Triggers & Scheduled Events

Sang Shin www.javapassion.com "Learning is fun!"



Topics

- What is a trigger?
- OLD and NEW values in UPDATE triggers
- Scheduled events

What is a Trigger?

What is a Trigger?

- A set of SQL statements that are automatically executed ("triggered") when a specific event occurs
- Used for automating database operations
 - Logging
 - Creating snapshots
 - Computation of interesting values
 - Changing data in one table in response to a change in another

Mechanism of a Trigger

- A trigger is always associated with a particular table
- A trigger can be executed either before or after an event
- An event type can be
 - > INSERT'ing record(s) into a table
 - > UPDATE'ing record(s) of a table
 - > DELETE'ing record(s) from a table
- There could be 6 possible triggers per table
 - > 2 (Before/After) * 3 types of events (INSERT/UPDATE/DELETE)

Trigger Example of INSERT event

```
/* Create a trigger - before INSERT - log the change */
CREATE TRIGGER person bi
 BEFORE INSERT ON person
 FOR EACH ROW
   INSERT INTO person log (user, message, time)
   VALUES (CURRENT USER(), 'New record is about to be added to
person table', NOW());
/* Create a trigger - after INSERT - log the change */
CREATE TRIGGER person ai
 AFTER INSERT ON person
 FOR EACH ROW
   INSERT INTO person_log (user, message, time)
   VALUES (CURRENT USER(), 'New record has been added to person
table', NOW());
```

Trigger Example of UPDATE event

```
/* Create a trigger - before UPDATE */
CREATE TRIGGER person bu
 BEFORE UPDATE ON person
 FOR EACH ROW
   INSERT INTO person log (user, message, time)
   VALUES (CURRENT USER(), 'A record is about to be updated in
person table', NOW());
/* Create a trigger - after UPDATE */
CREATE TRIGGER person au
 AFTER UPDATE ON person
 FOR EACH ROW
   INSERT INTO person_log (user, message, time)
   VALUES (CURRENT USER(), 'A record has been updated in person
table', NOW());
```

OLD and NEW Values in UPDATE Triggers

OLD and NEW Values

- In the UPDATE trigger
 - Old values can be accessible through OLD.<name-of-field>
 - New values can be accessible through NEW.<name-of-field>

OLD & NEW values in Update Trigger

```
/* Create a trigger - before UPDATE */
CREATE TRIGGER person bu
 BEFORE UPDATE ON person
 FOR EACH ROW
   INSERT INTO person log2 (user, old value, new value, time)
   VALUES (CURRENT USER(), OLD.first name, NEW.first name,
NOW());
/* Create a trigger - after UPDATE */
CREATE TRIGGER person au
 AFTER UPDATE ON person
 FOR EACH ROW
   INSERT INTO person log2 (user, old value, new value, time)
   VALUES (CURRENT USER(), OLD.first name, NEW.first name,
NOW());
```

Scheduled Events

What is Scheduled Event?

- A set of SQL statements can be executed on a time-based schedule through ON SCHEDULE <interval>
 - Once or repeatedly at predefined intervals
 - > Every day, every 90 minutes, every 2 hour
- Like triggers, a scheduled event is always associated with a particular table
- Usage examples
 - Log file rotation
 - > Statistics gathering
 - > Counter updates

What is Scheduled Event?

- A set of SQL statements can be executed on a time-based schedule through ON SCHEDULE <interval>
 - Once or repeatedly at predefined intervals
 - > Every day, every 90 minutes, every 2 hour
- Like triggers, a scheduled event is always associated with a particular table
- Usage examples
 - > Log file rotation
 - Statistics gathering
 - > Counter updates

Repeating (Recurring) Event Examples

```
/* Create an event handler for every minute */
CREATE EVENT myevent per minute
ON SCHEDULE EVERY 1 MINUTE
STARTS CURRENT TIMESTAMP
ENDS '2020-05-03 10:00:00'
ENABLE
DO
 INSERT INTO person log3 (user, message, time)
 VALUES (CURRENT USER(), 'Useless message', NOW());
/* Create an event handler for every hour */
CREATE EVENT myevent per hour
ON SCHEDULE EVERY 1 HOUR
STARTS CURRENT TIMESTAMP
ENABLE
DO
 TRUNCATE person log3;
```

One-time Event Example

```
/* Create onetime event handler */
CREATE EVENT myevent_onetime
ON SCHEDULE AT '2010-04-08 12:36'
ENABLE
DO
INSERT INTO person_log3 (user, message, time)
VALUES (CURRENT USER(), 'Onetime useless message', NOW());
```

Thank you!

Sang Shin
http://www.javapassion.com
"Learning is fun!"

