


MySQL
Conference & Expo



CS122B: Projects in Database Management
Spring 2008

Notes07: Using Stored Routines
for MySQL

Slides borrowed from Giuseppe Maxia at MySQL

Presented by
MySQL
Conference & Expo
O'REILLY

MySQL Stored routines Overview

- Stored on server
- Belong to database
- based on standard SQL specification

Presented by
MySQL
Conference & Expo
O'REILLY

MySQL Stored routines language

- PROCEDURE
- FUNCTION
- TRIGGER

Presented by
MySQL
Conference & Expo
O'REILLY

MySQL Stored routines language - procedure

- PARAMETERS
 - IN
 - OUT
 - INOUT
- may return one or more data sets
- can use dynamic SQL

Presented by
MySQL
Conference & Expo
O'REILLY

MySQL Stored routines language - function

- only input PARAMETERS
- MUST return one value of a given type
- can't use dynamic SQL
- can't return data sets

MySQL Stored routines language - trigger

- associated to table events (INSERT, UPDATE,DELETE)
- its parameters depend on the event
- does not return anything
- can't use dynamic SQL
- can't return data sets

MySQL Stored routines language

- CREATE PROCEDURE
- CREATE FUNCTION
- BEGIN .. END blocks
- IF ... THEN ... ELSE ... END IF
- CASE ... THEN ... THEN ... ELSE ... END CASE
- WHILE ... END WHILE
- LOOP ... END LOOP
- DECLARE

MySQL Stored routines language

- HANDLERS
- CURSORS

MySQL Stored routines language - procedure

```
•CREATE PROCEDURE how_is_it (IN x INT)
•BEGIN
•  IF (x > 5) THEN
•    SELECT CONCAT(x, " is higher") as
answer;
•  ELSE
•    SELECT CONCAT(x, " is lower") as answer;
•  END IF;
•END
```

MySQL Stored routines language - procedure

```
•CALL how_is_it(6);
•+-----+
•| answer      |
•+-----+
•| 6 is higher |
•+-----+
•CALL how_is_it(2);
```

MySQL Stored routines language - function

```
•CREATE FUNCTION is_bigger (x INT)
•RETURNS CHAR(3)
•BEGIN
•  IF (x > 5) THEN
•    RETURN 'YES';
•  ELSE
•    RETURN 'NO';
•  END IF;
•END
```

MySQL Stored routines language - function

```
•SELECT is_bigger(6);
•+-----+
•| is_bigger(6) |
•+-----+
•| YES          |
•+-----+
•SELECT is_bigger(2);
```

MySQL Stored routines language - triggers

```

CREATE TRIGGER salary_bi
BEFORE INSERT ON salary
FOR EACH ROW
BEGIN
  CASE
    WHEN new.work_done > 10
    THEN SET new.bonus = 5000;
    WHEN new.work_done > 5
    THEN SET new.bonus = 2500;
    WHEN new.work_done > 2
    THEN SET new.bonus = 1000;
    ELSE SET new.bonus = 0;
  END CASE;
END;

```

Presented by O'REILLY

MySQL Stored routines language - triggers

```

insert into salary (emp_no, work_done, bonus)
values (1,15,0);

Query OK, 1 row affected (0.00 sec)

insert into salary (emp_no, work_done, bonus)
values (2,5,0);

Query OK, 1 row affected (0.00 sec)

select emp_no, work_done, bonus from salary;

```

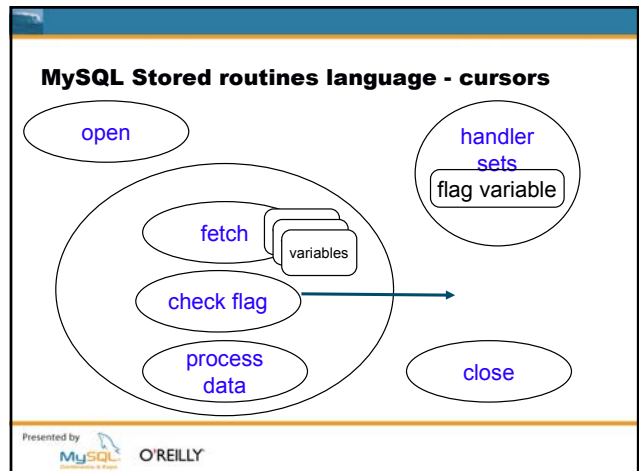
emp_no	work_done	bonus
1	15	5000
2	5	2500

Presented by O'REILLY

MySQL Stored routines language - cursors

- variables for cursor columns
- flag variable
- cursor
- continue handler

Presented by O'REILLY



Dynamic SQL

- Text converted to a query

- `set @query = 'select * from salary where emp_no= ? ';`

- `prepare myStat from @query;`

- `set @emp = 1;`

- `EXECUTE myStat USING @emp;`