CREATE TRIGGER teach AFTER INSERT ON teaches

FOR EACH ROW

BEGIN

IF NEW.ID IN (SELECT ID FROM teaches WHERE NEW.course\_id = course\_id AND NEW.sec\_id = sec\_id AND NEW.semester = semester AND NEW.`year` = `year`)

THEN

ROLLBACK;

END IF;

END;



CREATE TRIGGER time\_check2 AFTER UPDATE ON section

FOR EACH ROW

BEGIN

IF NEW.time\_slot\_id NOT IN (SELECT time\_slot\_id FROM time\_slot) THEN

ROLLBACK;

END IF;

END;

CREATE TRIGGER time\_check3 AFTER UPDATE ON time\_slot

FOR EACH ROW

BEGIN

IF OLD.time\_slot\_id IN (SELECT time\_slot\_id FROM section) THEN

ROLLBACK;

END IF;

END;



CREATE TRIGGER update1 AFTER UPDATE ON takes

FOR EACH ROW

BEGIN ATOMIC

IF NEW.ID <> OLD.ID AND NEW.course\_id = OLD.course\_id

THEN

UPADTE student SET tot\_cred = tot\_cred – (SELECT credits FROM course WHERE course\_id = NEW.course\_id) WHERE ID = OLD.ID;

UPADTE student SET tot\_cred = tot\_cred + (SELECT credits FROM course WHERE course\_id = NEW.course\_id) WHERE ID = NEW.ID;

ELSE

IF NEW.ID = OLD.ID AND NEW.course\_id <> OLD.course\_id

THEN

UPDATE student SET tot\_cred = tot\_cred + (SELECT credits FROM course WHERE course\_id = NEW.course\_id) - (SELECT credits FROM course WHERE course\_id = OLD.course\_id) WHERE ID = NEW.ID;

ELSE

UPDATE student SET tot\_cred + (SELECT credits FROM course WHERE course\_id = NEW.course\_id) WHERE ID = NEW.ID;

UPDATE student SET tot\_cred - (SELECT credits FROM course WHERE course\_id = OLD.course\_id) WHERE ID = OLD.ID;

END IF;

END IF;

END;



CREATE TRIGGER insert1 AFTER INSERT ON takes

FOR EACH ROW

BEGIN

UPDATE student SET tot\_cred = tot\_cred + (SELECT credits FROM course WHERE course\_id = NEW.course\_id) WHERE ID = NEW.ID;

END IF;

END;

1. 不修改已有学生学习的课程的course\_id和credits属性的值，新添加的课程信息，无学生学习。