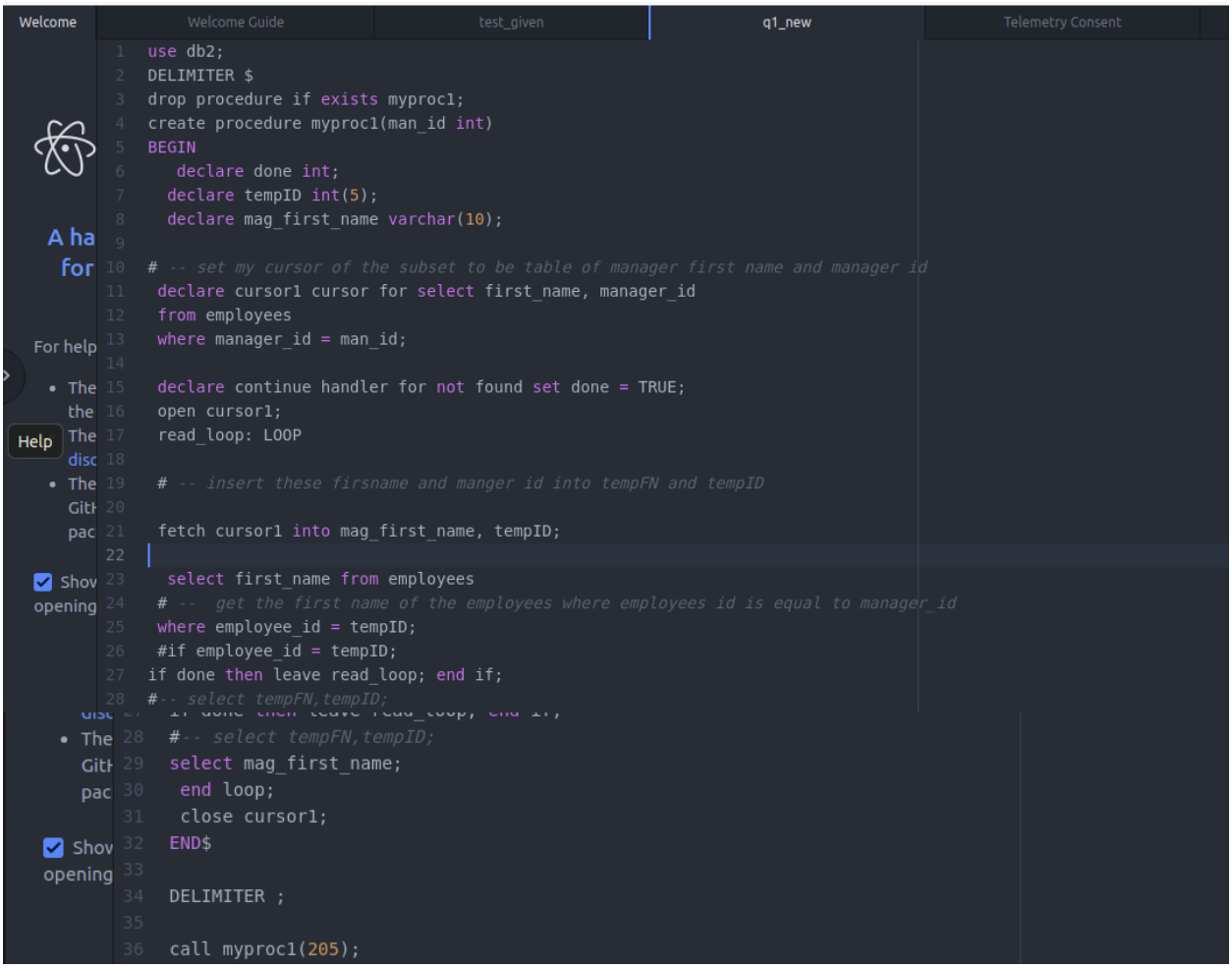
1. Create a procedure using a CURSOR that accepts a manager ID and displays the manager's name followed by the names of the employees who report to this manager.




