

Unit Code	COMP2350	Assignment#	3
StudentID Number	47772174	Student Name	Van Trang Nguyen
Tutor's Name	Louie Coughill	Workshop Date/Time	Tuesday 09AM – 11AM

#### TASK 4:

The trigger should be fired after updating a new row into Registration table. This is because we don't want prevent the system from updating the relevant tables when a student drops. In order to examine a student's status, it depends on whether the students is trying to drop from any of units and if yes, when did they drop and was that an advanced or non-advanced unit. Therefore, the trigger conditions will first check whether drop date is after the due date, and then check the status of the unit.

First of all, we need to check whether the student is trying to drop from any of the unit. If the student drops a unit after the due date (the dropped date without penalty deadlines, there would be two cases: the unit is at advanced level or not. If the unit is advanced, we automatically set the penalty fines for this student is \$600, and if not, we set the penalty fines at \$400.

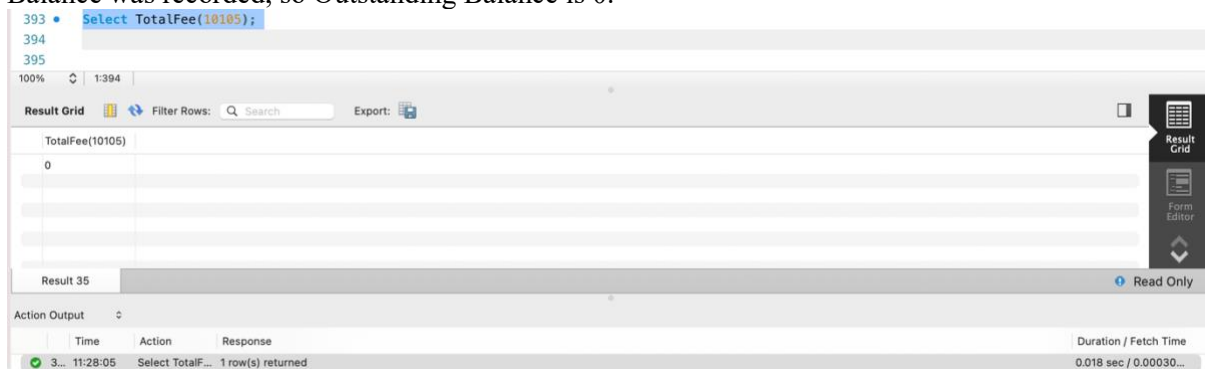
The trigger continues with updating the rows in the fee table, by changing the fee amount equal to the current fee\_amount, plus with the corresponding penalty fee applied with MATCHED student\_id and semester. Additionally, we also need to set the status for the fee to 'Dropped Unit Penalty'. Moreover, we also need to update the registration table, with status becoming 'dropped\_with\_penalty', and also update the drop date. Therefore, if students drops the unit before the due date or not drops from any, there will be nothing to update, and the trigger will fires an error.

To solve the problem of not-enough availability for other students to enroll, we need to update the table Unit as well if the student has dropped successfully. The enrollment should be set to enrollment = enrollment – 1, meaning that the number of students enrolling in that unit has decreased by 1.

One problem can happen is that after the trigger raised one error, it could not update a new value. However, the new value should still be updated to test other cases for the trigger. Thus, we might have NEW.Fee\_amount != 'CHANGE', as a lock (or a change mode) in order to run the trigger. To be more details, before updating the new values, I update NEW.Fee\_amount != 'CHANGE' so that the new values can be updated and other cases can be tested.

#### TASK 5:

*TEST CASE 1:* Since the student has not existed in database system, there was no OutStanding Balance was recorded, so Outstanding Balance is 0.



The screenshot shows a database query interface. At the top, a SQL query is entered: `Select TotalFee(10105);`. Below the query, a 'Result Grid' is displayed with a single row containing the value '0'. The grid has a column header 'TotalFee(10105)'. To the right of the grid, there are icons for 'Result Grid', 'Form Editor', and 'Read Only'. At the bottom, an 'Action Output' section shows a log entry: '3... 11:28:05 Select TotalF... 1 row(s) returned' with a duration of '0.018 sec / 0.00030...'.

