

# DBMS

## LAB SHEET – 4

ROLL NO – 20bcs077

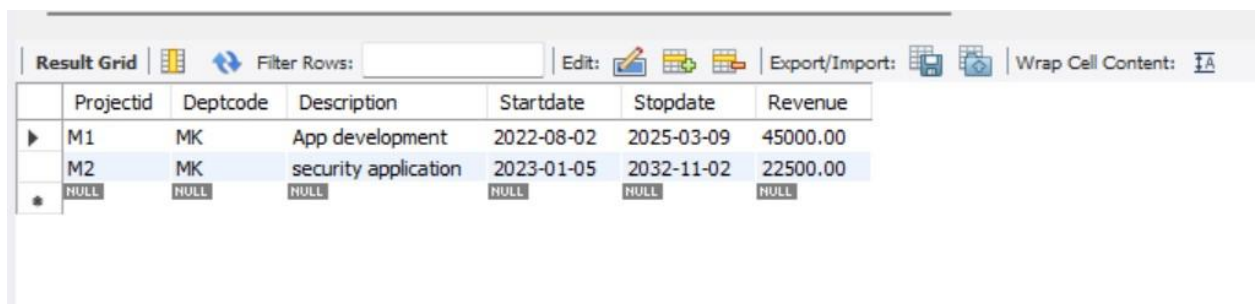
**AIM:** To write a single SQL query for each of the following based on database, finding duration and years of project. finding average, minimum, maximum salary, listing and printing details as given.

### EXPERIMENT:

- Listing the “magical” projects that have not started (indicated by a start date in the future or NULL) but are generating revenue.

```
select * from project;
```

```
where Startdate > '2022-02-25' or Startdate=null;
```



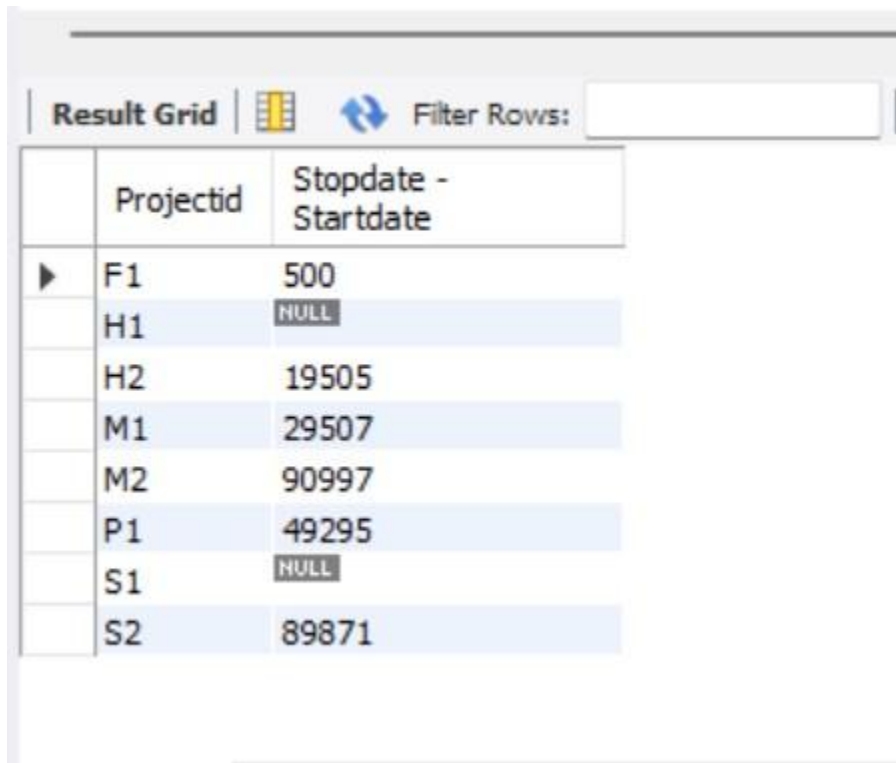
The screenshot shows a database application interface with a 'Result Grid' tab. The grid displays the following data:

