Database Systems Laboratory Lab 8 : Database Programming

Tanyawat Vittayapalotai 6031763021

Question 1

Each insertion of professor information, the data are inserted into not only professor table but also into faculty_insurance table that credit_limit value is calculated from 300% of his/her salary and ins_plan is "Group Insurance for Instructor". (**trigger name: new_professor_added)

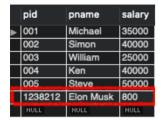
SQL

```
CREATE TRIGGER new_professor_added
AFTER INSERT ON Professor
FOR EACH ROW
INSERT INTO faculty_insurance (
    ref_id,
   ins_plan,
   credit limit,
   duedate,
   s_timestamp,
   status
)
VALUES (
    new.pid,
    "Group Insurance for Instructor",
    3 * new.salary,
    DATE_ADD(SYSDATE(), INTERVAL 4, YEAR),
    SYSDATE(),
    'A'
);
INSERT INTO Professor (pid, pname, salary)
VALUES ('1238212', 'Elon Musk', 800);
SELECT * FROM faculty_insurance;
```

Results

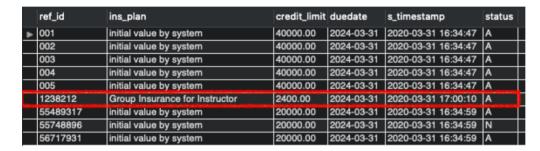
Professors Table

After Insert new professor



Faculty Insurance Table

After Trigger



Question 2

Convert the number declared in a numerical data type to other currencies using function named fn_currency(input_number, exchange_rate, currency_name) and return the result as string.

For example:

```
SELECT fn_currency (70,35.00 USD)
```

or try to test on professor table using

```
SELECT *, fn_currency (70,35.00 USD) FROM Professor"
```

Expected result of fn currency must be 2 USD

SQL

```
CREATE FUNCTION fn currency(
    input_number DECIMAL(65),
    exchange rate DECIMAL(65),
    currency_name VARCHAR(20)
)
RETURNS VARCHAR(99)
DETERMINISTIC
BEGIN
DECLARE res varchar(50);
SET res = CONCAT(
    input_number / exchange_rate,
    currency_name
);
RETURN res;
END$$
DELIMITER;
SELECT *, fn_currency(salary,35,"USD") from Professor;
```

Results

After apply fn_currency

```
SELECT *, fn_currency(70,35,"THB") from Professor
```

	pid	pname	salary	fn_currency(70,35,"TH	
▶	001	Michael	35000	2.0000 THB	
	002	Simon	40000	2.0000 THB	
	003	William	25000	2.0000 THB	
	004	Ken	40000	2.0000 THB	
	005	Steve	50000	2.0000 THB	
	1238212	Elon Musk	800	2.0000 THB	

After applying fn_currency to salary (I believe this would make more sense when selecting professor)

```
SELECT *, fn_currency(salary,35,"USD") from Professor
```

	pid	pname	salary	rn_currency(salary,35,"US
•	001	Michael	35000	1000.0000 USD
	002	Simon	40000	1142.8571 USD
	003	William	30250	864.2857 USD
	004	Ken	40000	1142.8571 USD
	005	Steve	50000	1428.5714 USD
	1238212	Elon Musk	1172	33.4857 USD

