1. Выбрать менеджера (работника, у которого в подчинении есть хотя бы один человек), имеющего наименьший стаж. Если таких менеджеров несколько, вывести всех.
   1. Поля: ID менеджера, Hire\_date
   2. Сортировка: ID менеджера

**Запрос:**

SELECT EMPLOYEE\_ID, HIRE\_DATE FROM EMPLOYEES

WHERE EMPLOYEE\_ID IN (SELECT DISTINCT MANAGER\_ID FROM EMPLOYEES WHERE MANAGER\_ID IS NOT NULL)

AND HIRE\_DATE=(SELECT MAX(HIRE\_DATE) FROM EMPLOYEES

WHERE EMPLOYEE\_ID IN (SELECT DISTINCT MANAGER\_ID FROM EMPLOYEES WHERE MANAGER\_ID IS NOT NULL))

**Результат:**

1. Выбрать сотрудников, имеющих оклад выше среднего оклада подразделения, где они работают. Исключить из выборки сотрудников, которые не приписаны ни к какому подразделению.
   1. Поля: идентификатор сотрудника, фамилия сотрудника, оклад, идентификатор подразделения, к которому приписан сотрудник
   2. Сортировка: идентификатор подразделения, оклад, фамилия сотрудника

**Запрос:**

SELECT EMPLOYEES.EMPLOYEE\_ID, EMPLOYEES.FIRST\_NAME, EMPLOYEES.SALARY, DEPARTMENTS.DEPARTMENT\_ID FROM DEPARTMENTS

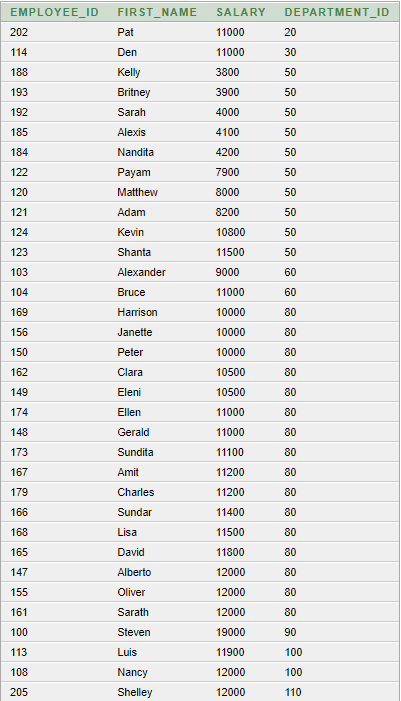
JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID

JOIN (SELECT DEPARTMENT\_ID, AVG(SALARY) AS AVERAGE\_SALARY FROM EMPLOYEES GROUP BY DEPARTMENT\_ID) A ON EMPLOYEES.DEPARTMENT\_ID=A.DEPARTMENT\_ID

WHERE EMPLOYEES.SALARY > A.AVERAGE\_SALARY

ORDER BY DEPARTMENTS.DEPARTMENT\_ID, EMPLOYEES.SALARY, EMPLOYEES.FIRST\_NAME

**Результат:**



1. Выбрать сотрудников, имеющих оклад равный минимальному окладу подразделения, где они работают. Исключить из выборки сотрудников, которые не приписаны ни к какому подразделению.
   1. Поля: идентификатор сотрудника, фамилию сотрудника, оклад, установленный сотруднику, идентификатор подразделения, к которому приписан сотрудник
   2. Сортировка: идентификатор подразделения, оклад, фамилия сотрудника

**Запрос:**

SELECT EMPLOYEES.EMPLOYEE\_ID, EMPLOYEES.FIRST\_NAME, EMPLOYEES.SALARY, DEPARTMENTS.DEPARTMENT\_ID FROM DEPARTMENTS

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID

JOIN (SELECT DEPARTMENT\_ID, MIN(SALARY) AS MINIMAL\_SALARY FROM EMPLOYEES GROUP BY DEPARTMENT\_ID) M ON EMPLOYEES.DEPARTMENT\_ID=M.DEPARTMENT\_ID