```
use db2;
DELIMITER $
drop procedure if exists myproc1;
create procedure myproc1(man_id int)
BEGIN
   declare done int;
  declare tempID int(5);
  declare mag_first_name varchar(10);

# -- set my cursor of the subset to be table of manager first name and manager id
 declare cursor1 cursor for select first_name, manager_id
 from employees
 where manager_id = man_id;

 declare continue handler for not found set done = TRUE;
 open cursor1;
 read_loop: LOOP

 # -- insert these firsname and manger id into tempFN and tempID

 fetch cursor1 into mag_first_name, tempID;

  select first_name from employees
 # --  get the first name of the employees where employees id is equal to manager_id
 where employee_id = tempID;
 #if employee_id = tempID;
if done then leave read_loop; end if;
#-- select tempFN,tempID;
 #-- select tempFN,tempID;
 select mag_first_name;
  end loop;
  close cursor1;
 END$

 DELIMITER ;

 call myproc1(205);
```

```
mysql> source q1_new;
Database changed
Query OK, 0 rows affected (0.02 sec)

Query OK, 0 rows affected, 1 warning (0.02 sec)

+------------+
| first_name |
+------------+
| Shelley    |
+------------+
1 row in set (0.00 sec)

+-----------------+
| mag_first_name |
+-----------------+
| William         |
+-----------------+
1 row in set (0.01 sec)

+------------+
| first_name |
+------------+
| Shelley    |
+------------+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.01 sec)
```