Time	Action	Response	Duration / Fetch Time
3... 11:28:05	Select TotalF...	1 row(s) returned	0.018 sec / 0.00030...

*TEST CASE 2:* Student has ONLY ONE fee that has not paid, so that is the total one

425 • `Select TotalFee(10104);`  
24:426

TotalFee(10104)
3000

Result 130

**TEST CASE 3:** Student has not paid the balance from previous semesters, so the outstanding balance is the total one (This student has \$3000 from 2023S1 and 2024S1, both unpaid).

424 • `Select TotalFee(10103);`  
12:419

TotalFee(10103)
6000

Result 131

**TEST CASE 4:** The student has already paid all of the fee, so the outstanding balance is 0.

426 • `Select TotalFee(10107);`  
24:434

TotalFee(10107)
0

Result 122

**TEST CASE 5:** Student has outstanding balance from previous semesters, they couldn't enrol in any unit (message shown below)

427 • `Call BRules (10102, 'IT404', '2024-09-12', 'YES', null, '2024S2', 4000.00, 'ENROL');`  
85:427

Time	Action	Response	Duration / Fetch Time
1... 22:30:29	Call BRules (10102, 'IT404', '2024-09-12', 'YES', null, '2024S2', 4000.00, 'ENROL');	Error Code: 1644. This student cannot enroll in any unit due to outstanding fees	0.024 sec

**TEST CASE 6:** Student has NO outstanding balance (Pass through BR2), but failed in more than 2 units, so they couldn't enrol in any ADVANCED unit

428 • `Call BRules (10100, 'IT405', '2024-09-15', 'YES', null, '2025S1', 4000.00, 'ENROL');`  
35:434

Time	Action	Response	Duration / Fetch Time
1... 22:31:20	Call BRules (10100, 'IT405', '2024-09-15', 'YES', null, '2025S1', 4000.00, 'ENROL');	Error Code: 1644. This student cannot enroll in any advanced level	0.0035 sec

**TEST CASE 7:** Student has satisfied all the business rules (BR2 and BR6), so they are allowed to enrol in all units they want.

BEFORE:

```

436 • Select r.student_id, r.unit_code, r.semester, r.registration_date, r.status, r.no_penalty_deadline, r.dropped_out_date, f.fee_Amount, f.description
437 From Registration r
438 join Student s
439 on s.student_id = r.student_id
440 join Fee f
441 on f.student_id = s.student_id
442 where r.student_id = 10107
443 and r.semester = '2025S1';

```

28:441

Result Grid

student_id	unit_code	semester	registration_date	status	no_penalty_deadline	dropped_out_date	fee_Amount	description

Result 123

Action Output

Time	Action	Response	Duration / Fetch Time
1... 22:24:32	Select r.student_id, r.unit_code, r.semester, r.registration...	0 row(s) returned	0.028 sec / 0.00029...

AFTER:

```

434 • Call BRules (10107, 'IT404', '2025-01-12', 'YES', NULL, '2025S1', 4000.00, 'ENROL');

```

28:439

Result Grid

message	
This student has successfully enrolled.	

Result 124

Action Output

Time	Action	Response	Duration / Fetch Time
1... 22:24:48	Call BRules (10107, 'IT404', '2025-01-12', 'YES', NULL, '2...	1 row(s) returned	0.031 sec / 0.000002...

```

436 • Select distinct r.student_id, r.unit_code, r.semester, r.registration_date, r.status, r.no_penalty_deadline, r.dropped_out_date, f.description
437 From Registration r
438 join Student s
439 on s.student_id = r.student_id
440 join Fee f
441 on f.student_id = s.student_id
442 where r.student_id = 10107

```

27:442

Result Grid

student_id	unit_code	semester	registration_date	status	no_penalty_deadline	dropped_out_date	description
10107	IT404	2025S1	2025-01-12	active	2025-02-11	NULL	Tuition Fee

Result 129

Action Output

Time	Action	Response	Duration / Fetch Time
1... 22:25:53	Select distinct r.student_id, r.unit_code, r.semester, r.re...	1 row(s) returned	0.0034 sec / 0.0000...

