BA700 Winter 2021

Project 2 (25%)

Due March14, 10PM

Upload a single SQL Query Script containing all of the responses to the questions below. Be sure to comment to denote each question (i.e., /\*Question 1\*/) All tables are in the Project 1 database we downloaded earlier in the semester.

Insert a screenshot of your result set panel and action panel for each of the following questions.

Example submission:

Graphical user interface, text, application

Description automatically generated

**Use employee\_master to complete the following:**

**Question 1 (2 points)**

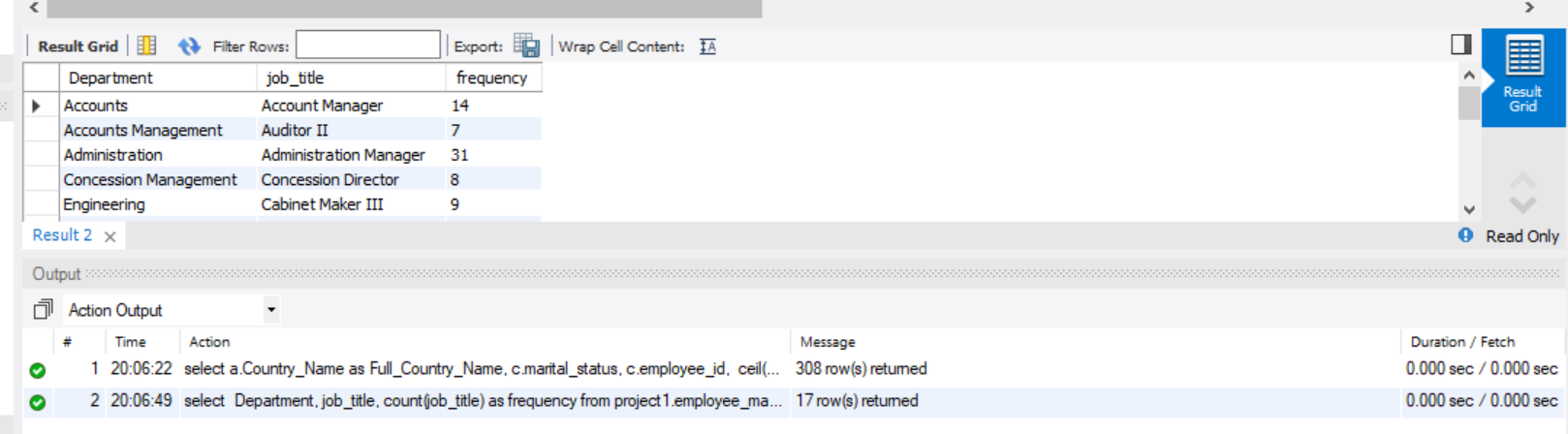
Write a query that returns the number of job titles within each department. Sort by department and then by frequency of job title.

select Department, job\_title, count(job\_title) as frequency

from project1.employee\_master

group by department

order by department;



**Question 2 (3 points)**

Generate a list of employee names, id numbers and job titles for employees born after January 1st 1970 who have a SALES job title (i.e., their job title includes the word sales somewhere in it). Sort by date of birth.

