

5.1 Requerimiento de la Gramática

Introducción

El lenguaje de programación HTML-es (HyperText Markup Language) es un lenguaje de marcado utilizado para estructurar y presentar contenido en la web. HTML-es es una versión en español del estándar HTML, y permite a los desarrolladores web definir la estructura de las páginas web mediante una serie de elementos y atributos. Estos elementos incluyen títulos, párrafos, enlaces, imágenes, tablas y otros componentes que forman el contenido visual y estructural de una página web. HTML-es es esencial para la creación de sitios web, ya que proporciona el esqueleto sobre el cual se aplica el diseño y la funcionalidad adicional a través de CSS y JavaScript.

Propósito del analizador sintáctico y su rol en el proceso de compilación o interpretación del lenguaje

El analizador sintáctico, también conocido como parser, es una parte crucial en el proceso de compilación o interpretación de cualquier lenguaje de programación, incluido HTML-es. Su propósito principal es tomar la secuencia de tokens generada por el análisis léxico y construir una estructura de datos, generalmente un árbol de sintaxis abstracta (AST, por sus siglas en inglés), que representa la estructura gramatical del código fuente de acuerdo con las reglas del lenguaje.

Funciones clave del analizador sintáctico:

1. **Verificación de la estructura gramatical:** El parser asegura que el código fuente sigue las reglas sintácticas del lenguaje. Por ejemplo, en HTML-es, verifica que los elementos estén correctamente anidados y que los atributos tengan valores válidos.
2. **Construcción del árbol de sintaxis abstracta (AST):** El AST es una representación jerárquica del código fuente que refleja su estructura lógica. Este árbol es utilizado por etapas posteriores del compilador o intérprete para realizar optimizaciones, generación de código intermedio, y ejecución del código.
3. **Detección de errores:** Durante el proceso de análisis sintáctico, se detectan y reportan errores gramaticales, lo que ayuda a los desarrolladores a identificar y corregir problemas en el código fuente.

Rol en el proceso de compilación o interpretación:

En el contexto de un compilador, el analizador sintáctico es la etapa que sigue al análisis léxico. Tras convertir el código fuente en tokens, el parser organiza estos tokens en una estructura que puede ser fácilmente manipulada por las etapas siguientes del compilador, como la optimización y la generación de código máquina.

En un intérprete, el análisis sintáctico también sigue al análisis léxico. Una vez construido el AST, el intérprete puede recorrer este árbol para ejecutar directamente las instrucciones, lo que permite la ejecución inmediata del código sin necesidad de convertirlo a un lenguaje intermedio o máquina.

Lista Completa de los Símbolos Terminales del Lenguaje HTML-es

Palabras clave

- html
- head
- title
- body
- h1, h2, h3, h4, h5, h6
- p
- a
- div
- span
- ul, ol, li
- table, tr, td, th
- form, input, button, label
- img
- link
- script
- style
- meta
- nav
- header
- footer
- section
- article
- aside
- main
- figure
- figcaption
- video
- audio

- source

Operadores

HTML en sí no incluye operadores como un lenguaje de programación tradicional, pero si se considera su uso dentro de scripts, como JavaScript, los operadores comunes serían:

- +, -, *, /, %
- ==, !=, ===, !==
- <, <=, >, >=
- &&, ||, !
- =, +=, -=, *=, /=

Delimitadores

- <, >
- </, />
- {, }
- [,]
- (,)
- ;
- :
- ,
- ", '

Literales

- **Números:** Enteros y decimales: 123, 45.67
- **Cadenas:** Cadenas de texto: "Hola, ¡mundo!", 'Este es un texto'

Identificadores

- **Variables y nombres de funciones en scripts:** miVariable, miFuncion
- **Nombres de etiquetas personalizadas:** mi-componente
- **Atributos y valores de atributos:** class, id, href, src

Símbolos No Terminales de HTML-es

No Terminales:

- **Document:** Representa un documento HTML completo.
- **Element:** Representa un elemento HTML.
- **Tag:** Representa una etiqueta HTML.
- **AttrList:** Lista de atributos de un elemento.
- **Attr:** Atributo individual de un elemento.
- **AttrValue:** Valor de un atributo.
- **Content:** Contenido de un elemento, que puede incluir texto y otros elementos.
- **Text:** Texto dentro de un elemento.
- **Script:** Código JavaScript embebido.
- **Style:** CSS embebido.
- **Comment:** Comentarios dentro del HTML.
- **Expr:** Expresión en JavaScript dentro de una etiqueta <script>.
- **Stmnt:** Declaración en JavaScript dentro de una etiqueta <script>.
- **StmntList:** Lista de declaraciones en JavaScript dentro de una etiqueta <script>.
- **FormElement:** Elemento de formulario (e.g., input, button, select).
- **List:** Lista de elementos HTML (ul, ol).
- **ListItem:** Elemento de una lista (li).
- **Table:** Tabla HTML.
- **TableRow:** Fila de una tabla (tr).
- **TableCell:** Celda de una tabla (td, th).

