Huibo Zhao

hz2480

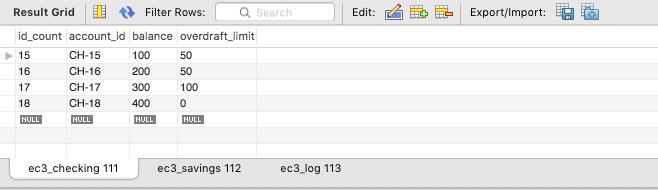
COMS4111 SEC003

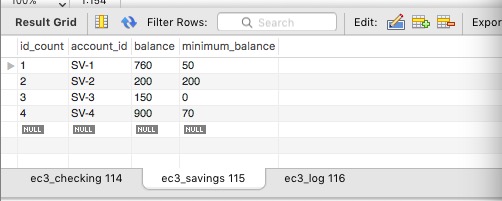
EC3

Dec.12th.2017

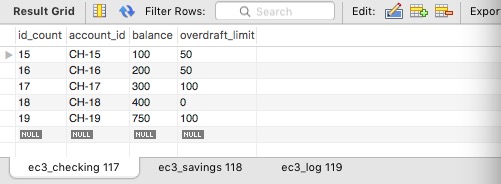
The code is in the appendix, which is at the end of this document.

Here is the initial checking and saving table:

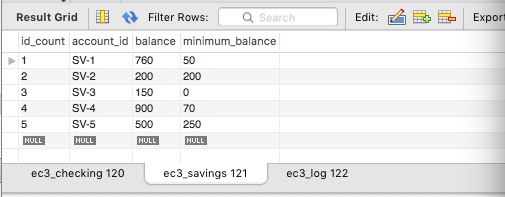




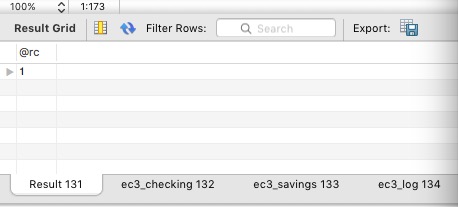
After Test 1:

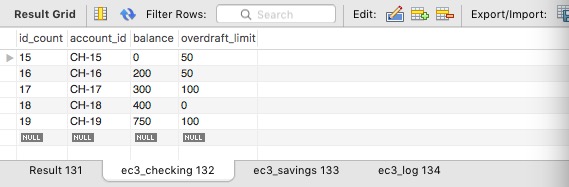


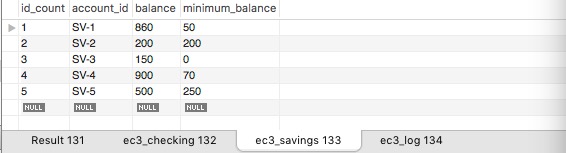
After Test 2:



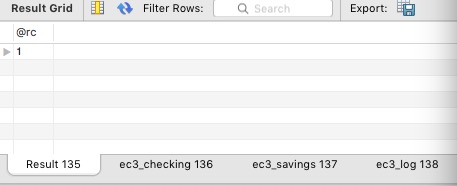
After Test 3:

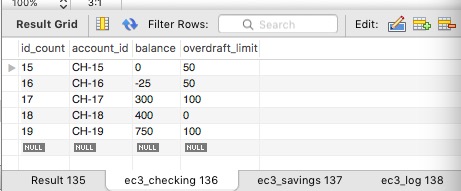


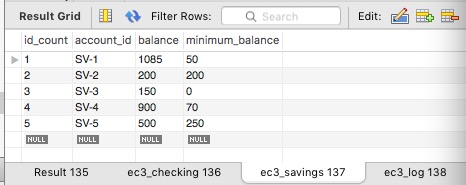




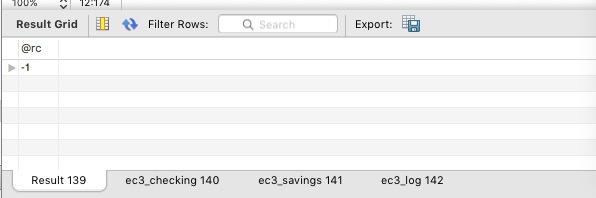
After test 4:





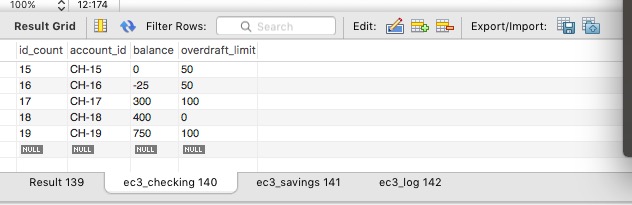


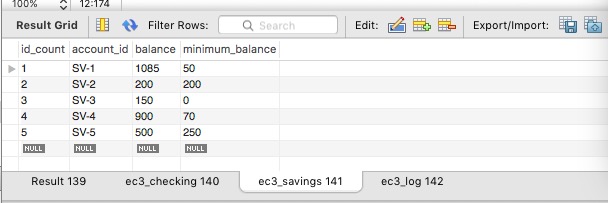
After test 5:

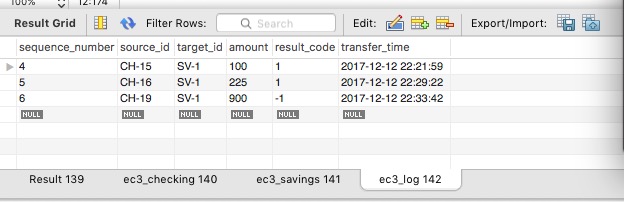


Tables have no change.

Final result:







Appendix:

/\*

DROP TABLE IF EXISTS `ec3\_checking`;

CREATE TABLE `ec3\_checking` (

`id\_count` int(11) NOT NULL AUTO\_INCREMENT,

`account\_id` varchar(12) DEFAULT NULL,

`balance` float NOT NULL,

`overdraft\_limit` float NOT NULL,

PRIMARY KEY (`id\_count`),

UNIQUE KEY `id\_count\_UNIQUE` (`id\_count`)

) ENGINE=InnoDB AUTO\_INCREMENT=14 DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `ec3\_savings`;

CREATE TABLE `ec3\_savings` (

`id\_count` int(11) NOT NULL AUTO\_INCREMENT,

`account\_id` varchar(12) DEFAULT NULL,

`balance` float NOT NULL,

`minimum\_balance` float NOT NULL,

PRIMARY KEY (`id\_count`),

UNIQUE KEY `id\_count\_UNIQUE` (`id\_count`)

) ENGINE=InnoDB AUTO\_INCREMENT=1 DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `ec3\_log`;

CREATE TABLE `ec3\_log` (

`sequence\_number` int(11) NOT NULL AUTO\_INCREMENT,

`source\_id` varchar(12) DEFAULT NULL,

`target\_id` varchar(12) DEFAULT NULL,

`amount` float DEFAULT NULL,

`result\_code` int DEFAULT NULL,

`transfer\_time` timestamp,

PRIMARY KEY (`sequence\_number`)

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8;

DROP TRIGGER IF EXISTS `trigger\_insert\_checking`;

DELIMITER $$

CREATE TRIGGER `trigger\_insert\_checking`

BEFORE INSERT ON `ec3\_checking` FOR EACH ROW

BEGIN

IF NOT EXISTS (select \* from `ec3\_checking`) THEN

SET New.account\_id = "CH-14";

ELSE

SET New.account\_id = CONCAT("CH-",(select max(id\_count) from `ec3\_checking`)+1);

end if;

END $$

DELIMITER ;

DROP TRIGGER IF EXISTS `trigger\_insert\_saving`;

DELIMITER $$

CREATE TRIGGER `trigger\_insert\_saving`

BEFORE INSERT ON `ec3\_savings` FOR EACH ROW

BEGIN

IF NOT EXISTS (select \* from `ec3\_savings`) THEN

SET New.account\_id = "SV-1";

ELSE

SET New.account\_id = CONCAT("SV-",(select max(id\_count) from `ec3\_savings`)+1);

end if;

END $$

DELIMITER ;

DROP PROCEDURE IF EXISTS `transfer\_commit`;

DELIMITER $$

CREATE PROCEDURE `transfer\_commit` (source\_id VARCHAR(12),target\_id VARCHAR(12),

amount float,OUT rc int)

BEGIN

DECLARE temp\_balance float;

IF source\_id LIKE "CH-%" THEN

IF ((select balance from `ec3\_checking` where `account\_id` = source\_id) - amount) >

(-1.0 \* (select overdraft\_limit from `ec3\_checking` where `account\_id` = source\_id)) THEN

