

SQL PROJECT

BY RASHID GHANSAR

84

36

[illegible]

FIND THE AVERAGE SALARY OF EMPLOYEES GROUP BY DEPARTMENT

206

207

208 • `SELECT department,AVG(salary) AS avg_salary FROM employees group by department;`

209

210

211

Result Grid   Filter Rows: | Export:  | Wrap Cell Content: 

	department	avg_salary
▶	IT	81818.181818
	HR	58422.222222
	Finance	74600.000000
	Marketing	54000.000000
	Sales	62222.222222








Modify the Employee table to add email to the table

```
82
83
84 • ALTER TABLE employees ADD COLUMN email VARCHAR(100);
85 • select * from employees;
86
```

Result Grid		Filter Rows:		Edit:		Export/Import:		Wrap C
	id	first_name	last_name	age	department	salary	hire_date	email
	2	Jane	Smith	34	HR	55000.00	2018-07-22	NULL
	3	Michael	Johnson	40	Finance	75000.00	2015-09-30	NULL
	4	Emily	Davis	27	Marketing	50000.00	2021-01-10	NULL
	5	Robert	Wilson	45	IT	90000.00	2010-05-18	NULL
	6	Sarah	Miller	31	Finance	70000.00	2017-11-25	NULL
	7	David	Brown	36	Sales	62000.00	2016-06-14	NULL
	8	Laura	Jones	29	HR	52000.00	2019-08-05	NULL
	9	James	Garcia	38	IT	85000.00	2014-12-12	NULL
	10	Olivia	Martinez	25	Marketing	48000.00	2022-03-08	NULL


UPDATE SALARY FOR EMPLOYEES IN THE 'HR' DEPARTMENT BY 10%

```
0
1 • UPDATE employees
2 SET salary = salary * 1.10
3 WHERE department = 'HR';
4
5
```

Result Grid   Filter Rows: <input type="text"/> Edit:    Export/Import:   View								
id	first_name	last_name	age	department	salary	hire_date	email	
1	John	Doe	29	IT	65000.00	2020-03-15	NULL	
2	Jane	Smith	34	HR	60500.00	2018-07-22	NULL	
3	Michael	Johnson	40	Finance	75000.00	2015-09-30	NULL	
4	Emily	Davis	27	Marketing	50000.00	2021-01-10	NULL	
5	Robert	Wilson	45	IT	90000.00	2010-05-18	NULL	
6	Sarah	Miller	31	Finance	70000.00	2017-11-25	NULL	
7	David	Brown	36	Sales	62000.00	2016-06-14	NULL	
8	Laura	Jones	29	HR	57200.00	2019-08-05	NULL	
9	James	Garcia	38	IT	85000.00	2014-12-12	NULL	

FILTERS EMPLOYEES WORKING
IN SALES OR MARKETING

```
86
87 • SELECT first_name, last_name, department
88 FROM employees
89 WHERE department IN ('Sales', 'Marketing');
90
```


Result Grid   Filter Rows: Export:  Wrap

	first_name	last_name	department
▶	Emily	Davis	Marketing
	David	Brown	Sales
	Olivia	Martinez	Marketing
	Matthew	Gonzalez	Sales
	Ethan	Taylor	Marketing
	Ava	Jackson	Sales
	Mia	Harris	Marketing
	William	Lewis	Sales
	Amelia	Allen	Marketing

employees 7 x

FINDS EMPLOYEES WITH 'SON' IN
THEIR LAST NAME

```
78
79
80 • SELECT first_name, last_name
81 FROM employees
82 WHERE last_name LIKE '%son%';
83
84
85
```

Result Grid   Filter Rows:

	first_name	last_name
▶	Michael	Johnson
	Robert	Wilson
	Isabella	Anderson
	Ava	Jackson
	Scarlett	Nelson
	Michael	Johnson
	Robert	Wilson
	Isabella	Anderson
	Ava	Jackson

employees 8 x

FINDS EMPLOYEES EARNING ABOVE
THE COMPANY'S AVERAGE SALARY

```
77
78 • SELECT first_name, last_name, salary
79 FROM employees
80 WHERE salary > (SELECT AVG(salary) FROM employees);
81
82
83
```

Result Grid   Filter Rows: | Export:  | Wrap Cell Content

	first_name	last_name	salary
▶	Michael	Johnson	75000.00
	Robert	Wilson	90000.00
	Sarah	Miller	70000.00
	James	Garcia	85000.00
	Daniel	Hernandez	68000.00
	Sophia	Lopez	72000.00
	Isabella	Anderson	73000.00
	Mason	Thomas	88000.00
	Mia	Harris	69000.00

