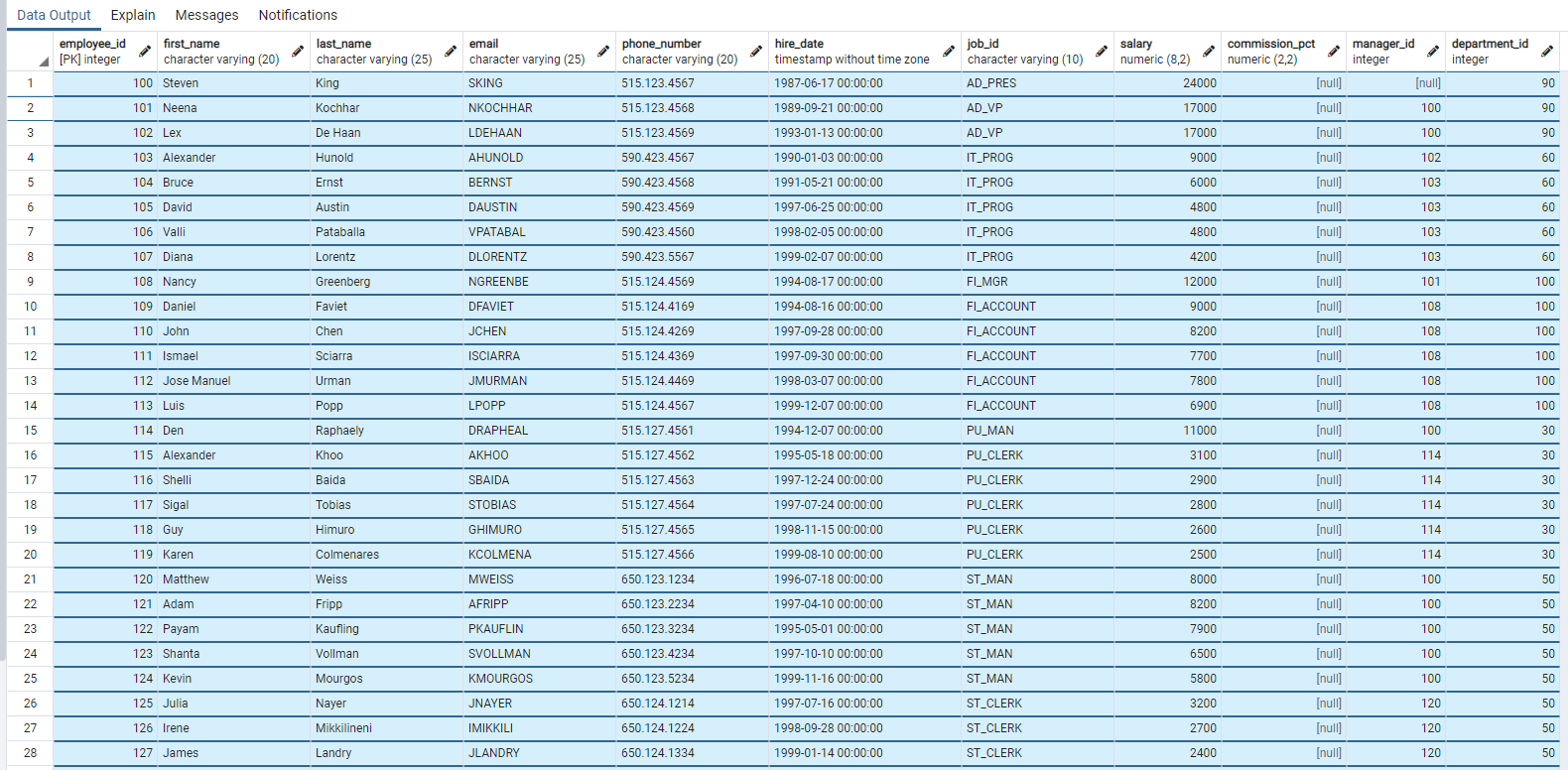
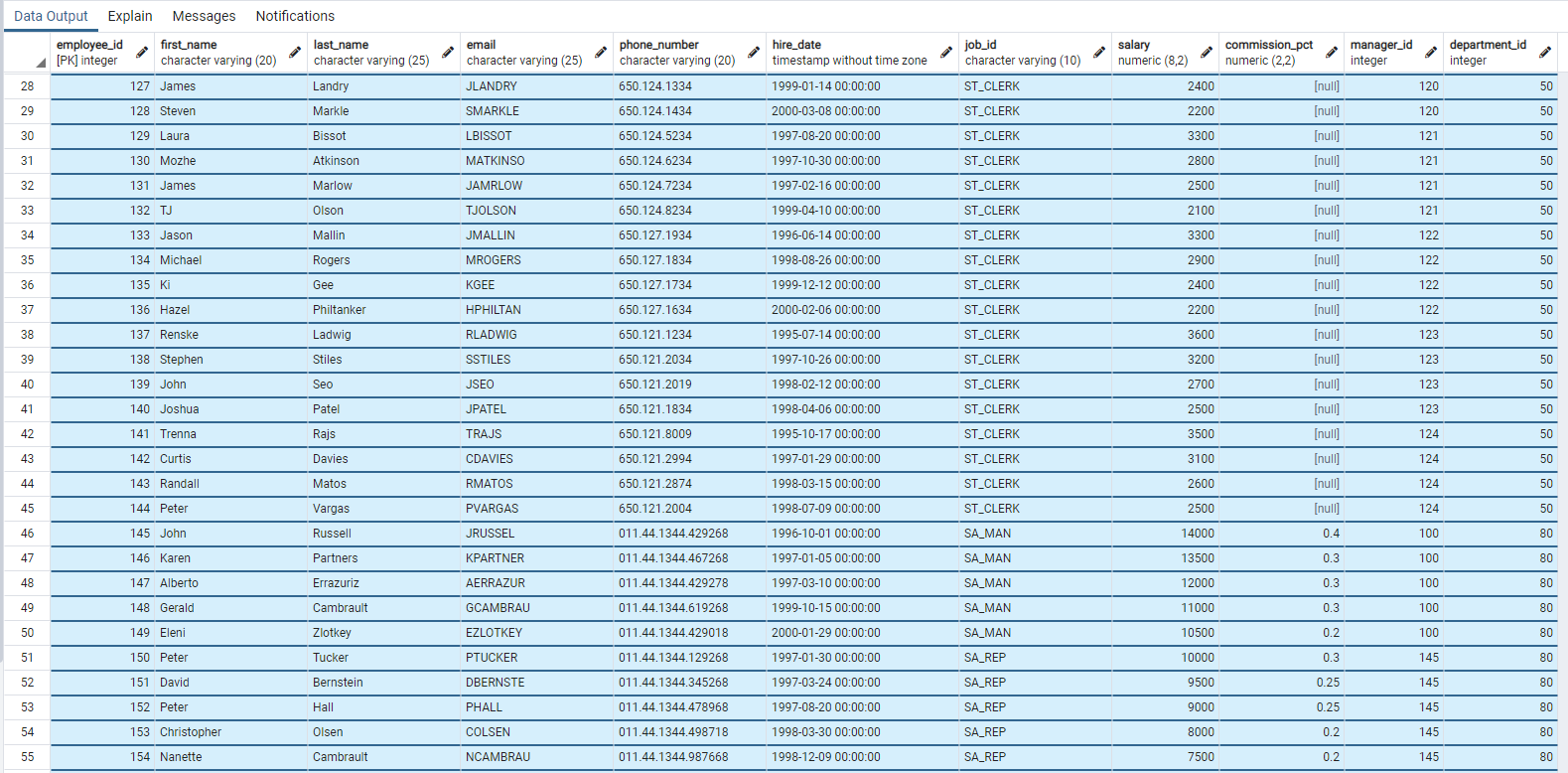
**Restricting and Sorting Data**

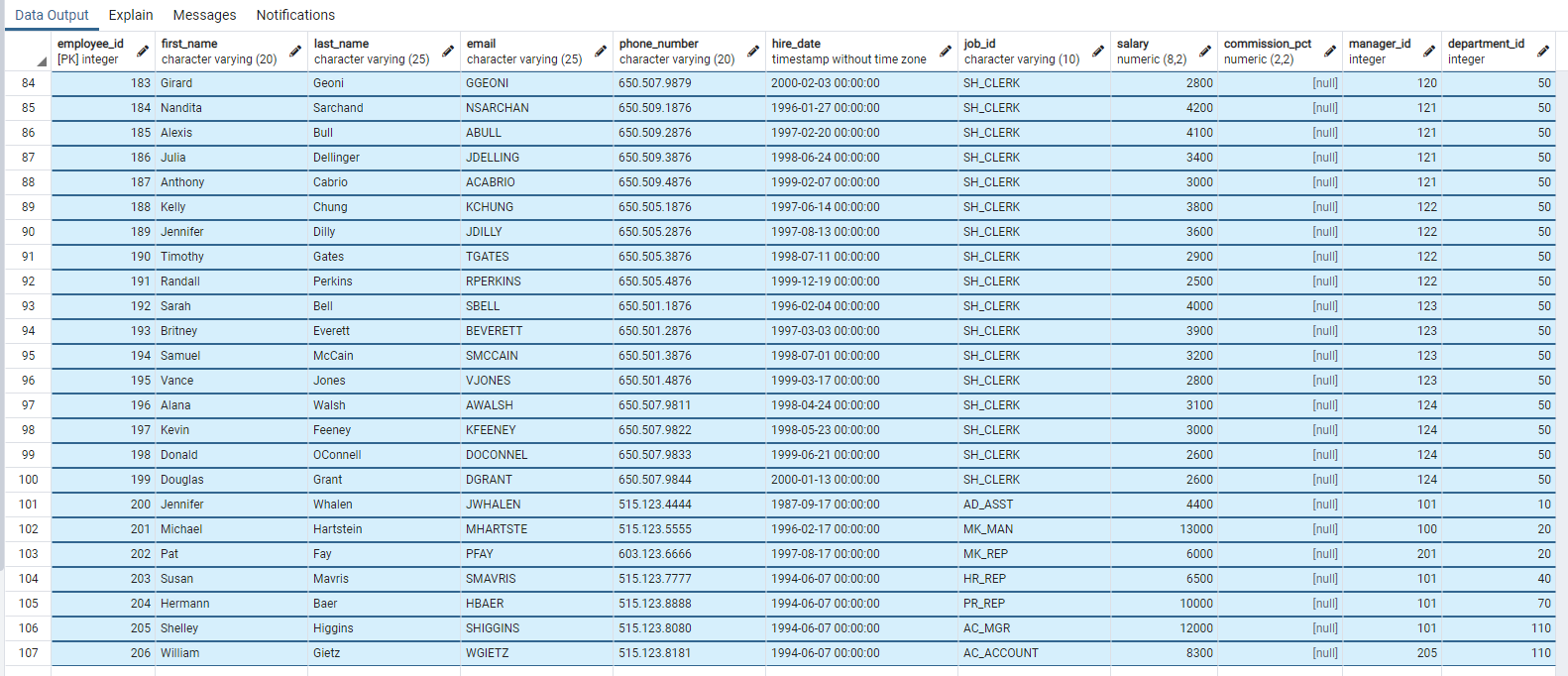
1. Таблица Employees. Получить список с информацией обо всех сотрудниках