WHERE EMPLOYEES.SALARY = M.MINIMAL\_SALARY

ORDER BY DEPARTMENTS.DEPARTMENT\_ID, EMPLOYEES.SALARY, EMPLOYEES.FIRST\_NAME

**Результат:**



1. Выбрать среди работников Америки (region\_name = "Americas") тех, чья зарплата превосходит среднюю зарплату менеджеров из Европы (region\_name = "Europe") с наибольшим количеством подчиненных.
   1. Поля: Фамилия сотрудника, Имя сотрудника, зарплата
   2. Сортировка: Фамилия сотрудника, Имя сотрудника

**Запрос:**

WITH EUROPE\_MANAGERS AS

(SELECT EMPLOYEE\_ID, MANAGER\_ID FROM REGIONS

JOIN COUNTRIES ON REGIONS.REGION\_NAME='Europe' AND REGIONS.REGION\_ID=COUNTRIES.REGION\_ID

JOIN LOCATIONS ON COUNTRIES.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID

WHERE MANAGER\_ID IS NOT NULL),

MANAGERS\_SUBORDINATES AS

(SELECT MANAGER\_ID, COUNT(EMPLOYEE\_ID) AS SUBORDINATE\_NUMBER FROM EUROPE\_MANAGERS

GROUP BY MANAGER\_ID),

MAX\_SUBORDINATES AS

(SELECT MANAGER\_ID FROM MANAGERS\_SUBORDINATES

WHERE MANAGERS\_SUBORDINATES.SUBORDINATE\_NUMBER=(SELECT MAX(SUBORDINATE\_NUMBER) FROM MANAGERS\_SUBORDINATES)),

AVERAGE\_SALARY AS

(SELECT AVG(SALARY) AS AVG\_S FROM EMPLOYEES, MAX\_SUBORDINATES

WHERE EMPLOYEE\_ID = MAX\_SUBORDINATES.MANAGER\_ID)

SELECT EMPLOYEES.FIRST\_NAME, EMPLOYEES.LAST\_NAME, EMPLOYEES.SALARY FROM AVERAGE\_SALARY, REGIONS

JOIN COUNTRIES ON REGIONS.REGION\_NAME='Americas' AND REGIONS.REGION\_ID=COUNTRIES.REGION\_ID

JOIN LOCATIONS ON COUNTRIES.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID

WHERE EMPLOYEES.SALARY > AVERAGE\_SALARY.AVG\_S

ORDER BY EMPLOYEES.FIRST\_NAME, EMPLOYEES.LAST\_NAME

**Результат:**



1. Сформировать список подразделений с информацией о сумме окладов всех сотрудников и проценте, который эта сумма составляет от суммы окладов всех сотрудников компании. Если в подразделении нет сотрудников, то считать, что сумма их окладов равна нулю. В отчете отдельной строкой учесть сведения о сотрудниках, которые не приписаны ни к какому подразделению компании (null в поле подразделения). При вычислении процентов должно быть использовано округление результата до 2-х знаков после запятой.
   1. Поля: идентификатор подразделения компании, сумма окладов сотрудников подразделения компании, процент
   2. Сортировка: процент по убыванию

**Запрос:**

WITH SUMMARY\_SALARIES AS

(SELECT DEPARTMENTS.DEPARTMENT\_ID,

SUM(CASE

WHEN SALARY IS NULL THEN 0

ELSE SALARY END) AS SUMMARY\_SALARY

FROM DEPARTMENTS

LEFT JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID

GROUP BY DEPARTMENTS.DEPARTMENT\_ID)

SELECT SUMMARY\_SALARIES.DEPARTMENT\_ID, SUMMARY\_SALARY,

CASE WHEN SUMMARY\_SALARY!=0 THEN ROUND(100\*EMPLOYEES.SALARY/SUMMARY\_SALARY,2)