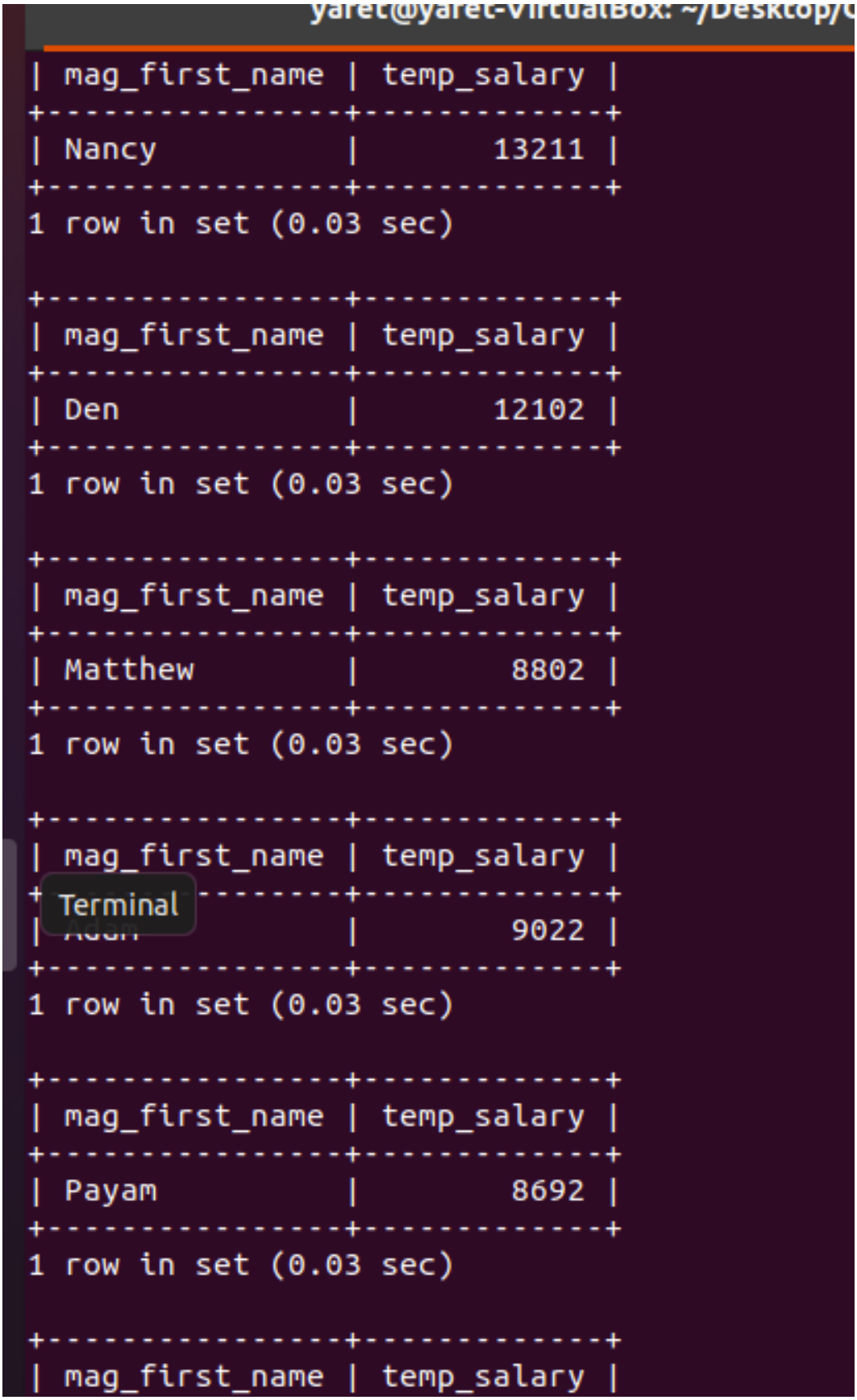
2. Create a procedure with a cursor that provides a 10% salary increase to department managers.

The screen shot below is the result Before the code run

```
mysql>
mysql>
mysql> select distinct mgr.first_name, mgr.salary
    -> from employees emp join employees mgr
    -> on emp.manager_id = mgr.employee_id;
+------------+----------+
| first_name | salary   |
+------------+----------+
| Steven     | 24000.00 |
| Neena      | 17000.00 |
| Lex        | 17000.00 |
| Alexander  |  9000.00 |
| Nancy      | 12008.00 |
| Den        | 11000.00 |
| Matthew    |  8000.00 |
| Adam       |  8200.00 |
| Payam      |  7900.00 |
| Shanta     |  6500.00 |
| Kevin      |  5800.00 |
| John       | 14000.00 |
| Karen      | 13500.00 |
| Alberto    | 12000.00 |
| Gerald     | 11000.00 |
| Eleni      | 10500.00 |
| Michael    | 13000.00 |
| Shelley    | 12008.00 |
+------------+----------+
18 rows in set (0.00 sec)

mysql>
```

```
use db2;
DELIMITER $
drop procedure if exists myproc2;
create procedure myproc2(salary_rise int)
BEGIN
   declare done int;
  declare temp_salary int(5);
  declare mag_first_name varchar(10);
# -- set my cursor of the subset to be table of manager first name and manager salary
 declare cursor1 cursor for select distinct mgr.first_name, mgr.salary
 from employees emp join employees mgr
 on emp.manager_id = mgr.employee_id;
 declare continue handler for not found set done = TRUE;
 open cursor1;
 read_loop: LOOP
# -- as the cursor is been read fetch the rows and read them as mag_first_name and temp_salary
 fetch cursor1 into mag_first_name, temp_salary;

 update employees
   set salary = temp_salary * (1 + (salary_rise/100))
    where salary = temp_salary;

if done then leave read_loop; end if;
select mag_first_name,temp_salary;
 end loop;
 close cursor1;
END$
DELIMITER ;
call myproc2(10);
```

```
mysql> source q2;
Database changed
