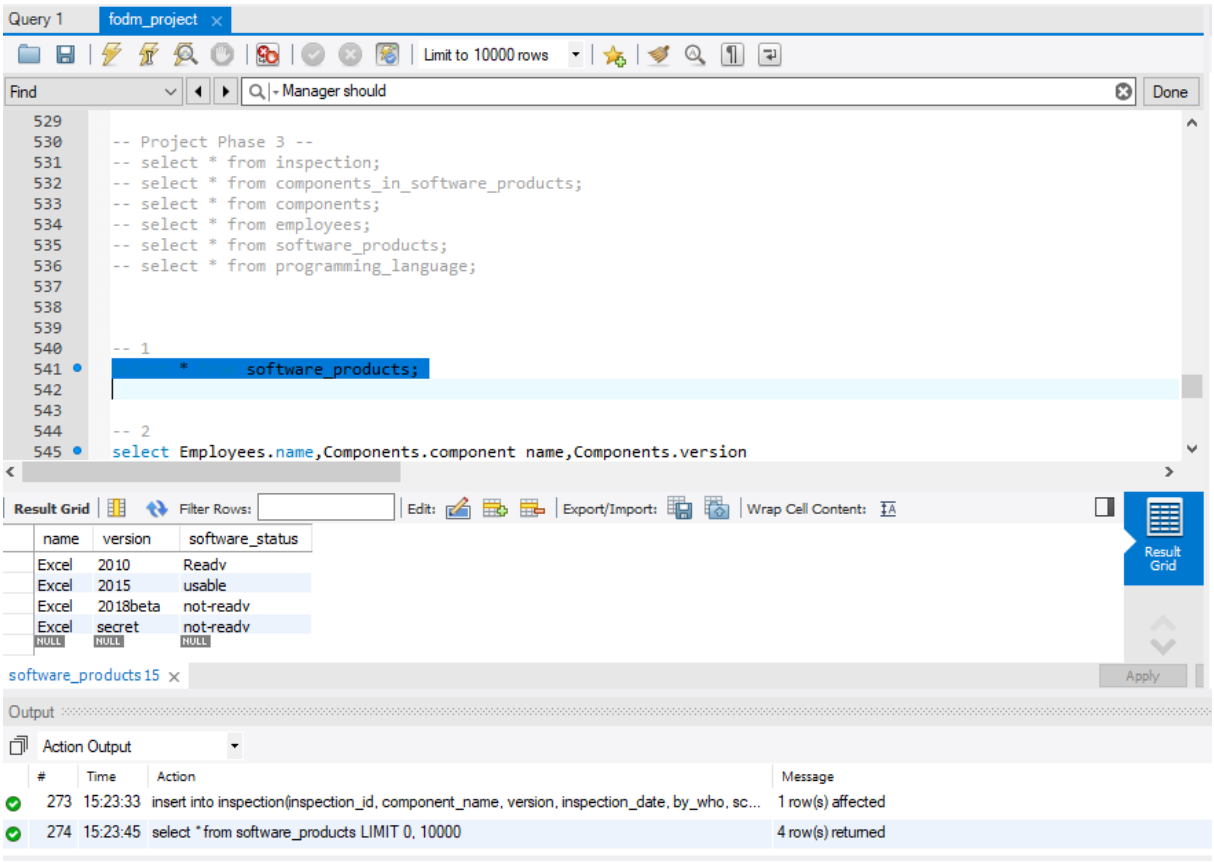
CSE 5330/7330 Fall 2017
Phase 3 Functional Requirements
DHAVAL GOGRI
47444609

Using your database populated with the data provided.

Everyone: Write Queries (and show the results) to answer the following questions:

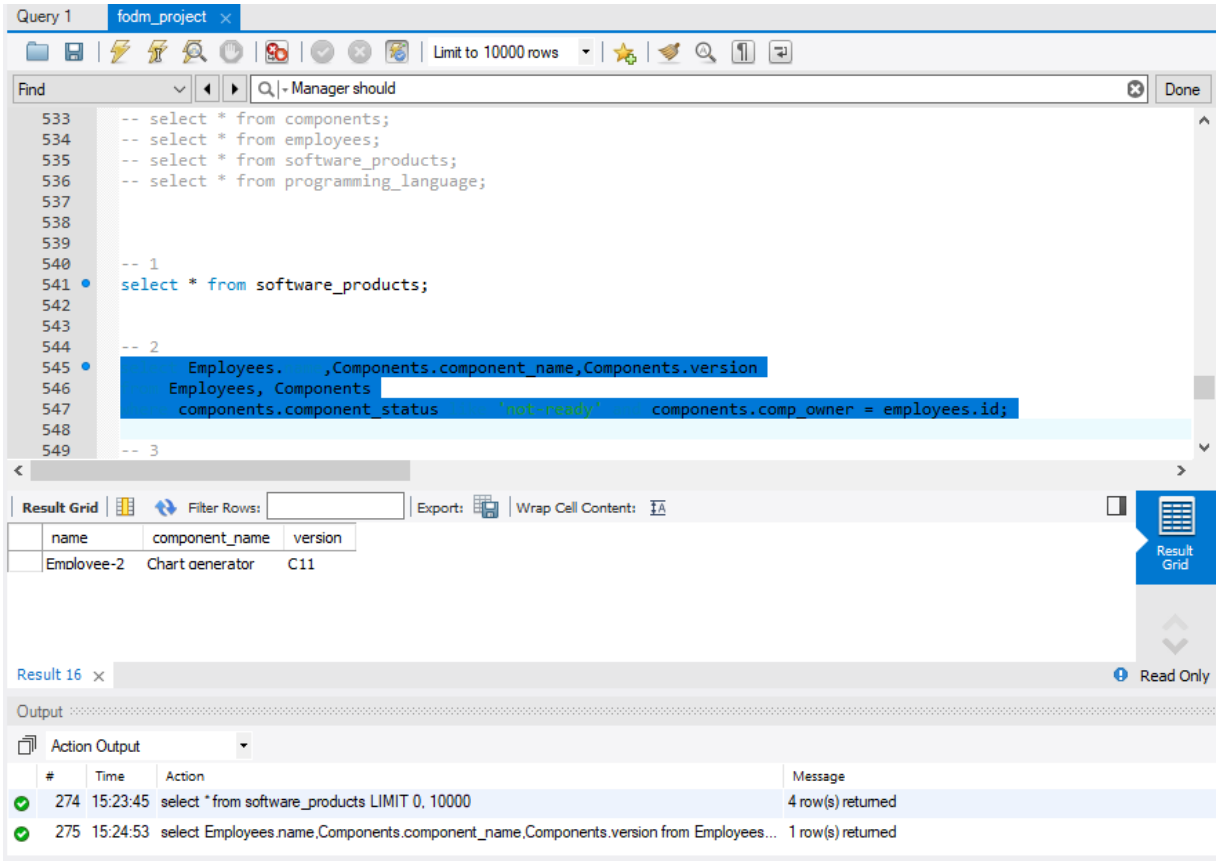
1. List all software product names and versions and current product status.

select * from software_products;



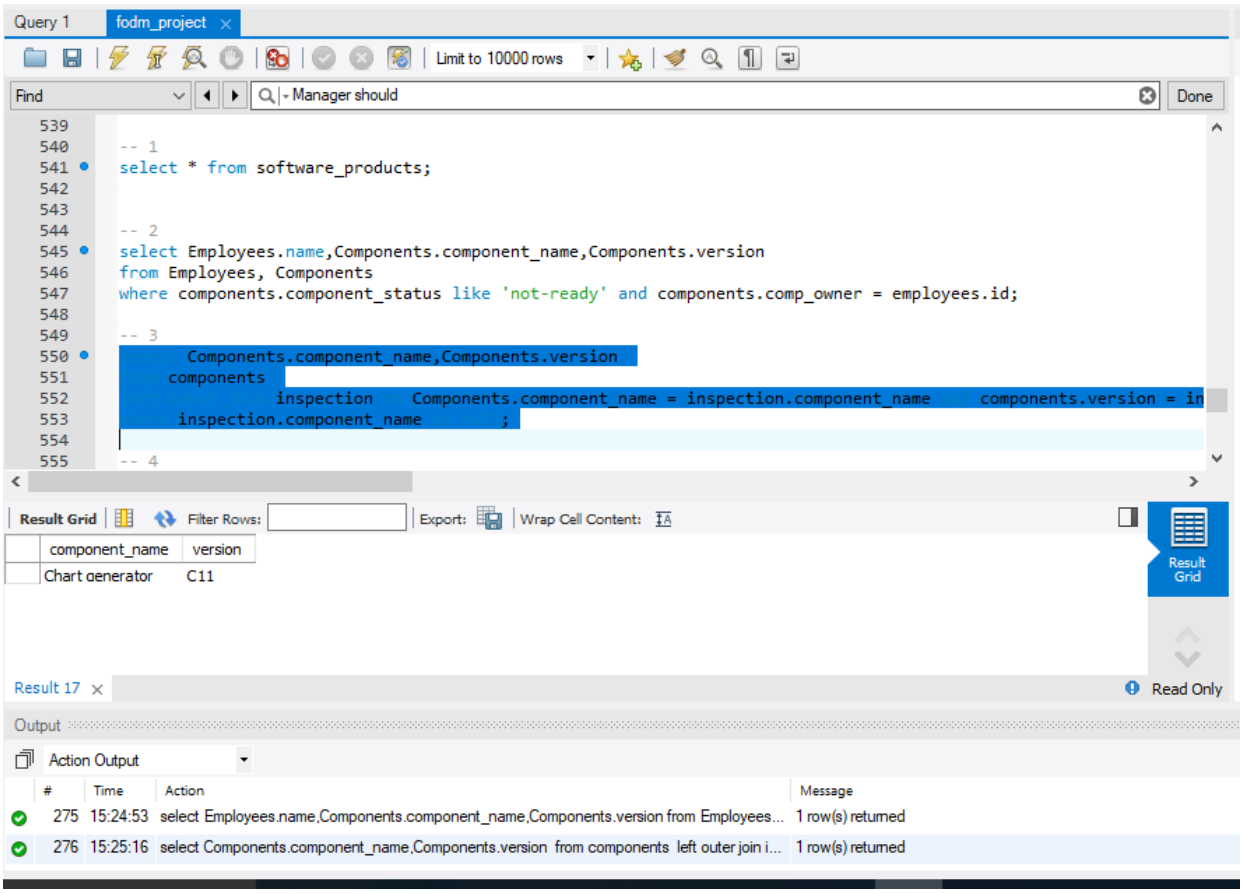
2. List the owner name, component name & version of all “not ready” components.

```
select Employees.name,Components.component_name,Components.version
from Employees, Components
where components.component_status like 'not-ready' and components.comp_owner =
employees.id;
```