**SELECT** \* **FROM** employees



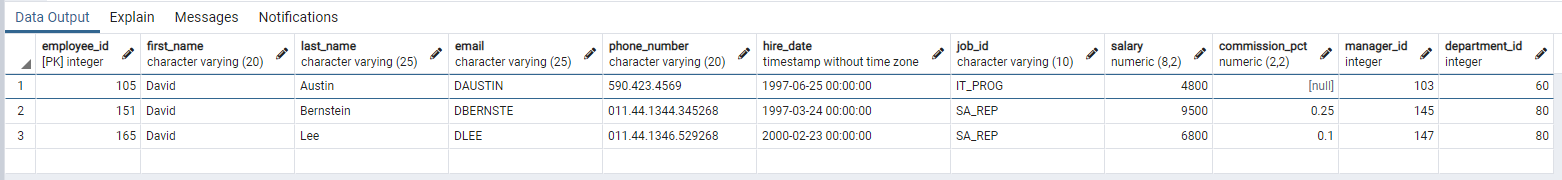






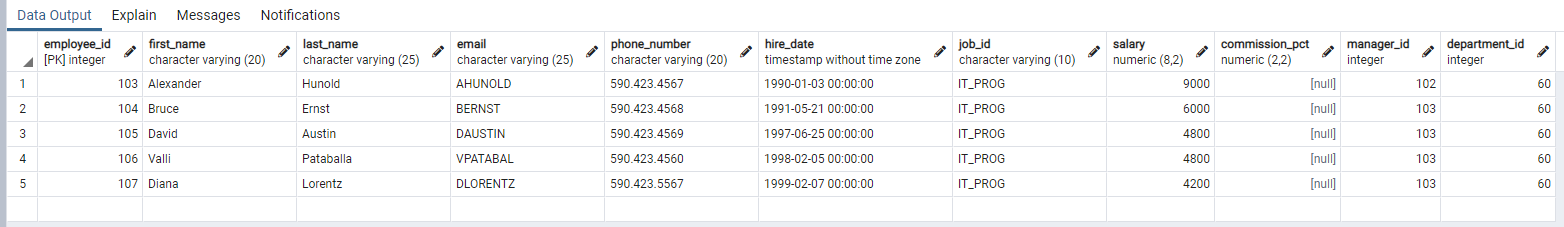
2. Таблица Employees. Получить список всех сотрудников с именем 'David'

**SELECT** \* **FROM** employees **WHERE** first\_name = 'David'



3. Таблица Employees. Получить список всех сотрудников с job\_id равным 'IT\_PROG'

**SELECT** \* **FROM** employees **WHERE** job\_id = 'IT\_PROG'



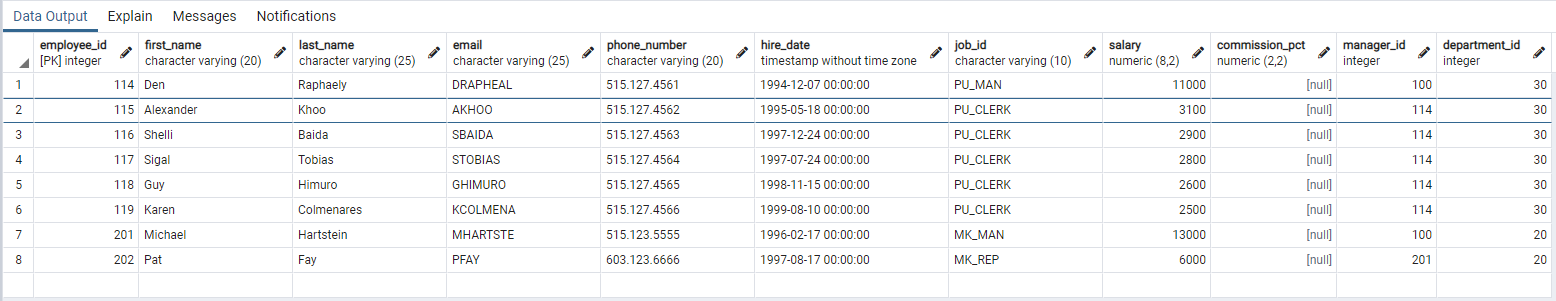
4. Таблица Employees. Получить список всех сотрудников из 50го отдела (department\_id) с зарплатой(salary), большей 4000

**SELECT** \* **FROM** employees **WHERE** depatment\_id = 50 **AND** salary > 4000



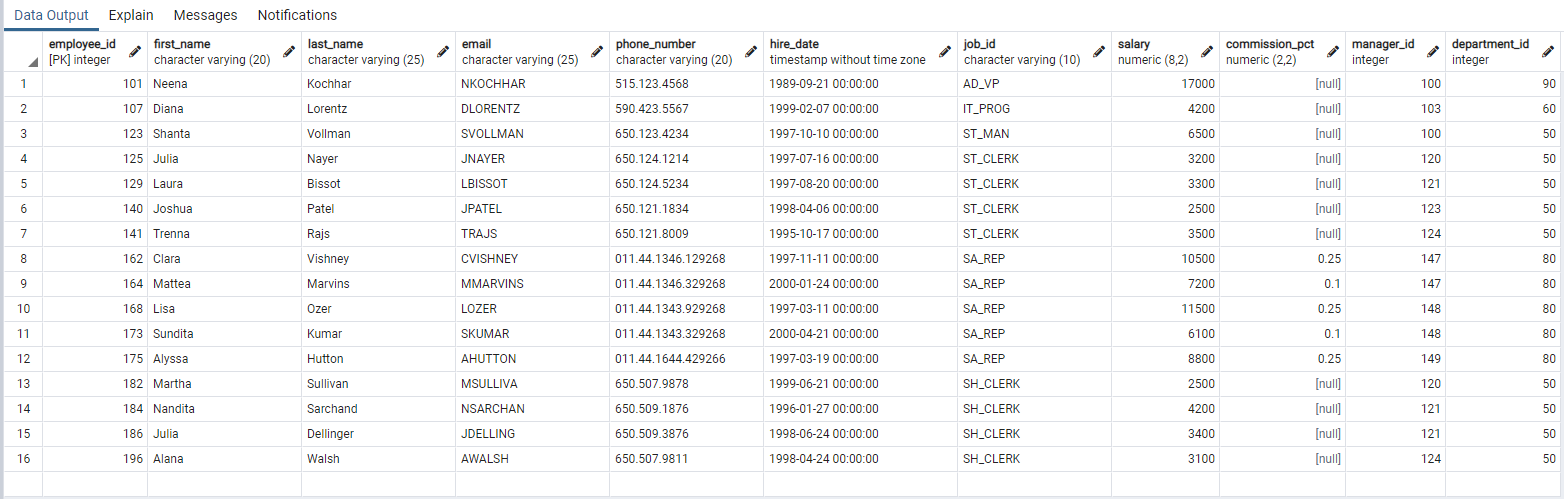
5. Таблица Employees. Получить список всех сотрудников из 20го и из 30го отдела (department\_id)

**SELECT** \* **FROM** employees **WHERE** department\_id = 20 **OR** department\_id = 30



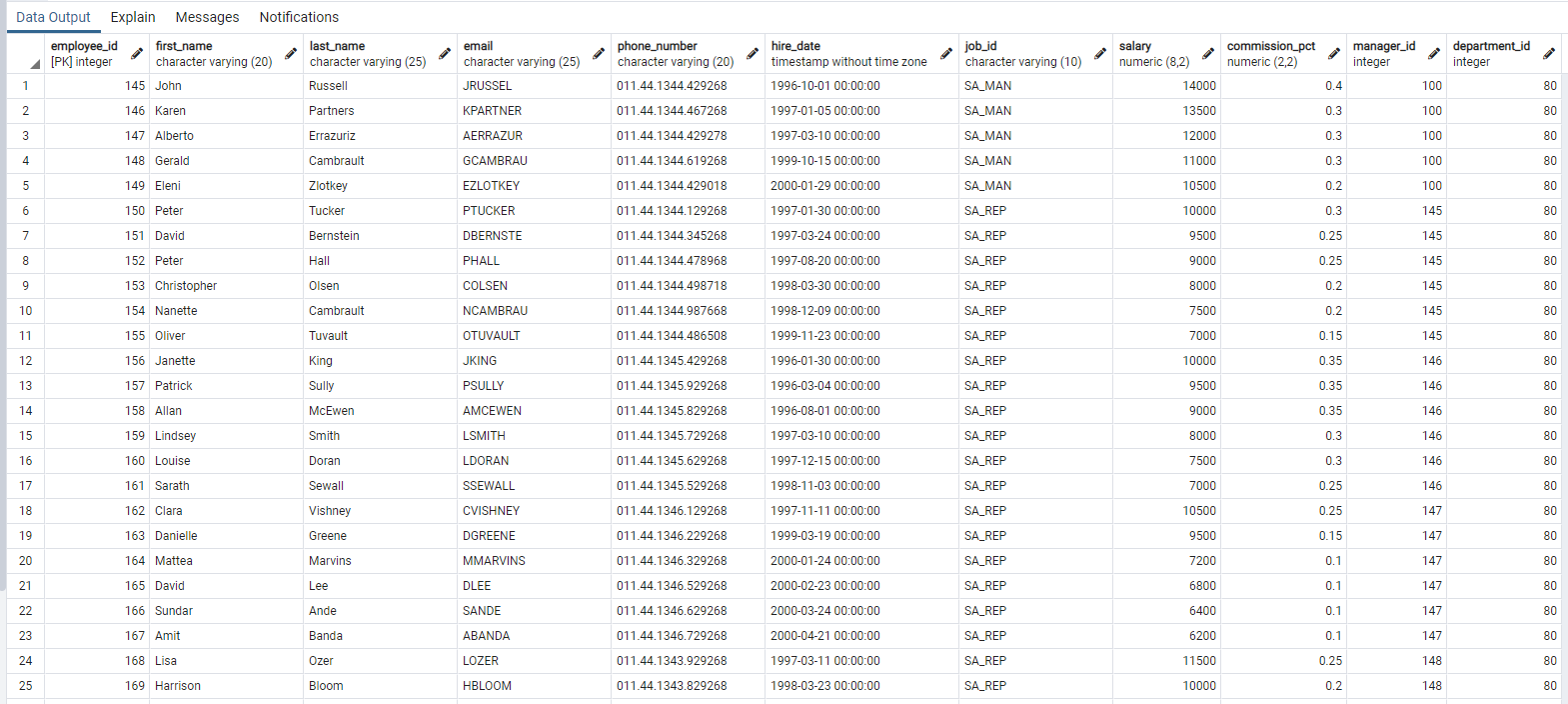
6. Таблица Employees. Получить список всех сотрудников у которых последняя буква в имени равна 'a'

