**PROCEDURE UPDATE\_STUDENT**

DELIMITER //

CREATE PROCEDURE UPDATE\_STUDENT(

IN v\_student\_id INT,

IN v\_first\_name VARCHAR(50),

IN v\_last\_name VARCHAR(50),

IN v\_street VARCHAR(100),

IN v\_city VARCHAR(50),

IN v\_state VARCHAR(50),

IN v\_gender CHAR(1),

IN v\_email VARCHAR(100),

IN v\_birthdate DATE,

IN v\_dept\_id INT

)

BEGIN

UPDATE student

SET

first\_name = v\_first\_name,

last\_name = v\_last\_name,

street = v\_street,

city = v\_city,

state = v\_state,

gender = v\_gender,

email = v\_email,

birthdate = v\_birthdate,

dept\_id = v\_dept\_id

WHERE student\_id = v\_student\_id;

END //

DELIMITER ;

**PROCEDURE UPDATE\_COURSE**

DELIMITER //

CREATE PROCEDURE UPDATE\_COURSE(

IN v\_course\_id INT,

IN v\_course\_name VARCHAR(100),

IN v\_credits INT,

IN v\_description TEXT,

IN v\_dept\_id INT

)

BEGIN

UPDATE COURSE

SET

course\_name = v\_course\_name,

credits = v\_credits,

description = v\_description,

dept\_id = v\_dept\_id

WHERE

course\_id = v\_course\_id;

END //

DELIMITER ;

**PROCEDURE UPDATE\_STUDENT\_PHONE**

DELIMITER //

CREATE PROCEDURE UPDATE\_STUDENT\_PHONE(

IN v\_student\_id INT,

IN v\_old\_phone VARCHAR(20),

IN v\_new\_phone VARCHAR(20)

)

BEGIN

UPDATE STUDENT\_PHONE

SET

phone = v\_new\_phone

WHERE

student\_id = v\_student\_id AND phone = v\_old\_phone;

END //

DELIMITER ;

**PROCEDURE UPDATE\_ASSIGNED\_GPA**

DELIMITER //

CREATE PROCEDURE UPDATE\_ASSIGNED\_GPA(

IN v\_student\_id INT,

IN v\_course\_id INT,

IN v\_year INT,

IN v\_semester VARCHAR(20),

IN v\_new\_gpa DECIMAL(3,2)

)

BEGIN

UPDATE assigned

SET

gpa = v\_new\_gpa

WHERE

student\_id = v\_student\_id AND course\_id = v\_course\_id AND year = v\_year AND semester = v\_semester;

END //

DELIMITER ;

**PROCEDURE UPDATE\_DEPARTMENT\_NAME**

DELIMITER //

CREATE PROCEDURE UPDATE\_DEPARTMENT\_NAME(

IN v\_dept\_id INT,

IN v\_new\_dept\_name VARCHAR(50)

)

BEGIN

UPDATE DEPARTMENT

SET

dept\_name = v\_new\_dept\_name

WHERE

dept\_id = v\_dept\_id;

END //

DELIMITER ;

**PROCEDURE INSERT\_DEPARTMENT**

DELIMITER //

CREATE PROCEDURE INSERT\_DEPARTMENT(

IN v\_dept\_id INT,

IN v\_dept\_name VARCHAR(50)

)

BEGIN

INSERT INTO DEPARTMENT (dept\_id, dept\_name)

VALUES (v\_dept\_id, v\_dept\_name);

END //

DELIMITER ;

**PROCEDURE INSERT\_STUDENT**

DELIMITER //

CREATE PROCEDURE INSERT\_STUDENT(

IN v\_student\_id INT,

IN v\_first\_name VARCHAR(50),

IN v\_last\_name VARCHAR(50),

IN v\_street VARCHAR(100),

IN v\_city VARCHAR(50),

IN v\_state VARCHAR(50),

IN v\_gender CHAR(1),

IN v\_email VARCHAR(100),

IN v\_birthdate DATE,

IN v\_dept\_id INT

)

