KATARINA CEHOVSKI

**Request 1**

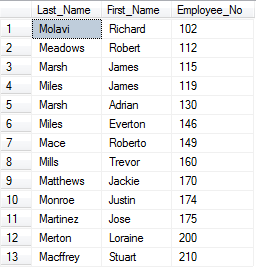
List the last name, first name and employee number of all employees that have a last name starting with M.

**Expected**  
  
3 columns  
Last\_Name, First\_Name, Employee\_No  
Marsh, Bob, 124  
…

**Query**

SELECT Last\_Name, First\_Name, Employee\_No FROM Employees

WHERE Last\_Name LIKE 'M%'



**Request 2**

List the last name, first name and employee number of all programmers who were hired on or before 11 Feburary 1994 sorted in descending order of last name.

**Expected**  
3 columns  
Last\_Name, First\_Name, Employee\_No  
Cehovski, Katarina, 106  
…

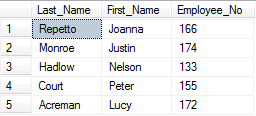
**Query**

SELECT e.Last\_Name, e.First\_Name, e.Employee\_No FROM Employees e

JOIN Jobs j ON j.Job\_ID = e.Job\_ID

WHERE j.Job\_Title = 'Programmer' AND e.Hire\_Date <= '1994-02-11'

ORDER BY e.Last\_Name DESC



**Request 3**

List all the data for all jobs where the maximum salary is greater than 12000 sorted in ascending order of the maximum salary.

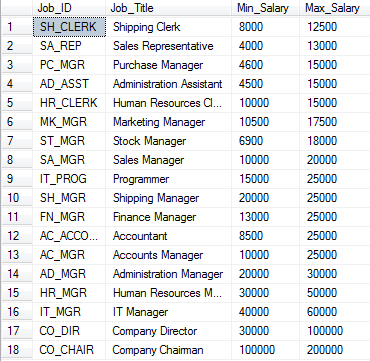
**Expected**

4 columns  
Job\_ID, Job\_Title, Min\_Salary, Max\_Salary  
IT\_SOFT\_DEV, Software Developer, 8000, 15000  
…  
  
**Query**

SELECT \* FROM Jobs

WHERE Max\_Salary > 12000

ORDER BY Max\_Salary ASC



**Request 4**

List all the data for all jobs where the minimum salary is less than or equal to 4500 sorted in descending order of the minimum salary.

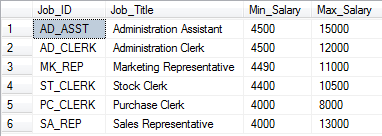
**Expected**

4 columns  
Job\_ID, Job\_Title, Min\_Salary, Max\_Salary  
AD\_CLERK, Administration Clerk, 4500, 10000  
…  
  
**Query**

SELECT \* FROM Jobs

WHERE Min\_Salary <= 4500

ORDER BY Min\_Salary DESC



**Request 5**

List the department name, location, last name and salary of employees who work in location 1700 sorted in ascending order of department name.

**Expected**

4 columns  
Department\_Name, Location\_ID, Last\_Name, Annual\_Salary  
Accounting, 1700, Cehovski, 15000  
…  
  
**Query**

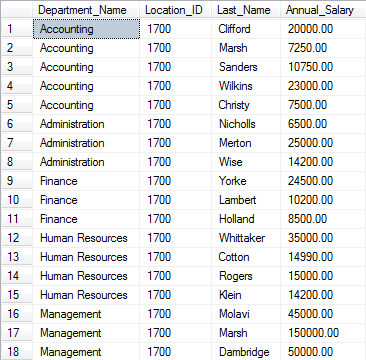
SELECT d.Department\_Name, d.Location\_ID, e.Last\_Name, e.Annual\_Salary

FROM Employees e

JOIN Departments d ON e.Department\_No= d.Department\_No

WHERE d.Location\_ID =1700

ORDER BY Department\_Name ASC



**Request 6**

Which jobs are found in the Marketing and Accounting departments?

