Cypher is the declarative query language for Neo4j, the world’s leading graph database.

Any principles and capabilities of Cypher are as follows:

- Cypher matches patterns of nodes and relationships in the graph, to extract information or modify the data.
- Cypher has the concept of variables which denote stored names and parameters.
- Cypher can create, update, and remove nodes, relationships, labels, and properties.
- Cypher manages indexes and constraints.

You can try Cypher snippets live in the Neo4j Cloud or on the Neo4j documentation in the Snap Editor. For the graph models using Cypher check out [Cypher.org](https://cypher.org).

The Cypher Reference is also available in PDF format.

### Note:

Cypher is the declarative query language for Neo4j. You can use it to find the Neo4j Cypher documentation in the Snap Editor. For the graph models using Cypher check out [Cypher.org](https://cypher.org).

`CREATE` is the basic operation to create a new node with the given properties.

`WITH` allows you to bind elements and parameters.

`SET` allows you to set properties.

### Syntax

#### Read-only Query Structure

Example: `MATCH (n:Person) WHERE n.name = 'foo' RETURN n.name`  
This example finds all nodes labeled `Person` and returns the `name` property of those nodes.

#### Read-Write Query Structure

Example: `CREATE (n:Person {name: 'Tobias'})`  
This example creates a new node labeled `Person` with the `name` property set to `'Tobias'`.

### Functions

#### Mathematical

- `ABS(n)`: Absolute value of `n`.
- `acos(n)`, `asin(n)`, `atan(n)`, `atan2(y, x)`: Trigonometric functions.
- `ceil(n)`, `floor(n)`: Ceiling and floor functions.
- `cos(n)`, `sin(n)`, `tan(n)`: Trigonometric functions.
- `exp(n)`, `ln(n)`: Exponential and logarithmic functions.
- `round(n)`, `trunc(n)`: Round and truncate functions.
- `sqrt(n)`: Square root.

#### String

- `length(s)`: Length of string.
- `trim(s)`: Trim all whitespace.
- `toUpper(s)`, `toLower(s)`: Convert to uppercase or lowercase.
- `replace(s, from, to)`: Replace occurrences of `from` with `to`.
- `split(s, separator)`: Split string into an array of substrings.

#### Logical

- `AND(x, y)`, `OR(x, y)`, `XOR(x, y)`: Logical operators.
- `NOT(x)`: Logical negation.

#### Other

- `now()`: Current date and time.
- `random()`: Random number.
- `coalesce(a, b)`: Returns the first non-null argument.

###考える

- `MATCH (n:Person) WHERE n.name = 'Tobias' RETURN n.name`  
  これ例は、すべてノードラベルをPersonとし、そのノードのnameプロパティを返します。

- `CREATE (n:Person {name: 'Tobias'})`  
  この例はノードを新しく作成し、そのノードをラベルをPersonとし、nameプロパティを'Tobias'に設定します。
Cypher is the declarative query language for Neo4j, the world’s leading graph database.

- **Syntax**
  - **Graph Patterns**
    - Node patterns can contain labels and properties.
    - Any pattern can be used in RETURN.
    - Patterns can be properties.
    - Assigns a path to a variable.
  - **Read-Write Query Structure**
    - CRUD (Create, Read, Update, Delete) operations.
    - CRUD operations support properties.
    - CRUD operations can create a relationship with the given type, direction, and bind a variable to it.
  - **Write-Only Query Structure**
    - CRUD (Create, Read, Update) operations.
    - CRUD operations support properties.
    - CRUD operations can create a relationship with the given type and direction.
  - **FOREACH**
    - Executes a mutating operation for each element in a list.
    - Executes a mutating operation for each pair in a list.
    - Executes a mutating operation for each relationship in a path.
  - **WITH**
    - Allows the use of map fields, or nested map fields.
    - Allows the use of map fields, or nested map fields.
    - Allows the use of map fields, or nested map fields.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.
  - **DELETE**
    - Deletes nodes and relationships from the database.
    - Deletes nodes and relationships from the database.
  - **MATCH**
    - Matches relationships with the declared properties.
    - Match a pattern or create it if it does not exist.
    - Match a pattern or create it if it does not exist.
  - **SET**
    - Sets or updates properties.
  - **CREATE**
    - Creates a node with the given properties.
    - Creates a node with the given properties.
  - **RETURN**
    - Returns the value of all variables.
    - Returns the value of all variables.
    - Returns the value of all variables.