Projectid	Deptcode	Description	Startdate	Stopdate	Revenue
M1	MK	App development	2022-08-02	2025-03-09	45000.00
M2	MK	security application	2023-01-05	2032-11-02	22500.00
NULL	NULL	NULL	NULL	NULL	NULL

Projectid	Deptcode	Description	Startdate	Stopdate	Revenue
F2	FIN	A study on Cost And Costing Models in Company	2023-01-11	2025-06-12	60000.00
O1	OPT	Multiple creative products on one platform improves efficiency	2022-03-29	2023-03-29	50000.00
NULL	NULL	NULL	NULL	NULL	NULL

- Finding the project ID and duration of each project.

select Projectid, Stopdate - Startdate from project;





The screenshot shows a 'Result Grid' window with a toolbar containing a 'Filter Rows' button and a search input field. The grid displays the results of the SQL query 'select Projectid, Stopdate - Startdate from project;'. The table has two columns: 'Projectid' and 'Stopdate - Startdate'. The data is as follows:

Projectid	Stopdate - Startdate
F1	500
H1	NULL
H2	19505
M1	29507
M2	90997
P1	49295
S1	NULL
S2	89871

- **Finding the years a project started. Remove duplicates.**

select distinct year(Startdate) from project;


Result Grid |   Filter Rows:


	year(Startdate)
▶	2022
	NULL
	2021
	2023
	2019
	2020

- Finding the IDs of employees assigned to a project that is more than 20 hours per week. Write three queries using 20, 40 and 60 hour work weeks.

```
select EmployeeID from workson
where Assignedtime*7 > 20;
```

Result Grid





Filter Rows:

	Employeeid
▶	3
	5
	6
	7
	8

select EmployeeID from workson

where Assignedtime\*7 > 40;



The screenshot shows a 'Result Grid' window with a toolbar containing a 'Filter Rows' button. The grid has a single column labeled 'Employeeid'. It contains two rows: the first row has the value '5' and a small black triangle icon to its left; the second row has the value '7' and is highlighted with a light blue background.

	Employeeid
▶	5
	7

select Employeeid from workson

where Assignedtime\*7 > 60;

