

SQL

Paolo Bettelini

Contents

1	Users	2
2	Triggers	2
3	Cursors	2
4	Stored Procedures	2
4.1	Definition	2
4.2	Call	3
5	Events	3
5.1	Enable	3
5.2	Definition	3
6	Functions	4
6.1	Definition	4
6.2	Call	4

1 Users

```
CREATE USER 'username'@'localhost' IDENTIFIED BY 'password';
% access from everywhere
CREATE USER 'username'@'\%' IDENTIFIED BY 'password';
% grant all + can grant
GRANT ALL ON database.* TO 'username'@'localhost' [WITH GRANT OPTION];
% grant single actions
GRANT SELECT, UPDATE, INSERT, DELETE ON database.table TO 'username'@'localhost';
% for triggers
GRANT EXECUTE ON database.* TO 'username'@'localhost';
% grant action on column
GRANT UPDATE(column) ON database.table TO 'username'@'localhost';
```

2 Triggers

```
DELIMITER //
CREATE
[DEFINER = <user>]
TRIGGER triggerName
BEFORE [INSERT | UPDATE | DELETE]
ON some_table
FOR EACH ROW BEGIN
    IF NEW.field != 'P' AND NEW.field != 'D' THEN
        SET NEW.ruolo = 'NV';
    END IF;
END;
//
DELIMITER ;
```

3 Cursors

```
BEGIN
% variables

DECLARE done INT DEFAULT FALSE;
DECLARE cur CURSOR FOR SELECT id, nome, cognome FROM persone;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;

OPEN cur;
iter: WHILE NOT done DO
    FETCH cur INTO var1, var2, var3;

    % ...
END WHILE iter;
END;
```

4 Stored Procedures

4.1 Definition

```
DELIMITER //
CREATE PROCEDURE addDummyData()
BEGIN
    INSERT INTO dummyTable VALUES ();
END
//
DELIMITER ;
```

```

DELIMITER //
CREATE PROCEDURE log (var1 VARCHAR(25), var2 VARCHAR(25), var3 VARCHAR(25))
BEGIN
    INSERT INTO dummyTable (field1, field2, field3) VALUES (var1, var2, var3);
END;
// DELIMITER ;

```

```

DELIMITER //
CREATE PROCEDURE doSomething (IN inVar INT, OUT outVar INT)
BEGIN
    SELECT COUNT(id) INTO outVar FROM dummyTable WHERE field = inVar;
END;
//
DELIMITER ;

```

4.2 Call

```

CALL addDummyData();
CALL doSomething(42, @res);
SELECT @res;

```

5 Events

5.1 Enable

```

SET GLOBAL event_scheduler = ON;
SET @@global.event_scheduler = ON;
SET GLOBAL event_scheduler = 1;
SET @@global.event_scheduler = 1;

```

5.2 Definition

```

CREATE EVENT eventName
ON SCHEDULE EVERY 1 SECOND
COMMENT 'Adds data'
DO
    CALL someProcedure();

```

```

CREATE EVENT eventName
ON SCHEDULE EVERY 1 SECOND
DO BEGIN
    DELETE FROM dummyTable;
END

```

```

CREATE EVENT eventName
ON SCHEDULE AT CURRENT_TIMESTAMP + INTERVAL 10 SECOND
DO BEGIN
    DELETE FROM dummyTable;
END

```

ON SCHEDULE EVERY interval STARTS timestamp [+INTERVAL] ENDS timestamp [+INTERVAL]

6 Functions

6.1 Definition

```
DELIMITER //
CREATE FUNCTION countSomething (name VARCHAR(25))
RETURNS INT
[DETERMINISTIC]
[READS SQL DATA]
BEGIN
    RETURN (SELECT COUNT(*) FROM dummyTable WHERE user_name = name);
END;
// DELIMITER ;
```

6.2 Call

```
SELECT countSomething('Paolo');
```