

که یک گرامر برای زیرمجموعه ای از زبان برنامه نویسی انتخاب شده نوشته شود.

گرامر مجموعه ای از قوانین است که نحو و ساختار یک زبان را تعریف می کند. زیربرنامه یک بلوک کد نامگذاری شده است که می تواند توسط قسمت های دیگر برنامه اجرا شود.

SQL زبانی برای پرس و جو و دستکاری داده ها در پایگاه داده های رابطه ای است. یک زیربرنامه از SQL می تواند یک رویه ذخیره شده یا یک تابع باشد. رویه ذخیره شده زیربرنامه ای است که وظیفه خاصی را انجام می دهد و می تواند مقدار صفر یا بیشتر را برگرداند. تابع یک زیربرنامه است که یک مقدار واحد را برمی گرداند و می تواند در عبارات SQL استفاده شود .

یکی از راه های ممکن برای نوشتن گرامر برای یک زیر برنامه از SQL به صورت زیر است:

```
Subprogram ::= procedure | function
procedure ::= CREATE PROCEDURE name (parameters) AS BEGIN statements END
function ::= CREATE FUNCTION name (parameters) RETURNS type AS BEGIN RETURN expression
END
parameters ::= parameter | parameter, parameters
parameter ::= name type
statements ::= statement | statement; statements
statement ::= assignment | control | query | call
assignment ::= name := expression
control ::= IF condition THEN statements ELSE statements END IF | WHILE condition LOOP
statements END LOOP
query ::= SELECT columns FROM tables WHERE condition
call ::= name (arguments)
columns ::= column | column, columns
column ::= name | name AS alias
tables ::= table | table, tables
table ::= name | name AS alias
condition ::= expression comparison expression | condition AND condition | condition OR
condition | NOT condition
comparison ::= = | <> | < | > | <= | >=
expression ::= term | term + term | term - term | term * term | term / term | term %
term | (expression) | name | literal | function (arguments)
term ::= name | literal | function (arguments)
arguments ::= argument | argument, arguments
argument ::= expression
type ::= INT | FLOAT | CHAR | VARCHAR | DATE | BOOLEAN
name ::= identifier
alias ::= identifier
identifier ::= letter | letter identifier
letter ::= A | B | ... | Z | a | b | ... | z
literal ::= number | string | date | boolean
number ::= digit | digit number
digit ::= 0 | 1 | ... | 9
string ::= 'character' | 'character string'
character ::= any printable ASCII character
date ::= 'YYYY-MM-DD'
```

boolean ::= TRUE | FALSE

از این دستور زبان می توان برای ایجاد زیربرنامه های SQL استفاده کرد که از نحو و ساختار تعریف شده توسط قوانین پیروی می کنند. به عنوان مثال، زیر یک زیربرنامه معتبر از SQL است که از این دستور زبان استفاده می کند:

```
CREATE FUNCTION average_salary (dept_id INT) RETURNS FLOAT AS
BEGIN
    RETURN (SELECT AVG(salary) FROM employees WHERE department_id = dept_id);
END
```

توضیحات گرامر:

- خط اول قانون زیربرنامه را تعریف می کند که می گوید یک زیربرنامه می تواند یک رویه یا یک تابع باشد.

- خط دوم قانون رویه را تعریف می کند که می گوید یک رویه با کلمه کلیدی CREATE PROCEDURE شروع می شود و به دنبال آن یک نام، لیستی از پارامترهای داخل پرانتز، کلمه کلیدی AS، کلمه کلیدی BEGIN، دنباله ای از عبارات و کلمه کلیدی END قرار می گیرد.

- خط سوم قانون تابع را تعریف می کند که می گوید یک تابع با کلمه کلیدی CREATE FUNCTION شروع می شود و پس از آن یک نام، لیستی از پارامترهای داخل پرانتز، کلمه کلیدی RETURNS، یک نوع، کلمه کلیدی AS، کلمه کلیدی BEGIN، کلمه کلیدی RETURN، یک عبارت و کلمه کلیدی END.

- خط چهارم قانون پارامترها را تعریف می کند که می گوید یک لیست از پارامترها می تواند یک پارامتر باشد یا یک پارامتر به دنبال یک کاما و یک لیست دیگر از پارامترها.

- خط پنجم قانون پارامتر را تعریف می کند که می گوید یک پارامتر از یک نام و یک نوع تشکیل شده است.

- خط ششم قاعده عبارات را تعریف می کند که می گوید دنباله ای از دستورات می تواند یک دستور منفرد یا یک دستور به دنبال آن یک نقطه ویرگول و دنباله ای دیگر از دستورات باشد.

