

EXERCISES:

1. **Write a trigger program to update the total capacity in the WorkCenterStats table before a new work center is inserted into the WorkCenter table:**

```
DELIMITER $$

CREATE TRIGGER before_workcenters_insert
BEFORE INSERT
ON WorkCenters FOR EACH ROW
BEGIN
    DECLARE rowcount INT;

    SELECT COUNT(*)
    INTO rowcount
    FROM WorkCenterStats;

    IF rowcount > 0 THEN
        UPDATE WorkCenterStats
        SET totalCapacity = totalCapacity + new.capacity;
    ELSE
        INSERT INTO WorkCenterStats(totalCapacity)
        VALUES(new.capacity);
    END IF;

END $$
```

Checking the trigger functionality

```
INSERT INTO WorkCenters(name, capacity) VALUES('Mold Machine',100);
SELECT * FROM WorkCenterStats;
```

2. **Write a trigger program to inserts a reminder into the reminders table if the birth date of the member is NULL.**

```
DELIMITER $$

CREATE TRIGGER after_members_insert
AFTER INSERT
ON members FOR EACH ROW
BEGIN
```

```

        IF NEW.birthDate IS NULL THEN
            INSERT INTO reminders(memberId, message)
                VALUES (new.id,CONCAT('Hi ', NEW.name, ', please update your date of
birth.'));
        END IF;
    END$$

```

```
DELIMITER ;
```

checking the trigger functionality

```

INSERT INTO members(name, email, birthDate)
VALUES
    ('John Doe', 'john.doe@example.com', NULL),
    ('Jane Doe', 'jane.doe@example.com','2000-01-01');

SELECT * FROM members;
SELECT * FROM reminders;

```

3. Create MySQL Trigger that inserts a new row into the SalaryArchives table before a row from the Salaries table is deleted.

```

DELIMITER $$

CREATE TRIGGER before_salaries_delete
BEFORE DELETE
ON salaries FOR EACH ROW
BEGIN
    INSERT INTO SalaryArchives(employeeNumber,validFrom,amount)
    VALUES (OLD.employeeNumber,OLD.validFrom,OLD.amount);
END$$

DELIMITER ;

```

checking the trigger functionality

```

DELETE FROM salaries
WHERE employeeNumber = 1002;

SELECT * FROM SalaryArchives;
The trigger was invoked and inserted a new row into
the SalaryArchives table.
DELETE FROM salaries;

```

```
SELECT * FROM SalaryArchives;
```

4. Create MySQL Trigger updates the total salary in the SalaryBudgets table after a row is deleted from the Salaries table

```
CREATE TRIGGER after_salaries_delete  
AFTER DELETE  
ON Salaries FOR EACH ROW  
UPDATE SalaryBudgets  
SET total = total - old.salary;  
DELETE FROM Salaries  
WHERE employeeNumber = 1002;  
SELECT * FROM SalaryBudgets;
```

Credit: All these exercises are taken from *mysqldata.org* web site.

You can find more **exercise problems** in,

- ✓ <http://www.cse.chalmers.se/edu/year/2017/course/TDA357/VT2016-dontuse/tutorials/4-triggers/index.html>
- ✓ https://cs.unibg.it/mutti/teaching/exe5a_polimi_180116_parabosc.pdf
- ✓ <https://www.wiseowl.co.uk/sql/exercises/standard/triggers/4275/>

References:

- <https://www.mysqltutorial.org/mysql-triggers/>