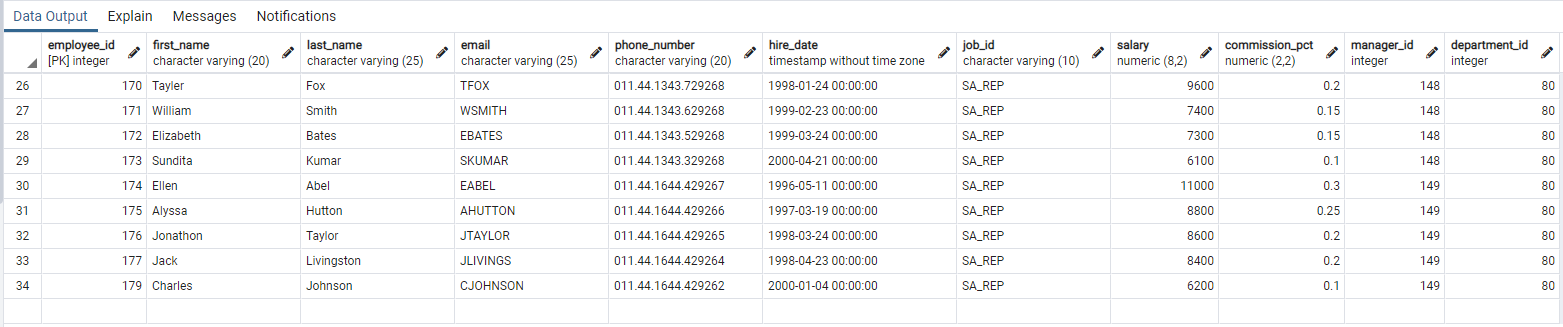
**SELECT** \* **FROM** employees **WHERE** first\_name **LIKE** '%a'



7. Таблица Employees. Получить список всех сотрудников из 50го и из 80го отдела (department\_id) у которых есть бонус (значение в колонке commission\_pct не пустое)

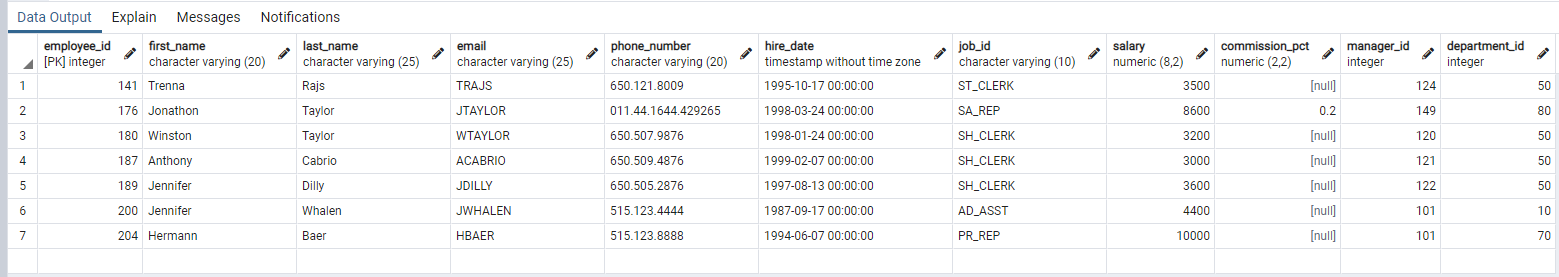
**SELECT** \* **FROM** employees **WHERE** (depatment\_id = 50 **OR** depatment\_id = 80) **AND** commission\_pct **IS NOT** NULL





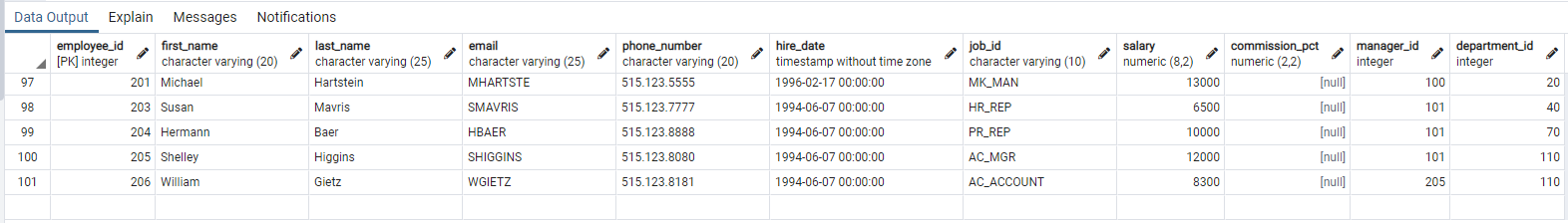
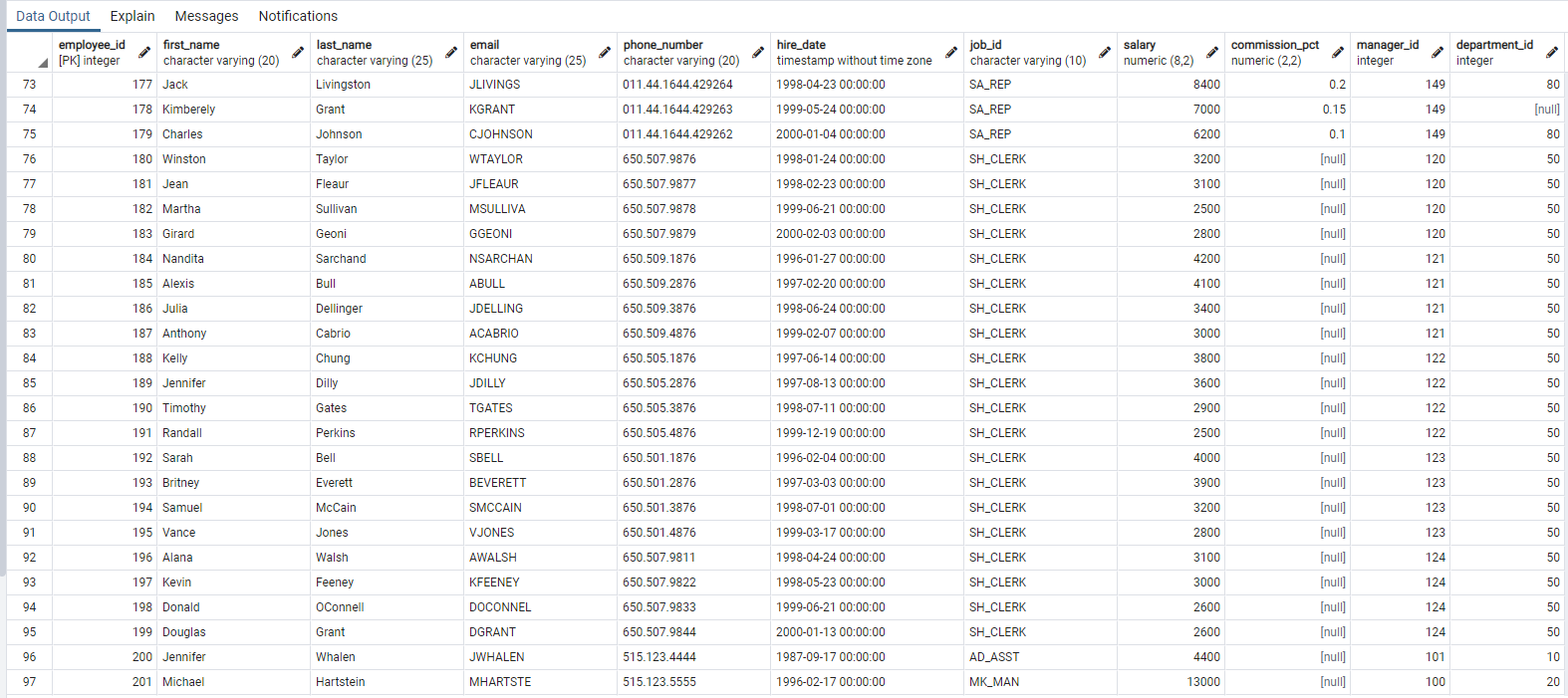
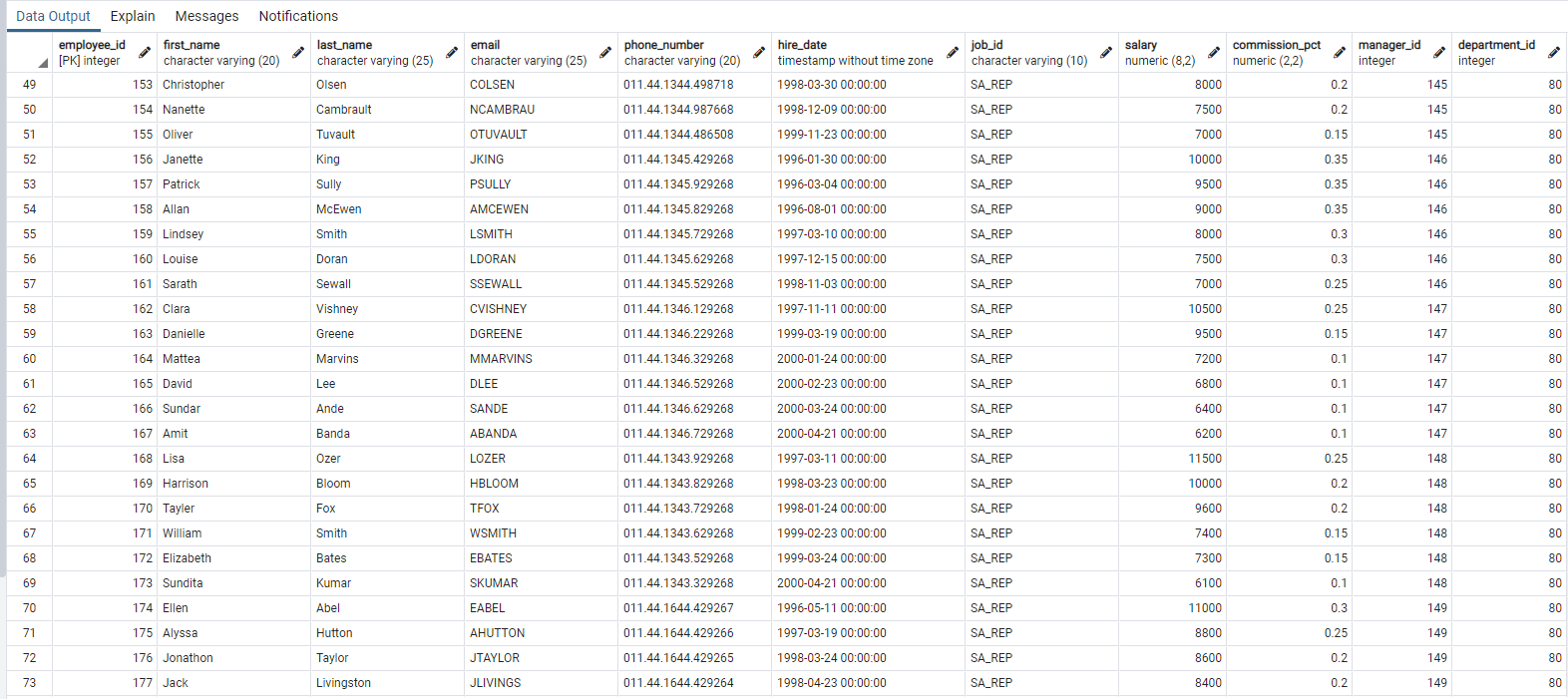
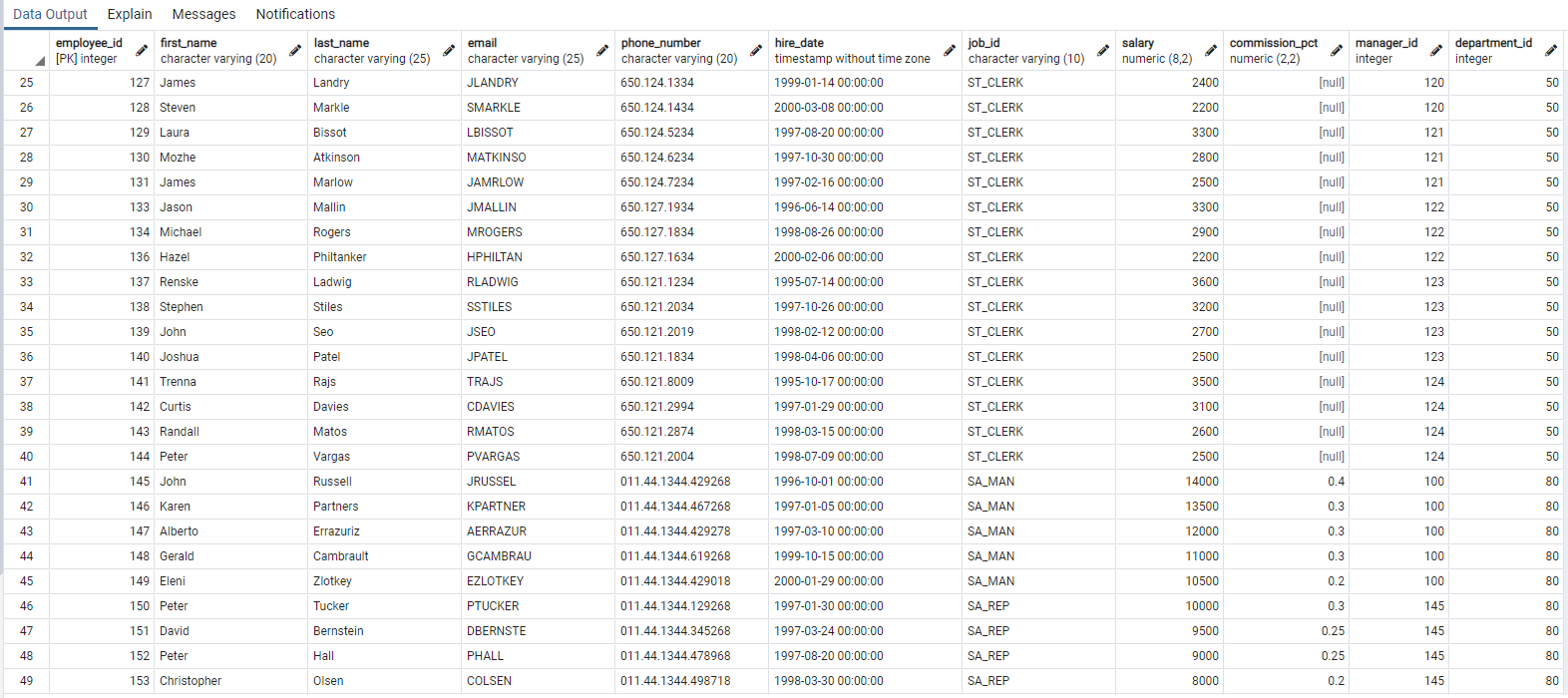
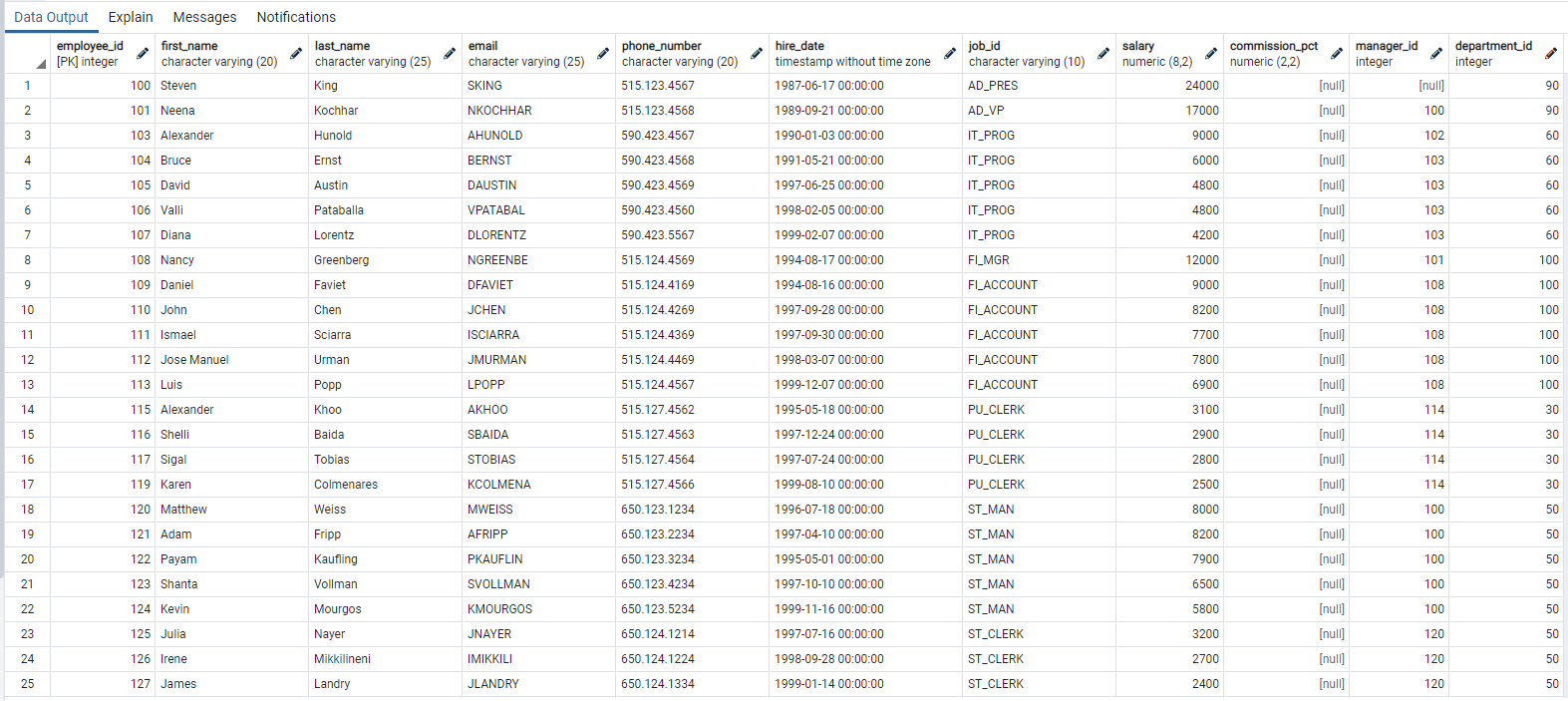
8. Таблица Employees. Получить список всех сотрудников у которых в имени содержатся минимум 2 буквы 'n'