**Expected**  
1 column  
Job\_Title  
Accounts Clerk  
…

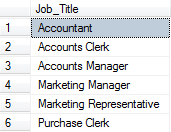
**Query**

SELECT DISTINCT j.Job\_Title FROM Jobs j

JOIN Employees e ON j.Job\_ID = e.Job\_ID

JOIN Departments d ON d.Department\_No =e.Department\_No

WHERE d.Department\_Name IN ('Marketing', 'Accounting')



**Request 7**

Show the total salaries figure for one week displayed with no decimal places.

**Expected**  
1 column with number  
Total\_Salary\_Week  
25465  
  
**Query**  
  
SELECT CAST(SUM(Annual\_Salary/52) AS INT) AS Total\_Salary\_Week FROM Employees



**Request 8**

Show the total number of jobs.

**Expected**  
  
1 column with number  
Total\_Number\_of\_Jobs  
51  
  
**Query**  
  
SELECT COUNT(Job\_ID) AS Total\_Number\_of\_Jobs FROM Jobs



**Request 9**

List the department number, department name and the number of employees for each department that has less than 4 employees grouping by department number and department name.

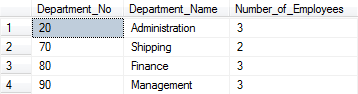
**Expected**  
3 columns  
Department\_No, Department\_Name, Number\_of\_Employees  
50, IT, 2  
…  
  
**Query**SELECT d.Department\_No, d.Department\_Name, COUNT(\*) AS Number\_of\_Employees

FROM Departments d JOIN Employees e

ON d.Department\_No = e.Department\_No

GROUP BY d.Department\_Name, d.Department\_No

HAVING COUNT(\*) < 4



**Request 10**

List the department number, department name and the number of employees for the department that has the lowest number of employees using appropriate grouping.

**Expected**  
  
3 columns  
Department\_No, Department\_Name, Number\_of\_Employees  
50, IT, 2  
  
**Query**  
  
SELECT TOP 1 WITH TIES d.Department\_No, d.Department\_Name, COUNT(\*) AS Number\_of\_Employees

From Departments d JOIN Employees e

ON d.Department\_No = e.Department\_No

GROUP BY d.Department\_No, d.Department\_Name

ORDER BY COUNT(\*) ASC



**Request 11**

List the department number and name for all departments where no sales representatives work.

**Expected**  
  
2 columns  
Department\_No, Department\_Name  
50, IT  
…  
  
**Query**  
  
SELECT d.Department\_No, d.Department\_Name FROM Departments d

WHERE d.Department\_No NOT IN

(

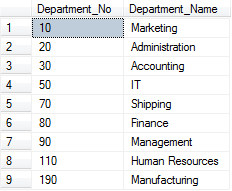
SELECT d.Department\_No FROM Departments d

JOIN Employees e ON e.Department\_No = d.Department\_No

JOIN Jobs j ON e.Job\_ID = j.Job\_ID

WHERE j.Job\_Title = 'Sales Representative'

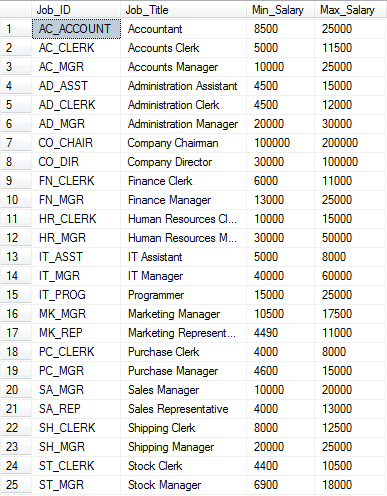
)