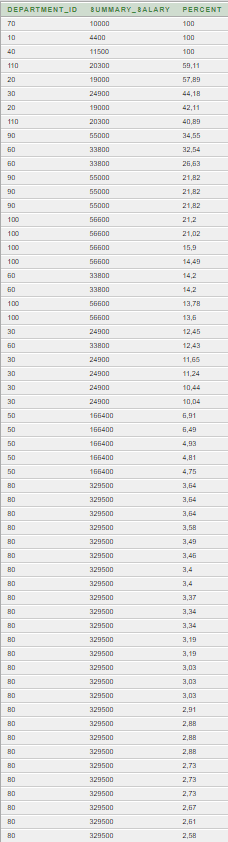
ELSE 0 END PERCENT

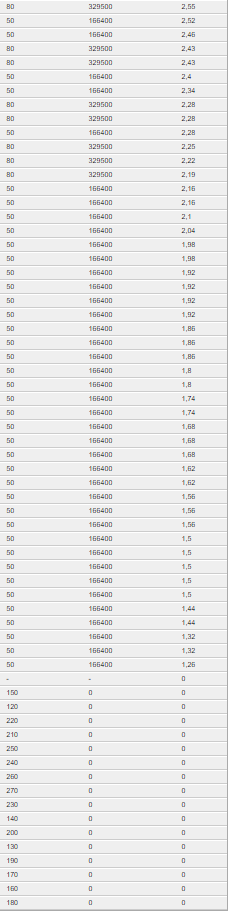
FROM EMPLOYEES

FULL OUTER JOIN SUMMARY\_SALARIES ON EMPLOYEES.DEPARTMENT\_ID=SUMMARY\_SALARIES.DEPARTMENT\_ID

ORDER BY PERCENT DESC

**Результат:**





1. Сформировать список подразделений с информацией о сумме среднего оклада сотрудников по подразделениям компании и проценте, который эта сумма составляет от средней величины окладов всех сотрудников компании. Если в подразделении нет сотрудников, то считать, что сумма их окладов равна нулю. В отчете отдельной строкой учесть сведения о сотрудниках, которые не приписаны ни к какому подразделению компании
   1. Поля: идентификатор подразделения компании, средняя зарплата по подразделению, процент
   2. Сортировка: идентификатор подразделения компании, средняя зарплата по подразделению, процент по убыванию

**Запрос:**

WITH AVERAGE\_SALARIES AS

(SELECT DEPARTMENTS.DEPARTMENT\_ID,

ROUND(AVG(CASE

WHEN SALARY IS NULL THEN 0

ELSE SALARY END),2) AS AVERAGE\_SALARY

FROM DEPARTMENTS

LEFT JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID

GROUP BY DEPARTMENTS.DEPARTMENT\_ID)

SELECT AVERAGE\_SALARIES.DEPARTMENT\_ID, AVERAGE\_SALARY,

CASE WHEN AVERAGE\_SALARY!=0 THEN ROUND(100\*EMPLOYEES.SALARY/AVERAGE\_SALARY, 2)

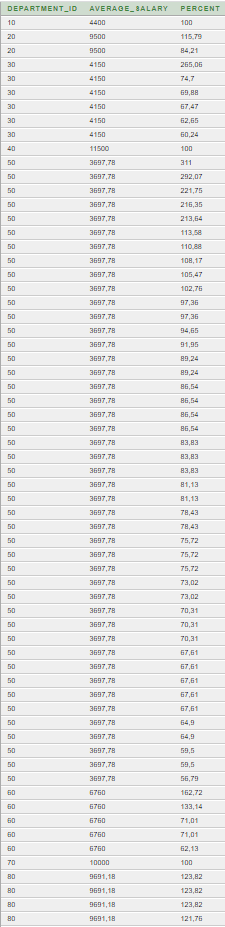
ELSE 0 END PERCENT

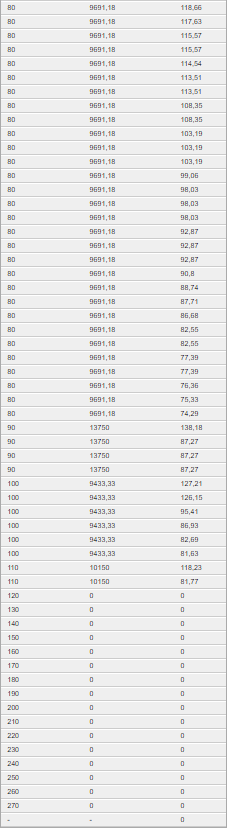
FROM EMPLOYEES

FULL OUTER JOIN AVERAGE\_SALARIES ON EMPLOYEES.DEPARTMENT\_ID=AVERAGE\_SALARIES.DEPARTMENT\_ID

ORDER BY AVERAGE\_SALARIES.DEPARTMENT\_ID, AVERAGE\_SALARY, PERCENT DESC

**Результат:**





1. Вывести имена сотрудников и количество работников с одним именем
   1. Поля: Имя, Количество работников
   2. Сортировка: Количество работников, Имя