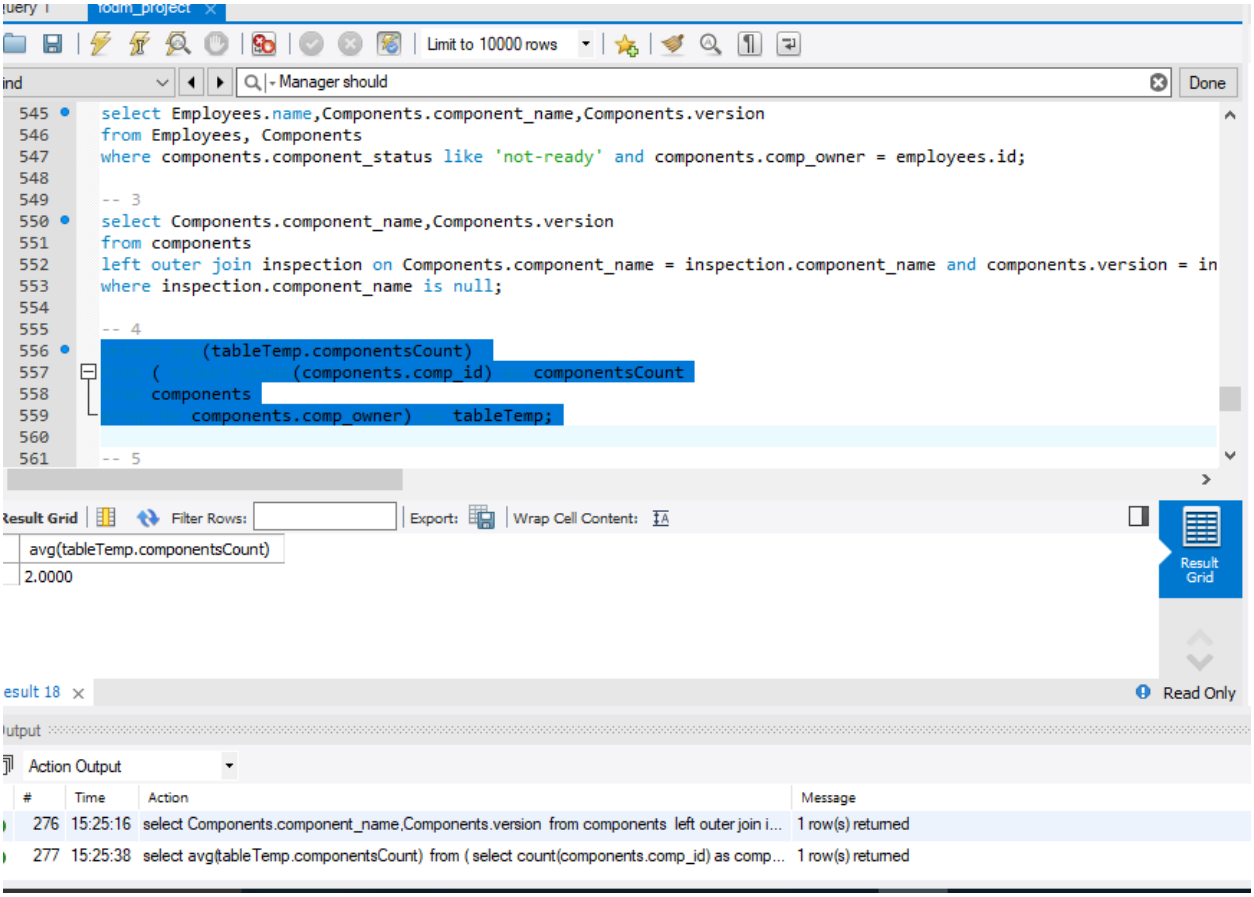
3. List all component names and versions that have not been inspected.

```
select Components.component_name,Components.version
from components
left outer join inspection on Components.component_name = inspection.component_name and
components.version = inspection.version
where inspection.component_name is null;
```



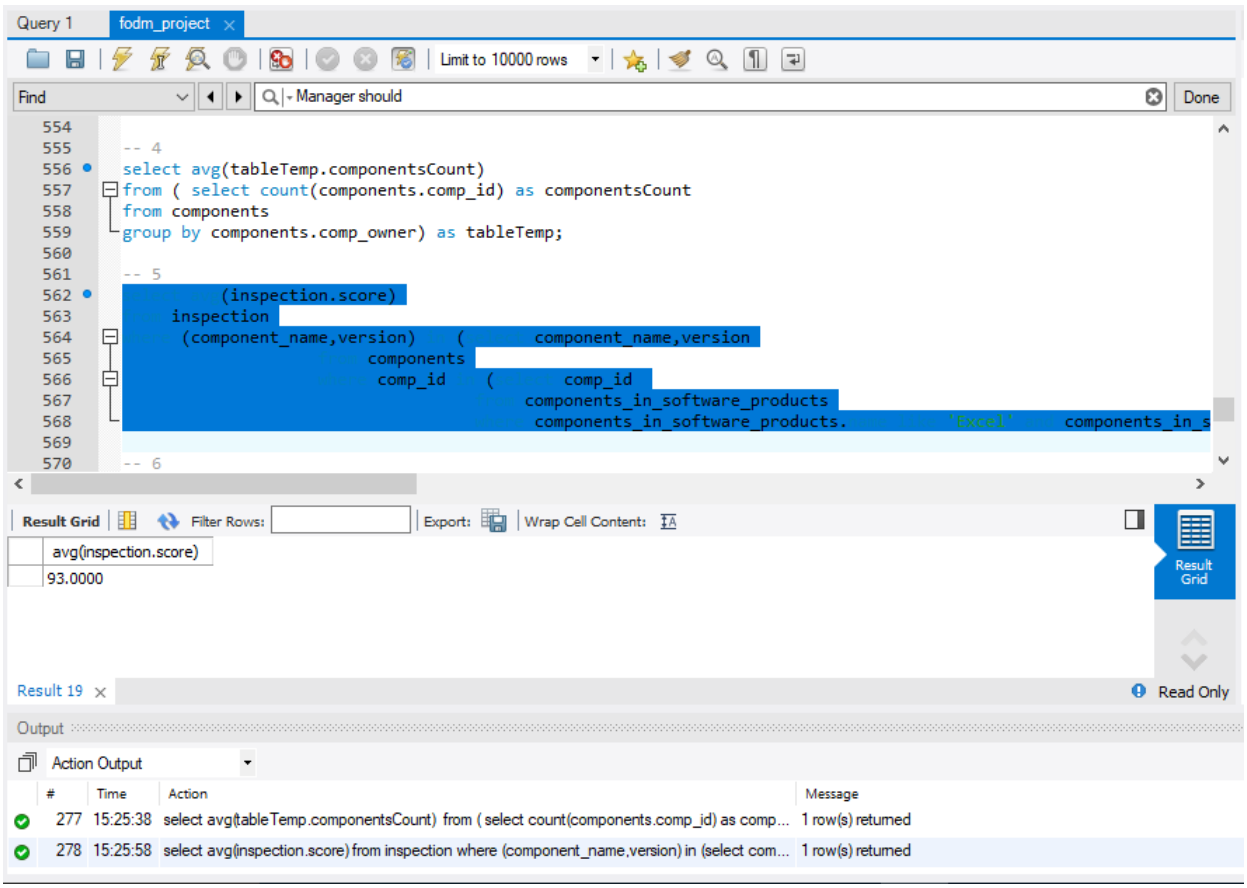
4. What is the average number of components owned per person?

```
select avg(tableTemp.componentsCount)
from ( select count(components.comp_id) as componentsCount
from components
group by components.comp_owner) as tableTemp;
```