**Request 12**

Add the following new job IT\_ASST, IT Assistant, 5000, 8000

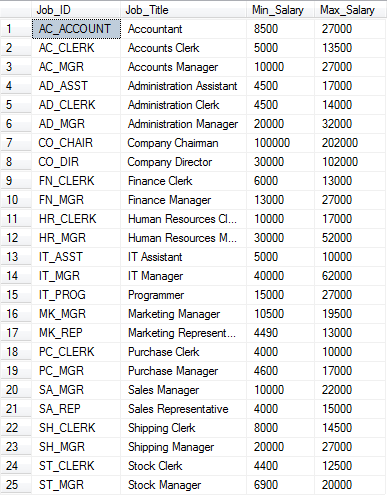
**Expected**  
New job will be added.  
Job\_ID: IT\_ASST, Job\_Title: IT Assistant, Min\_Salary: 5000, Max\_Salary: 8000  
  
**Query**  
  
INSERT INTO Jobs VALUES ('IT\_ASST', 'IT Assistant', 5000, 8000)



**Request 13**

Update all the maximum salaries for jobs with an increase of 2000

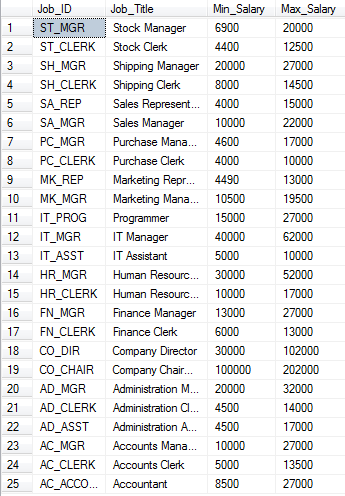
**Expected**  
All Max\_Salary will be increased.  
Before: 27000, After: 29000  
  
**Query**  
  
UPDATE Jobs SET Max\_Salary += 2000



**Request 14**

List all the data for jobs sorted in descending order of job id.

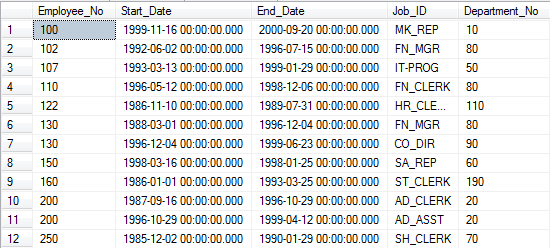
**Expected**  
4 columns  
Job\_ID, Job\_Title, Min\_Salary, Max\_Salary  
IT\_SOF\_DEV, Software Developer, 8000, 15000  
…  
  
  
**Query**  
SELECT \* FROM Jobs ORDER BY Job\_ID DESC



**Request 15**

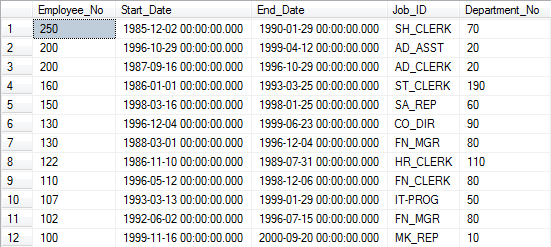
a) The job history for employee number 125 is no longer required. Delete this record.

**Expected**  
Employee wit Emloyee\_No 125 will no longer exist in table Job\_History.  
  
**Query**  
  
DELETE FROM Job\_History WHERE Employee\_No = 125



b) List all the data for job history sorted in descending order of employee number.

**Expected**  
5 columns  
Employee\_No, Start\_Date, End\_Date, Job\_ID, Department\_No  
356, 12/12/2001, 05/04/2011, IT\_SOFT\_DEVE, 50  
…  
  
**Query**  
SELECT \* FROM Job\_History ORDER BY Employee\_No DESC



**Request 16**

Create a new view for clerks’s details only using all the fields from the employee table.