**Запрос:**

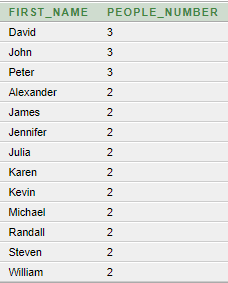
SELECT \* FROM (SELECT FIRST\_NAME, COUNT(EMPLOYEE\_ID) AS PEOPLE\_NUMBER FROM EMPLOYEES

GROUP BY FIRST\_NAME

ORDER BY PEOPLE\_NUMBER DESC, FIRST\_NAME)

WHERE PEOPLE\_NUMBER > 1

**Результат:**



1. Вывести для каждого отдела из Америки (region\_name = "Americas") количество работников в иерархии до третьего уровня. Первый уровень - работники без руководителя, второй уровень - это их подчиненные, а третий уровень - подчиненные работников второго уровня.
   1. Поля: Название отдела, количество работников I уровня, количество работников II уровня, количество работников III уровня
   2. Сортировка: Название отдела

**Запрос:**

WITH ALL\_EMPLOYEES AS

(SELECT DEPARTMENT\_ID, EMPLOYEE\_ID, MANAGER\_ID FROM REGIONS

JOIN COUNTRIES ON REGIONS.REGION\_NAME='Americas' AND REGIONS.REGION\_ID=COUNTRIES.REGION\_ID

JOIN LOCATIONS ON COUNTRIES.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID),

A\_LVL AS

(SELECT DEPARTMENT\_ID, EMPLOYEE\_ID FROM ALL\_EMPLOYEES

WHERE MANAGER\_ID IS NULL),

B\_LVL AS

(SELECT ALL\_EMPLOYEES.DEPARTMENT\_ID, ALL\_EMPLOYEES.EMPLOYEE\_ID FROM ALL\_EMPLOYEES

JOIN A\_LVL ON ALL\_EMPLOYEES.MANAGER\_ID=A\_LVL.EMPLOYEE\_ID),

C\_LVL AS

(SELECT ALL\_EMPLOYEES.DEPARTMENT\_ID, ALL\_EMPLOYEES.EMPLOYEE\_ID FROM ALL\_EMPLOYEES

JOIN B\_LVL ON ALL\_EMPLOYEES.MANAGER\_ID=B\_LVL.EMPLOYEE\_ID),

A\_LVL\_NUMBER AS

(SELECT DEPARTMENTS.DEPARTMENT\_ID, DEPARTMENTS.DEPARTMENT\_NAME, COUNT(A\_LVL.EMPLOYEE\_ID) AS TOP\_NUMBER FROM DEPARTMENTS

LEFT JOIN A\_LVL ON DEPARTMENTS.DEPARTMENT\_ID=A\_LVL.DEPARTMENT\_ID

GROUP BY DEPARTMENTS.DEPARTMENT\_ID, DEPARTMENTS.DEPARTMENT\_NAME),

B\_LVL\_NUMBER AS

(SELECT DEPARTMENTS.DEPARTMENT\_ID, DEPARTMENTS.DEPARTMENT\_NAME, COUNT(B\_LVL.EMPLOYEE\_ID) AS MEDIUM\_NUMBER FROM DEPARTMENTS

LEFT JOIN B\_LVL ON DEPARTMENTS.DEPARTMENT\_ID=B\_LVL.DEPARTMENT\_ID

GROUP BY DEPARTMENTS.DEPARTMENT\_ID, DEPARTMENTS.DEPARTMENT\_NAME),

C\_LVL\_NUMBER AS

(SELECT DEPARTMENTS.DEPARTMENT\_ID, DEPARTMENTS.DEPARTMENT\_NAME, COUNT(C\_LVL.EMPLOYEE\_ID) AS LOW\_NUMBER FROM DEPARTMENTS

LEFT JOIN C\_LVL ON DEPARTMENTS.DEPARTMENT\_ID=C\_LVL.DEPARTMENT\_ID

GROUP BY DEPARTMENTS.DEPARTMENT\_ID, DEPARTMENTS.DEPARTMENT\_NAME)