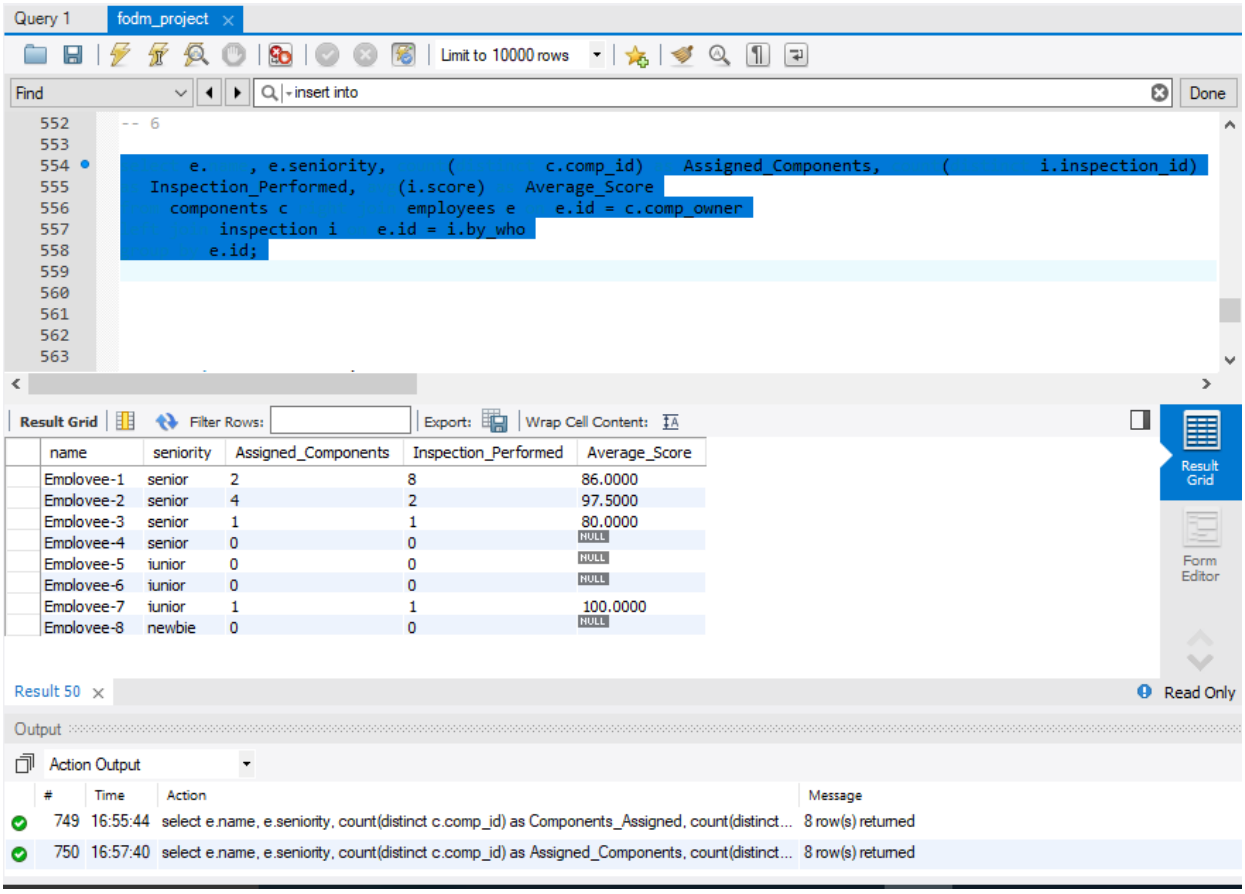
5. What is the average score of all inspections for Excel secret?

```
select avg(inspection.score)
from inspection
where (component_name,version) in (select component_name,version
                                   from components
                                   where comp_id in (select comp_id
                                                     from
components_in_software_products
                                                     where
components_in_software_products.name like 'Excel' and
components_in_software_products.version like 'secret'));
```



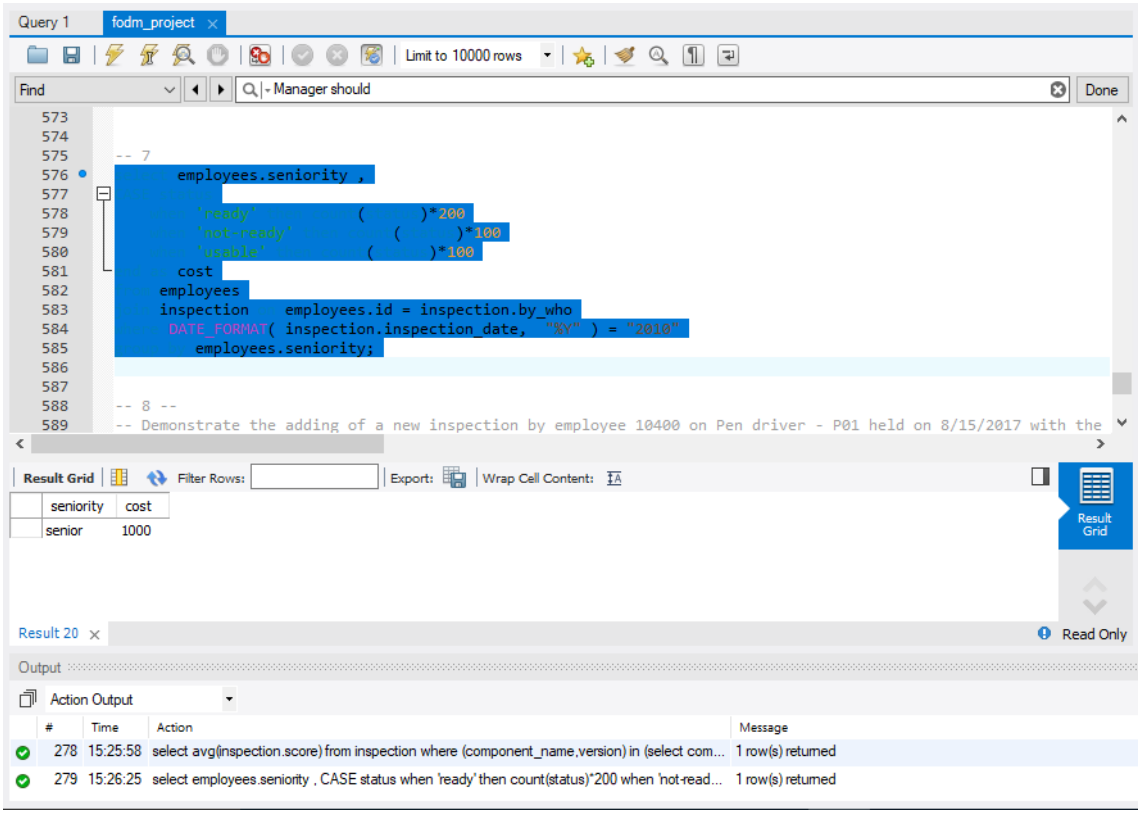
6. List all employees by name, seniority, count of components assigned to them, count of inspections performed by them and their average inspection score.

```
select e.name, e.seniority, count(distinct c.comp_id) as Assigned_Components, count(distinct
i.inspection_id)
as Inspection_Performed, avg(i.score) as Average_Score
from components c right join employees e on e.id = c.comp_owner
left join inspection i on e.id = i.by_who
group by e.id;
```



7. Assume an inspection that results in a “ready” status costs $200, and all other inspections cost $100 each. How much did *OSF* in 2010 for inspections conducted by each seniority level?

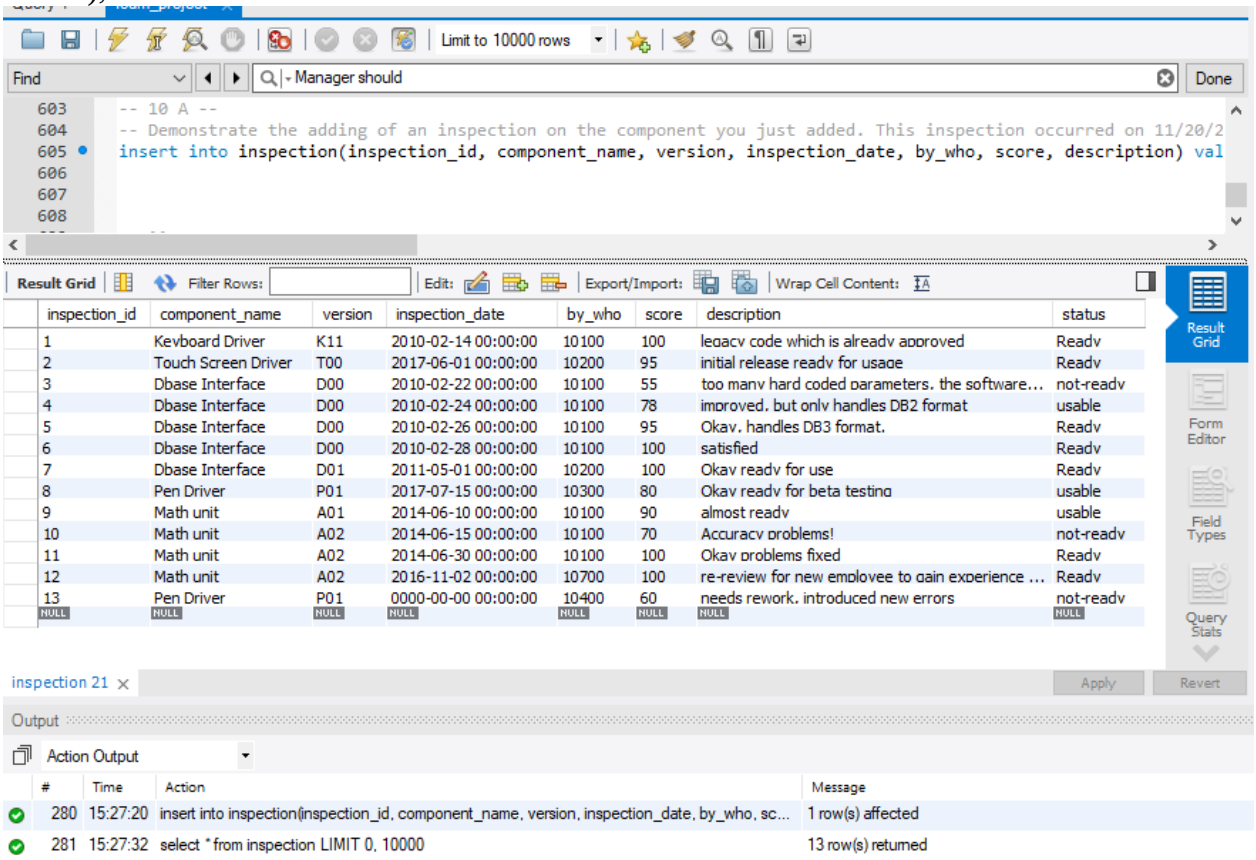
```
select employees.seniority ,
CASE status
        when 'ready' then count(status)*200
        when 'not-ready' then count(status)*100
        when 'usable' then count(status)*100
end as cost
from employees
join inspection on employees.id = inspection.by_who
where DATE_FORMAT( inspection.inspection_date,  "%Y" ) = "2010"
group by employees.seniority;
```



