Skilaverkefni 3 – Kolbeinn Ingólfsson

/\*1\*/

create database 0908002640\_company;

use 0908002640\_company;

/\*2\*/

create table deptsal as select dept\_no, 0 as totalsalary from dept;

/\*quickly execute commands

----------------------------\*/

select \* from deptsal;

select \* from dept;

select \* from employee;

/\*--------------------------\*/

/\*3.1\*/

delimiter //

/\*3.2\*/

create procedure updateSalary(in department\_number int)

begin

update deptsal

set totalsalary = (select sum(salary) from employee where department\_number = dept\_no)

where department\_number = dept\_no;

end //

/\*3.3\*/

delimiter ;

/\*3.4\*/

call updateSalary(3);

/\*3.5\*/

select \* from deptsal;

/\*4\*/

select \* from information\_schema.routines

where routine\_type = 'PROCEDURE'

and routine\_schema = "0908002640\_company";

/\*5\*/

drop procedure updateSalary;

/\*6\*/

update deptsal

set totalsalary = 0;

/\*7\*/

delimiter //

CREATE PROCEDURE updateSalaryLoop()

BEGIN

DECLARE v INT;

DECLARE total int;

SET v = 1;

WHILE v < (select (count(\*)+1) from dept) DO

set total = (select sum(salary) from employee where v = dept\_no);

if total is not null then

update deptsal

set totalsalary = (select sum(salary) from employee where v = dept\_no)

where v = dept\_no;

else

update deptsal

set totalsalary = 0

where v = dept\_no;

end if;

SET v = v + 1;

END WHILE;

END; //

delimiter ;

call updateSalaryLoop;

/\*8\*/

delimiter //

create procedure updateSalaryCursor()

begin

declare done int default 0;

declare current\_dept\_no int;

declare total int;

declare dept\_no\_cursor CURSOR FOR SELECT dept\_no FROM deptsal;

declare continue handler for not found set done = 1;

open dept\_no\_cursor;

repeat

fetch dept\_no\_cursor into current\_dept\_no;

set total = (select sum(salary) from employee where dept\_no = current\_dept\_no);

if total is not null then

update deptsal

set totalsalary = total

where dept\_no = current\_dept\_no;

else

update deptsal

set totalsalary = 0

where dept\_no = current\_dept\_no;

end if;

until done

end repeat;

close dept\_no\_cursor;

end //

delimiter ;

drop procedure updateSalaryCursor;

call updateSalaryCursor;

/\*9\*/

delimiter //

create procedure raise10()

BEGIN

update deptsal

set totalsalary = (totalsalary \* 1.10);

END //

delimiter ;

call raise10();

/\*10. a\*/

delimiter //

create trigger update\_salaryA

after insert on employee

for each row

begin

if new.dept\_no is not null then

update deptsal

set totalsalary = totalsalary + new.salary

where dept\_no = new.dept\_no;

end if;

end ; //

delimiter ;

/\*test deptsal before insert into employee\*/

select \* from deptsal;

insert into employee

values (9, "Kolbeinn Ingólfsson", NULL, 4, 500000);

/\*test deptsal after insert into employee\*/

select \* from deptsal;

/\*10. b\*/

delimiter //

create trigger update\_salaryB

after delete on employee

for each row

begin

update deptsal

set totalsalary = 0;

call updateSalaryLoop;

end ; //

delimiter ;

/\*test deptsal before deleting an employee\*/

select \* from deptsal;

delete from employee where emp\_id = 9;

/\*test deptsal after deleting an employee\*/

select \* from deptsal;