Query OK, 0 rows affected (0.01 sec)

Query OK, 0 rows affected, 1 warning (0.02 sec)

+----------------+-------------+
| mag_first_name | temp_salary |
+----------------+-------------+
| Steven         |       26403 |
+----------------+-------------+
1 row in set (0.01 sec)

+----------------+-------------+
| mag_first_name | temp_salary |
+----------------+-------------+
| Neena          |       18702 |
+----------------+-------------+
1 row in set (0.01 sec)

+----------------+-------------+
| mag_first_name | temp_salary |
+----------------+-------------+
| Lex            |       18702 |
+----------------+-------------+
1 row in set (0.01 sec)

+----------------+-------------+
| mag_first_name | temp_salary |
+----------------+-------------+
| Alexander      |        9902 |
+----------------+-------------+
1 row in set (0.01 sec)

+----------------+-------------+
```


yaret@yaret-VirtualBox: ~/Desktop/C
| mag_first_name | temp_salary |
+----------------+-------------+
| Nancy | 13211 |
+----------------+-------------+
1 row in set (0.03 sec)
+----------------+-------------+
| mag_first_name | temp_salary |
+----------------+-------------+
| Den | 12102 |
+----------------+-------------+
1 row in set (0.03 sec)
+----------------+-------------+
| mag_first_name | temp_salary |
+----------------+-------------+
| Matthew | 8802 |
+----------------+-------------+
1 row in set (0.03 sec)
+----------------+-------------+
| mag_first_name | temp_salary |
+----------------+-------------+
| Adam | 9022 |
+----------------+-------------+
1 row in set (0.03 sec)
+----------------+-------------+
| mag_first_name | temp_salary |
+----------------+-------------+
| Payam | 8692 |
+----------------+-------------+
1 row in set (0.03 sec)
+----------------+-------------+
| mag_first_name | temp_salary |
Terminal

```
+-----------------+-------------+
| mag_first_name | temp_salary |