BEGIN

INSERT INTO STUDENT (student\_id, first\_name, last\_name, street, city, state, gender, email, birthdate, dept\_id)

VALUES (v\_student\_id, v\_first\_name, v\_last\_name, v\_street, v\_city, v\_state, v\_gender, v\_email, v\_birthdate, v\_dept\_id);

END //

DELIMITER ;

**PROCEDURE INSERT\_COURSE**

DELIMITER //

CREATE PROCEDURE INSERT\_COURSE(

IN v\_course\_id INT,

IN v\_course\_name VARCHAR(100),

IN v\_credits INT,

IN v\_description TEXT,

IN v\_dept\_id INT

)

BEGIN

INSERT INTO COURSE (course\_id, course\_name, credits, description, dept\_id)

VALUES (v\_course\_id, v\_course\_name, v\_credits, v\_description, v\_dept\_id);

END //

DELIMITER ;

**PROCEDURE INSERT\_ASSIGNED**

DELIMITER //

CREATE PROCEDURE INSERT\_ASSIGNED(

IN v\_student\_id INT,

IN v\_course\_id INT,

IN v\_year INT,

IN v\_semester VARCHAR(20),

IN v\_gpa DECIMAL(3, 2)

)

BEGIN

INSERT INTO assigned (student\_id, course\_id, year, semester, gpa)

VALUES (v\_student\_id, v\_course\_id, v\_year, v\_semester, v\_gpa);

END //

DELIMITER ;

DELIMITER //

**PROCEDURE INSERT\_STUDENT\_PHONE**

CREATE PROCEDURE INSERT\_STUDENT\_PHONE(

IN v\_student\_id INT,

IN v\_phone VARCHAR(20)

)

BEGIN

INSERT INTO STUDENT\_PHONE (student\_id, phone)

VALUES (v\_student\_id, v\_phone);

END //

DELIMITER ;

**PROCEDURE DELETE\_STUDENT**

DELIMITER //

CREATE PROCEDURE DELETE\_STUDENT(

IN v\_student\_id INT

)

BEGIN

DELETE FROM STUDENT

WHERE student\_id = v\_student\_id;

END //

DELIMITER ;

**PROCEDURE DELETE\_DEPARTMENT**

DELIMITER //

CREATE PROCEDURE DELETE\_DEPARTMENT(

IN v\_dept\_id INT

)

BEGIN

DELETE FROM DEPARTMENT

WHERE dept\_id = v\_dept\_id;

END //

DELIMITER ;

DELIMITER //

**PROCEDURE DELETE\_COURSE**

CREATE PROCEDURE DELETE\_COURSE(

IN v\_course\_id INT

)

BEGIN

DELETE FROM COURSE

WHERE course\_id = v\_course\_id;

END //

DELIMITER ;

**PROCEDURE DELETE\_STUDENT\_PHONE**

DELIMITER //

CREATE PROCEDURE DELETE\_STUDENT\_PHONE(

IN v\_student\_id INT,

IN v\_phone VARCHAR(20)

)

BEGIN

DELETE FROM STUDENT\_PHONE

WHERE student\_id = v\_student\_id AND phone = v\_phone;

END //

DELIMITER ;

**PROCEDURE DELETE\_ASSIGNED**

DELIMITER //

CREATE PROCEDURE DELETE\_ASSIGNED(

IN v\_student\_id INT,

IN v\_course\_id INT,

IN v\_year INT

)

BEGIN

DELETE FROM assigned

WHERE student\_id = v\_student\_id AND course\_id = v\_course\_id AND year = v\_year;

END //

DELIMITER ;

**TRIGGER `T\_CUM\_GPA\_DELETE`**

CREATE DEFINER=`root`@`localhost` TRIGGER `T\_CUM\_GPA\_DELETE` AFTER DELETE ON `assigned` FOR EACH ROW BEGIN