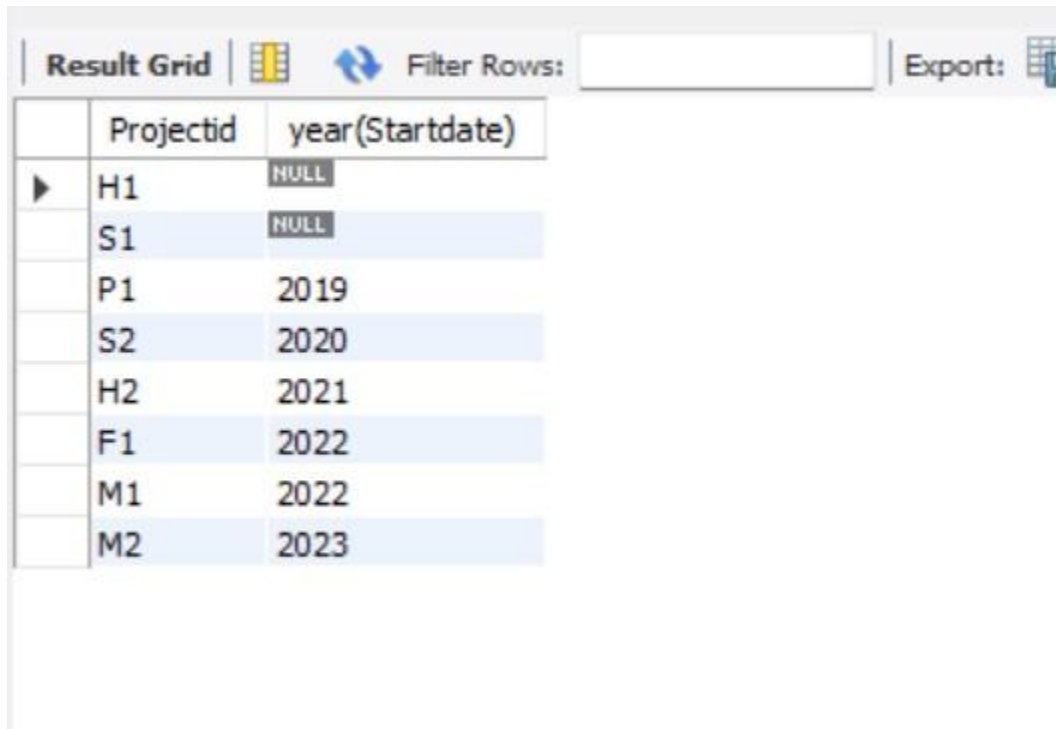


The screenshot shows a 'Result Grid' window with a toolbar containing a 'Filter Rows' button. The grid has a single column labeled 'Employeeid'. It contains one row with the value '7' and a small black triangle icon to its left.

	Employeeid
▶	7

- For each project, listing the ID and year the project started. Keeping the results in ascending order by year.

```
select Projectid, year(Startdate)
from project
order by year(Startdate);
```



The screenshot shows a database query result grid. At the top, there is a toolbar with a 'Result Grid' tab, a grid icon, a refresh icon, a 'Filter Rows:' input field, and an 'Export:' button. Below the toolbar is a table with two columns: 'Projectid' and 'year(Startdate)'. The table contains nine rows of data, ordered by the year. The first two rows, H1 and S1, have NULL values for the year. The subsequent rows show projects P1 through M2 with years ranging from 2019 to 2023. The rows are alternatingly highlighted in white and light blue.

	Projectid	year(Startdate)
▶	H1	NULL
	S1	NULL
	P1	2019
	S2	2020
	H2	2021
	F1	2022
	M1	2022
	M2	2023

- **Finding the average salary for all employees.**

```
select avg(Revenue)
from project;
```

Result Grid		Filter
	avg(Revenue)	
▶	71087.500000	

- **Finding the minimum salary for an employee.**

```
select min(Salary)
from employee;
```

Result Grid		Filter Rows:
	min(Salary)	
▶	60000.00	

- **Finding the maximum salary for an employee.**

```
select max(Salary)
from employee;
```

Result Grid		Filter Rows:
	max(Salary)	
▶	180000.00	

- **Finding list of employee whose name has exactly 6 chracters.**

select Firstname from employee

where Firstname like '\_\_\_\_\_';

Result Grid		Filter Rows:
	Firstname	
▶	Gaurav	
	Manish	

- **Finding list of employees whose name starts with 'a'.**

select Firstname, Lastname from employee

where Firstname like 'a%';

Result Grid			Filter Rows:	
	Firstname	Lastname		
▶	Amit	Kumar		

- **Finding list of employees who works for more than one department/project.**

```

select e.Firstname, e.Lastname, w.Projectid
from employee e inner join workson w
on e.EmployeeID = w.Employeeid
where e.EmployeeID
in (select Employeeid from workson
group by Employeeid having count(Employeeid) > 1);

```




Result Grid			
Filter Rows:			
	Firstname	Lastname	Projectid
▶	Amit	Kumar	H1
	Amit	Kumar	H1
	Bhanwar	Yuvraj	M1
	Bhanwar	Yuvraj	M1
	Gaurav	Patel	M2
	Gaurav	Patel	M2
	Joe	Root	S1
	Joe	Root	S1
	Kristofer	Henry	S2
	Kristofer	Henry	S2
	Lora	Lane	P1
	Lora	Lane	P1
	Manish	Singh	H2
	Manish	Singh	H2
	Oven	Proven	F1
	Oven	Proven	F1

- **Printing all details of employees except the minimum and maximum paid employees.**

```
select * from employee
```