Everyone: Demonstrate ≡ show the SQL command(s) and result

8. Demonstrate the adding of a new inspection by employee 10400 on Pen driver - P01 held on 8/15/2017 with the score of 60 and description of “needs rework, introduced new errors”.

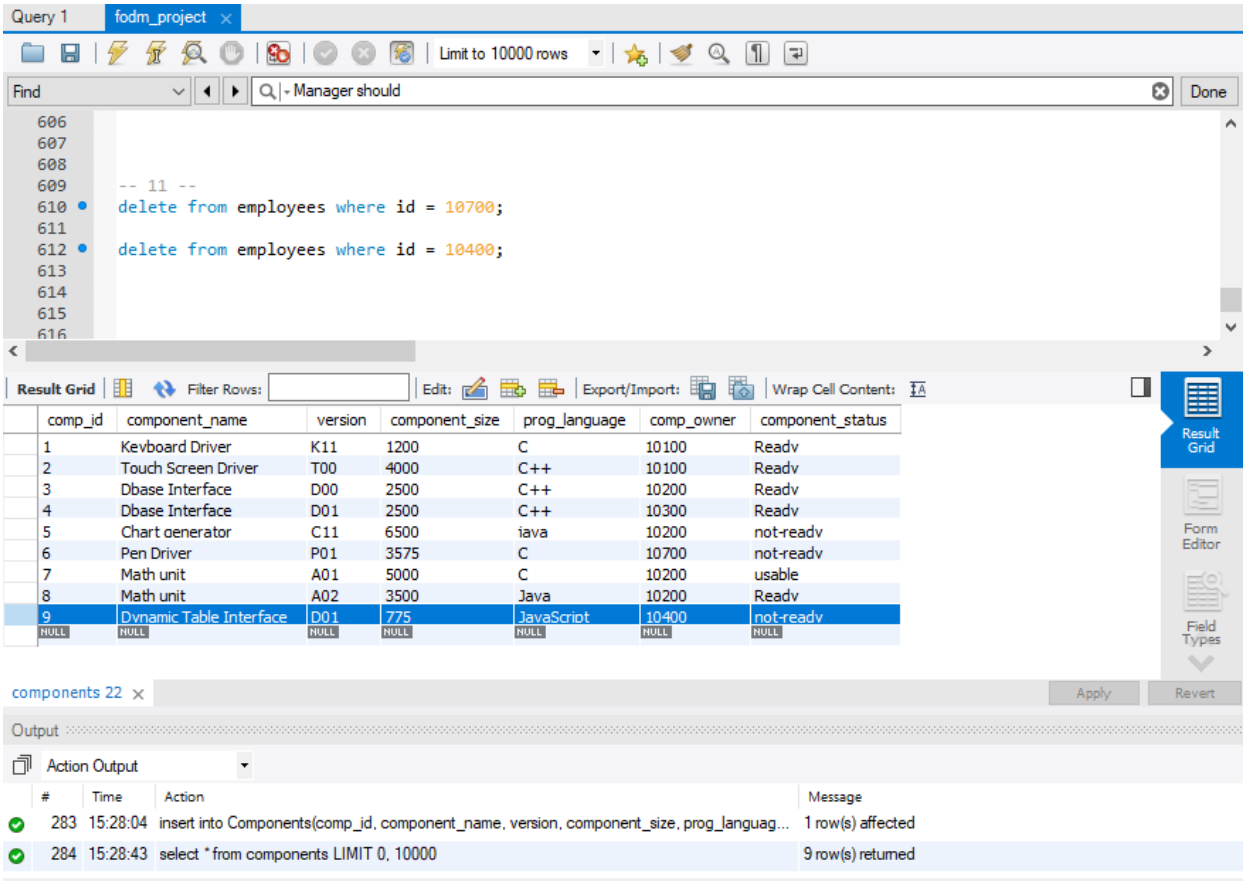
insert into inspection(inspection_id, component_name, version, inspection_date, by_who, score, description) values(13, "Pen Driver", "P01", 08/15/2017, 10400, 60, "needs rework, introduced new errors");



9. A) Demonstrate adding a new component to Excel 2018beta. This new component is named “Dynamic Table Interface”, version D01, and was written in javascript by person 10400, size = 775.

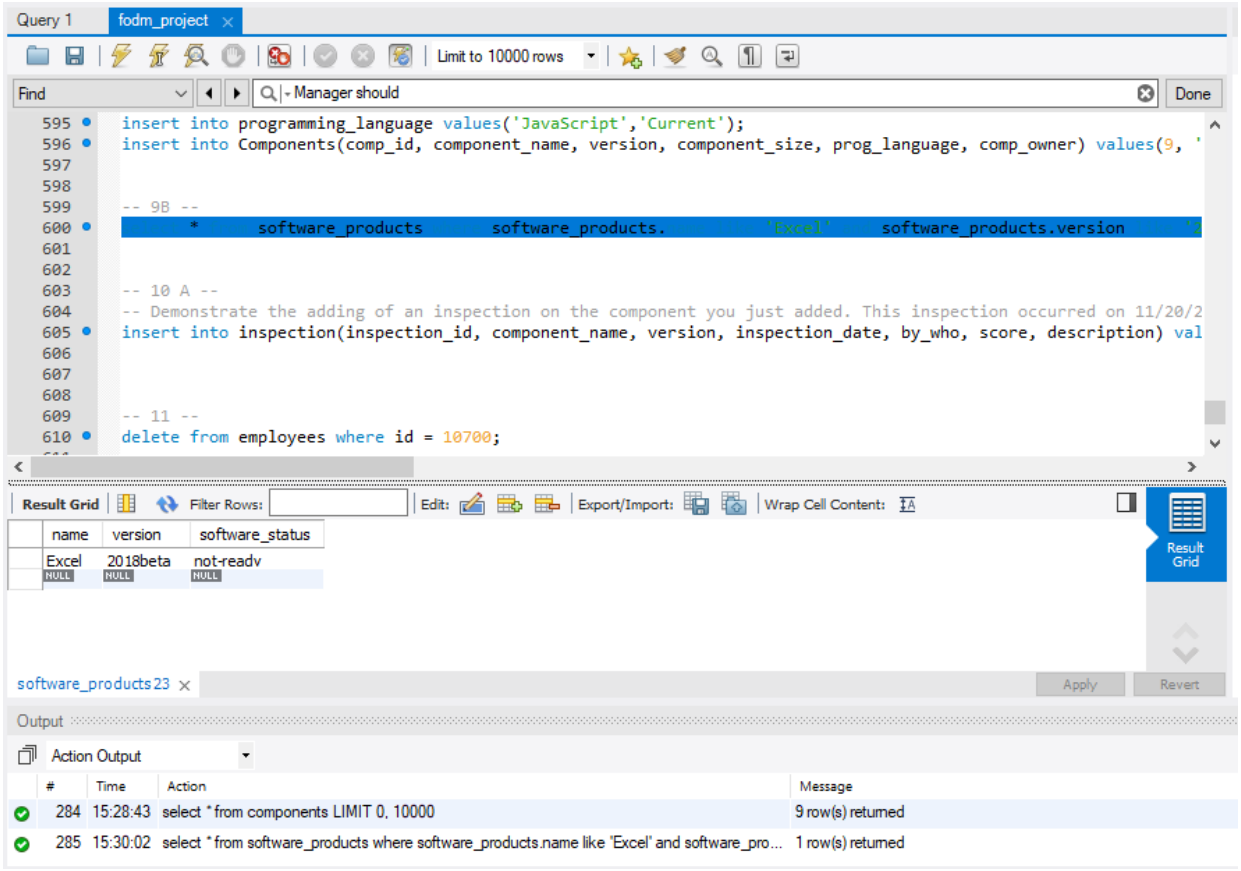
1. insert into programming_language values('JavaScript','Current');

