

<Command> ::= <CommandType> ";"

<CommandType> ::= <Use>|<Create>|<Drop>|<Alter>|<Insert>|<Select>|<Update>|<Delete>|<Join>

1) USE

<Use> ::= "USE " [DatabaseName]

2) CREATE

<Create> ::= <CreateDatabase> | <CreateTable>

- <CreateDatabase> ::= "CREATE " "DATABASE " [DatabaseName]
 - <CreateTable> ::= "CREATE " "TABLE " [TableName] |
"CREATE " "TABLE " [TableName] "(" <AttributeList> ")"
 - <AttributeList> ::= [AttributeName] | [AttributeName] ";" <AttributeList>
 - [AttributeName] ::= [PlainText]
-

3) DROP

<Drop> ::= "DROP " "DATABASE " [DatabaseName] | "DROP " "TABLE " [TableName]

4) ALTER

<Alter> ::= "ALTER " "TABLE " [TableName] " " <AlterationType> " " [AttributeName]

- <AlterationType> ::= "ADD" | "DROP"
-

5) INSERT

<Insert> ::= "INSERT " "INTO " [TableName] " VALUES "(" <ValueList> ")"

- <ValueList> ::= [Value] | [Value] ";" <ValueList>
 - [Value] ::= "" [StringLiteral] "" | [BooleanLiteral] | [FloatLiteral] | [IntegerLiteral] | "NULL"
-

6) SELECT

<Select> ::= "SELECT " <WildAttribList> " FROM " [TableName] |

"SELECT " <WildAttribList> " FROM " [TableName] " WHERE " <Condition>

- <WildAttribList> ::= <AttributeList> | "*"
 - <AttributeList> ::= [AttributeName] | [AttributeName] ";" <AttributeList>
-

7) UPDATE

<Update> ::= "UPDATE " [TableName] " SET " <NameValueList> " WHERE " <Condition>

- <NameValueList> ::= <NameValuePair> | <NameValuePair> ";" <NameValueList>
 - <NameValuePair> ::= [AttributeName] "=" [Value]
-

8) DELETE

<Delete> ::= "DELETE " "FROM " [TableName] " WHERE " <Condition>

9) JOIN

<Join> ::= "JOIN " [TableName] " AND " [TableName] " ON " [AttributeName] " AND " [AttributeName]

[TableName] ::= [PlainText]

[AttributeName] ::= [PlainText]

[DatabaseName] ::= [PlainText]

[PlainText] ::= [Letter] | [Digit] | [PlainText] [Letter] | [PlainText] [Digit]

- [Digit] ::= "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"

- [Letter] ::= [Uppercase] | [Lowercase]

- [Uppercase] ::= "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" | "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" | "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z"

- [Lowercase] ::= "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" | "i" | "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" | "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z"

[Value] ::= "" [StringLiteral] "" | [BooleanLiteral] | [FloatLiteral] | [IntegerLiteral] | "NULL"

- [StringLiteral] ::= "" | [CharLiteral] | [StringLiteral] [CharLiteral]

- [CharLiteral] ::= [Space] | [Letter] | [Symbol] | [Digit]

- [Symbol] ::= "!" | "#" | "\$" | "%" | "&" | "(" | ")" | "*" | "+" | ";" | "-" | ":" | "/" | ":" | ";" | ">" | "=" | "<" | "?" | "@" | "[" | "\" | "]" | "^" | "_" | "`" | "{" | "}" | "~"

- [Space] ::= " "

- [BooleanLiteral] ::= "TRUE" | "FALSE"

- [FloatLiteral] ::= [DigitSequence] "." [DigitSequence] | "-" [DigitSequence] "." [DigitSequence] | "+" [DigitSequence] "." [DigitSequence]

- [IntegerLiteral] ::= [DigitSequence] | "-" [DigitSequence] | "+" [DigitSequence]
 - [DigitSequence] ::= [Digit] | [Digit] [DigitSequence]

We have made an adjustment to the following rule - in it's original form, some valid queries would be very hard to parse

For example, the following query was valid according to BNF: SELECT * FROM STUDENTS WHERE AGE>10ANDMARK>70;

With the revised rules, the above would NOT be valid and would need to be replaced by: SELECT * FROM STUDENTS WHERE AGE>10 AND MARK>70;

<OLD-Condition> ::= "(" <Condition> <BoolOperator> <Condition> ")" | <Condition> <BoolOperator> <Condition> | "(" [AttributeName] <Comparator> [Value] ")" | [AttributeName] <Comparator> [Value]

<Condition> ::= "(" <Condition> ")" | <FirstCondition> <BoolOperator> <SecondCondition> | [AttributeName] <Comparator> [Value]

- <FirstCondition> ::= <Condition> " " | "(" <Condition> ")"

- <SecondCondition> ::= " " <Condition> | "(" <Condition> ")"

- <BoolOperator> ::= "AND" | "OR"

- <Comparator> ::= "==" | ">" | "<" | ">=" | "<=" | "!=" | "LIKE"

Note:

<name> denotes a rule which may contain arbitrary additional whitespace within the symbol.

[name] indicates a rule that cannot contain additional whitespace.

For rules where additional whitespace is permitted, this can only occur before/after/between tokens and NOT _inside_ the tokens themselves...

For example, the following is valid: CREATE DATABASE marks;

Whereas the following is not valid: CRE ATE DATA BASE marks;