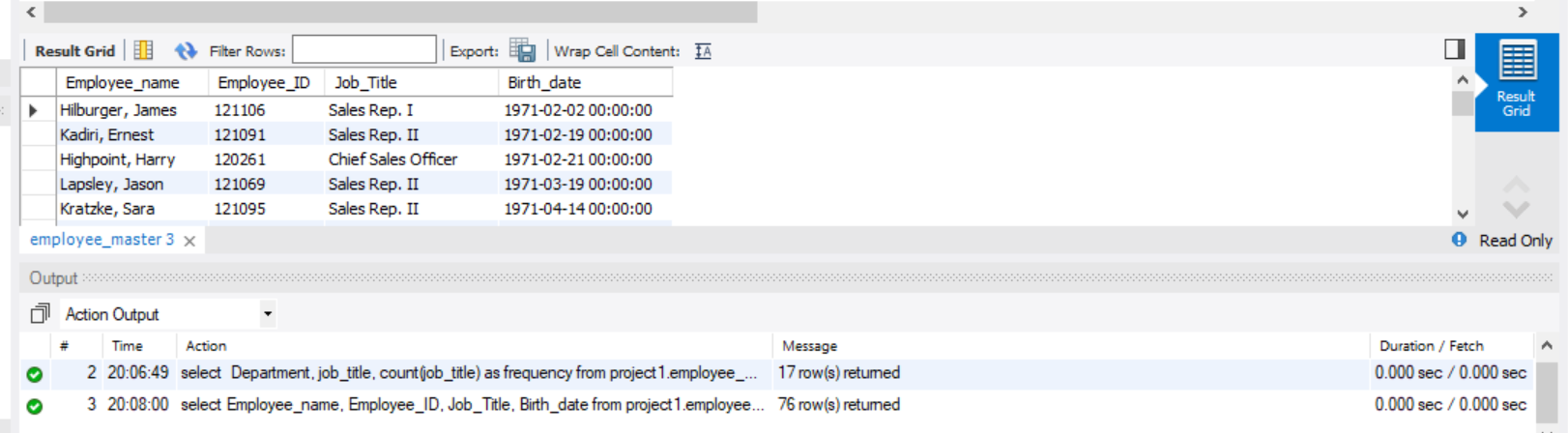
select Employee\_name, Employee\_ID, Job\_Title, Birth\_date

from project1.employee\_master

where Birth\_date between '1970-01-01' and curdate()

having Job\_Title like '%sales%'

order by Birth\_date;



**Question 3 (1 point)**

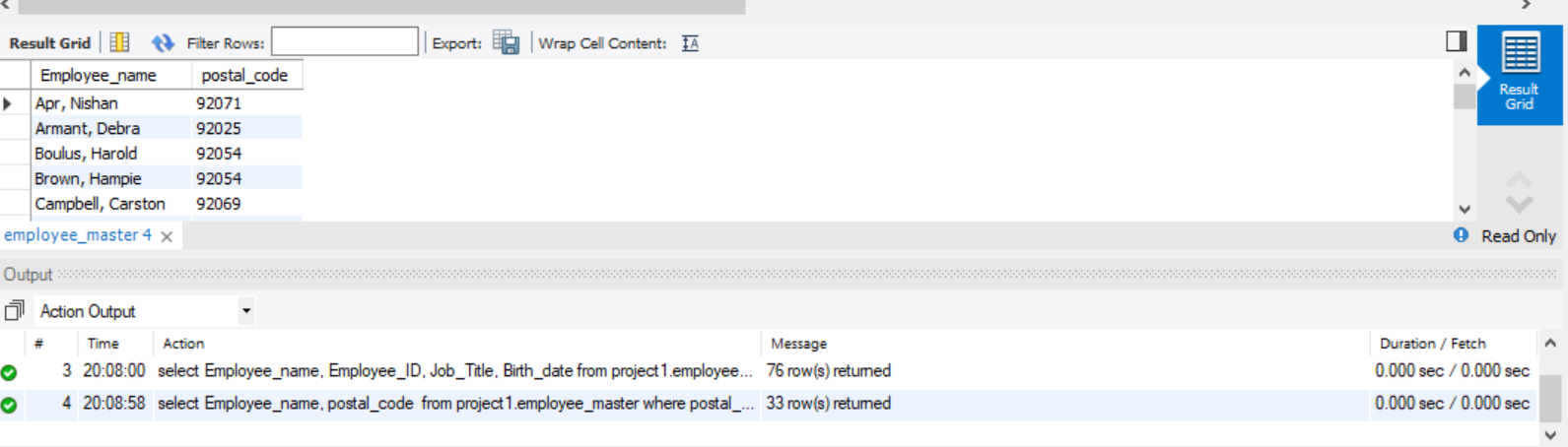
Generate a list of San Diego employees in the 920 postal code area (i.e., postal code starts with 920). Select all columns.