+-----------------+-------------+
| Shanta          |        7152 |
+-----------------+-------------+
1 row in set (0.03 sec)

+-----------------+-------------+
| mag_first_name | temp_salary |
+-----------------+-------------+
| Kevin           |        6382 |
+-----------------+-------------+
1 row in set (0.03 sec)

+-----------------+-------------+
| mag_first_name | temp_salary |
+-----------------+-------------+
| John            |       15402 |
+-----------------+-------------+
1 row in set (0.03 sec)
```

Terminal

```
+-----------------+-------------+
| mag_first_name | temp_salary |
+-----------------+-------------+
| Karen           |       14852 |
+-----------------+-------------+
1 row in set (0.03 sec)

+-----------------+-------------+
| mag_first_name | temp_salary |
+-----------------+-------------+
| Alberto         |       13202 |
+-----------------+-------------+
1 row in set (0.03 sec)
```

```
+-----------------+-------------+
| Alberto         |       13202 |
+-----------------+-------------+
1 row in set (0.03 sec)

+-----------------+-------------+
| mag_first_name | temp_salary |
+-----------------+-------------+
| Gerald          |       12102 |
+-----------------+-------------+
1 row in set (0.04 sec)

+-----------------+-------------+
| mag_first_name | temp_salary |
+-----------------+-------------+
| Eleni           |       11552 |
+-----------------+-------------+
1 row in set (0.05 sec)

+-----------------+-------------+
| mag_first_name | temp_salary |
+-----------------+-------------+
| Michael         |       14302 |
+-----------------+-------------+
1 row in set (0.06 sec)

+-----------------+-------------+
| mag_first_name | temp_salary |
+-----------------+-------------+
| Shelley         |       13211 |
+-----------------+-------------+
1 row in set (0.06 sec)

Query OK, 0 rows affected (0.06 sec)

mysql>
```

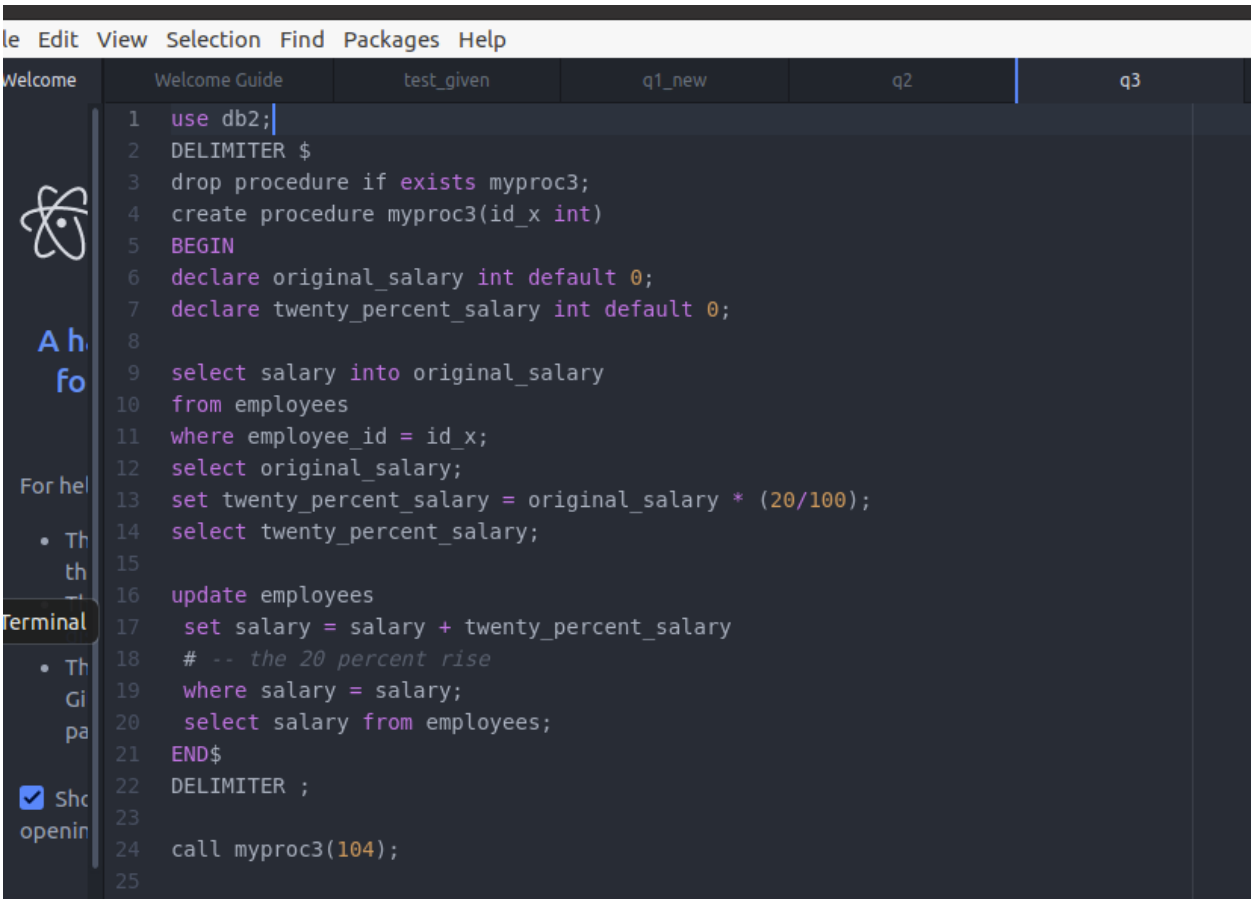
Atom

