

SQL

For QA Engineers

Part 2

(Continue...)

SELECT Statement

To retrieve data from database (Tables) we use the SELECT statement:

```
SELECT [ALL | DISTINCT] column1[,column2]
      FROM      <table1> [,<table2>]
      [WHERE    "conditions"]
      [GROUP BY "column-list"]
      [HAVING   "conditions"]
      [ORDER BY "column-list" [ASC | DESC]]
```

We reviewed SELECT, FROM, WHERE, and ORDER BY. What is GROUP BY?

SELECT...GROUP BY

GROUP BY clause allow to group the results by some criteria.

Group1 (**grouped by** department 30):
4 employees

Group2 (**grouped by** department 80):
3 employees

Group3(**grouped by** department 90):
1 employee

Group4(**grouped by** department 100):
3 employees

Department_ID	Last Name
30	Patel
30	Johnson
30	Beyer
30	Ivanov
80	Chen
80	Subramanian
80	Trevorson
90	Pham
100	Nguen
100	Stevenson
100	Malikov

SELECT...GROUP BY

How many people work in each department?

```
Select      department_id, count(*)  
from       employees  
group by   department_id
```

Group1 (grouped by department 30):
4 employees

Group2 (grouped by department 80):
3 employees

Group3 (grouped by department 90):
1 employee

Group4 (grouped by department 100):
3 employees

Department	Last Name
30	Patel
30	Johnson
30	Beyer
30	Ivanov
80	Chen
80	Subramanian
80	Trevorson
90	Pham
100	Nguen
100	Stevenson
100	Malikov

Quiz

- 1) Find out total salary per each department.
- 2) Retrieve maximum and minimum salary in the departments 80, 90, 100, ordered by department;
- 3) Display all departments in order. Within each department show salary and number of employees, getting the same salary.

SELECT Statement

To retrieve data from database (Tables) we use the SELECT statement:

```
SELECT [ALL | DISTINCT] column1[,column2]
      FROM          <table1> [,<table2>]
      [WHERE        "conditions"]
      [GROUP BY "column-list"]
      [HAVING    "conditions"]
      [ORDER BY "column-list" [ASC | DESC]]
```

We reviewed all keywords, except HAVING. What is it for?

Having

HAVING was added to use *aggregate functions* (sum, max, min, avg...).

WHERE could not be used with aggregate functions

```
SELECT      DEPARTMENT_ID, max(SALARY)
FROM        EMPLOYEES
GROUP BY    DEPARTMENT_ID
HAVING     MAX(SALARY) > 10000
ORDER BY    DEPARTMENT_ID
```

Exercise

Show only those departments, where the maximum salary is greater than \$10,000:

```
SELECT      DEPARTMENT_ID, MAX(SALARY)
FROM        EMPLOYEES
GROUP BY    DEPARTMENT_ID
having      max(salary) > 10000
ORDER BY    DEPARTMENT_ID
```

Quiz

- 1) Display all the departments with more than 10 employees;
- 2) Show all employees with the same first name;

Data Manipulation

We learned how to query (read) data using SELECT statement:



Great Job!

SELECT

<column(s)>

FROM

<table(s)>

[WHERE

<condition(s)>]

[GROUP BY

<column(s)>]

[HAVING

<condition(s)>]

ORDER BY

<column(s)> [ASC | DESC]

How to see ALL TABLES?

To list ALL TABLES that your account has access to, select through the ALL_TABLES:

```
select *  
from all_tables
```



Let's
continue...



Read (join) information from
MORE THAN ONE TABLE

SELECT ... WHERE

(Related Tables)

- 1) How to read (join) information from more than one table?

SELECT WHERE

- 2) How to use JOIN? Types of JOINS
- 3) How to use UNION?

More Than One Table

So far we looked at the data from **ONE** table only.

What if the information that is located in **SEVERAL** tables?



We are moving

More Than One Table

Getting information from related tables is called **JOIN** operation.