select Employee\_name, postal\_code

from project1.employee\_master

where postal\_code like '920%'

order by Employee\_name;



**Question 4 (2 points)**

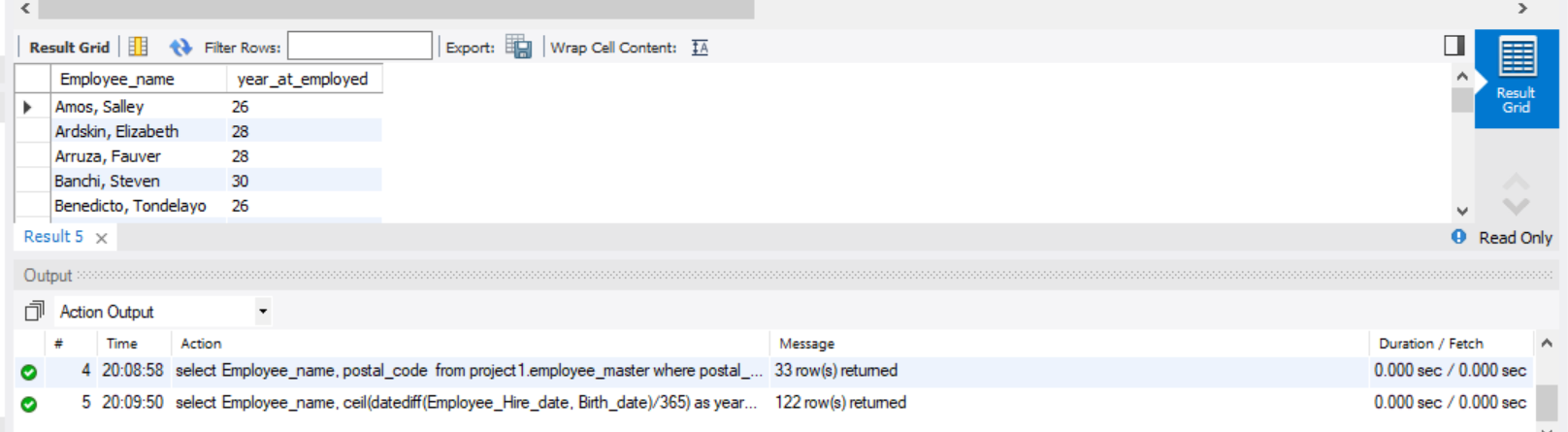
Using the employee master table, return a list of all employees (all fields) where their age at the time of being hired was at least 25 years old.

select Employee\_name, ceil(datediff(Employee\_Hire\_date, Birth\_date)/365) as year\_at\_employed

from project1.employee\_master

having year\_at\_employed > 25

order by Employee\_name;