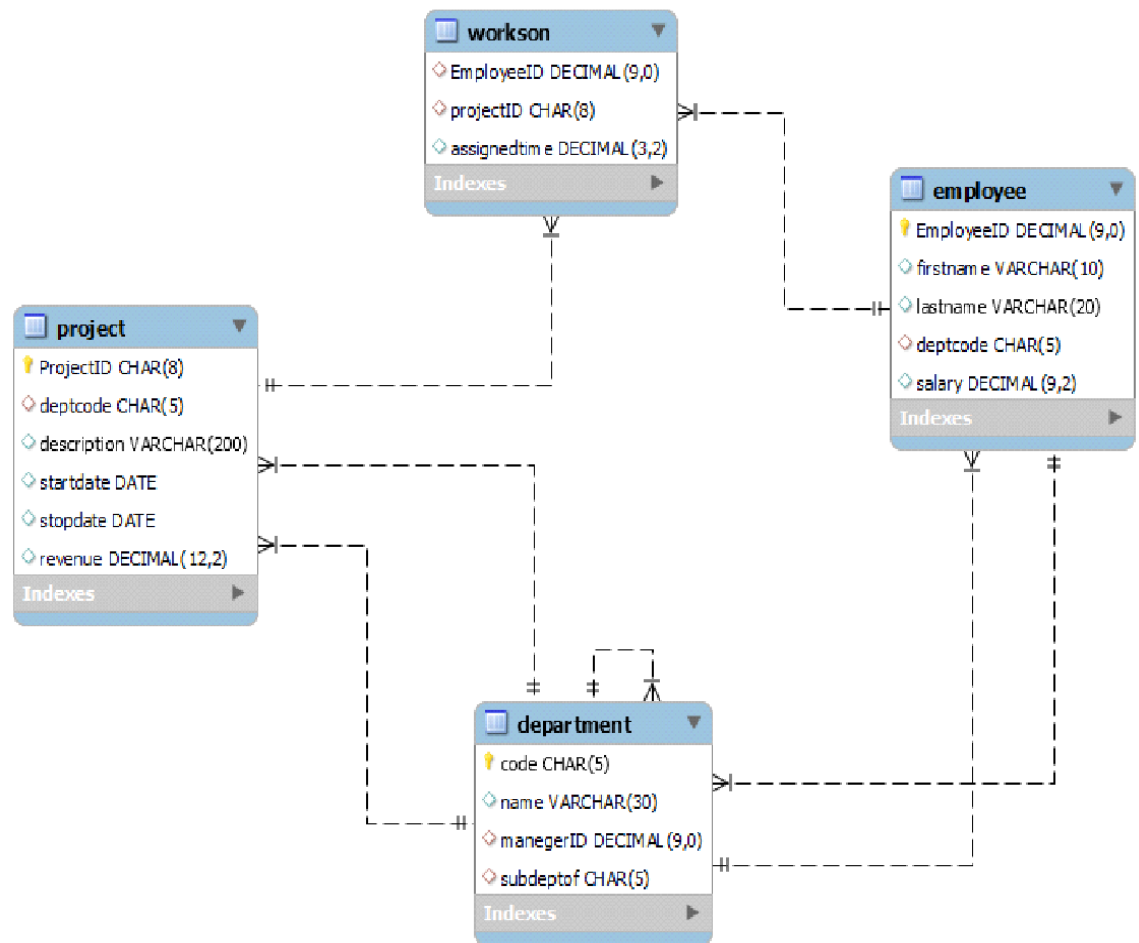
```
where Salary != (select max(Salary) from employee)
```

```
and Salary != (select min(Salary) from employee);
```

Result Grid					
Filter Rows: <input type="text"/>					
Edit: 					
	EmployeeID	Firstname	Lastname	Deptcode	Salary
▶	1	Amit	Kumar	GM	70000.00
	2	Bhanwar	Yuvraj	MK	60000.00
	3	Gaurav	Patel	OP	78500.00
	4	Joe	Root	FN	66000.00
	5	Kristofer	Henry	HR	99000.00
	6	Lora	Lane	PU	138000.00
	7	Manish	Singh	PR	67500.00
	8	Oven	Proven	SL	180000.00
•	NULL	NULL	NULL	NULL	NULL