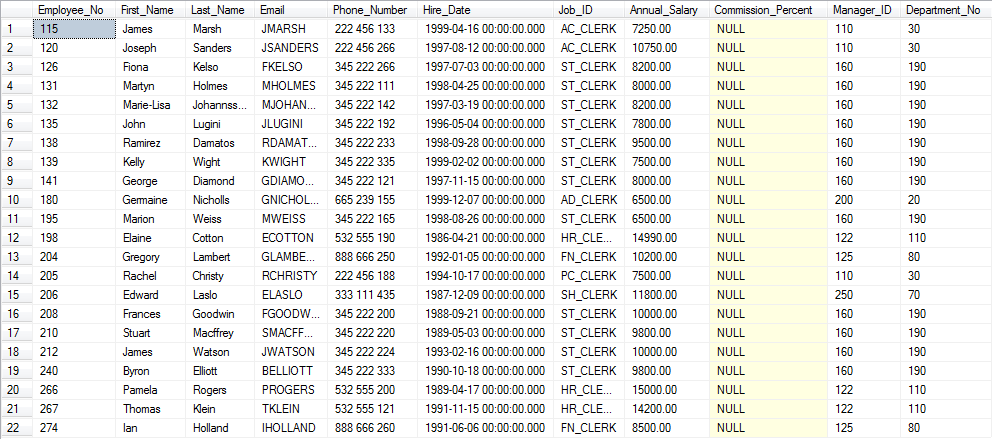
**Expected**  
New view table with name Clerks\_Details will be created.  
  
  
**Query**  
  
CREATE VIEW Clerks\_Details AS

(

SELECT \* FROM Employees

WHERE Job\_ID LIKE '%CLERK'

)

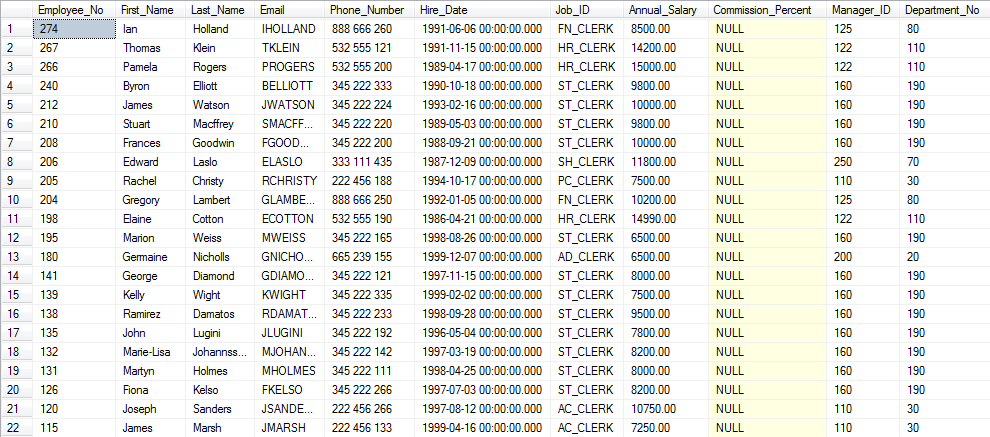


**Request 17**

Show all the fields and all the clerks using the view for clerks sorted in descending order of employee number.

**Expected**  
11 columns  
Employee\_No, First\_Name, Last\_Name, Email, Phone\_Number, Hire\_Date, Job\_ID, Annual\_Salary, Commission\_Percent, Manager\_ID, Department\_No  
  
102, Katarina, Cehovski, KCEHO, 456-421-123, 12/12/2012, ST\_CLERK, 7600.00, 0.05, 120, 30  
…  
  
**Query**  
  
SELECT \* FROM Clerks\_Details

ORDER BY Employee\_No DESC



**Request 18**

Grant the authority to all other users to access the view for clecks for SELECT statements only.

**Expected**  
All users will be able to use only SELECT statement.  
  
**Query**GRANT SELECT ON Clerks\_Details TO PUBLIC



**Request 19**

Create an index named LOC\_POSTAL\_CODE on the Postal Code in the locations table. Provide a printout showing that the index has been created.

**Expected**  
New index with name LOCATION\_POSTAL\_CODE will be added.  
  
**Query**  
  
CREATE INDEX LOCATION\_POSTAL\_CODE ON Locations (Postal\_Code)