**TEST CASE 8:** Student drops from A NON- ADVANCED unit after no penalty deadline, so the penalty fee goes into the total fee\_amount for the student (now the fee\_amount = tuition fee + penalty fee), and the dropped\_out\_date and status in registration table is changed

**BEFORE**

418 • `Select r.student_id, r.unit_code, r.semester, r.dropped_out_date, r.status, f.fee_amount`  
 419 `From Registration r`  
 420 `join Student s`  
 421 `on s.student_id = r.student_id`  
 422 `join Fee f`  
 423 `on f.student_id = s.student_id`  
 424 `where r.student_id = 10101`  
 425 `and r.semester = '2024S2';`

Result Grid

student_id	unit_code	semester	dropped_out_date	status	fee_amount
10101	IT103	2024S2		active	3000.00

Result 100

Action Output

Time	Action	Response	Duration / Fetch Time
1... 20:52:41	Select * From Registration r join Student s on s.student...	1 row(s) returned	0.051 sec / 0.00028 s...
1... 20:53:43	Select r.student_id, r.unit_code, r.semester, r.dropped_o...	1 row(s) returned	0.0026 sec / 0.00002...

## AFTER

message

This student has dropped successfully but with penalty

Result 102

Action Output

Time	Action	Response	Duration / Fetch Time
1... 20:54:42	Select r.student_id, r.unit_code, r.semester, r.dropped_o...	1 row(s) returned	0.0011 sec / 0.00001...
1... 20:54:48	Call Rules (10101, 'IT103', '2024-10-23', 'NO', '2024-12...	1 row(s) returned	0.013 sec / 0.000003...

418 • `Select r.student_id, r.unit_code, r.semester, r.dropped_out_date, r.status, f.fee_amount`  
 419 `From Registration r`  
 420 `join Student s`  
 421 `on s.student_id = r.student_id`  
 422 `join Fee f`  
 423 `on f.student_id = s.student_id`  
 424 `where r.student_id = 10101`  
 425 `and r.semester = '2024S2';`

Result Grid

student_id	unit_code	semester	dropped_out_date	status	fee_amount
10101	IT103	2024S2	2024-12-26	dropped_with_penalty	3400.00

Result 103

Action Output

Time	Action	Response	Duration / Fetch Time
1... 20:54:48	Call Rules (10101, 'IT103', '2024-10-23', 'NO', '2024-12...	1 row(s) returned	0.013 sec / 0.000003...
1... 20:55:06	Select r.student_id, r.unit_code, r.semester, r.dropped_o...	1 row(s) returned	0.0017 sec / 0.00000...

**TEST CASE 9:** Student drops from an ADVANCED unit after no penalty deadline, so the penalty fee goes into the total fee\_amount for the student (now the fee\_amount = tuition fee + penalty fee), and the dropped\_out\_date and status in registration table is changed

422 • `Select r.student_id, r.unit_code, r.semester, r.registration_date, r.no_penalty_deadline, r.dropped_out_date, r.status, f.fee_amount, f.description`  
 423 `From Registration r`  
 424 `join Student s`  
 425 `on s.student_id = r.student_id`  
 426 `join Fee f`  
 427 `on f.student_id = s.student_id`  
 428 `where r.student_id = 10105;`  
 429  
 430  
 431  
 432  
 433  
 434

Result Grid

student_id	unit_code	semester	registration_date	no_penalty_deadline	dropped_out_date	status	fee_amount	description
10105	IT401	2022S2	2022-06-10	2022-07-10	NULL	active	4000.00	Tuition Fee

Result 112

Action Output

Time	Action	Response	Duration / Fetch Time
1... 21:24:53	Select * From Registration r join Student s on s.student...	1 row(s) returned	0.013 sec / 0.000013...
1... 21:26:01	Select r.student_id, r.unit_code, r.semester, r.registratio...	1 row(s) returned	0.0018 sec / 0.00001...

421 • `Call BRules (10105, 'IT401', '2022-06-10', 'YES', '2022-07-18', '2022S2', 4000.00, 'DROP');`  
 422  
 423  
 424  
 425  
 426  
 427  
 428  
 429

Result Grid

message
This student has dropped successfully but with penalty

Result 113

Action Output

Time	Action	Response	Duration / Fetch Time
1... 21:27:09	Call BRules (10105, 'IT401', '2022-06-10', 'YES', '2022-0...	1 row(s) returned	0.012 sec / 0.000008...