It is one of the most powerful operations in SQL language.

More Than One Table (JOIN)

We already met JOIN:

ID	First Name	Last Name	Sex		
1	Anna	Razgul	F		
2	Boris	Gurich	M		
3	Sri				
4	Anna	EntryNumber	OwnerId	PhoneNumber	PhoneType
		1001	1	(111) 111 1111	Mobile
		1002	2	(222) 222 2222	Home
		1003	3	(333) 333 3333	Home
		1004	1	(432) 555 4444	Work

To join:

Student.Id = Phone.OwnerId

More Than One Table (JOIN)

Another example.

We need a list of people and their department names. However, people names and department names are stored in different tables.

So we have to get one piece of information from one table, and another – from the second table.

More Than One Table (JOIN)

EMPLOYEES

Last_Name	First_Name	Department_Id
Patel	Anu	15
Ivanov	Sergey	30
Beyer	David	80
Chen	Alice	80
Johnson	John	80
Pham	Julia	80
Sumar	Avinash	90
Smirnoff	Peter	90
Patel	Ash	90
Patel	Anjana	110
Alison	David	120

DEPARTMENTS

Department_Id	Department_Name
10	Accounting
20	Payroll
30	Finance
80	IT
90	Marketing
110	Sales
112	Legal
130	Manufacturing

There is a DEPARTMENT_ID column, but not DEPARTMENT_NAME

More Than One Table (Resultset)

EMPLOYEES + DEPARTMENTS

Last_Name	First_Name	Department_Name
Ivanov	Sergey	Finance
Beyer	David	IT
Chen	Alice	IT
Johnson	John	IT
Pham	Julia	IT
Sumar	Avinash	Marketing
Smirnoff	Peter	Marketing
Patel	Ash	Marketing
Patel	Anjana	Sales

SELECT...WHERE

- 1) We *start* with specifying tables, where our data is located:

SELECT

FROM EMPLOYEES, DEPARTMENTS

WHERE

SELECT...WHERE

2) We merge tables, using PK-FK relation:

```
SELECT ....  
FROM   EMPLOYEES, DEPARTMENTS  
WHERE  DEPARTMENTS.department_ID =  
       EMPLOYEES.department_ID
```

SELECT...WHERE

3) Last step, after the tables are merged, choose what columns you need:

```
SELECT EMPLOYEES.FIRST_NAME,  
       EMPLOYEES.LAST_NAME,  
       DEPARTMENTS.DEPARTMENT_NAME  
FROM   EMPLOYEES, DEPARTMENTS  
WHERE  DEPARTMENTS.department_ID =  
       EMPLOYEES.department_ID
```

SELECT...WHERE (JOIN)

EMPLOYEES

FIRST_NAME	LAST_NAME	DEPARTMENT_ID
Steven	King	90
Neena	Kochhar	90
Lex	De Haan	90
Alexander	Hunold	60
Bruce	Ernst	60
David	Austin	~~
Valli	Pataballa	
Diana	Lorentz	
Nancy	Greenberg	
Daniel	Faviet	
More than 10 rows available. Increase rows selector to view more rows.		

DEPARTMENT_ID	DEPARTMENT_NAME
10	Administration
20	Marketing
30	Purchasing
40	Human Resources
50	Shipping
60	IT
70	Public Relations
80	Sales
90	Executive
100	Finance

More than 10 rows available. Increase rows selector to view more rows.

FIRST_NAME	LAST_NAME	DEPARTMENT_NAME
Ellen	Abel	Administration
Sundar	Ande	Administration
Mozhe	Atkinson	Administration
David	Austin	Administration
Hermann	Baer	Administration
Shelli	Baida	Administration
Amit	Banda	Administration
Elizabeth	Bates	Administration
Sarah	Bell	Administration
David	Bernstein	Administration

```
SELECT EMPLOYEES.FIRST_NAME,  
       EMPLOYEES.LAST_NAME,  
       DEPARTMENTS.DEPARTMENT_NAME  
  FROM EMPLOYEES, DEPARTMENTS  
 WHERE DEPARTMENTS.department_ID = EMPLOYEES.department_ID
```

Alias

Query can be very complex with long and complicated table and columns names. You can give a table or a column another name by using an **alias**.