**SELECT** \* **FROM** employees **WHERE** first\_name **LIKE** '%n%n%'



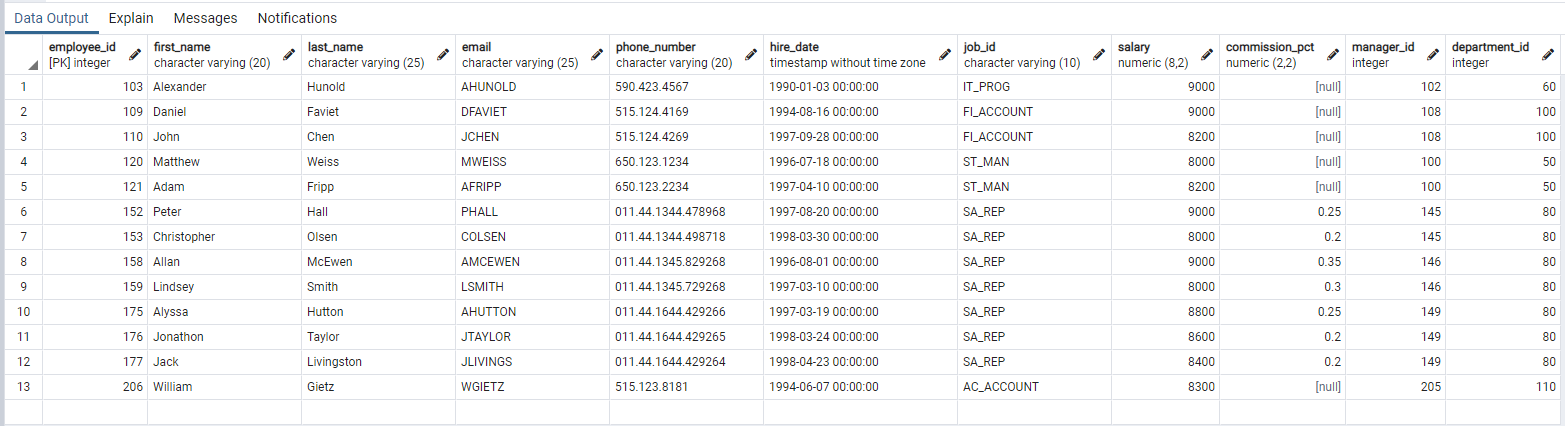
9. Таблица Employees. Получить список всех сотрудников у которых длина имени больше 4 букв

**SELECT** \* **FROM** employees **WHERE** first\_name **LIKE** '%\_\_\_\_%'



10. Таблица Employees. Получить список всех сотрудников у которых зарплата находится в промежутке от 8000 до 9000 (включительно)

**SELECT** \* **FROM** employees **WHERE** salary **BETWEEN** 8000 **AND** 9000



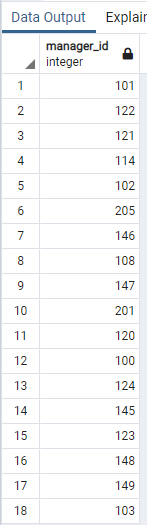
11. Таблица Employees. Получить список всех сотрудников у которых в имени содержится символ '%'