422 • `Select r.student_id, r.unit_code, r.semester, r.registration_date, r.no_penalty_deadline, r.dropped_out_date, r.status, f.fee_amount, f.description`  
 423 `From Registration r`  
 424 `join Student s`  
 425 `on s.student_id = r.student_id`  
 426 `join Fee f`  
 427 `on f.student_id = s.student_id`  
 428 `where r.student_id = 10105;`  
 429

Result Grid

student_id	unit_code	semester	registration_date	no_penalty_deadline	dropped_out_date	status	fee_amount	description
10105	IT401	2022S2	2022-06-10	2022-07-10	2022-07-18	dropped_with...	4000.00	Dropped Unit Penalty

**TEST CASE 10:** The student drops from the unit BEFORE the penalty deadline, so no penalty fee was charged, though the status and the dropped\_out\_date are changed BEFORE:

428 • `Select r.student_id, r.unit_code, r.semester, r.registration_date, r.status, r.no_penalty_deadline, r.dropped_out_date, f.fee_Amount, f.description`  
 429 `From Registration r`  
 430 `join Student s`  
 431 `on s.student_id = r.student_id`  
 432 `join Fee f`  
 433 `on f.student_id = s.student_id`  
 434 `where r.student_id = 10106;`  
 435  
 436  
 437

Result Grid

student_id	unit_code	semester	registration_date	status	no_penalty_deadline	dropped_out_date	fee_Amount	description
10106	IT102	2024S1	2024-01-13	active	2024-02-12	NULL	3000.00	Tuition Fee

Result 119

Action Output

Time	Action	Response	Duration / Fetch Time
1... 22:02:45	Select r.student_id, r.unit_code, r.semester, r.registratio...	1 row(s) returned	0.0014 sec / 0.00001...

AFTER:

1:427

Result Grid

Filter Rows: Search Export:

message

This student has dropped successfully without penalty

Result 120 Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
1...	22:03:30	Call BRules (10106, 'IT102', '2024-01-10', 'NO', '2024-0...	1 row(s) returned	0.023 sec / 0.000013...

```
428 • Select r.student_id, r.unit_code, r.semester, r.registration_date, r.status, r.no_penalty_deadline, r.dropped_out_date, f.fee_Amount, f.description
429 From Registration r
430 join Student s
431 on s.student_id = r.student_id
432 join Fee f
433 on f.student_id = s.student_id
434 where r.student_id = 10106;
435
```

1:436

Result Grid

Filter Rows: Search Export:

student_...	unit_code	semester	registration_date	status	no_penalty_deadline	dropped_out_date	fee_Amount	description
10106	IT102	2024S1	2024-01-13	dropped_without_penalty	2024-02-12	2024-01-21	3000.00	Tuition Fee

Result 121 Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
1...	22:03:48	Select r.student_id, r.unit_code, r.semester, r.registratio...	1 row(s) returned	0.0011 sec / 0.00001...

TEST CASE 11: The data entry in the parameters for the procedure is somewhat not corresponding with those existed, a message was shown so that viewers know they pass the wrong data.

```
427 • Call BRules (10104, 'IT401', '2022-06-10', 'YES', '2022-07-18', '202252', 4000.00, 'DROP');
428
```

92:427

Action Output

	Time	Action	Response	Duration / Fetch Time
1...	22:05:37	Call BRules (10104, 'IT401', '2022-06-10', 'YES', '2022-0...	Error Code: 1644. The registration record does not exist for the given student, unit, and semester	0.016 sec

TEST CASE 12: The registration date passed is null, meaning that the student has never registered, which is incorrect, since no student could drop if he/she hasn't enrolled yet

```
428 • Call BRules (10102, 'IT405', null, 'YES', '2022-07-18', '202252', 4000.00, 'DROP');
429
```

1:429

Action Output

	Time	Action	Response	Duration / Fetch Time
1...	22:15:08	Call BRules (10102, 'IT405', null, 'YES', '2022-07-18', '2...	Error Code: 1644. An error since no registration date has been found	0.031 sec