## RESULTS:

- Schema diagram



- **Tables**

- **Employee**

	EmployeeID	firstname	lastname	deptcode	salary
▶	1	Narendra	Gandhi	GEM	40000.00
	2	Rahul	Kejriwal	MKT	50000.00
	3	Arvind	Yadav	OPT	34500.00
	4	Akhilesh	Modi	FIN	100000.00
	5	Joe	Clint	HRD	55000.00
	6	Taylor	Swift	PUR	70000.00
	7	Camila	Cabello	PRE	48500.00
	8	Hailee	Steinfeld	SAL	160000.00
⌵	NULL	NULL	NULL	NULL	NULL
Employee 8 ✕					

- **Department**

	code	name	manegerID	subdeptof
▶	FIN	Finance	4	GEM
	GEM	General Management	1	GEM
	HRD	Human Resource	5	GEM
	MKT	Marketing	2	GEM
	OPT	Operations	3	GEM
	PRE	Public Relations	7	GEM
	PUR	Purchasing	6	GEM
	SAL	Sales	8	GEM
⌵	NULL	NULL	NULL	NULL
Department 9 ✕				

- **Project**

86 ('F1', 'FIN', 'Investment awareness in financial asset and preference of financial  
87 ... Cost And Costing Models in Company ...

< [Filter Rows: ] | Edit: [Icons] | Export/Import: [Icons] | Wrap Cell Content: [Icon]

	ProjectID	deptcode	description	startdate	stopdate	revenue
▶	F1	FIN	Investment awareness in financial asset and pr...	2021-11-11	2022-08-22	750000.00
	F2	FIN	A study on Cost And Costing Models in Company	2023-01-11	2025-06-12	60000.00
	H1	HRD	Policies, procedures, safety rules and other imp...	2021-10-26	2022-03-14	74700.00
	H2	HRD	A study on identification of non performing deal...	2022-01-01	2023-06-04	148500.00
	H3	HRD	Industrial relations - how to manage union and ...	2022-01-01	2024-03-14	74700.00
	M1	MKT	A study on the factors affecting dealer perform...	2022-01-29	2023-03-29	50000.00
	M2	MKT	Emergence of Internet Marketing -Origins, Nee...	2021-12-05	2022-11-02	34500.00
	O1	OPT	Multiple creative products on one platform impr...	2022-03-29	2023-03-29	50000.00
	O2	OPT	Managing feedback from multiple stakeholders b...	2021-11-25	2022-11-02	34500.00
	P1	PUR	To provide quality services to business joinING ...	2021-07-16	2022-09-28	70000.00
	S1	SAL	IMPACT OF CELEBRITY ENDORSEMENT ON CO...	2022-02-13	2023-06-12	60000.00
	S2	SAL	DIGITAL MARKETING AS A KEY DRIVER OF SAL...	2021-09-30	2022-04-07	55000.00

\* [Project 23] x [Apply]

Output

	EmployeeID	projectID	assignedtime
▶	1	H1	1.32
	2	M1	4.55
	3	M2	3.00
	4	S1	2.15
	5	S2	8.22
	6	P1	9.30
	7	H2	6.20
	8	F1	5.45
	5	H3	1.00
	3	O1	4.50
	4	O2	3.00
	1	F2	2.25

SQL databases are the most prominent databases, in which data can be inserted in the form of tables with the help of some commands. Through these commands one can create tables, and in each table one

can declare some entities. The 'SELECT' statement is used to select data from a database with or without conditions which gives us a better option to render data. The data returned is stored in a result table, called the result-set. 'WHERE' clause is a keyword used to specify the exact criteria of data or rows that will be affected by a specified SQL statement. A 'GROUP BY' statement in SQL specifies that a SQL SELECT statement partitions result rows into groups, based on their values in one or several columns.