**Question 5 (3 points)**

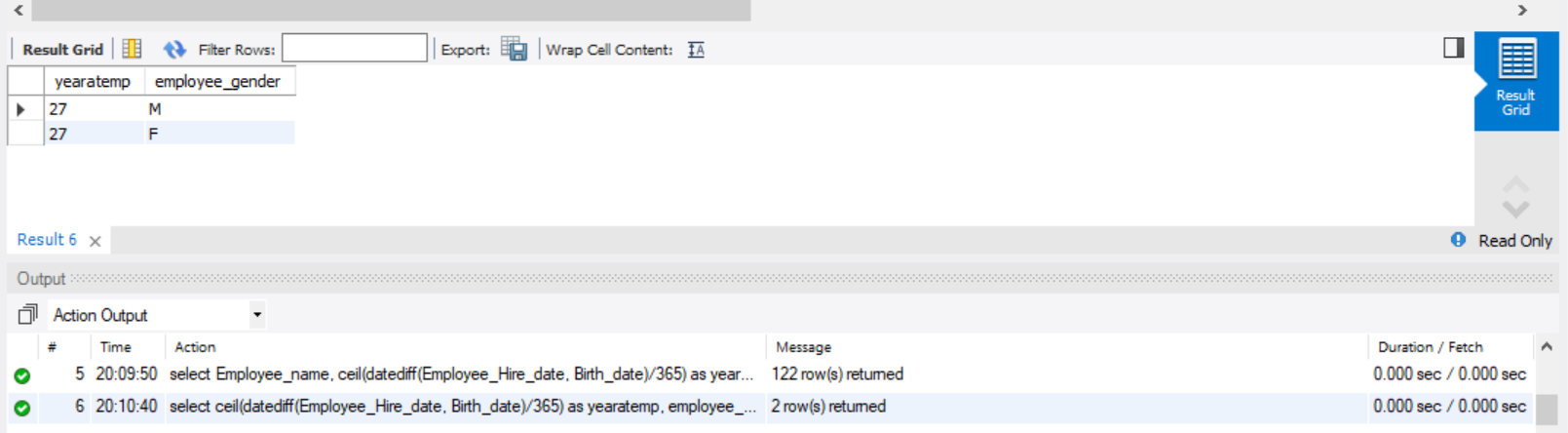
Using the employee master table, what is the average age at the time the employee was hired for each gender.

select ceil(datediff(Employee\_Hire\_date, Birth\_date)/365) as yearatemp, employee\_gender

from project1.employee\_master

group by employee\_gender

having avg(yearatemp);



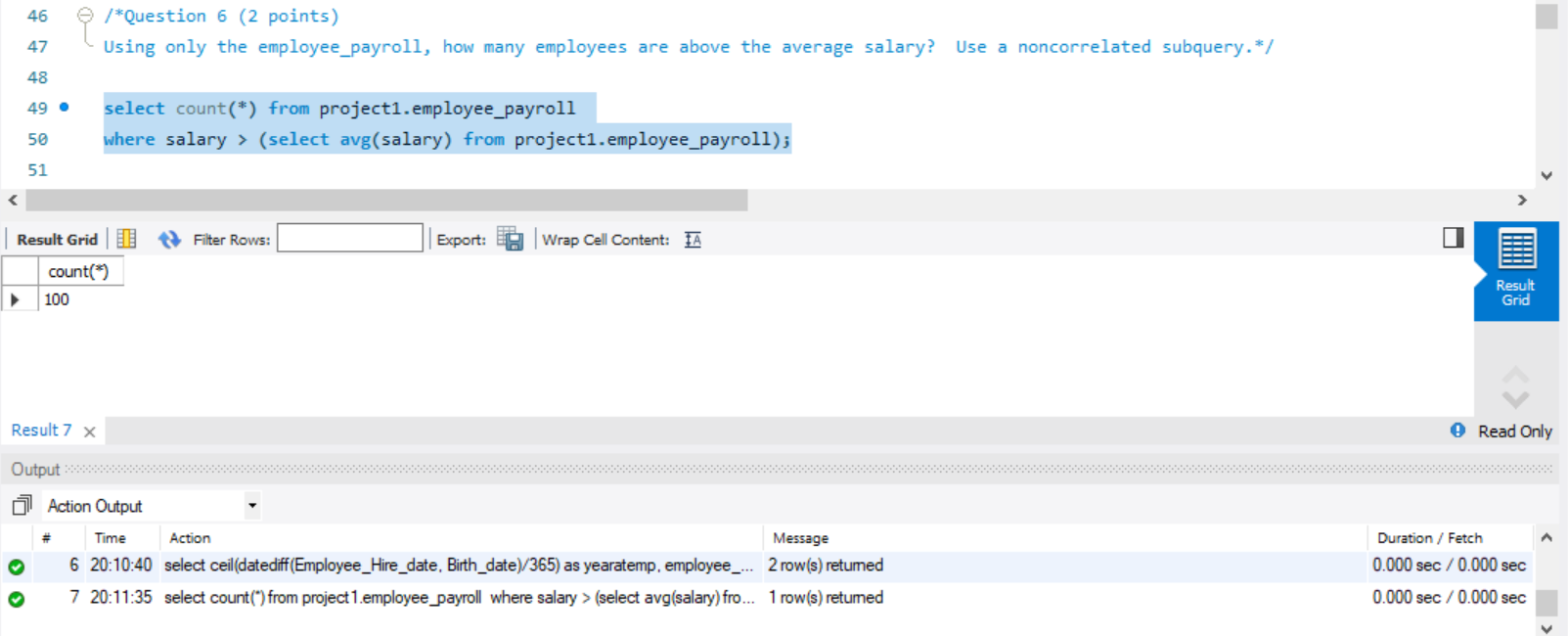
**Use employee\_payroll to complete the following:**

