

## **CORRRECT QUERIES:**

**ALTER TABLE bus\_lifion\_ops\_002.ScheduledBusinessEvents**

ADD COLUMN mask VARCHAR(50);

ALTER TABLE bus\_lifion\_ops\_002.ScheduledBusinessEventsExecution

ADD COLUMN mask VARCHAR(50);

ALTER TABLE bus\_lifion\_ops\_002.SorTriggerInstance

ADD COLUMN mask VARCHAR(50);

ALTER TABLE business\_events.ScheduledBusinessEvents

ADD COLUMN mask VARCHAR(50);

ALTER TABLE business\_events.ScheduledBusinessEventsExecution

ADD COLUMN mask VARCHAR(50);

ALTER TABLE business\_events.SorTriggerInstance

ADD COLUMN mask VARCHAR(50);

select \* from ScheduledBusinessEvents where businesseventId =  
'ca176d913655438bb39566aae0ef7459';

bus\_sre\_test\_c\_59e8

use bus\_lifion\_ops\_002;

SET GLOBAL event\_scheduler = ON;

DROP EVENT IF EXISTS business\_events.jobs\_polling\_event\_scheduler;

DROP PROCEDURE IF EXISTS business\_events.transferJobs;

```

DROP PROCEDURE IF EXISTS transferJobs;
DELIMITER $$
CREATE PROCEDURE transferJobs()
BEGIN

DECLARE exit handler for sqlException
BEGIN
ROLLBACK;
END;

START TRANSACTION;
PREPARE stmt1 FROM 'UPDATE ScheduledBusinessEvents SET status =
"InProgress" WHERE executionTime <= NOW() AND status IS NULL';
EXECUTE stmt1;
PREPARE stmt2 FROM 'INSERT IGNORE INTO
business_events.ScheduledBusinessEventsExecution(scheduledBusinessEvent
Key,clientId,eventId,businessEventId,entityId,executionTime,systemActor,schem
aEntityMapping,context,sorTriggerType,dateCreated,lastModified,mask)
SELECT
scheduledBusinessEventKey,clientId,eventId,businessEventId,entityId,execution
Time,systemActor,schemaEntityMapping,context,sorTriggerType,dateCreated,la
stModified,mask FROM ScheduledBusinessEvents WHERE status =
"InProgress"';
EXECUTE stmt2;
PREPARE stmt3 FROM 'UPDATE ScheduledBusinessEvents SET status =
"Completed" WHERE executionTime <= NOW() AND status = "InProgress"';
EXECUTE stmt3;
COMMIT;
END
$$
DELIMITER ;

```

```

DROP EVENT IF EXISTS jobs_polling_event_scheduler;
DELIMITER $$

```

```

CREATE EVENT IF NOT EXISTS jobs_polling_event_scheduler
ON SCHEDULE EVERY 20 SECOND

```

```
ON COMPLETION PRESERVE  
DO  
BEGIN
```

```
CALL transferJobs();
```

```
END$$  
DELIMITER ;
```