2. insert into Components(comp_id, component_name, version, component_size, prog_language, comp_owner) values(9, 'Dynamic Table Interface', 'D01', 775, 'JavaScript', 10400);



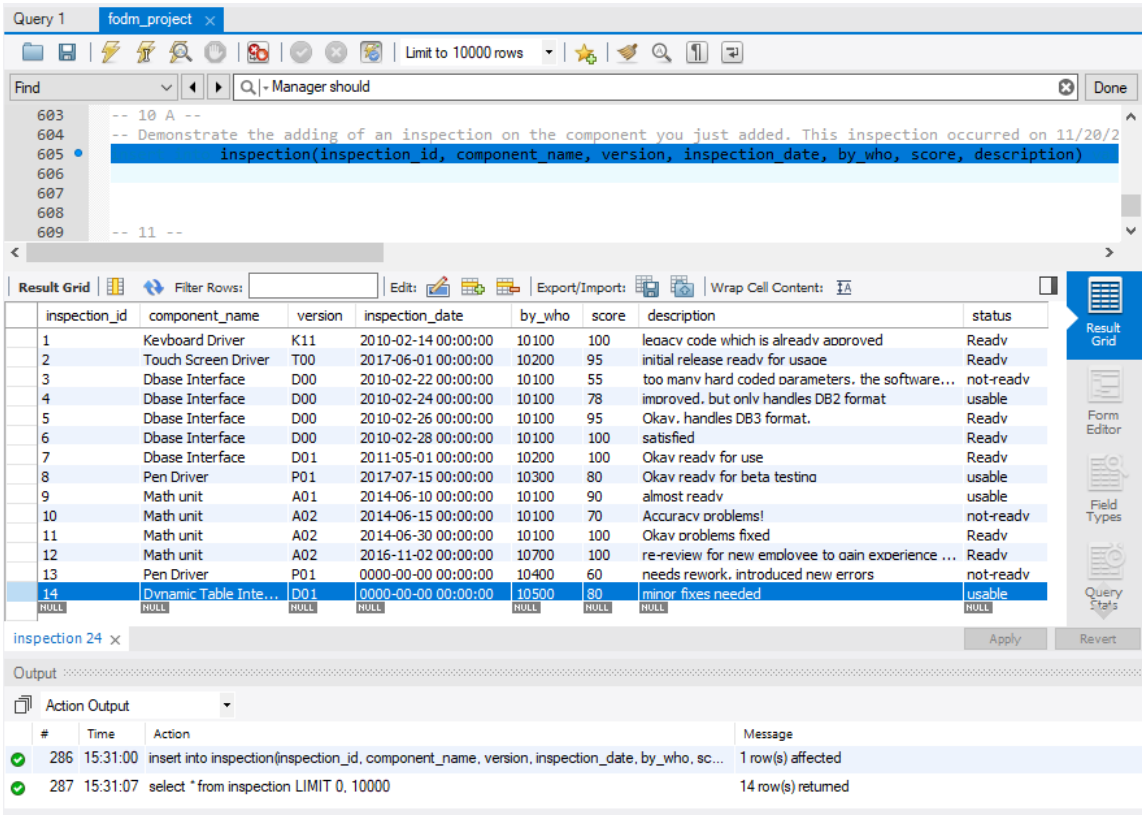
B) What is the Excel 2018beta product status?

select * from software_products where software_products.name like 'Excel' and software_products.version like '2018beta';



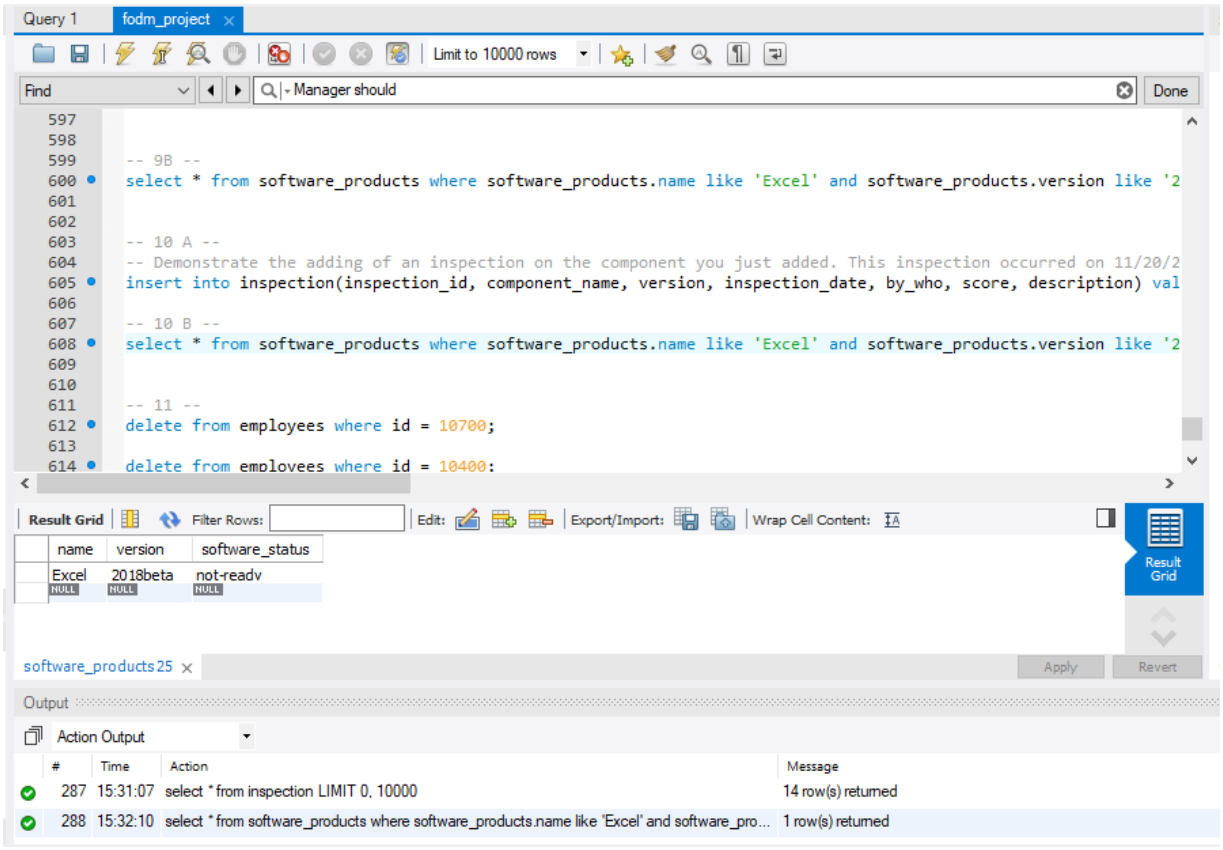
10. A) Demonstrate the adding of an inspection on the component you just added. This inspection occurred on 11/20/2017 by inspector 10500, with a score of 80, and note of “minor fixes needed”.

insert into inspection(inspection_id, component_name, version, inspection_date, by_who, score, description) values(14, "Dynamic Table Interface", "D01", 11/20/2017, 10500, 80, "minor fixes needed");



B) What is the Excel 2018beta product status?

select * from software_products where software_products.name like 'Excel' and software_products.version like '2018beta';



GRADUATE:

11. Person 10700 has decided to leave *OSF* for other employment. Implement a solution for this situation.

Assumption:
1. When an employee leaves the company, it's work on the project should not be deleted. I.e. the components the employee has made or the inspection the employee has performed on any project.