**Question 6 (2 points)**

Using only the employee\_payroll, how many employees are above the average salary? Use a noncorrelated subquery.

select count(\*) from project1.employee\_payroll

where salary > (select avg(salary) from project1.employee\_payroll);

****

**The following use multiple tables:**

**Question 7 (3 points)**

Using the employee payroll table and the employee address table, return a result set that contains the employee name as well as a new column called salary\_range based on salary from the payroll table where salary:

Less than 35000 is label “low earner”

Between 35000 and 65000 is labeled ‘mid-level earner’

Above 65000 is labeled ‘high earner’

Sort your results by descending salary

select b.Employee\_Name,

case

when salary <35000 then 'low earner'

when salary between 35000 and 65000 then 'mid-level earner'

when salary >= '65000' then 'high earner'

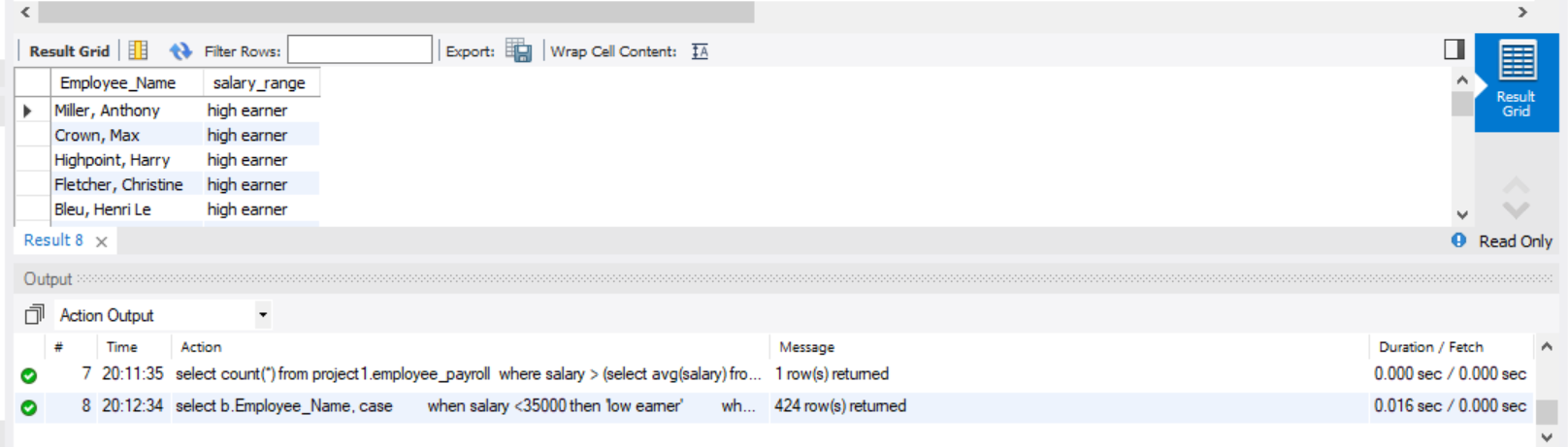
END AS salary\_range

from project1.employee\_payroll a

inner JOIN project1.employee\_addresses b

ON a.employee\_id=b.employee\_id

order by salary desc;



