



Taller I - Fundamentos de la computación

Contenido: Reconocedor de tokens con Flex

Objetivo

Realizar la implementación de un analizador léxico para reconocer tokens (grupo de caracteres que conforman una unidad léxica) de comandos básicos de SQL, que reconozca palabras clave de SQL con soporte para identificadores, operadores relacionales y lógicos, literales numéricos y string. Para ello, se deberá utilizar el generador de analizador léxico **Flex**. La finalidad de esta etapa es reconocer los tokens y verificar si estos corresponden a las especificaciones de la gramática.

Convenciones léxicas

1. Las palabras clave de SQL son insensibles a mayúsculas/minúsculas, es decir, "SELECT", "select" y "Select" son válidos y equivalentes.
2. Los identificadores (nombres de tablas, columnas) deben comenzar con una letra o guion bajo, seguido de letras, números o guiones bajos.
3. Los string literales deben estar encerrados entre comillas simples o dobles.
4. Para separar, solo consideraremos los delimitadores (; | , | ())
5. Los espacios en blanco se componen de espacios, retorno de línea y tabulaciones. Deben ser ignorados, excepto cuando separen tokens.
6. En este caso solo utilizaremos comentarios de línea que inician con "--" y no de bloque.

Entregables

- a. Código fuente en Flex (.l)
- b. Código fuente compilado en lenguaje C (.c)
- c. Realizar un modo interactivo que permita analizar directamente por consola y mostrar resultados.
- d. Cinco archivos de texto de ejemplo con comandos SQL (3 válidos y 2 con errores)
- e. README indicando claramente cómo correr el reconocedor.
- f. Informe técnico en LaTeX, este debe incluir:
 - Portada.
 - Índice.
 - Introducción.
 - Diseño e implementación.
 - Pruebas y validación.
 - Conclusiones y trabajo futuro.
 - Referencias bibliográficas.

Tokens

- 1) <KEYWORD> := "SELECT" | "INSERT" | "WHERE" | "UPDATE" | "DELETE" | "FROM"
- 2) <IDENTIFIER> := <LETTER> (<LETTER> | <DIGIT> | "_")*
- 3) <INTEGER> := <DIGIT>+
- 4) <DECIMAL> := <DIGIT>+ "." <DIGIT>+
- 5) <STRING> := ""<CHAR>* "" | ""<CHAR>* ""
- 6) <RELATIONAL_OP> := "=" | "<" | ">" | "<=" | ">="
- 7) <LOGICAL_OP> := "AND" | "OR" | "NOT"
- 8) <DELIMITER> := ";" | "," | "(" | ")"
- 9) <COMMENT> := "--" <ANY_CHAR>*
- 10) <WHITE SPACE> := (" " | "\t" | "\n" | "\r")+ (se ignora en la salida)

Definiciones auxiliares

- 1) <LETTER> := [a-zA-Z_]
- 2) <DIGIT> := [0-9]
- 3) <CHAR> := cualquier carácter excepto la comilla correspondiente
- 4) <ANY_CHAR> := cualquier carácter

Ejemplos

Cadena de entrada	Salida
Select nombre FROM alumnos;	<KEYWORD>: Select <IDENTIFIER>: nombre <KEYWORD>: FROM <IDENTIFIER>: alumnos <IDENTIFIER>: ;
FROM SELECT INSERT hola; -- comentario	<KEYWORD>: FROM <KEYWORD>: SELECT <KEYWORD>: INSERT <IDENTIFIER>: hola <COMMENT>: -- comentario
DELETE FROM clientes -- eliminando registros	<KEYWORD>: DELETE <KEYWORD>: FROM <IDENTIFIER>: clientes <COMMENT>: -- eliminando registros

Cabe mencionar que es un reconocedor de tokens, por lo tanto, el ejemplo 2 está léxicamente correcto, pero sintácticamente no.

Plazo de entrega

- La fecha límite de entrega es domingo 13 de noviembre a las 23:59 h.
- **No se aceptarán entregas fuera de plazo.**

Formato de entrega

- La resolución del taller debe ser resuelta en grupos de 2 a 3 personas.
- Para el informe en LaTeX, se sugiere trabajar en cualquiera de las siguientes 3 plantillas:
 - [**Coventry University Technical Report Template**](#)
 - [**Technical Document Template**](#)
 - [**LaTeX Template for Technical Report**](#)
- Las consultas sobre el taller serán respondidas mediante correo electrónico.
- El taller deberá ser entregado en campus virtual, incluyendo todos los entregables solicitados.
- La copia del taller será sancionada con nota 1,0 y los antecedentes serán reportados a la jefatura de carrera y al registro curricular.

Correos de contacto

- **Profesor:** Claudio Meneses (cmeneses@ucn.cl)
- **Ayudante:** Byron Letelier (byron.letelier@alumnos.ucn.cl)