SELECT A\_LVL\_NUMBER.DEPARTMENT\_NAME, A\_LVL\_NUMBER.TOP\_NUMBER, B\_LVL\_NUMBER.MEDIUM\_NUMBER, C\_LVL\_NUMBER.LOW\_NUMBER FROM EMPLOYEES

JOIN A\_LVL\_NUMBER ON EMPLOYEES.DEPARTMENT\_ID=A\_LVL\_NUMBER.DEPARTMENT\_ID

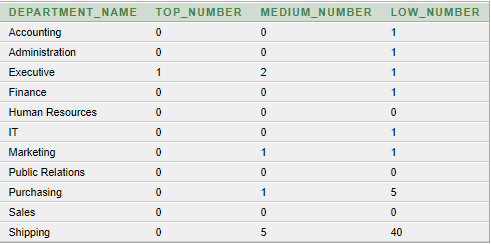
JOIN B\_LVL\_NUMBER ON EMPLOYEES.DEPARTMENT\_ID=B\_LVL\_NUMBER.DEPARTMENT\_ID

JOIN C\_LVL\_NUMBER ON EMPLOYEES.DEPARTMENT\_ID=C\_LVL\_NUMBER.DEPARTMENT\_ID

GROUP BY A\_LVL\_NUMBER.DEPARTMENT\_NAME, A\_LVL\_NUMBER.TOP\_NUMBER, B\_LVL\_NUMBER.MEDIUM\_NUMBER, C\_LVL\_NUMBER.LOW\_NUMBER

ORDER BY A\_LVL\_NUMBER.DEPARTMENT\_NAME

**Результат:**



1. Вывести названия всех отделов, в которых наименьшая зарплата выше средней зарплаты в Америке (region\_name = "Americas")
   1. Поля: Название отдела
   2. Сортировка: Название отдела

**Запрос:**

WITH AVERAGE\_SALARY AS

(SELECT ROUND(AVG(SALARY),2) AS AVG\_SALARY FROM REGIONS

JOIN COUNTRIES ON REGIONS.REGION\_NAME='Americas' AND REGIONS.REGION\_ID=COUNTRIES.REGION\_ID

JOIN LOCATIONS ON COUNTRIES.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID),

MIN\_SALARIES AS

(SELECT DEPARTMENT\_NAME, MIN(SALARY) AS MIN\_SALARY FROM DEPARTMENTS

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID

GROUP BY DEPARTMENTS.DEPARTMENT\_NAME)

SELECT DEPARTMENT\_NAME FROM MIN\_SALARIES, AVERAGE\_SALARY

WHERE MIN\_SALARY > AVG\_SALARY

ORDER BY DEPARTMENT\_NAME

**Результат:**



1. Вывести общую зарплату работников, работающих в отделе с наименьшим количеством сотрудников
   1. Поля: Номер отдела, общая сумма зарплаты
   2. Сортировка: Номер отдела

**Запрос:**

WITH DEPARTMENT\_SALARY AS

(SELECT DEPARTMENTS.DEPARTMENT\_ID, DEPARTMENTS.DEPARTMENT\_NAME, COUNT(EMPLOYEE\_ID) AS EMPLOYEE\_NUMBER, SUM(SALARY) AS SUM\_SALARY FROM DEPARTMENTS

LEFT JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID

GROUP BY DEPARTMENTS.DEPARTMENT\_ID, DEPARTMENTS.DEPARTMENT\_NAME

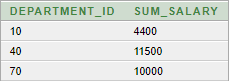
ORDER BY DEPARTMENTS.DEPARTMENT\_ID, DEPARTMENTS.DEPARTMENT\_NAME, EMPLOYEE\_NUMBER)

SELECT DEPARTMENT\_ID, SUM\_SALARY FROM DEPARTMENT\_SALARY

WHERE EMPLOYEE\_NUMBER = (SELECT MIN(EMPLOYEE\_NUMBER) FROM DEPARTMENT\_SALARY WHERE EMPLOYEE\_NUMBER!=0)

ORDER BY DEPARTMENT\_ID

**Результат:**



1. Вывести однофамильцев. Если встречается более двух - вывести их количество.
   1. Поля: Фамилия, Количество работников
   2. Сортировка: Фамилия