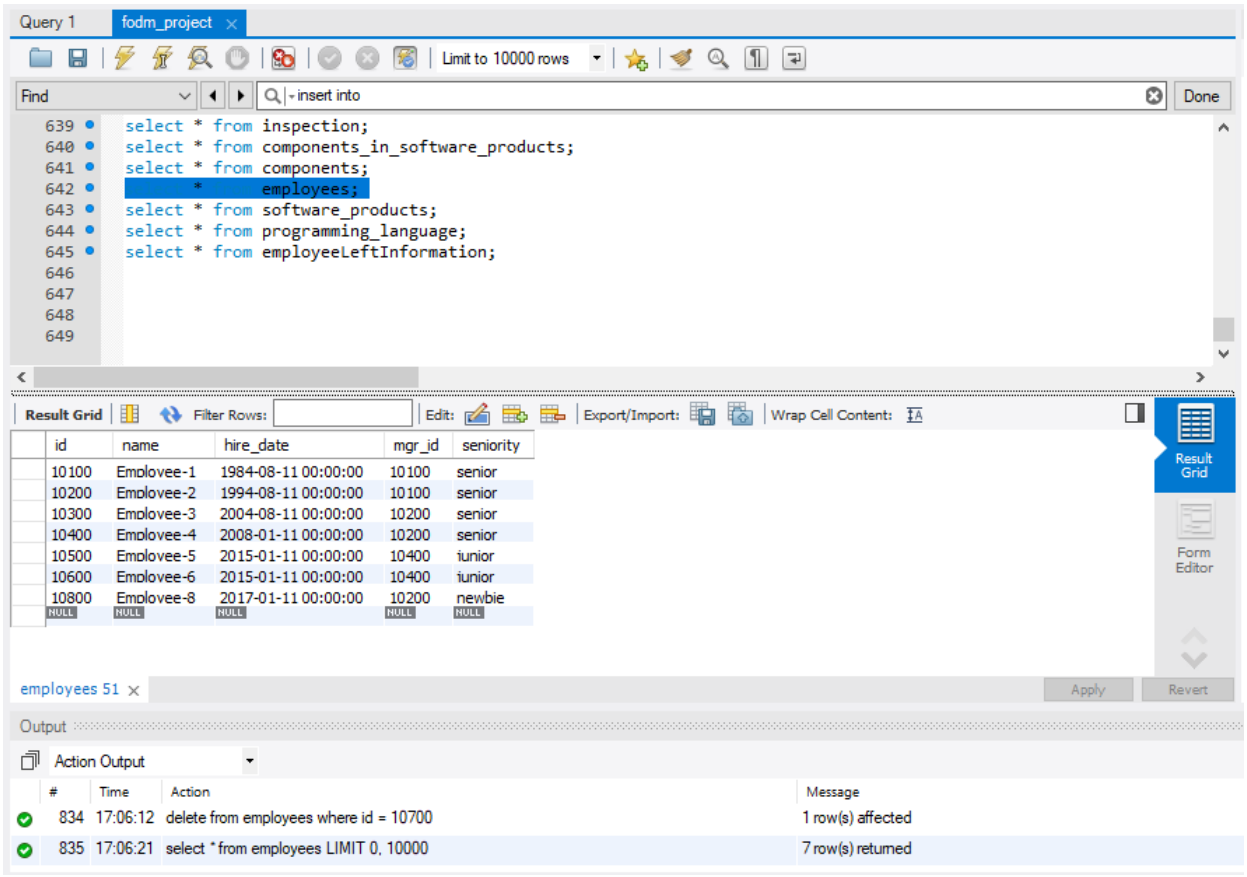
So I have created a new table employeeLeftInformation. In this table all the employees that leave the company, their basic information like Employee ID and and Name is stored in the table.

If the employee is a manager for other employees then, all the CURRENT employees with that manager would be set to null and the manager field would be updated when the new manager is assigned.

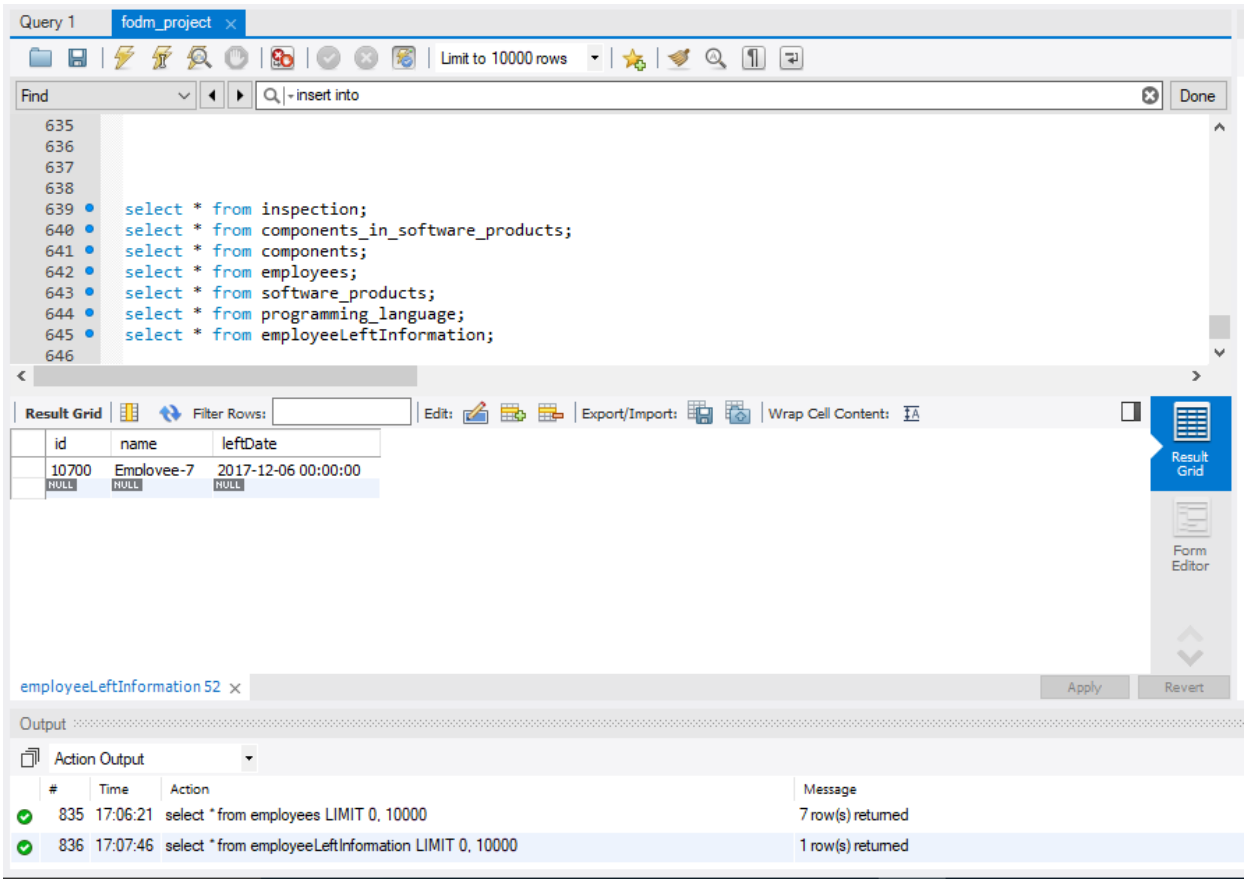
So even when component table is updated and if comp_owner and by_who is changed it will check both the tables to check if the data entered is correct or wrongly entered. A trigger will handle this.

delete from employees where id = 10700;

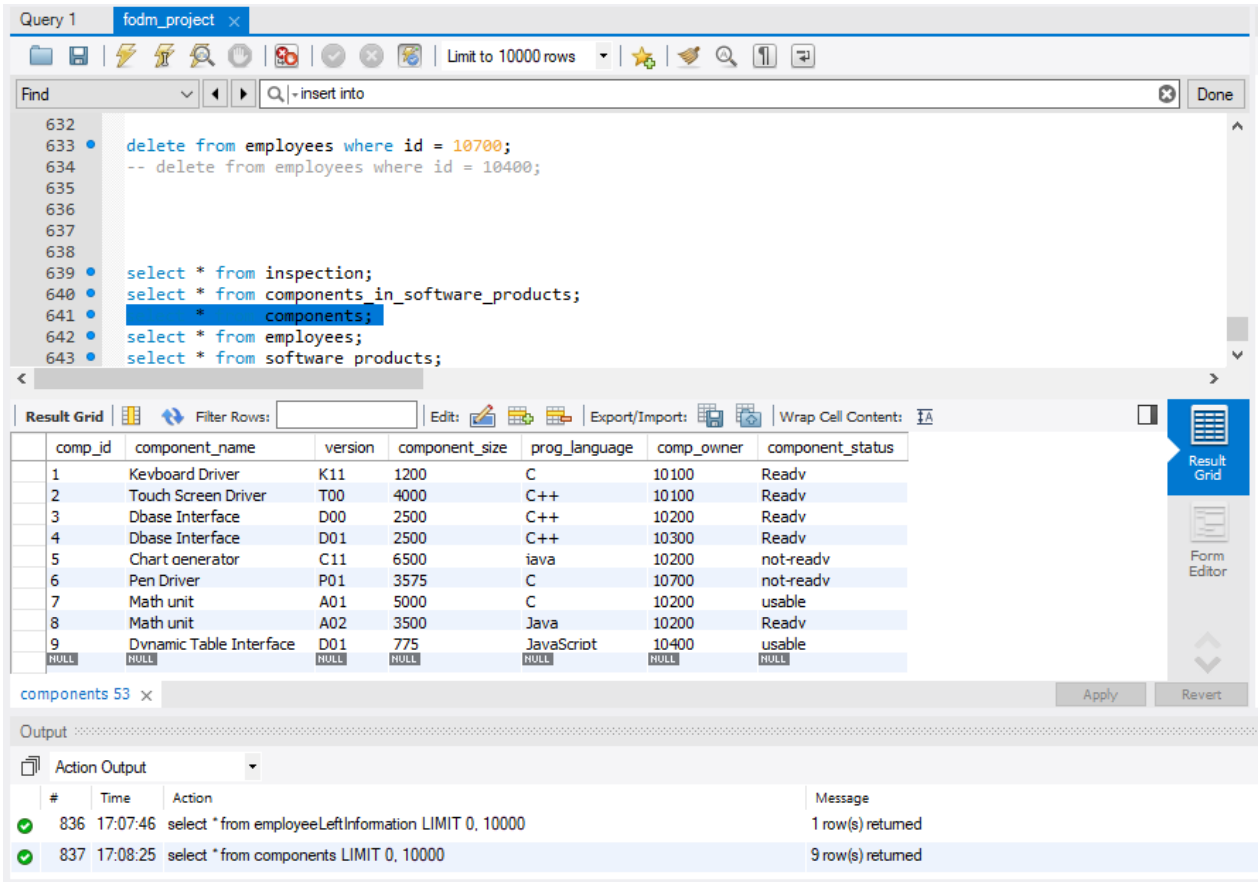
select * from Employees;