DECLARE v\_cum\_gpa DECIMAL(3,2);

SELECT SUM(gpa \* credits) / SUM(credits) INTO v\_cum\_gpa

FROM assigned a

INNER JOIN course c ON a.course\_id = c.course\_id

WHERE student\_id = OLD.student\_id;

UPDATE student

SET cum\_gpa = v\_cum\_gpa

WHERE student\_id = OLD.student\_id;

END

**TRIGGER `T\_CUM\_GPA\_UPDATE`**

CREATE DEFINER=`root`@`localhost` TRIGGER `T\_CUM\_GPA\_UPDATE` AFTER UPDATE ON `assigned` FOR EACH ROW BEGIN

DECLARE v\_cum\_gpa DECIMAL(3,2);

IF NEW.gpa IS NOT NULL THEN

SELECT SUM(gpa \* credits) / SUM(credits) INTO v\_cum\_gpa

FROM assigned a

INNER JOIN course c ON a.course\_id = c.course\_id

WHERE student\_id = NEW.student\_id;

UPDATE student

SET cum\_gpa = v\_cum\_gpa

WHERE student\_id = NEW.student\_id;

END IF;

END

**TRIGGER `T\_CUM\_GPA\_INSERT**

CREATE TRIGGER `T\_CUM\_GPA\_INSERT` AFTER INSERT ON `assigned` FOR EACH ROW BEGIN

DECLARE v\_cum\_gpa DECIMAL(3,2);

IF NEW.gpa IS NOT NULL THEN

select SUM(t2.gpa \* t1.credits) / SUM(t1.credits) into v\_cum\_gpa

from assigned t2 inner join(

SELECT (a.student\_id), (a.course\_id), max(a.year) year, max(c.credits) credits

FROM assigned a

INNER JOIN course c ON a.course\_id = c.course\_id

WHERE student\_id = 15 AND gpa IS NOT NULL

group by (a.student\_id), (a.course\_id)) t1

on t2.course\_id = t1.course\_id and t2.student\_id = t1.student\_id and t1.year = t2.year;

UPDATE student

SET cum\_gpa = v\_cum\_gpa

WHERE student\_id = NEW.student\_id;

END IF;

END

**TRIGGER delete\_cascade\_department**

CREATE TRIGGER delete\_cascade\_department

BEFORE DELETE ON DEPARTMENT

FOR EACH ROW

BEGIN

-- Delete from ASSIGNED table

DELETE FROM ASSIGNED WHERE course\_id IN (SELECT course\_id FROM COURSE WHERE dept\_id = OLD.dept\_id);

-- Delete from STUDENT\_PHONE table using a subquery

DELETE FROM STUDENT\_PHONE WHERE student\_id IN (SELECT student\_id FROM STUDENT WHERE dept\_id = OLD.dept\_id);

-- Delete from STUDENT table

DELETE FROM STUDENT WHERE dept\_id = OLD.dept\_id;

-- Delete from COURSE table

DELETE FROM COURSE WHERE dept\_id = OLD.dept\_id;

END;

**TRIGGER delete\_cascade\_course**

CREATE TRIGGER delete\_cascade\_course

BEFORE DELETE ON COURSE

FOR EACH ROW

BEGIN

-- Delete from assigned table

DELETE FROM assigned WHERE course\_id = OLD.course\_id;

END;

**TRIGGER delete\_cascade\_ student**

CREATE TRIGGER delete\_cascade\_ student

BEFORE DELETE ON STUDENT

FOR EACH ROW

BEGIN

-- Delete from assigned table

DELETE FROM assigned WHERE student\_id = OLD.student\_id;

-- Delete from STUDENT\_PHONE table

DELETE FROM STUDENT\_PHONE WHERE student\_id = OLD.student\_id;

END;

RESOURCES:

How to calc GPA

<https://www.youtube.com/watch?v=mrSrOscIBuQ>