**Запрос:**

SELECT LAST\_NAME,

CASE

WHEN PEOPLE\_NUMBER > 2 THEN PEOPLE\_NUMBER

ELSE NULL END AS PEOPLE\_NUMBER

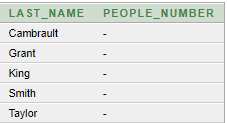
FROM (SELECT \* FROM (SELECT LAST\_NAME, COUNT(EMPLOYEE\_ID) AS PEOPLE\_NUMBER FROM EMPLOYEES

GROUP BY LAST\_NAME

ORDER BY LAST\_NAME)

WHERE PEOPLE\_NUMBER > 1)

**Результат:**



1. В стране с наибольшим количеством подразделений (департаментов) выбрать работника с максимальной зарплатой. Если таких стран несколько, то найти такого работника для каждой из них (то же касается нескольких работников).
   1. Поля: ID страны, ID работника
   2. Сортировка: ID страны, ID работника

**Запрос:**

WITH DEPARTMENTS\_NUMBER AS

(SELECT COUNTRIES.COUNTRY\_ID, COUNT(DEPARTMENTS.DEPARTMENT\_ID) AS DEPARTMENT\_NUMBER FROM COUNTRIES

JOIN LOCATIONS ON COUNTRIES.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

GROUP BY COUNTRIES.COUNTRY\_ID),

MAX\_DEPARTMENTS AS

(SELECT COUNTRY\_ID FROM DEPARTMENTS\_NUMBER

WHERE DEPARTMENT\_NUMBER=(SELECT MAX(DEPARTMENT\_NUMBER) FROM DEPARTMENTS\_NUMBER)),

ALL\_EMPLOYEES AS

(SELECT COUNTRIES.COUNTRY\_ID, EMPLOYEES.EMPLOYEE\_ID, SALARY FROM COUNTRIES

JOIN MAX\_DEPARTMENTS ON COUNTRIES.COUNTRY\_ID=MAX\_DEPARTMENTS.COUNTRY\_ID

JOIN LOCATIONS ON MAX\_DEPARTMENTS.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID)

SELECT COUNTRY\_ID, EMPLOYEE\_ID FROM ALL\_EMPLOYEES

WHERE SALARY=(SELECT MAX(SALARY) FROM ALL\_EMPLOYEES)

**Результат:**



1. В стране с наибольшим количеством подразделений (департаментов) выбрать работника, у которого в подчинении больше всего человек. Если таких стран несколько, то найти такого работника для каждой из них.
   1. Поля: ID работника; количество подчиненных сотрудников
   2. Сортировка: ID работника

**Запрос:**

WITH DEPARTMENTS\_NUMBER AS

(SELECT COUNTRIES.COUNTRY\_ID, COUNT(DEPARTMENTS.DEPARTMENT\_ID) AS DEPARTMENT\_NUMBER FROM COUNTRIES

JOIN LOCATIONS ON COUNTRIES.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

GROUP BY COUNTRIES.COUNTRY\_ID),

MAX\_DEPARTMENTS AS

(SELECT COUNTRY\_ID FROM DEPARTMENTS\_NUMBER

WHERE DEPARTMENT\_NUMBER=(SELECT MAX(DEPARTMENT\_NUMBER) FROM DEPARTMENTS\_NUMBER)),

MANAGERS AS

(SELECT MANAGER\_ID, COUNT(EMPLOYEE\_ID) AS SUBORDINATE\_NUMBER FROM COUNTRIES

JOIN MAX\_DEPARTMENTS ON COUNTRIES.COUNTRY\_ID=MAX\_DEPARTMENTS.COUNTRY\_ID

JOIN LOCATIONS ON MAX\_DEPARTMENTS.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID

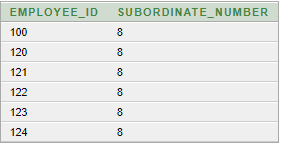
GROUP BY MANAGER\_ID)

SELECT MANAGER\_ID AS EMPLOYEE\_ID, SUBORDINATE\_NUMBER FROM MANAGERS

WHERE SUBORDINATE\_NUMBER=(SELECT MAX(SUBORDINATE\_NUMBER) FROM MANAGERS)

ORDER BY EMPLOYEE\_ID

**Результат:**



1. В стране, в которой проживает сотрудник с наименьшим стажем (если таких сотрудников несколько, то рассмотреть страну для каждого из них), выбрать работника, в подчинении которого больше всего человек.
   1. Поля: ID работника, количество подчиненных сотрудников
   2. Сортировка: ID работника