**SELECT** \* **FROM** employees **WHERE** first\_name **LIKE** '%\%%' **ESCAPE** '\'



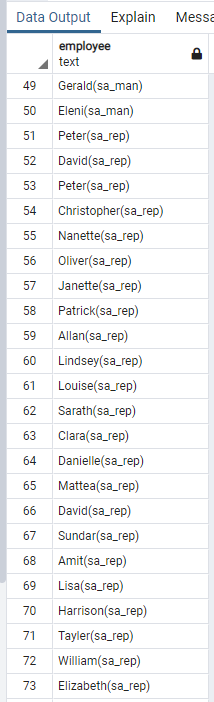
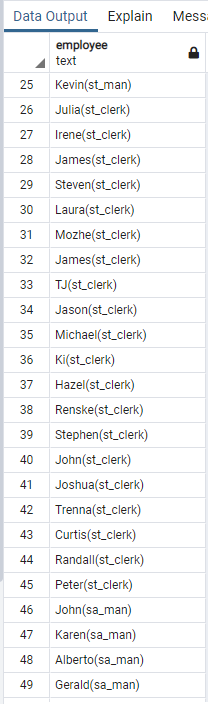
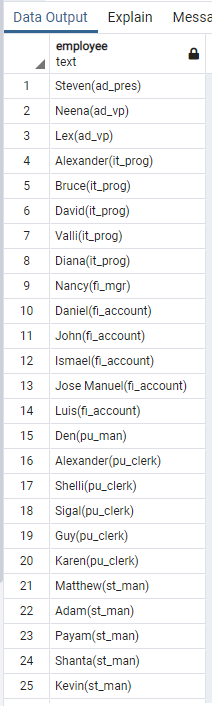
12. Таблица Employees. Получить список всех ID менеджеров

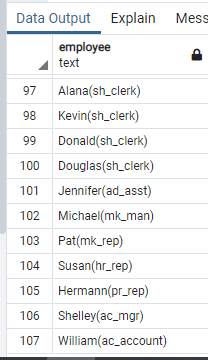
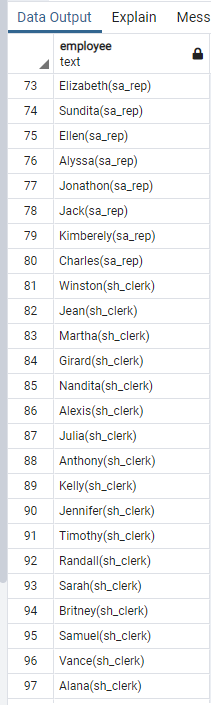
**SELECT DISTINCT** manager\_id **FROM** employees **WHERE** manager\_id **IS NOT** NULL



13. Таблица Employees. Получить список работников с их позициями в формате: Donald(sh\_clerk)

**SELECT** first\_name || '(' || **LOWER** (job\_id) || ')' employee **FROM** employees

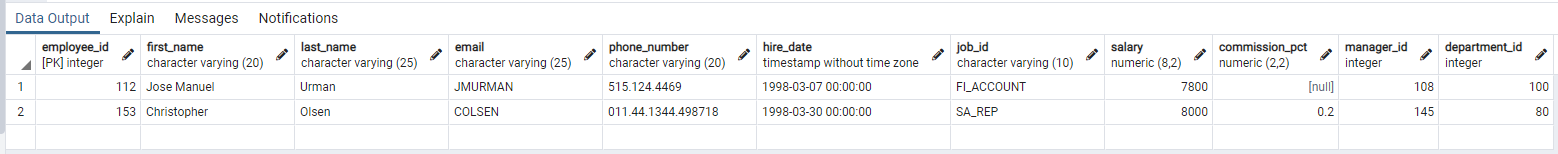




**Using Single-Row Functions to Customize Output**

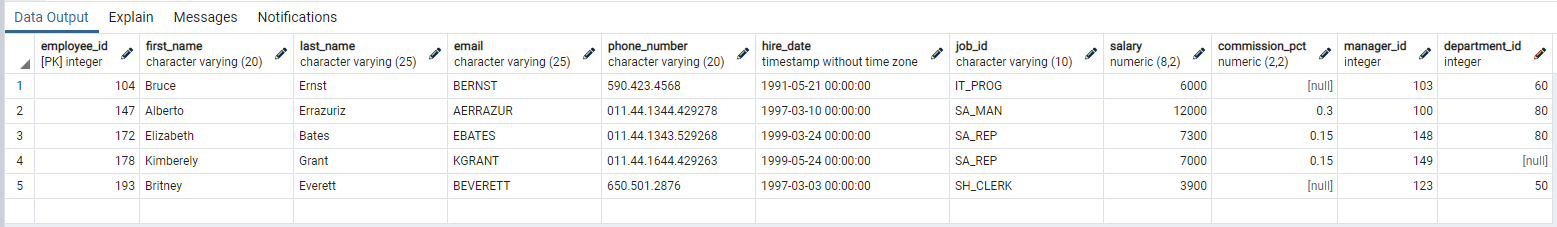
14. Таблица Employees. Получить список всех сотрудников у которых длина имени больше 10 букв

**SELECT** \* **FROM** employees **WHERE LENGTH** (first\_name) > 10



15. Таблица Employees. Получить список всех сотрудников у которых в имени есть буква 'b' (без учета регистра)

**SELECT** \* **FROM** employees **WHERE** STRPOS (**LOWER** (first\_name), 'b') > 0

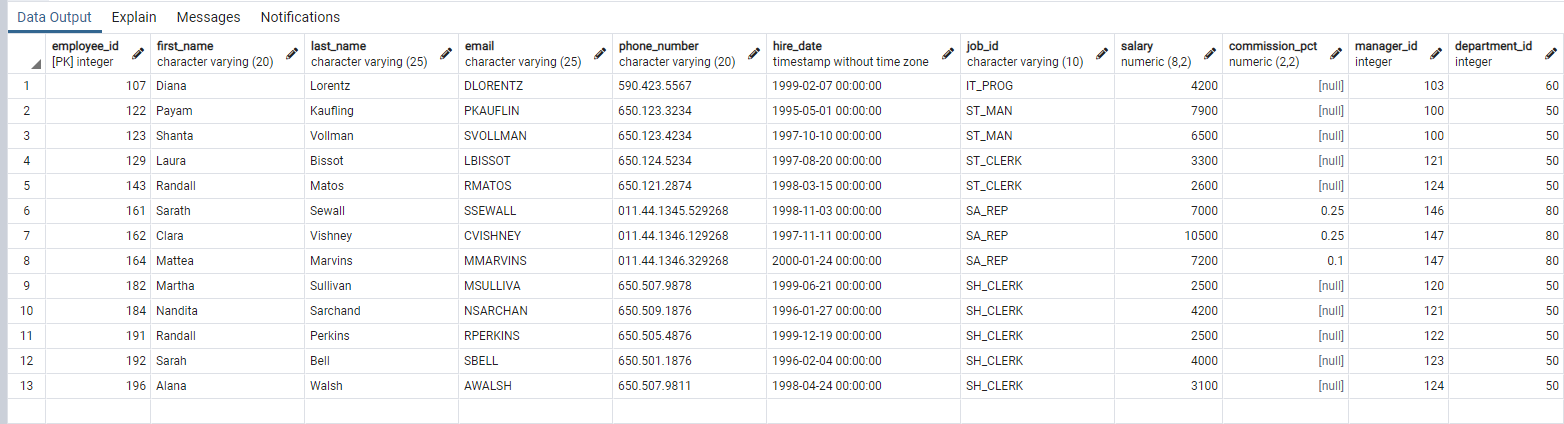


16. Таблица Employees. Получить список всех сотрудников у которых в имени содержатся минимум 2 буквы 'a'

SELECT \* FROM employees WHERE first\_name LIKE '%a%a%'

**SELECT** \* **FROM** employees **WHERE** first\_name **LIKE** '%а%а%'

**Я так это вижу(**



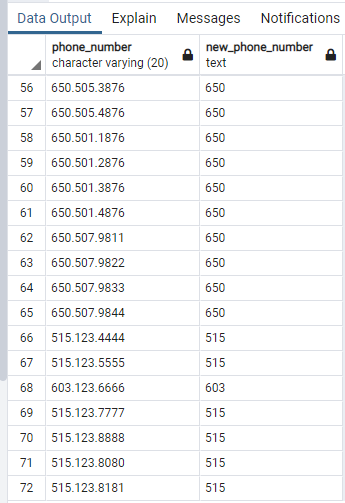
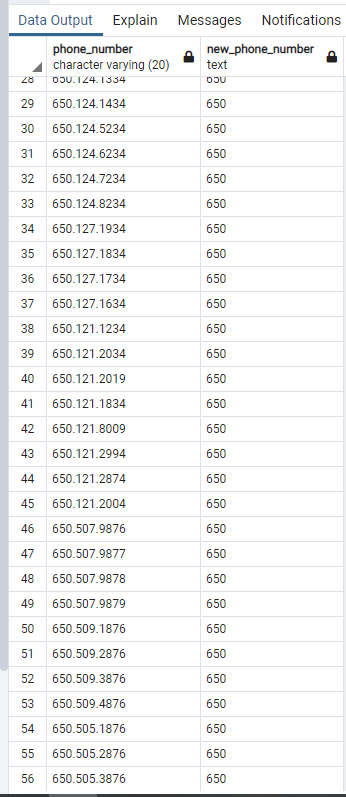
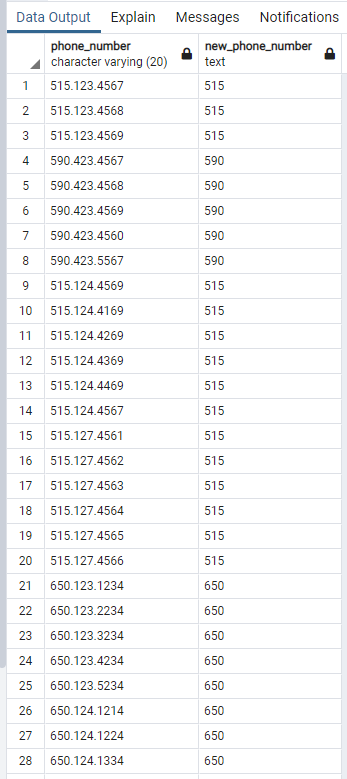
17. Таблица Employees. Получить список всех сотрудников зарплата которых кратна 1000

**SELECT** \* **FROM** employees **WHERE MOD** (salary, 1000) = 0



18. Таблица Employees. Получить первое 3х значное число телефонного номера сотрудника если его номер в формате ХХХ.ХХХ.ХХХХ

**SELECT** phone\_number, **SUBSTR** (phone\_number, 1, 3) new\_phone\_number **FROM** employees **WHERE** phone\_number **LIKE** '\_\_\_.\_\_\_.\_\_\_\_'