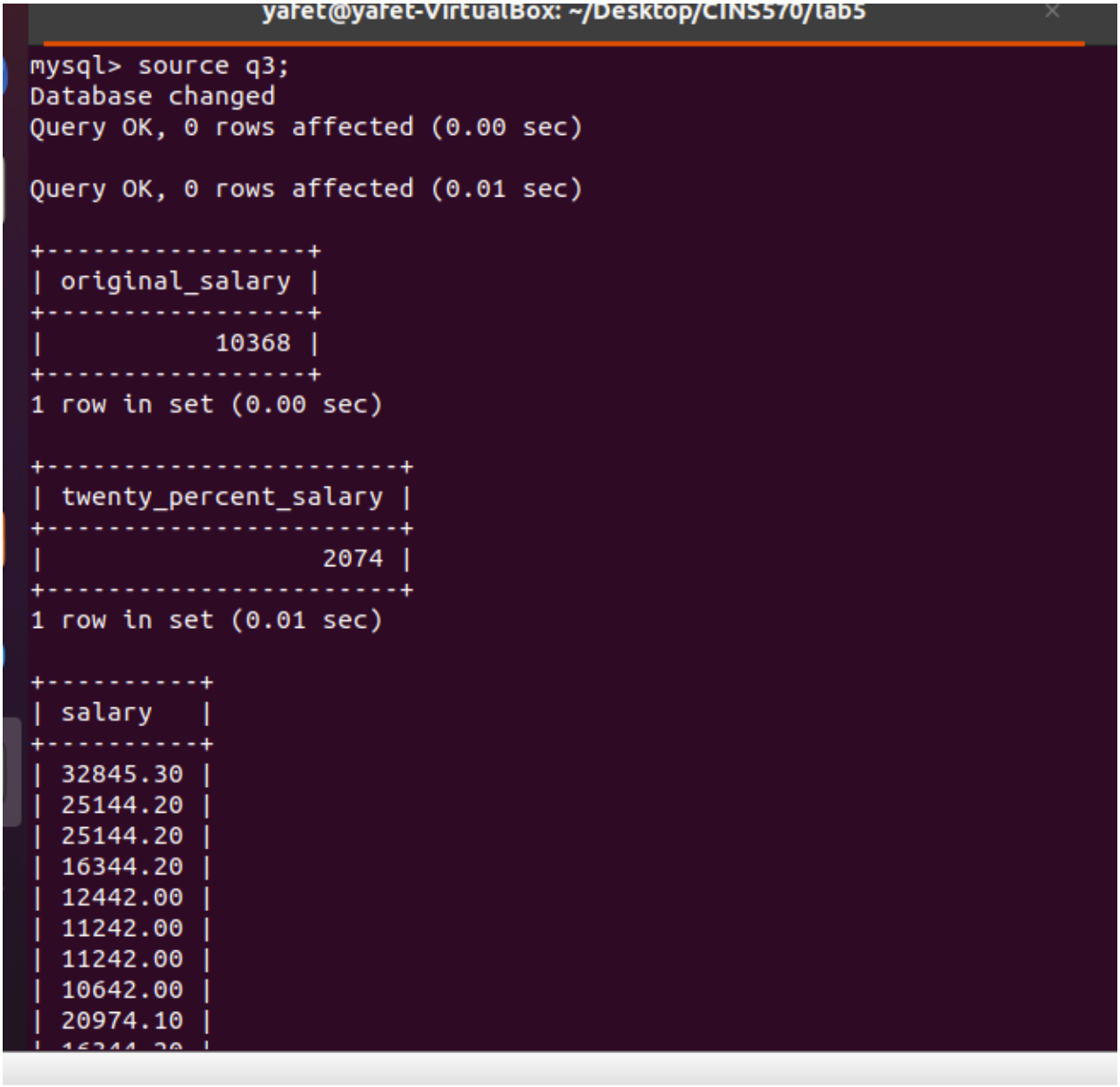
3. Create a procedure that determine and display the amount X needed to give all employees 20% of his/her salary.

The screen shot below is the result Before the code run

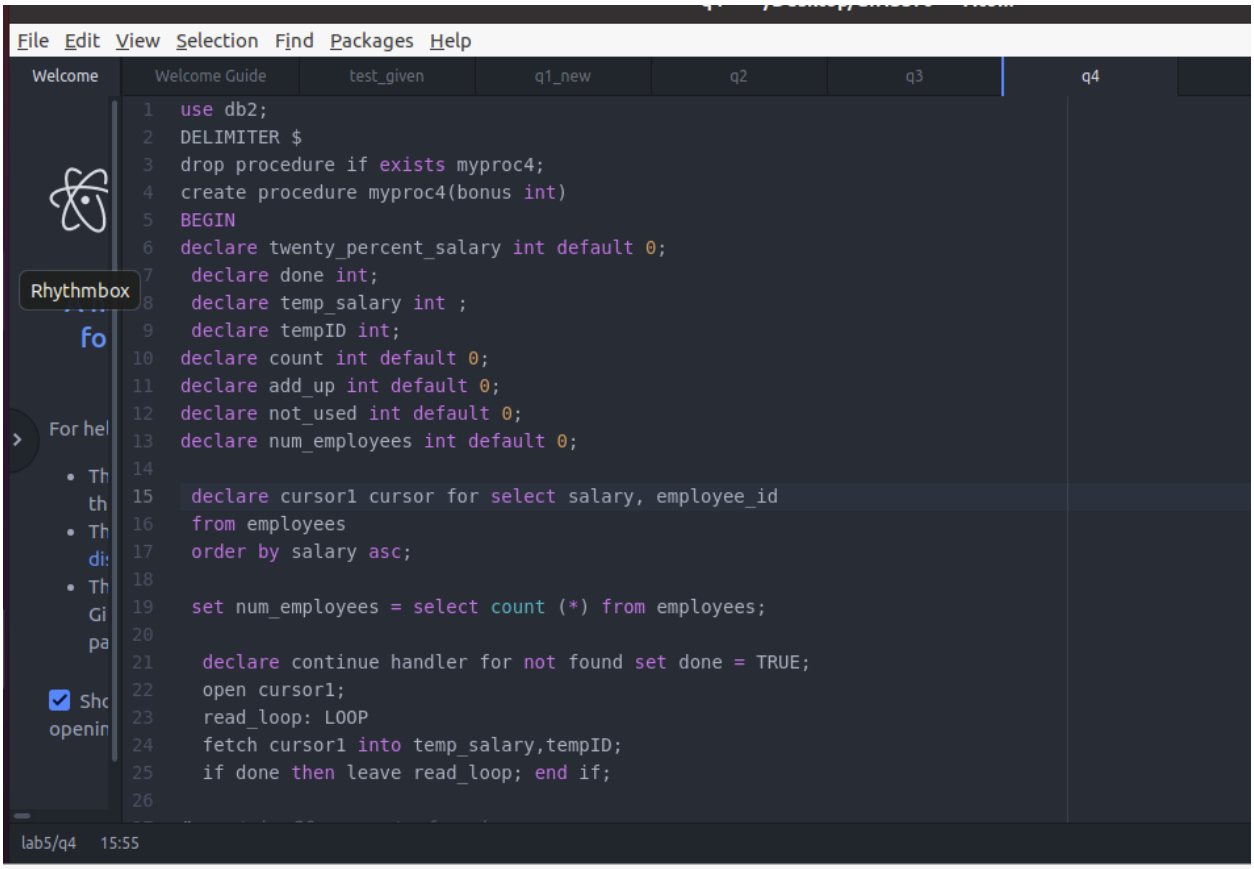
```
Query OK, 107 rows affected (0.06 sec)

mysql> select first_name, employee_id, salary from employees;
+-------------+-------------+----------+
| first_name  | employee_id | salary   |
+-------------+-------------+----------+
| Steven      |         100 | 30771.30 |
| Neena       |         101 | 23070.20 |
| Lex         |         102 | 23070.20 |
| Alexander   |         103 | 14270.20 |
| Bruce       |         104 | 10368.00 |
| David       |         105 |  9168.00 |
| Valli       |         106 |  9168.00 |
| Diana       |         107 |  8568.00 |
| Nancy       |         108 | 18900.10 |
| Daniel      |         109 | 14270.20 |
|             |         110 | 13390.20 |
| Ismael      |         111 | 12068.00 |
| Jose Manuel |         112 | 12168.00 |
| Luis        |         113 | 11268.00 |
| Den         |         114 | 16470.20 |
| Alexander   |         115 |  7468.00 |
| Shelli      |         116 |  7268.00 |
| Sigal       |         117 |  7168.00 |
| Guy         |         118 |  6968.00 |
| Karen       |         119 |  6868.00 |
| Matthew     |         120 | 13170.20 |
```

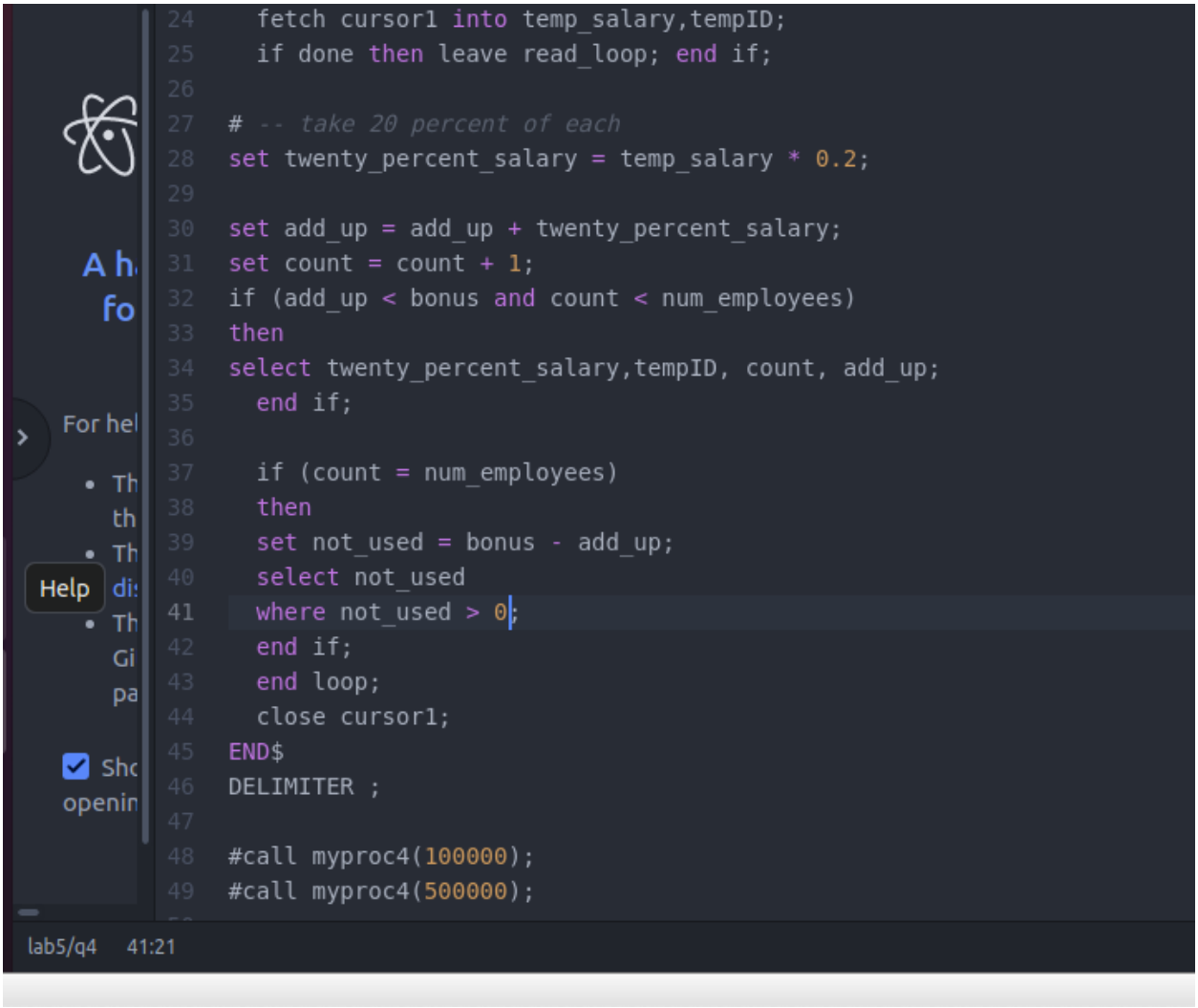
Help


le Edit View Selection Find Packages Help
Welcome
Welcome Guide
test_given
q1_new
q2
q3
Terminal
use db2;
DELIMITER $
drop procedure if exists myproc3;
create procedure myproc3(id_x int)
BEGIN
declare original_salary int default 0;
declare twenty_percent_salary int default 0;
select salary into original_salary