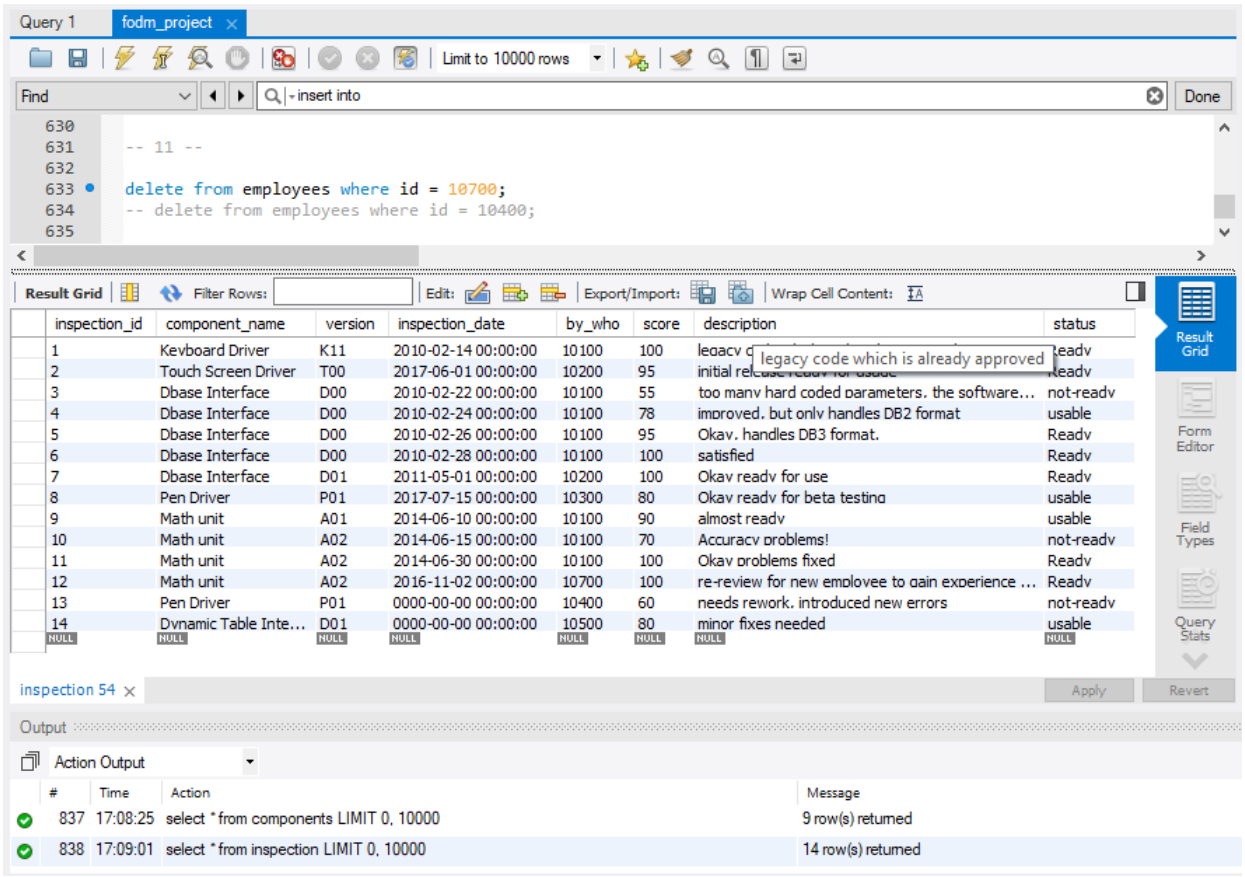
select * from employeeLeftInformation



select * from Components;



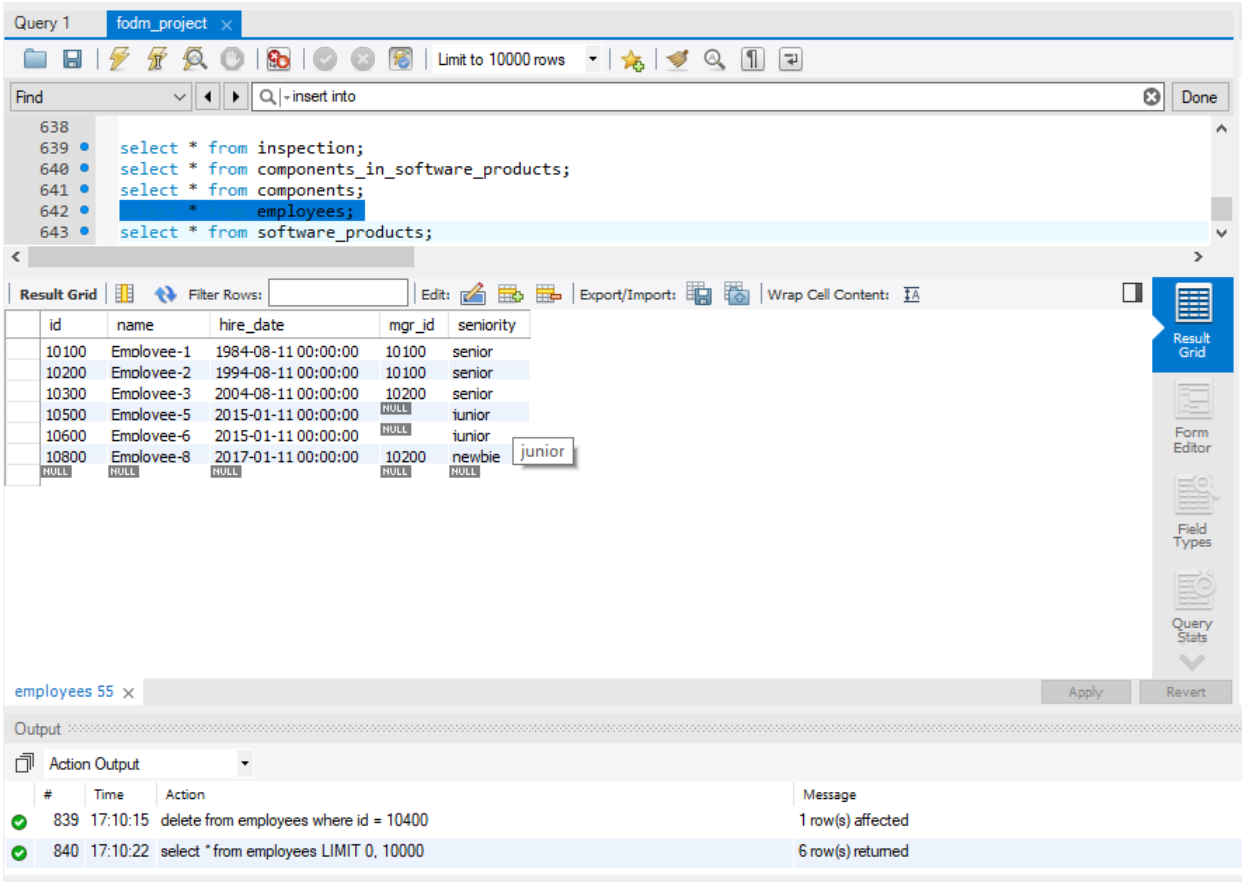
select * from Inspection;



To make my design show proof I have also tried to delete another employee with ID = 10400 who is a manager to few employees.