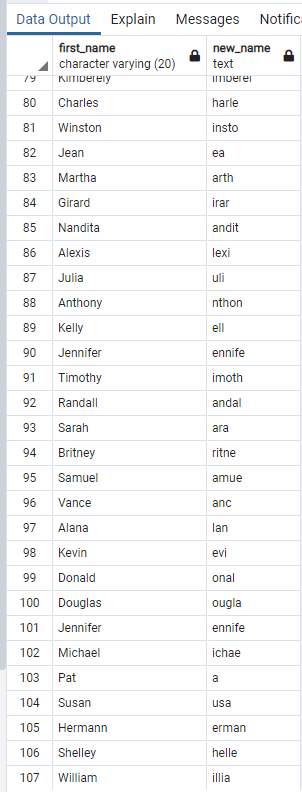
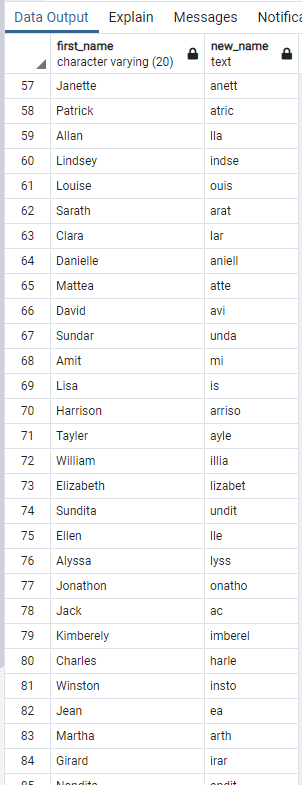
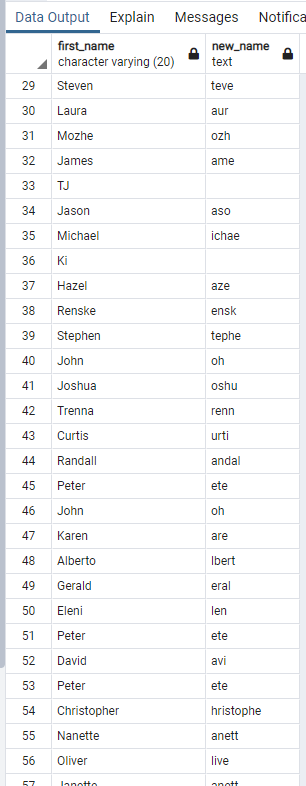
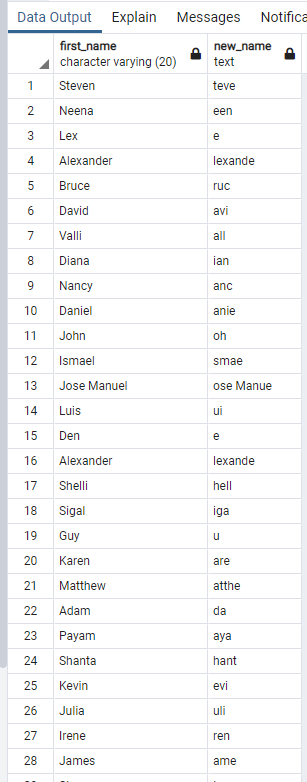
19. Таблица Departments. Получить первое слово из имени департамента для тех у кого в названии больше одного слова

**SELECT** department\_name, SUBSTR (department\_name, 1, STRPOS (department\_name, ' ')- 1) first\_word **FROM** departments **WHERE** STRPOS (department\_name, ' ') > 0



20. Таблица Employees. Получить имена сотрудников без первой и последней буквы в имени

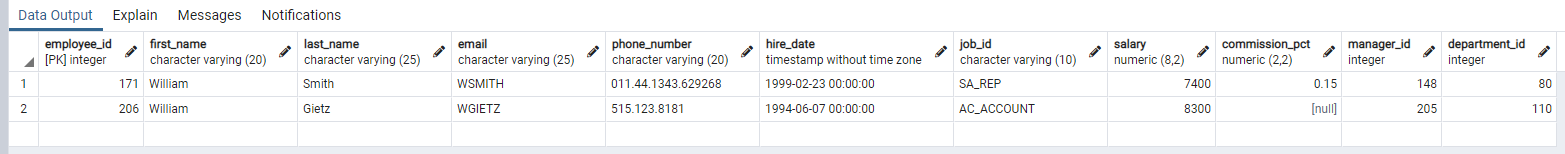
**SELECT** first\_name, SUBSTR (first\_name, 2, **LENGTH** (first\_name) – 2) new\_name **FROM** employees



21. Таблица Employees. Получить список всех сотрудников у которых последняя буква в имени равна 'm' и длинной имени большей 5ти

**SELECT** \* **FROM** employees **WHERE LENGTH**(first\_name) > 5 **AND** first\_name **LIKE** '%5'

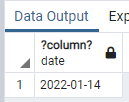
**Знаю, что не правильно, но у меня заработало только так:**



22. Таблица Dual. Получить дату следующей пятницы

**Не работает как в примере, заработало только так:**

**SELECT** 'TOMORROW'::date + (5 – **extract** ( dow **FROM** 'TOMORROW'::date))::**int%**7



23. Таблица Employees. Получить список всех сотрудников которые работают в компании больше 17 лет

**SELECT** \* **FROM** employees **WHERE** MONTHS\_BETWEEN (**SYSDATE**, hire\_date) / 12>17

**Не смог(**

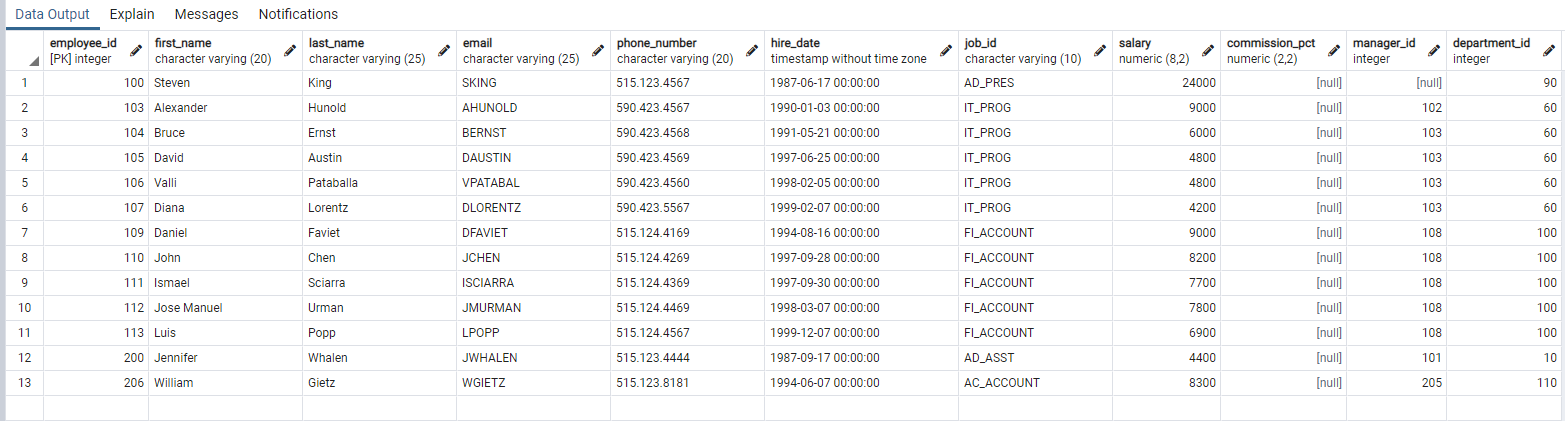
24. Таблица Employees. Получить список всех сотрудников у которых последня цифра телефонного номера нечетная и состоит из 3ех чисел разделенных точкой

**SELECT** \* **FROM** employees **WHERE MOD (SUBSTR** (phone\_number, -1), 2) != 0 **AND INSTR** (phone\_number, '.', 1,3) = 0 **AND INSTR** (phone\_number, '.', 1,2) > 0

**Не смог( Не понимаю что тут нужно вместо INSTR, но тут еше отвалился MOD**

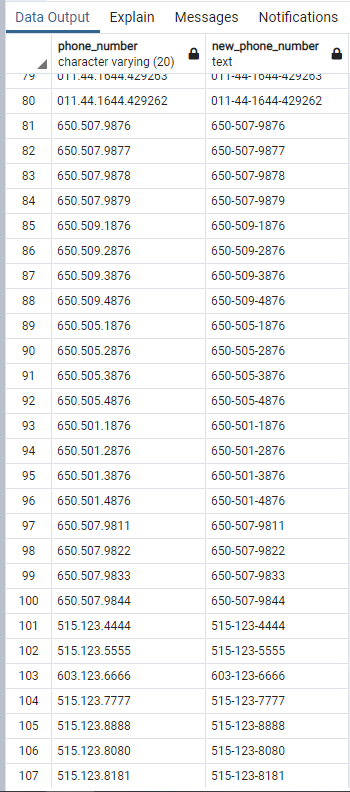
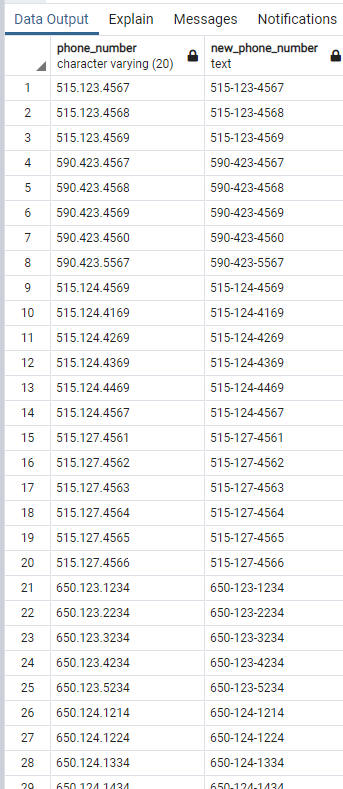
25. Таблица Employees. Получить список всех сотрудников у которых в значении job\_id после знака '\_' как минимум 3 символа но при этом это значение после '\_' не равно 'CLERK'

**SELECT** \* **FROM** employees **WHERE LENGTH (SUBSTR** (job\_id, STRPOS (job\_id, '\_') + 1)) > 3 **AND SUBSTR** (job\_id, STRPOS (job\_id, '\_') + 1) != 'CLERK'



26. Таблица Employees. Получить список всех сотрудников заменив в значении PHONE\_NUMBER все '.' на '-'

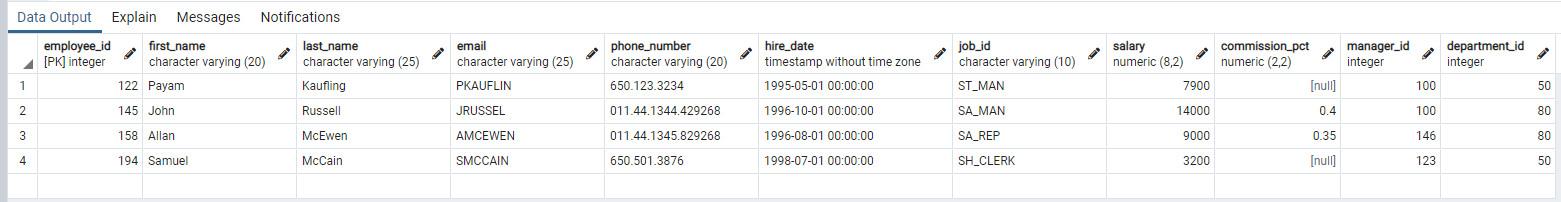
**SELECT** phone\_number, **REPLACE (**phone\_number, '.', '-') new\_phone\_number **FROM** employees



**Using Conversion Functions and Conditional Expressions**

27. Таблица Employees. Получить список всех сотрудников которые пришли на работу в первый день месяца (любого)