Question 3

Update salary of all professors who earns salary less than 30,000 up to 10% and update credit_limit of insurance up to 400 % of new salary and also insert log into system_log table that stores the old salary, new salary, old credit limit and new credit limit. Finally, the data stored procedure has to print the name, old salary, new salary and credit limit of all professor information that are updated. (procedure name: Proc_cal_professor_upvel)

```
CREATE PROCEDURE Proc cal professor upvel()
DETERMINISTIC
BEGIN
IF
(SELECT count(*) FROM Professor WHERE salary < 30000) > 0
# Create new temp table storing ID, old salary and old credit limit
CREATE TEMPORARY TABLE IF NOT EXISTS TEMP PROF OLD (
    PID varchar(16),
    salary INT,
    credit_limit decimal(10,2)
);
TRUNCATE TABLE TEMP PROF OLD;
INSERT INTO TEMP_PROF_OLD (PID, salary, credit_limit)
SELECT pid, salary, faculty_insurance.credit_limit
FROM Professor
INNER JOIN faculty insurance ON ref id = pid
WHERE salary < 30000;
# Update professors incresing salary by 10%
UPDATE Professor SET salary = salary * 1.1
WHERE pid IN (SELECT PID FROM TEMP PROF OLD);
# Update insurance credit limit by 400% of new salary
UPDATE faculty_insurance
INNER JOIN Professor ON ref_id = pid
SET credit_limit = 4 * salary
WHERE ref id IN (SELECT PID FROM TEMP PROF OLD);
# Insert all the values into system log
INSERT INTO system_log (user_log, remark, timestamp)
SELECT o.PID, CONCAT(
    'old salary: ', o.salary,
    ' new salary: ', p.salary,
    ' old credit limit:', o.credit_limit,
    ' new credit limit: ', f.credit_limit
    ). SYSDATE()
FROM TEMP_PROF_OLD o
INNER JOIN Professor p ON p.pid = o.PID
INNER JOIN faculty_insurance f on p.pid = f.ref_id;
# Select values to show
SELECT o.PID, p.name,
    o.salary as old_salary,
    p.salary as new_salary,
    o.credit_limit as old_credit_limit,
    f.credit limit as new credit limit,
    SYSDATE()
FROM TEMP PROF OLD o
INNER JOIN Professor p ON p.pid = o.PID
INNER JOIN faculty_insurance f on p.pid = f.ref_id;
ELSE
SELECT 'Professor < 30000 is empty';</pre>
END IF;
END$$
CALL Proc_cal_professor_upvel();
```

Results

Before Proc_cal_professor_upvel()

Professors table

	pid	pname	salary
•	001	Michael	35000
	002	Simon	40000
	003	William	25000
	004	Ken	40000
	005	Steve	50000
	1238212	Elon Musk	800
	NULL	NULL	NULL

After Proc_cal_professor_upvel()

Results from Proc_cal_professor_upvel()

	PID	pname	old_salary	new_salary	old_credit_limit	new_credit_limit	SYSDATE()
•	003	William	25000	27500	40000.00	110000.00	2020-03-31 18:28:18
	1238212	Elon Musk	800	880	2400.00	3520.00	2020-03-31 18:28:18

system_log table

	id	user_log	remark	timestamp
▶	1	55748896	get F	2020-03-31 16:44:13
	2	55748896	9	2020-03-31 16:44:22
	7	003	old salary: 25000 new salary: 27500 old credit limit:40000.00 new credit limit: 11000	2020-03-31 18:28:18
	8	1238212	old salary: 800 new salary: 880 old credit limit:2400.00 new credit limit: 3520.00	2020-03-31 18:28:18
	NULL	NULL	NULL	NULL

Professor table

	pid	pname	salary
▶	001	Michael	35000
	002	Simon	40000
	003	William	27500
	004	Ken	40000
	005	Steve	50000
	1238212	Elon Musk	880
	NULL	NULL	NULL

faculty_insurance table

	ref_id	ins_plan	credit_limit	duedate	s_timestamp	status
•	001	initial value by system	40000.00	2024-03-31	2020-03-31 16:34:47	Α
	002	initial value by system	40000.00	2024-03-31	2020-03-31 16:34:47	Α
	003	initial value by system	110000.00	2024-03-31	2020-03-31 16:34:47	Α
	004	initial value by system	40000.00	2024-03-31	2020-03-31 16:34:47	Α
	005	initial value by system	40000.00	2024-03-31	2020-03-31 16:34:47	Α
	1238212	Group Insurance for Instructor	3520.00	2024-03-31	2020-03-31 17:00:10	Α
	55489317	initial value by system	20000.00	2024-03-31	2020-03-31 16:34:59	Α
	55748896	initial value by system	20000.00	2024-03-31	2020-03-31 16:34:59	N
	56717931	initial value by system	20000.00	2024-03-31	2020-03-31 16:34:59	Α