| Hacer | PR\_Hacer |  |
| Caso | PR\_Conforme\_Caso |  |
| : | Operador\_Dos\_Puntos |  |
| En otro caso | PR\_Conforme\_Caso\_Predeterminado |  |
| FinConforme | PR\_Fin\_Condicional\_Conforme |  |
| ( | Parentesis\_Abierto |  |
| ) | Parentesis\_Cerrado |  |

Tabla 3 – Tabla de valores