# Debit the balance in the source account

SET temp\_balance = (select balance from `ec3\_checking` where `account\_id` = source\_id);

update `ec3\_checking` set balance = temp\_balance - amount where account\_id = source\_id;

# Increase the balance in the target account

IF target\_id LIKE "CH-%" THEN

SET temp\_balance = (select balance from `ec3\_checking` where `account\_id` = target\_id);

update `ec3\_checking` set balance = temp\_balance + amount where account\_id = target\_id;

END IF;

IF target\_id LIKE "SV-%" THEN

SET temp\_balance = (select balance from `ec3\_savings` where `account\_id` = target\_id);

update `ec3\_savings` set balance = temp\_balance + amount where account\_id = target\_id;

END IF;

# Write a log record containing the source account id,

# target account id, amount and an rc=1 to the log table

INSERT INTO `ec3\_log` (`source\_id`, `target\_id`, `amount`, `result\_code`)

VALUES (source\_id,target\_id,amount,1);

SET rc = 1;

ELSE

INSERT INTO `ec3\_log` (`source\_id`, `target\_id`, `amount`, `result\_code`)

VALUES (source\_id,target\_id,amount,-1);

SET rc = -1;

END IF;

ELSEIF source\_id LIKE "SV-%" THEN

IF ((select balance from `ec3\_savings` where `account\_id` = source\_id) - amount) >

(1.0 \* (select minimum\_balance from `ec3\_savings` where `account\_id` = source\_id)) THEN

# Debit the balance in the source account

SET temp\_balance = (select balance from `ec3\_savings` where `account\_id` = source\_id);

update `ec3\_savings` set balance = temp\_balance - amount where account\_id = source\_id;

# Increase the balance in the target account

IF target\_id LIKE "CH-%" THEN

SET temp\_balance = (select balance from `ec3\_checking` where `account\_id` = target\_id);

update `ec3\_checking` set balance = temp\_balance + amount where account\_id = target\_id;

END IF;

IF target\_id LIKE "SV-%" THEN

SET temp\_balance = (select balance from `ec3\_savings` where `account\_id` = target\_id);

update `ec3\_savings` set balance = temp\_balance + amount where account\_id = target\_id;

END IF;

# Write a log record containing the source account id,

# target account id, amount and an rc=1 to the log table

INSERT INTO `ec3\_log` (`source\_id`, `target\_id`, `amount`, `result\_code`)

VALUES (source\_id,target\_id,amount,1);

SET rc = 1;

ELSE

INSERT INTO `ec3\_log` (`source\_id`, `target\_id`, `amount`, `result\_code`)

VALUES (source\_id,target\_id,amount,-1);

SET rc = -1;

END IF;

END IF;

END $$

DELIMITER ;

# load initial data

INSERT INTO `ec3\_checking` (balance,overdraft\_limit) VALUES (0,0);

INSERT INTO `ec3\_checking` (balance,overdraft\_limit) VALUES (100,50);

INSERT INTO `ec3\_checking` (balance,overdraft\_limit) VALUES (200,50);

INSERT INTO `ec3\_checking` (balance,overdraft\_limit) VALUES (300,100);

INSERT INTO `ec3\_checking` (balance,overdraft\_limit) VALUES (400,0);

DELETE FROM `ec3\_checking` where id\_count = 14;

INSERT INTO `ec3\_savings` (balance,minimum\_balance) VALUES (760,50);

INSERT INTO `ec3\_savings` (balance,minimum\_balance) VALUES (200,200);

INSERT INTO `ec3\_savings` (balance,minimum\_balance) VALUES (150,0);

INSERT INTO `ec3\_savings` (balance,minimum\_balance) VALUES (900,70);

# test 1

INSERT INTO `ec3\_checking` (balance,overdraft\_limit) VALUES (750,100);

# test 2

INSERT INTO `ec3\_savings` (balance, minimum\_balance) values(500, 250);

# test 3

call transfer\_commit("CH-15", "SV-1",100,@rc);

select @rc;

# test 4

call transfer\_commit("CH-16", "SV-1",225,@rc);

select @rc;

\*/

# test 5:

call transfer\_commit("CH-19", "SV-1",900,@rc);

select @rc;

#call transfer\_commit("V-15", "SV-1",100,@rc);

#select @rc;

select \* from `ec3\_checking`;

select \* from `ec3\_savings`;

select \* from `ec3\_log`;