For column:

```
SELECT <column1> AS alias_name  
FROM   <table1>
```

For Table:

```
SELECT <column1>  
FROM   <table1> alias_name
```

ALIAS is a nickname for column or table, easy to remember

Alias Example

```
SELECT EMPLOYEES.FIRST_NAME,  
       EMPLOYEES.LAST_NAME,  
       DEPARTMENTS.DEPARTMENT_NAME  
FROM   EMPLOYEES, DEPARTMENTS  
WHERE DEPARTMENTS.department_ID = EMPLOYEES.department_ID
```

Using Table alias:

```
SELECT e.FIRST_NAME,  
       e.LAST_NAME,  
       d.DEPARTMENT_NAME  
FROM   EMPLOYEES  e,  
       DEPARTMENTS d  
WHERE d.department_ID = e.department_ID
```

Exercise – More than one table

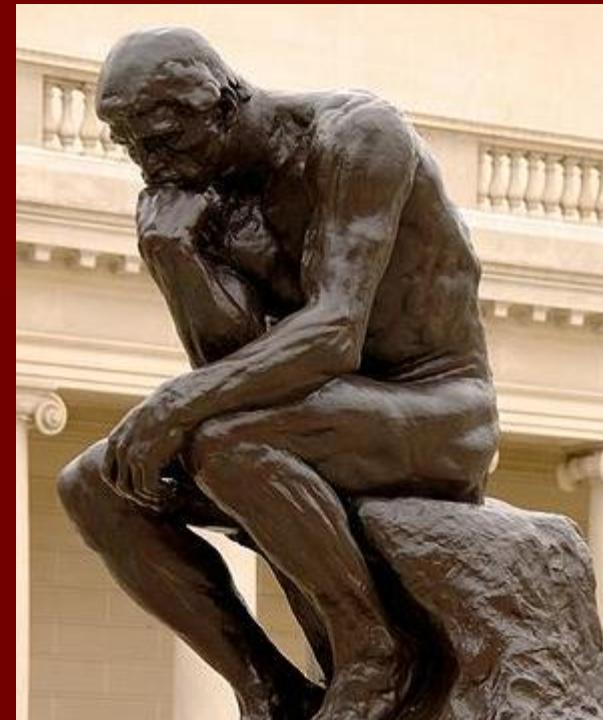
Display all employees (names, department id and department name) from the departments 30, 40, 50) and sort the resultset by the department_id, employee last name;

Exercise – More than one table

```
Select    e.department_id,  
        d.department_name,  
        e.last_name,  
        e.first_name  
From      employees    e,  
        departments  d  
Where    e.department_id = d.department_id  
and      e.department_id in (30, 40, 50)  
order by e.department_id, e.last_name
```

