Appendix 1 Tiny-SQL Grammar

```
letter ::= 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
digit ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
integer ::= digit+
comp-op ::= < | > | =
table-name ::= letter[digit | letter]*
attribute-name ::= letter[digit | letter]*
column-name ::= [table-name.]attribute-name
literal ::= "whatever-except-(")-and-(")"

statement ::= create-table-statement | drop-table-statement | select-statement
   | delete-statement | insert-statement

create-table-statement ::= CREATE TABLE table-name(attribute-type-list)

data-type ::= INT | STR20
attribute-type-list ::= attribute-name data-type
   | attribute-name data-type, attribute-type-list

drop-table-statement ::= DROP TABLE table-name

select-statement ::= SELECT [DISTINCT] select-list
                     FROM table-list
                     [WHERE search-condition]
                     [ORDER BY colume-name]

select-list ::= * | select-sublist
select-sublist ::= column-name | column-name, select-sublist
table-list ::= table-name | table-name, table-list

delete-statement ::= DELETE FROM table-name [WHERE search-condition]

insert-statement ::= INSERT INTO table-name(attribute-list) insert-tuples

insert-tuples ::= VALUES (value-list) | select-statement

attribute-list ::= attribute-name | attribute-name, attribute-list
value ::= literal | integer | NULL
value-list ::= value | value, value-list

search-condition ::= boolean-term | boolean-term OR search-condition
boolean-term ::= boolean-factor | boolean-factor AND boolean-term
boolean-factor ::= [NOT] boolean-primary
boolean-primary ::= comparison-predicate | "[" search-condition "]"
comparison-predicate ::= expression comp-op expression
expression ::= term | term + expression | term - expression
term ::= factor | factor * term | factor / term
factor ::= column-name | literal | integer | ( expression )
```