



Create Routine

CREATE PROCEDURE

```
CREATE  
[OR REPLACE]  
[DEFINER = { user | CURRENT_USER | role | CURRENT_ROLE }]  
PROCEDURE [IF NOT EXISTS] sp_name ([proc_parameter[,...]])  
[characteristic ...] routine_body
```

proc_parameter:

```
[ IN | OUT | INOUT ] param_name type
```

type:

Any valid MariaDB data type

characteristic:

```
LANGUAGE SQL  
| { CONTAINS SQL | NO SQL | READS SQL DATA | MODIFIES SQL DATA }  
| SQL SECURITY { DEFINER | INVOKER }  
| COMMENT 'string'
```

routine_body:

Valid SQL procedure statement



[DEFINER = { user | CURRENT_USER | role |
CURRENT_ROLE }]

- CREATE PROCEDURE procedure_name() DEFINER = 'user'@'host' BEGIN

PROCEDURE [IF NOT EXISTS] sp_name

- By default, a routine is associated with the default database
- To associate the routine explicitly with a given database, specify the name as db_name.sp_name

proc_parameter

[IN | OUT | INOUT] param_name type

- IN/OUT/INOUT
- param_name type

characteristic

- SQL SECURITY Clause:
 - SQL SECURITY=DEFINER
 - SQL SECURITY= INVOKER



CONTAINS SQL/NO SQL/READS SQL
DATA/MODIFIES SQL DATA

- Optional

Create Function

CREATE [OR REPLACE]

[DEFINER = {user | CURRENT_USER | role | CURRENT_ROLE}]

[AGGREGATE] FUNCTION [IF NOT EXISTS] func_name ([func_parameter[,...]])

RETURNS type

[characteristic ...]

RETURN func_body

func_parameter:

[IN | OUT | INOUT | IN OUT] param_name type

type:

Any valid MariaDB data type

characteristic:

LANGUAGE SQL

[NOT] DETERMINISTIC

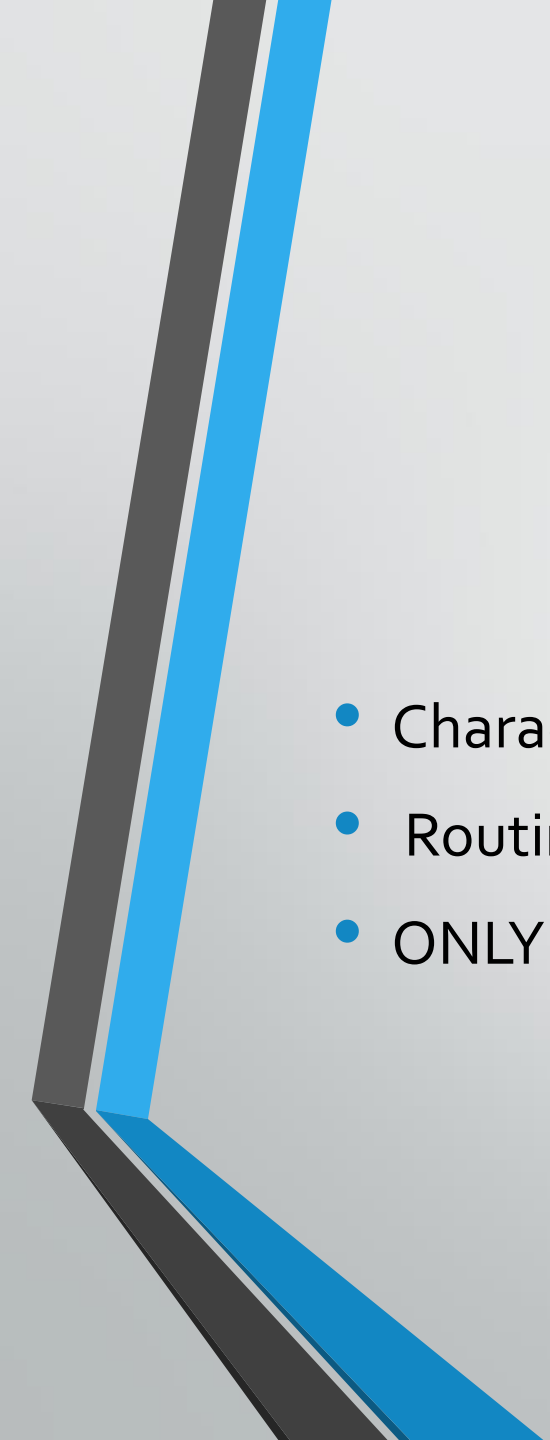
[{ CONTAINS SQL | NO SQL | READS SQL DATA | MODIFIES SQL DATA }]

[SQL SECURITY { DEFINER | INVOKER }]

[COMMENT 'string']

func_body:

Valid SQL procedure statement

- 
- Character Sets and Collations
 - Routines can contain DDL statements such as Create and Drop
 - ONLY functions can contain SQL transaction statements such as COMMIT