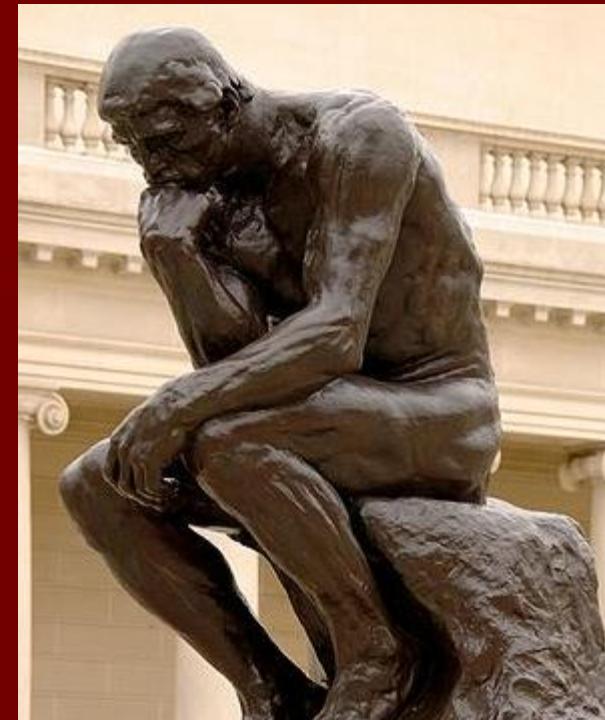
Quiz

1) List all employees, working
in 'Shipping' department;



Quiz

2) Show all department names
and number of employees
per department;



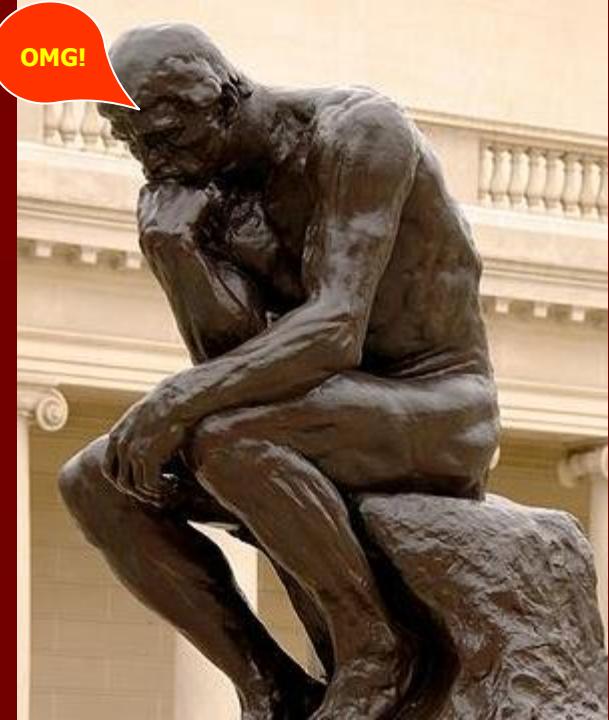
Quiz

OMG!

Advanced:

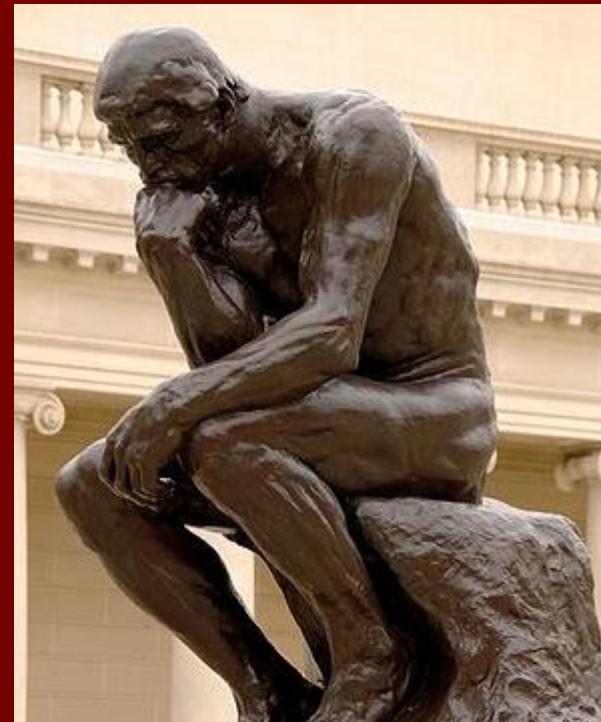
- 3) Retrieve all employees (first name, last name, department name, city and country), NOT residing in the US;