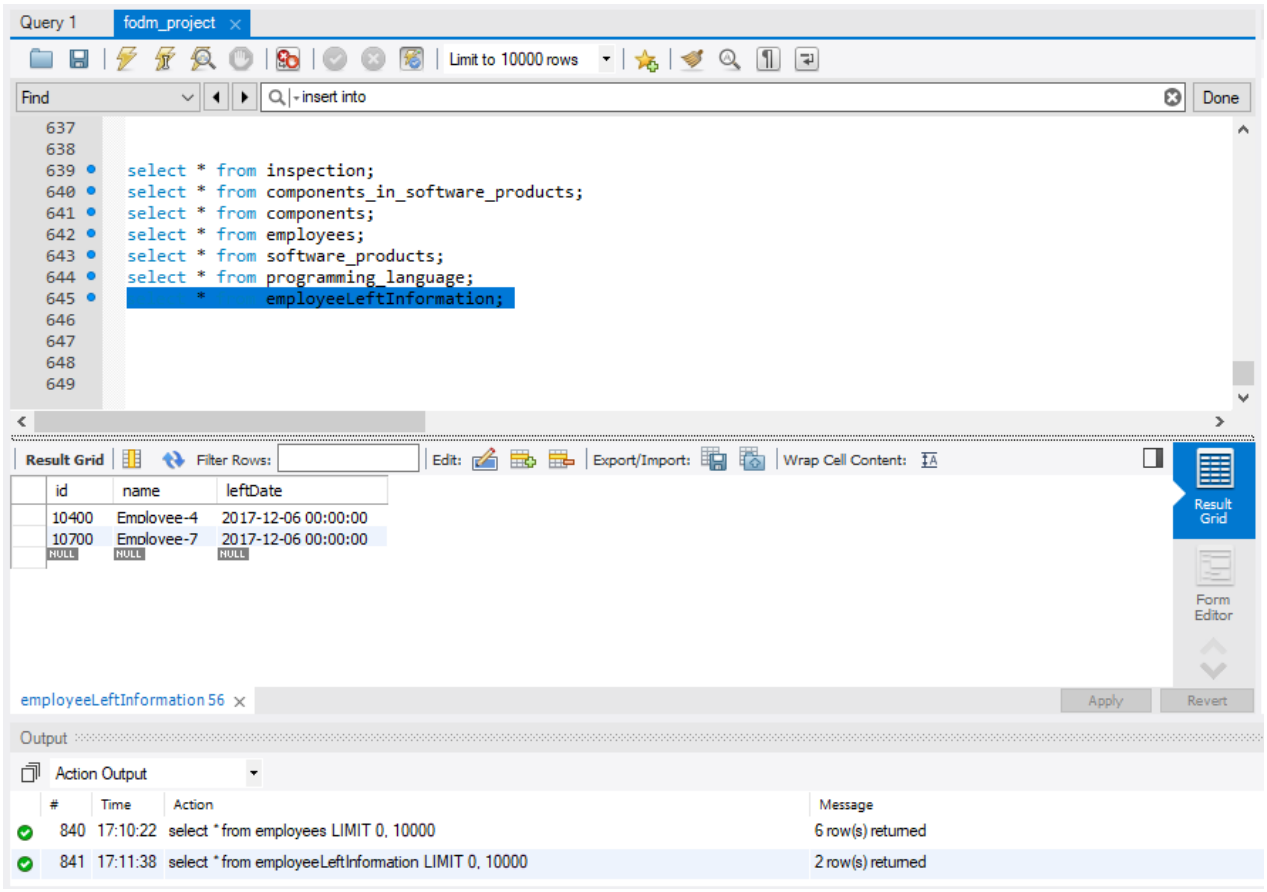
delete from employees where id = 10400;

select * from employees;

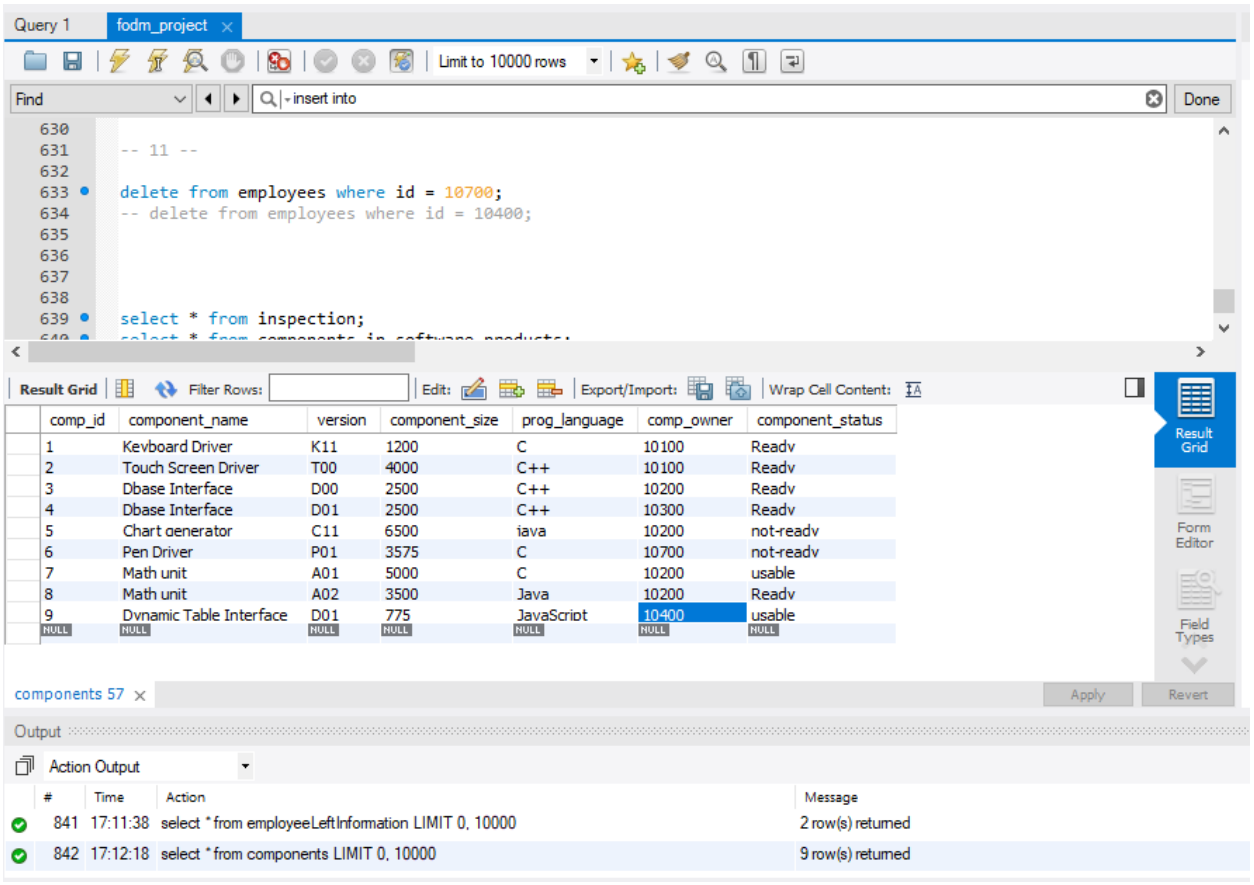


The employee with ID = 10400 was a manager to 2 employees. When he left the manager ID for other employees will be set to null until a new manager is assigned.

select * from employeeLeftInformation;

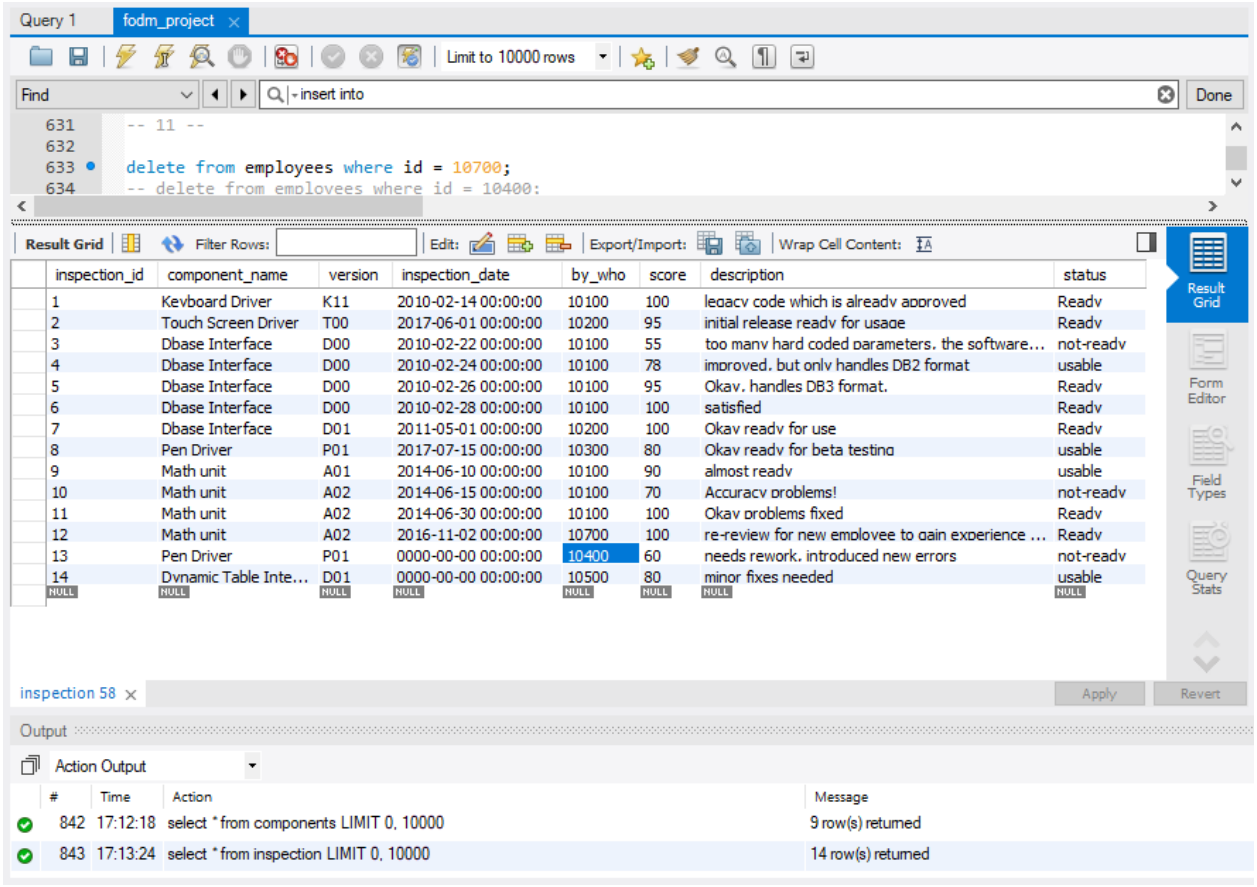


select * from Components;



The work that the employee Id 10400 did is not deleted as it is documented and should not be deleted.

select * from inspection;



(END)