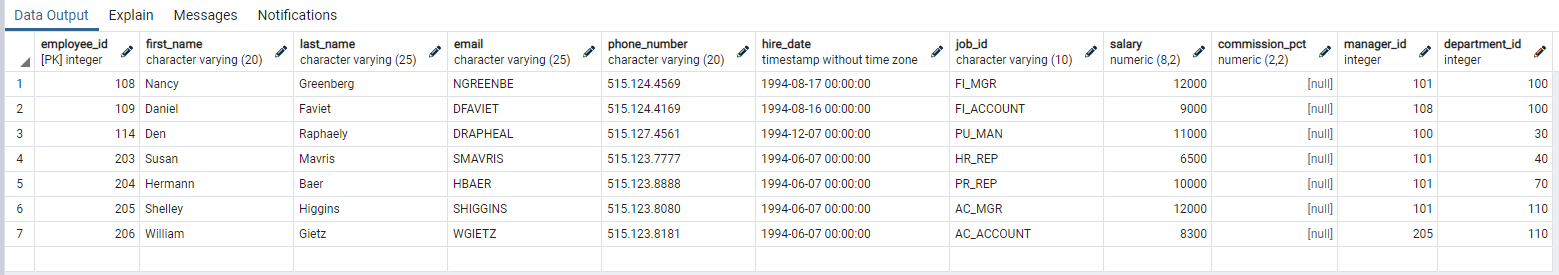
**SELECT** \* **FROM** employees **WHERE** TO\_CHAR (hire\_date, 'DD') = '01'



28. Таблица Employees. Получить список всех сотрудников которые пришли на работу в 2008ом году

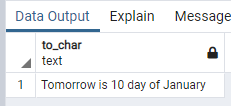
**SELECT** \* **FROM** employees **WHERE** TO\_CHAR (hire\_date, 'YYYY') = '1994'

В 2008-м никого, ради интереса изменил на 1994й



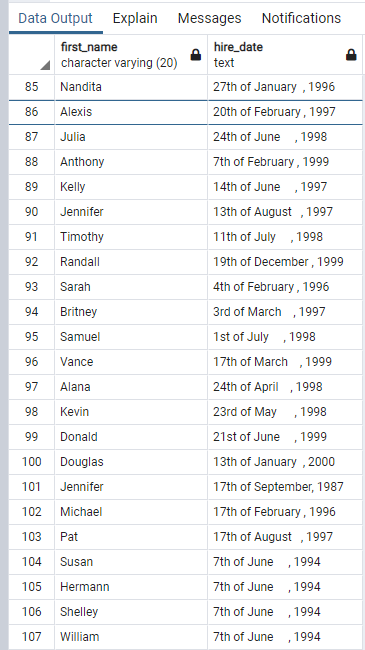
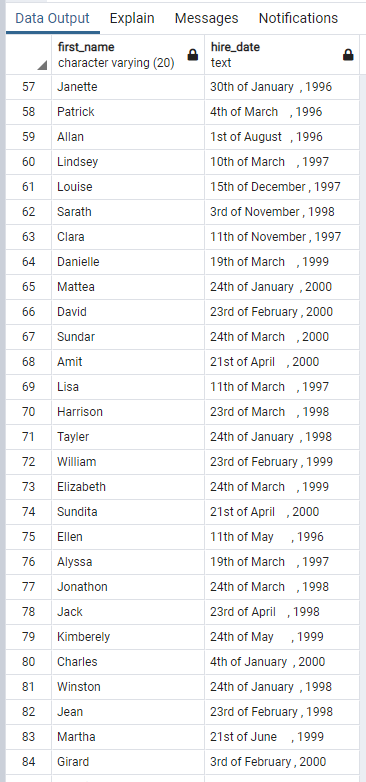
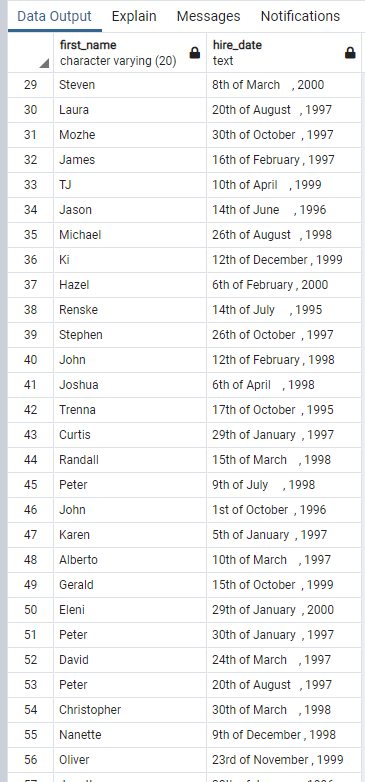
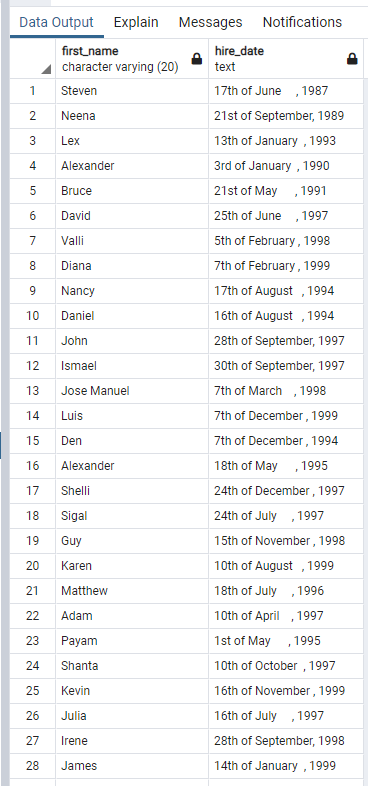
29. Таблица DUAL. Показать завтрашнюю дату в формате: Tomorrow is Second day of January

**SELECT** TO\_CHAR ('TOMORROW'::date, '"Tomorrow" "is" FMDD "day" of Month')



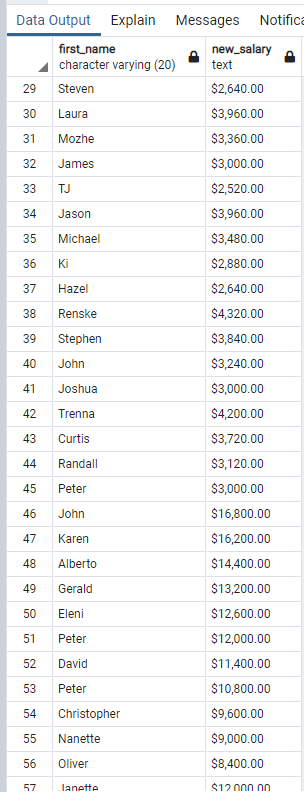
30. Таблица Employees. Получить список всех сотрудников и дату прихода на работу каждого в формате: 21st of June, 2007

**SELECT** first\_name, TO\_CHAR (hire\_date, 'fmddth ""of"" Month, YYYY') hire\_date **FROM** employees



31. Таблица Employees. Получить список работников с увеличенными зарплатами на 20%. Зарплату показать со знаком доллара

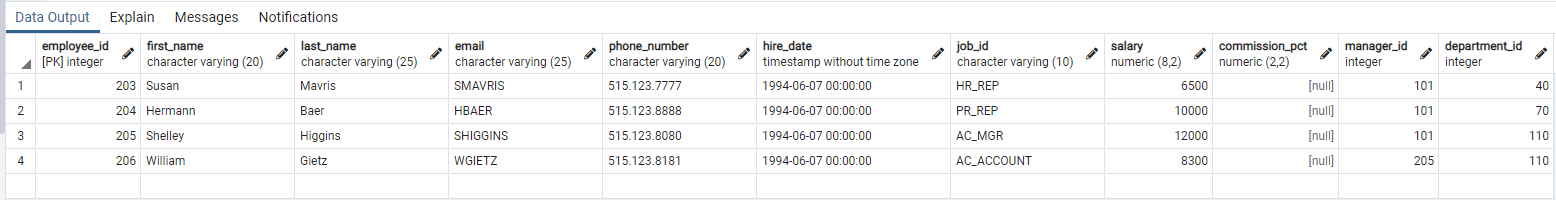
**SELECT** first\_name, TO\_CHAR (salary + salary \* 0.20, 'fm$999,999.00') new\_salary **FROM** employees



32. Таблица Employees. Получить список всех сотрудников которые пришли на работу в феврале 2007го года.

**SELECT** \* **FROM** employees **WHERE EXTRACT** (**MONTH FROM** hire\_date) = 6 **AND EXTRACT** (**YEAR FROM** hire\_date) = 1994

**Нет записей для февраля 2007. Для примера взял 06.1994**

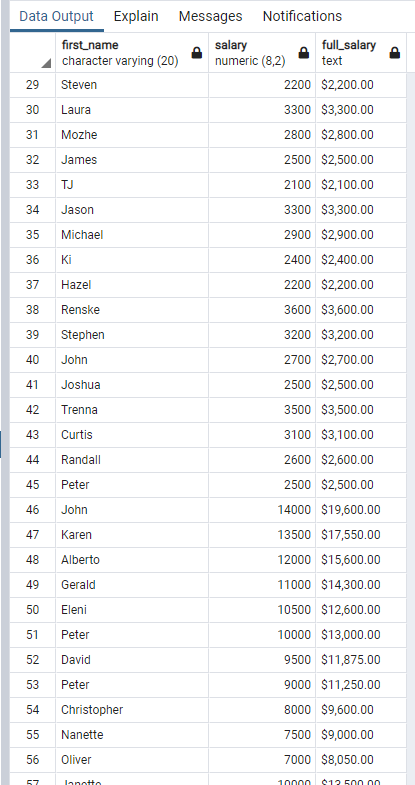


33. Таблица DUAL. Вывезти актуальную дату, + секунда, + минута, + час, + день, + месяц, + год