Hint: EMPLOYEES, DEPARTMENTS, LOCATIONS tables



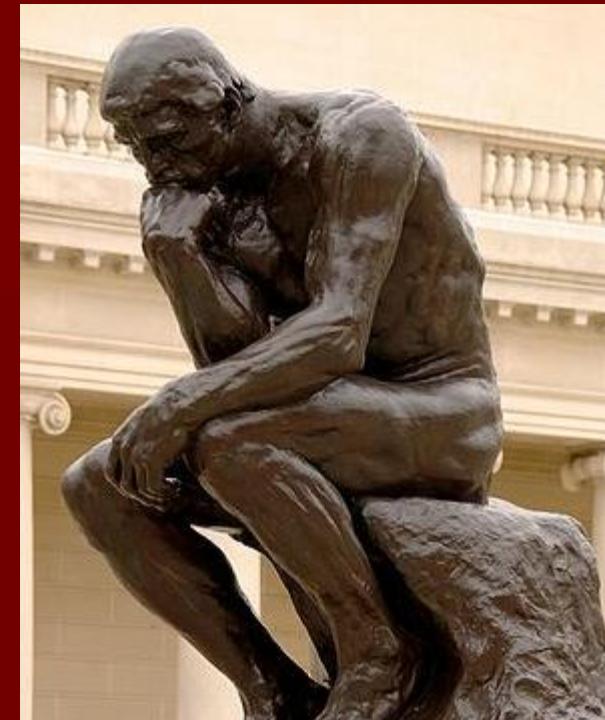
Quiz

4) List all Canadian employees, making over \$2000, showing highest paid employee first;



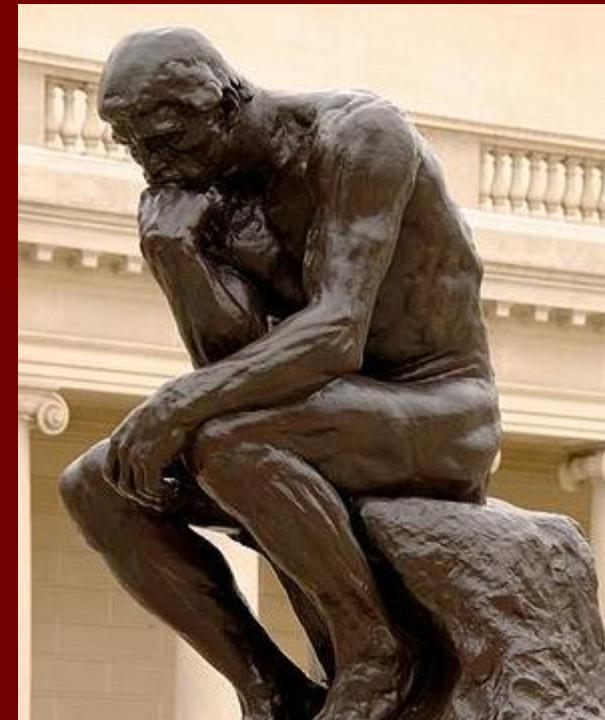
Quiz

5) Show in what country employees work on commissions;



Quiz

6) Find the maximum salary in IT department;



SELECT...WHERE (JOIN)

EMPLOYEES

Last_Name	First_Name	Department_Id
Patel	Anu	15
Beyer	David	80
Chen	Alice	80
Johnson	John	80
Pham	Julia	80
Alison	David	120

DEPARTMENTS

Department_Id	Department_Name
10	Accounting
20	Payroll
80	IT

RESULTSET

Last_Name	First_Name	Department_Name
Beyer	David	IT
Chen	Alice	IT
Johnson	John	IT
Pham	Julia	IT

Read employee and department name from two related tables:

```

SELECT e.FIRST_NAME,
       e.LAST_NAME,
       d.DEPARTMENT_NAME
  FROM EMPLOYEES e,
       DEPARTMENTS d
 WHERE d.department_ID = e.department_ID
  
```