CALCULATES TOTAL SALARIES
FOR EACH DEPARTMENT

```
77
78
79 • SELECT department, SUM(salary) AS total_salary
80 FROM employees
81 GROUP BY department;
82
83
```

Result Grid   Filter Rows: | Export:  | Wrap C

	department	total_salary
▶	IT	1800000.00
	HR	1051600.00
	Finance	1492000.00
	Marketing	972000.00
	Sales	1120000.00

FIND EMPLOYEES WHOSE NAMES START WITH 'A'



```
181
182 • SELECT first_name, last_name FROM employees WHERE first_name LIKE 'A%';
183
184
185
```

Result Grid   Filter Rows: Export:  Wrap Cell Content: 

	first_name	last_name
▶	Ava	Jackson
	Amelia	Allen
	Alexander	Wright
	Aiden	Carter
	Abigail	Mitchell
	Ava	Jackson
	Amelia	Allen
	Alexander	Wright
	Aiden	Carter

USE USECASE TO CATEGORIZE
EMPLOYEES BY SALARY

```
188
189 • SELECT first_name, last_name, salary,
190       CASE
191         WHEN salary > 90000 THEN 'High Salary'
192         WHEN salary BETWEEN 60000 AND 90000 THEN 'Medium Salary'
193         ELSE 'Low Salary'
194       END AS Salary_Category
195 FROM employees;
196
197
```

Result Grid   Filter Rows: Export:  Wrap Cell Content: 

	first_name	last_name	salary	Salary_Category
*	John	Doe	65000.00	Medium Salary
	Jane	Smith	60500.00	Medium Salary
	Michael	Johnson	75000.00	Medium Salary
	Emily	Davis	50000.00	Low Salary
	Robert	Wilson	90000.00	Medium Salary
	Sarah	Miller	70000.00	Medium Salary
	David	Brown	62000.00	Medium Salary
	Laura	Jones	57200.00	Low Salary
	James	Garcia	85000.00	Medium Salary

FIND EMPLOYEES HIRED IN THE LAST 5 YEARS

```
214
215
216 • SELECT first_name, last_name, hire_date
217 FROM employees
218 WHERE hire_date >= DATE_SUB(CURDATE(), INTERVAL 5 YEAR);
219
220
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	first_name	last_name	hire_date
▶	Emily	Davis	2021-01-10
	Olivia	Martinez	2022-03-08
	Sophia	Lopez	2020-06-30
	Ethan	Taylor	2021-05-21
	Noah	Clark	2020-09-25
	Sophia	Lee	2021-07-09
	Harper	King	2022-05-16
	Evelyn	Scott	2020-08-12
	Scarlett	Nelson	2021-06-28

FIND THE NUMBER OF EMPLOYEES WORKING IN EACH DEPARTMENT

```
206
207
208 • SELECT department, COUNT(*) AS num_employees
209 FROM employees
210 GROUP BY department
211 HAVING COUNT(*) > 3;
212
213
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	department	num_employees
▶	IT	22
	HR	18
	Finance	20
	Marketing	18
	Sales	18

CREATES A VIRTUAL TABLE OF EMPLOYEES
EARNING MORE THAN 80,000.



```
90
91 • CREATE VIEW HighSalaryEmployees AS
92   SELECT first_name, last_name, salary
93   FROM employees
94   WHERE salary > 80000;
95 • SELECT * FROM HighSalaryEmployees;
96
97
```

	first_name	last_name	salary
▶	Robert	Wilson	90000.00
	James	Garcia	85000.00
	Mason	Thomas	88000.00
	Lucas	Hall	89000.00
	Alexander	Wright	87000.00
	Sebastian	Baker	81000.00
	Elijah	Evans	86000.00
	Logan	Morris	90000.00
	Robert	Wilson	90000.00

HighSalaryEmployees 16 x

CALCULATE THE AGE DIFFERENCE BETWEEN
TWO EMPLOYEES

```
199
200
201
202 • SELECT first_name, last_name, ABS(age - 30) AS Age_Difference FROM employees;
203
204
```

Result Grid   Filter Rows: Export:  Wrap Cell Content: 

	first_name	last_name	Age_Difference
▶	John	Doe	1
	Jane	Smith	4
	Michael	Johnson	10
	Emily	Davis	3
	Robert	Wilson	15
	Sarah	Miller	1
	David	Brown	6
	Laura	Jones	1
	James	Garcia	8