**Question 8 (3 points)**

Using the appropriate tables, write a query that returns the employee id, marital status, full country name (spelled out) and the number of years employed. The number of years employed should be “dynamic” and get recalculated based on the date the query is run.

select a.Country\_Name as Full\_Country\_Name, c.marital\_status, c.employee\_id,

ceil(datediff(curdate() , d.Employee\_Hire\_date)/365) as Years\_employed

from project1.country\_lookup a

inner join project1.employee\_addresses b

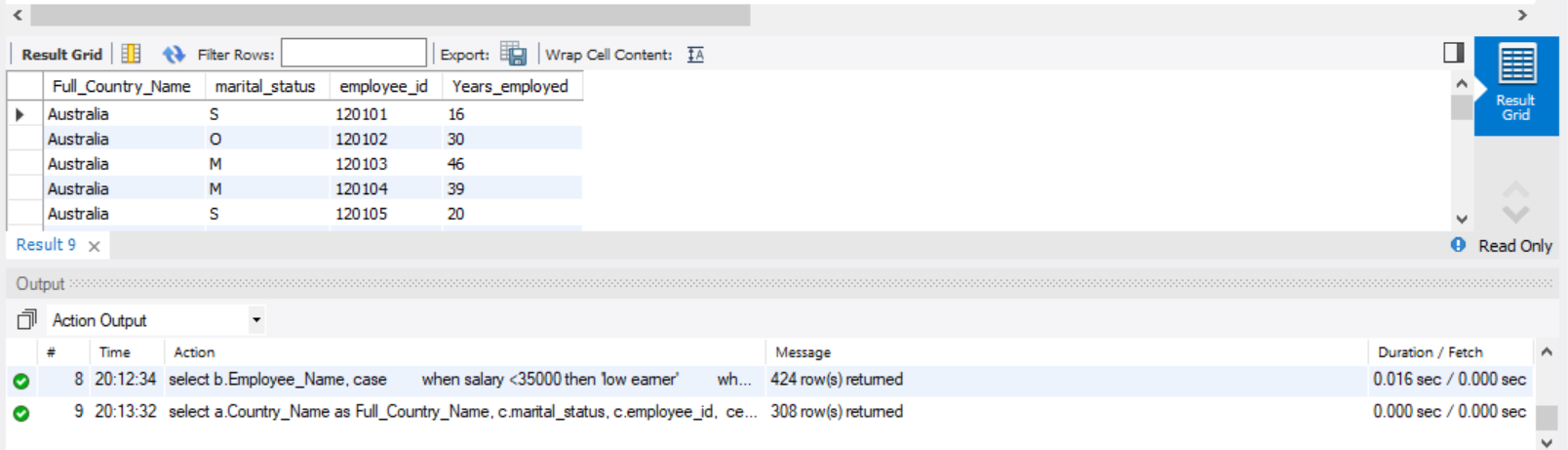
on a.Country\_Key=b.Country

inner join project1.employee\_payroll c

on b.employee\_id=c.employee\_id

inner join project1.employee\_master d

on d.Employee\_ID=c.Employee\_ID;



**Question 9 (2 points)**

Using the query from question 7, add a new column called LEGACY where it holds the value of “yes” if they have been employed at least 30 years and “no” if less than 30 years.

select a.Country\_Name as Full\_Country\_Name, c.marital\_status, c.employee\_id,

ceil(datediff(curdate() , d.Employee\_Hire\_date)/365) as Years\_employed,

case when ceil(datediff(curdate() , d.Employee\_Hire\_date)/365) >= 30 then 'YES' else 'NO'

end as LEGACY

from project1.country\_lookup a

inner join project1.employee\_addresses b

on a.Country\_Key=b.Country

inner join project1.employee\_payroll c

on b.employee\_id=c.employee\_id

inner join project1.employee\_master d

on d.Employee\_ID=c.Employee\_ID;