**Запрос:**

WITH ALL\_EMPLOYEES AS

(SELECT COUNTRIES.COUNTRY\_ID, HIRE\_DATE FROM COUNTRIES

JOIN LOCATIONS ON COUNTRIES.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID),

MAX\_DATE AS

(SELECT COUNTRY\_ID FROM ALL\_EMPLOYEES

WHERE HIRE\_DATE=(SELECT MAX(HIRE\_DATE) FROM ALL\_EMPLOYEES)),

MANAGERS AS

(SELECT MANAGER\_ID, COUNT(EMPLOYEE\_ID) AS SUBORDINATE\_NUMBER FROM COUNTRIES

JOIN MAX\_DATE ON COUNTRIES.COUNTRY\_ID=MAX\_DATE.COUNTRY\_ID

JOIN LOCATIONS ON MAX\_DATE.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID

GROUP BY MANAGER\_ID)

SELECT MANAGER\_ID AS EMPLOYEE\_ID, SUBORDINATE\_NUMBER FROM MANAGERS

WHERE SUBORDINATE\_NUMBER=(SELECT MAX(SUBORDINATE\_NUMBER) FROM MANAGERS)

ORDER BY EMPLOYEE\_ID

**Результат:**



1. В стране, в которой проживает сотрудник с наибольшим стажем (если таких несколько, то рассмотреть страну для каждого из них), выбрать работника, с наибольшей зарплатой его подчиненных.
   1. Поля: ID работника; Сумма зарплат подчиненных
   2. Сортировка: ID работника

**Запрос:**

WITH ALL\_EMPLOYEES AS

(SELECT COUNTRIES.COUNTRY\_ID, HIRE\_DATE FROM COUNTRIES

JOIN LOCATIONS ON COUNTRIES.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID),

MIN\_DATE AS

(SELECT COUNTRY\_ID FROM ALL\_EMPLOYEES

WHERE HIRE\_DATE=(SELECT MIN(HIRE\_DATE) FROM ALL\_EMPLOYEES)),

MANAGERS AS

(SELECT EMPLOYEE\_ID, MANAGER\_ID, SALARY FROM COUNTRIES

JOIN MIN\_DATE ON COUNTRIES.COUNTRY\_ID=MIN\_DATE.COUNTRY\_ID

JOIN LOCATIONS ON MIN\_DATE.COUNTRY\_ID=LOCATIONS.COUNTRY\_ID

JOIN DEPARTMENTS ON LOCATIONS.LOCATION\_ID=DEPARTMENTS.LOCATION\_ID

JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT\_ID=EMPLOYEES.DEPARTMENT\_ID

WHERE MANAGER\_ID IS NOT NULL),

REQUIRED\_MANAGERS AS

(SELECT DISTINCT MANAGER\_ID FROM MANAGERS

WHERE SALARY=(SELECT MAX(SALARY) FROM MANAGERS))

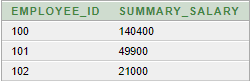
SELECT EMPLOYEES.MANAGER\_ID AS EMPLOYEE\_ID, SUM(SALARY) AS SUMMARY\_SALARY FROM EMPLOYEES, REQUIRED\_MANAGERS

WHERE EMPLOYEES.MANAGER\_ID IN REQUIRED\_MANAGERS.MANAGER\_ID

GROUP BY EMPLOYEES.MANAGER\_ID

ORDER BY EMPLOYEE\_ID

**Результат:**



1. Определить год, в котором было трудоустроено больше всего человек.
   1. Поля: Год; Кол-во трудоустроенных
   2. Сортировка: Год

**Запрос:**

SELECT \* FROM (SELECT HIRE\_DATE, COUNT(EMPLOYEE\_ID) AS PEOPLE\_NUMBER FROM EMPLOYEES

GROUP BY HIRE\_DATE)

WHERE PEOPLE\_NUMBER=(SELECT MAX(PEOPLE\_NUMBER) FROM (SELECT HIRE\_DATE, COUNT(EMPLOYEE\_ID) AS PEOPLE\_NUMBER FROM EMPLOYEES

GROUP BY HIRE\_DATE))

ORDER BY HIRE\_DATE

**Результат:**