Producciones para la Gramática de HTML-es

A continuación, se especifican las producciones que describen cómo los símbolos no terminales de HTML-es se descomponen en otros no terminales y terminales

Documento → HTML

(Un documento HTML se compone de un elemento HTML.)

HTML → <html> Cabeza Cuerpo </html>

(Un elemento HTML se compone de las etiquetas <html> y </html>, con un Cabeza y un Cuerpo en su interior.)

Cabeza → <head> ElementosCabeza </head>

(La sección de la cabeza se compone de las etiquetas <head> y </head>, y puede contener varios ElementosCabeza.)

Cuerpo → <body> ElementosCuerpo </body>

(La sección del cuerpo se compone de las etiquetas <body> y </body>, y puede contener varios ElementosCuerpo.)

ElementosCabeza → ElementoCabeza ElementosCabeza | ε

(La sección de la cabeza puede contener uno o más ElementoCabeza, o puede estar vacía.)

ElementosCuerpo → ElementoCuerpo ElementosCuerpo | ε

(La sección del cuerpo puede contener uno o más ElementoCuerpo, o puede estar vacía.)

ElementoCabeza → Meta | Título | Link

(Un ElementoCabeza puede ser una metaetiqueta, un título o un enlace.)

ElementoCuerpo → Parrafo | Enlace | Imagen

(Un ElementoCuerpo puede ser un párrafo, un enlace o una imagen.)

Meta → <meta>

(Una metaetiqueta se representa con la etiqueta <meta>.)

Título → <title> Texto </title>

(Un título se compone de las etiquetas <title> y </title>, con Texto en su interior.)

Link → <link>

(Un enlace se representa con la etiqueta <link>.)

Parrafo → <p> Texto </p>

(Un párrafo se compone de las etiquetas <p> y </p>, con Texto en su interior.)

Enlace → <a> Texto

(Un enlace se compone de las etiquetas <a> y , con Texto en su interior.)

Imagen →

(Una imagen se representa con la etiqueta .)

Texto → Caracter Texto | ε

(El texto se compone de uno o más Caracter, o puede estar vacío.)

Caracter → cualquier carácter

(Un carácter puede ser cualquier símbolo que no sea una etiqueta HTML.)

Notas:

Símbolos no terminales: Documento, HTML, Cabeza, Cuerpo, ElementosCabeza, ElementosCuerpo, ElementoCabeza, ElementoCuerpo, Meta, Título, Link, Párrafo, Enlace, Imagen, Texto, Caracter.

Símbolos terminales: Las etiquetas HTML (<html>, </html>, <head>, </head>, <body>, </body>, <meta>, <title>, </title>, <link>, <p>, </p>, <a>, ,) y los caracteres individuales dentro del texto.

Explicación de las Producciones:

La producción 1 indica que un documento se compone de un elemento HTML.

La producción 2 indica que un elemento HTML incluye una cabeza y un cuerpo.

La producción 3 define la estructura de la cabeza del documento.

La producción 4 define la estructura del cuerpo del documento.

La producción 5 indica que la cabeza puede contener múltiples elementos o estar vacía.

La producción 6 indica que el cuerpo puede contener múltiples elementos o estar vacío.

La producción 7 define los posibles elementos dentro de la cabeza.

La producción 8 define los posibles elementos dentro del cuerpo.

La producción 9 especifica la metaetiqueta.

La producción 10 especifica el título.

La producción 11 especifica el enlace.

La producción 12 especifica el párrafo.

La producción 13 especifica el enlace dentro del cuerpo.

La producción 14 especifica la imagen.

La producción 15 define que el texto puede estar compuesto por caracteres repetidos o estar vacío.

La producción 16 define que un carácter puede ser cualquier símbolo que no sea una etiqueta HTML.

Símbolo Inicial de la Gramática de HTML-es

El símbolo inicial de la gramática desde donde se empiezan las derivaciones es:

Document (Documento)

Precedencia y Asociatividad:

1. **Operadores multiplicativos (*, /, %):**
 - Mayor precedencia
 - Asociatividad izquierda
2. **Operadores aditivos (+, -):**
 - Menor precedencia que los multiplicativos
 - Asociatividad izquierda
3. **Operadores relacionales (<, <=, >, >=):**
 - Menor precedencia que los aditivos
 - Asociatividad izquierda
4. **Operadores de igualdad (==, !=, ===, !==):**
 - Menor precedencia que los relacionales
 - Asociatividad izquierda
5. **Operadores lógicos AND (&&):**
 - Menor precedencia que los operadores de igualdad
 - Asociatividad izquierda
6. **Operadores lógicos OR (||):**
 - Menor precedencia que los operadores lógicos AND
 - Asociatividad izquierda
7. **Operador de asignación (=, +=, -=, *=, /=):**
 - Menor precedencia que los operadores lógicos OR
 - Asociatividad derecha