- خط هفتم قانون دستور را تعریف می کند که می گوید یک دستور می تواند یک انتساب، یک کنترل، یک پرس و جو یا یک فراخوانی باشد.
- خط هشتم قانون انتساب را تعریف می کند که می گوید یک انتساب از یک نام تشکیل شده است که توسط عملگر انتساب =: و یک عبارت دنبال می شود.
- خط نهم قانون کنترل را تعریف می کند که می گوید یک دستور کنترل می تواند یک دستور IF یا یک دستور WHILE باشد.
- سطر دهم عبارت IF را تعریف می کند که می گوید یک دستور IF با کلمه کلیدی IF شروع می شود و پس از آن یک شرط، کلمه کلیدی THEN، دنباله ای از عبارات، کلمه کلیدی ELSE، دنباله ای دیگر از عبارات و کلمه کلیدی END IF قرار می گیرد.
- خط یازدهم عبارت WHILE را تعریف می کند که می گوید یک دستور WHILE با کلمه کلیدی WHILE شروع می شود و به دنبال آن یک شرط، کلمه کلیدی LOOP، دنباله ای از عبارات و کلمه کلیدی END LOOP قرار می گیرد.
- خط دوازدهم قانون query را تعریف می کند که می گوید یک query از کلمه کلیدی SELECT، لیستی از ستون ها، کلمه کلیدی FROM، لیست جداول، کلمه کلیدی WHERE و یک شرط تشکیل شده است.
- خط سیزدهم قانون فراخوانی (Call) را تعریف می کند که می گوید فراخوانی از یک نام تشکیل شده و به دنبال آن فهرستی از آرگومان های داخل پرانتز قرار می گیرد.
- خط چهاردهم قانون ستون ها (columns) را تعریف می کند که می گوید فهرست ستون ها می تواند تک ستونی یا ستونی باشد که با کاما و لیست دیگری از ستون ها همراه باشد.
- خط پانزدهم قانون هر ستون را تعریف می کند، که می گوید یک ستون می تواند یک نام باشد یا یک نام به دنبال آن کلمه کلیدی AS و نام مستعار.
- خط شانزدهم قانون جداول را تعریف می کند، که می گوید لیست جداول می تواند یک جدول باشد یا یک جدول به دنبال یک کاما و لیست دیگری از جداول.

- خط هفدهم قانون هر جدول را تعریف می کند که می گوید یک جدول می تواند یک نام باشد یا یک نام به دنبال آن کلمه کلیدی AS و نام مستعار.

- خط هجدهم قاعده شرط (condition) را تعریف می کند که می گوید شرط می تواند یا عبارتی باشد که بعد از آن یک عملگر مقایسه و یک عبارت دیگر قرار می گیرد یا یک شرط بعد از یک عملگر منطقی (AND, OR, NOT) و شرط دیگری.

- خط نوزدهم قانون مقایسه (comparison) را تعریف می کند که می گوید عملگر مقایسه می تواند یکی از نمادهای زیر باشد: = یا < یا > یا <= یا >=.

- خط بیستم قانون عبارت (expression) را تعریف می کند، که می گوید یک عبارت می تواند یک جمله باشد یا یک اصطلاح به دنبال یک عملگر حسابی (+، -، *، /، %) و یک عبارت دیگر، یا یک عبارت محصور در پرانتز، یا یک نام، یا یک کلمه، یا یک تابع به دنبال فهرستی از آرگومان ها در پرانتز.

- خط بیست و یکم اصطلاح قانون (term) را تعریف می کند که می گوید یک اصطلاح می تواند یک نام باشد یا یک کلمه یا یک تابع و به دنبال آن فهرستی از آرگومان های داخل پرانتز.

- خط بیست و دوم قانون آرگومان ها را تعریف می کند، که می گوید لیستی از آرگومان ها می تواند یک آرگومان واحد باشد یا یک آرگومان به دنبال یک کاما و یک لیست دیگر از آرگومان ها.

- خط بیست و سوم قانون هر آرگومان را تعریف می کند که می گوید آرگومان یک عبارت است.

- خط بیست و چهارم قانون نوع (type) را تعریف می کند که می گوید یک نوع می تواند یکی از کلیدواژه های زیر باشد:

INT, FLOAT, CHAR, VARCHAR, DATE, BOOLEAN.

- خط بیست و پنجم قانون نام را تعریف می کند که می گوید یک نام یک شناسه (identifier) است.

- خط بیست و ششم قانون مستعار (alias) را تعریف می کند که می گوید نام مستعار یک شناسه است.

- خط بیست و هفتم قاعده هر شناسه (identifier) را تعریف می کند که می گوید یک شناسه می تواند یک حرف باشد یا یک حرف و به دنبال آن یک شناسه دیگر.
- خط بیست و هشتم قانون حرف (letter) را تعریف می کند که می گوید یک حرف می تواند هر حرف بزرگ یا کوچکی از A تا Z باشد.
- خط بیست و نهم قاعده لفظی (literal) را تعریف می کند که می گوید لفظ می تواند عدد باشد یا رشته یا تاریخ یا بولی.
- خط سی ام قاعده اعداد (number) را تعریف می کند که می گوید یک عدد می تواند یک رقم باشد یا یک رقم به دنبال آن یک عدد دیگر.
- خط سی و یکم قانون رقمی (digit) را تعریف می کند که می گوید یک رقم می تواند هر رقمی از 0 تا 9 باشد.
- خط سی و دوم قانون رشته (String) را تعریف می کند، که می گوید یک رشته می تواند یک کاراکتر منفرد باشد که در گیومه های تکی محصور شده است یا یک کاراکتر به دنبال رشته دیگری.
- خط سی و سوم قانون کاراکتر (character) را تعریف می کند که می گوید یک کاراکتر می تواند هر یک از کاراکترهای جدول ASCII باشد.
- خط سی و چهارم قانون تاریخ (date) را تعریف می کند که می گوید تاریخ یک سال چهار رقمی، یک ماه دو رقمی و یک روز دو رقمی است که با خط فاصله از هم جدا شده و در گیومه های تکی قرار می گیرد.
- خط سی و پنجم قانون بولی (Boolean) را تعریف می کند که می گوید یک بولی می تواند کلمه کلیدی TRUE یا کلمه کلیدی FALSE باشد.