from employees
where employee_id = id_x;
select original_salary;
set twenty_percent_salary = original_salary * (20/100);
select twenty_percent_salary;
update employees
set salary = salary + twenty_percent_salary
# -- the 20 percent rise
where salary = salary;
select salary from employees;
END$
DELIMITER ;
call myproc3(104);



4. Create a procedure. The company wants to award $100,000 of profit by giving a bonus. The bonus is calculated as 20% of an employee's salary. Employees who earn lower salary values are given priority such that the $100,000 profit is distributed starting with the lowest salary earners in ascending order until the $100,000 is exhausted. If the $100,000 is exhausted, display the quantity of employees who received a bonus (condition 1) otherwise, display the portion from the $100,000 not used (condition 2). Run the procedure a second time with a different initial $100,000 amount to test the other condition. Clearly label both test runs.


File Edit View Selection Find Packages Help
Welcome
Welcome Guide
test_given
q1_new
q2
q3
q4
Rhythmbox
use db2;
DELIMITER $
drop procedure if exists myproc4;
create procedure myproc4(bonus int)
BEGIN
declare twenty_percent_salary int default 0;
declare done int;
declare temp_salary int ;
declare tempID int;
declare count int default 0;
declare add_up int default 0;
declare not_used int default 0;
declare num_employees int default 0;
declare cursor1 cursor for select salary, employee_id
from employees
order by salary asc;
set num_employees = select count (*) from employees;
declare continue handler for not found set done = TRUE;
open cursor1;
read_loop: LOOP
fetch cursor1 into temp_salary,tempID;
if done then leave read_loop; end if;
lab5/q4
15:55


fetch cursor1 into temp_salary,tempID;
if done then leave read_loop; end if;
# -- take 20 percent of each
set twenty_percent_salary = temp_salary * 0.2;
set add_up = add_up + twenty_percent_salary;
set count = count + 1;
if (add_up < bonus and count < num_employees)
then
select twenty_percent_salary,tempID, count, add_up;
end if;
if (count = num_employees)
then
set not_used = bonus - add_up;
select not_used
where not_used > 0;
end if;
end loop;
close cursor1;
END$
DELIMITER ;
#call myproc4(100000);
#call myproc4(500000);
Help
lab5/q4 41:21

```
mysql> call myproc4(100000);
```

```
+----------------------+--------+-------+--------+
| twenty_percent_salary | tempID | count | add_up |
+----------------------+--------+-------+--------+
|                 2248 |    106 |    49 |  93532 |
+----------------------+--------+-------+--------+
1 row in set (0.01 sec)

+----------------------+--------+-------+--------+
| twenty_percent_salary | tempID | count | add_up |
+----------------------+--------+-------+--------+
|                 2488 |    104 |    50 |  96020 |
+----------------------+--------+-------+--------+
1 row in set (0.01 sec)

+----------------------+--------+-------+--------+
| twenty_percent_salary | tempID | count | add_up |
+----------------------+--------+-------+--------+
|                 2488 |    202 |    51 |  98508 |
+----------------------+--------+-------+--------+
1 row in set (0.02 sec)

Empty set (0.02 sec)

Query OK, 0 rows affected (0.02 sec)
```

```
mysql> call myproc4(500000);
```

```
+----------+
| not_used |
+----------+
|   217513 |
+----------+
1 row in set (0.02 sec)

Query OK, 0 rows affected (0.02 sec)
```