**Не смог(**

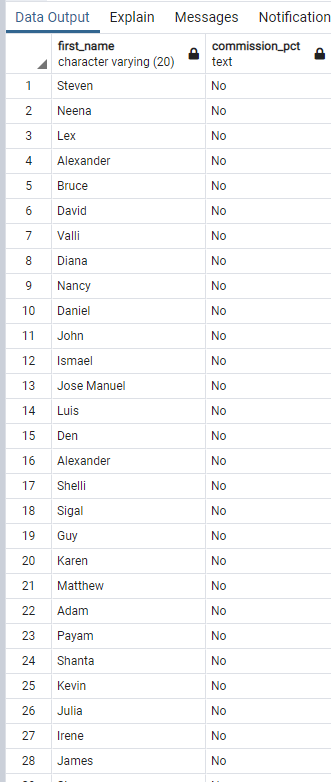
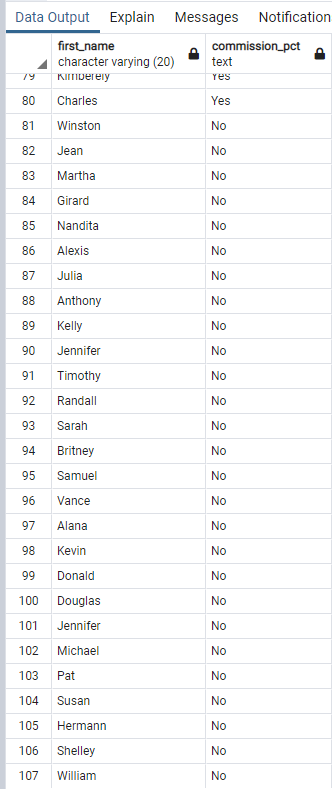
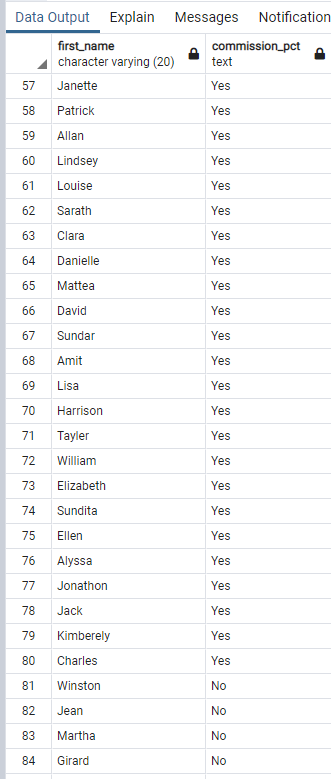
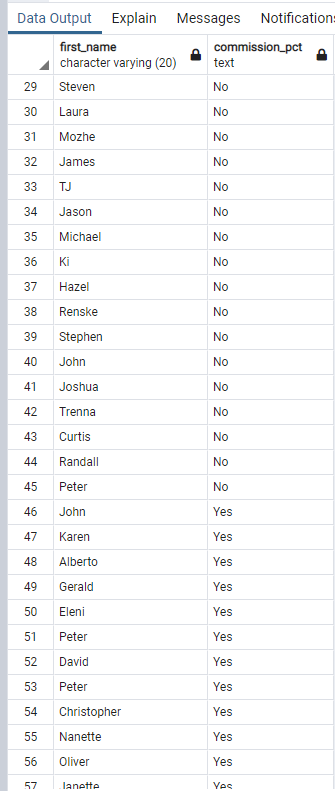
34. Таблица Employees. Получить список всех сотрудников с полными зарплатами (salary + commission\_pct(%)) в формате: $24,000.00

**SELECT** first\_name, salary, TO\_CHAR (salary + salary \* **COALESCE** (commission\_pct, 0), 'fm$99,999.00') full\_salary **FROM** employees



35. Таблица Employees. Получить список всех сотрудников и информацию о наличии бонусов к зарплате (Yes/No)

**SELECT** first\_name, **CASE WHEN** commission\_pct **IS** null **THEN** 'No' **ELSE** 'Yes' **END** commission\_pct **FROM** employees

****

36. Таблица Employees. Получить уровень зарплаты каждого сотрудника: Меньше 5000 считается Low level, Больше или равно 5000 и меньше 10000 считается Normal level, Больше или равно 10000 считается High level

**SELECT** first\_name,

salary,

**CASE**

**WHEN** salary < 5000 **THEN** 'Low'

**WHEN** salary >= 5000 **AND** salary < 10000 **THEN** 'Normal'

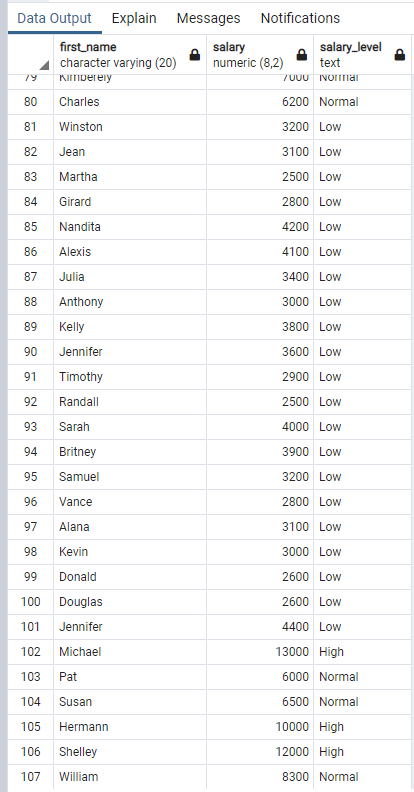
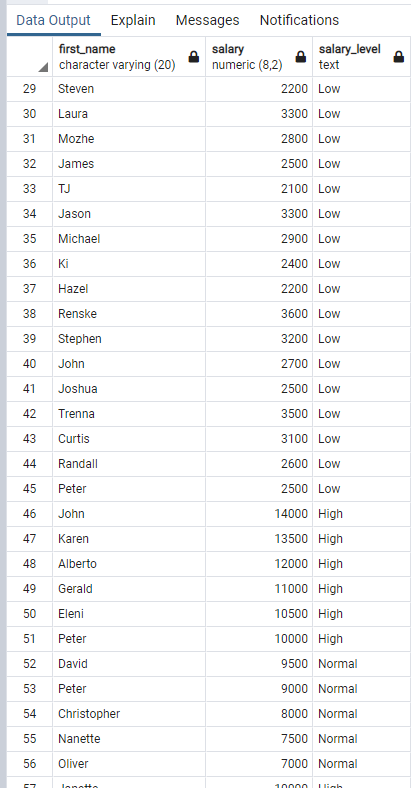
**WHEN** salary >= 10000 **THEN** 'High'

**ELSE** 'Unknown'

**END**

salary\_level

**FROM** employees



37. Таблица Countries. Для каждой страны показать регион в котором он находится: 1-Europe, 2-America, 3-Asia, 4-Africa (без Join)

**SELECT** country\_name

country,

**CASE** region\_id

**WHEN** 1 **THEN** 'Europe'

**WHEN** 2 **THEN** 'America'

**WHEN** 3 **THEN** 'Asia'

**WHEN** 4 **THEN** 'Africa'

**ELSE** 'Unknown'

**END**

region

**FROM** countries



**Reporting Aggregated Data Using the Group Functions**

38. Таблица Employees. Получить репорт по department\_id с минимальной и максимальной зарплатой, с ранней и поздней датой прихода на работу и с количествов сотрудников. Сорировать по количеству сотрудников (по убыванию)

**SELECT** department\_id,

**MIN** (salary) min\_salary,

**MAX** (salary) max\_salary,

**MIN** (hire\_date) min\_hire\_date,

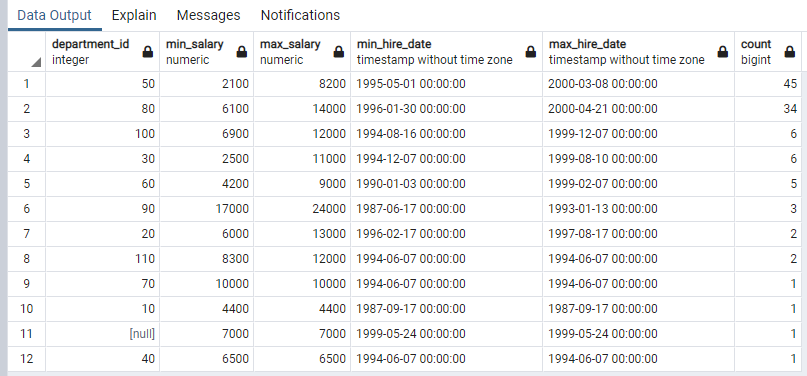
**MAX** (hire\_date) max\_hire\_Date,

**COUNT** (\*) **count**

**FROM** employees

**GROUP** **BY** department\_id

**order** **by** **count**(\*) **desc**



39. Таблица Employees. Сколько сотрудников имена которых начинается с одной и той же буквы? Сортировать по количеству. Показывать только те где количество больше 1

**SELECT** **SUBSTR** (first\_name, 1, 1) first\_char, **COUNT** (\*)

**FROM** employees

**GROUP** **BY** **SUBSTR** (first\_name, 1, 1)

**HAVING** **COUNT** (\*) > 1

**ORDER** **BY** 2 **DESC**



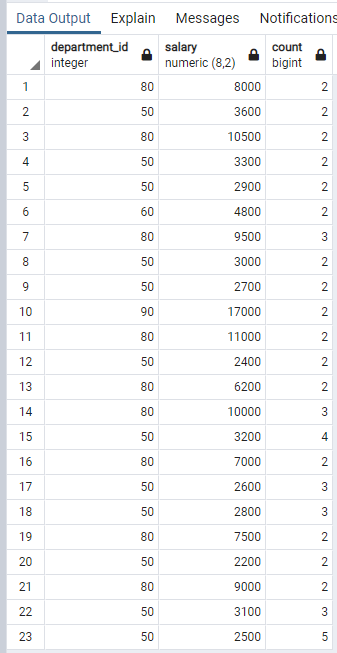
40. Таблица Employees. Сколько сотрудников которые работают в одном и тоже отделе и получают одинаковую зарплату?

**SELECT** department\_id, salary, **COUNT** (\*)

**FROM** employees

**GROUP** **BY** department\_id, salary

**HAVING** **COUNT** (\*) > 1



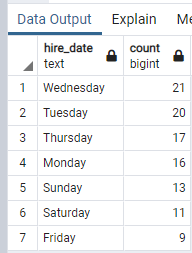
41. Таблица Employees. Получить репорт сколько сотрудников приняли на работу в каждый день недели. Сортировать по количеству

**SELECT** TO\_CHAR (hire\_Date, 'Day') hire\_Date, **COUNT** (\*)

**FROM** employees

**GROUP** **BY** TO\_CHAR (hire\_Date, 'Day')

**ORDER** **BY** 2 **DESC**



42. Таблица Employees. Получить репорт сколько сотрудников приняли на работу по годам. Сортировать по количеству

**SELECT** TO\_CHAR (hire\_date, 'YYYY') hire\_date, **COUNT** (\*)

**FROM** employees

**GROUP** **BY** TO\_CHAR (hire\_date, 'YYYY')



43. Таблица Employees. Получить количество департаментов в котором есть сотрудники

**Не смог(**

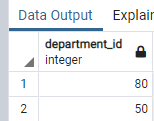
44. Таблица Employees. Получить список department\_id в котором работают больше 30 сотрудников

**SELECT** department\_id

**FROM** employees

**GROUP** **BY** department\_id

**HAVING** **COUNT** (\*) > 30

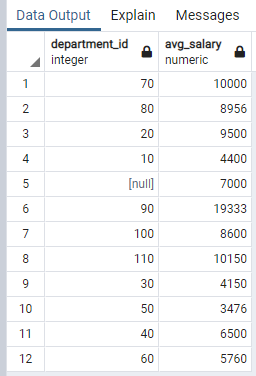


45. Таблица Employees. Получить список department\_id и округленную среднюю зарплату работников в каждом департаменте.

**SELECT** department\_id, **ROUND** (**AVG** (salary)) avg\_salary

**FROM** employees

**GROUP** **BY** department\_id



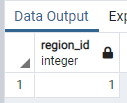
46. Таблица Countries. Получить список region\_id сумма всех букв всех country\_name в котором больше 60ти

**SELECT** region\_id

**FROM** countries

**GROUP** **BY** region\_id

**HAVING** **SUM** (**LENGTH** (country\_name)) > 60



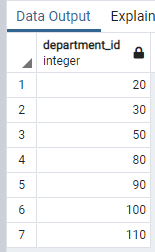
47. Таблица Employees. Получить список department\_id в котором работают работники нескольких (>1) job\_id

**SELECT** department\_id

**FROM** employees

**GROUP** **BY** department\_id

**HAVING** **COUNT** (**DISTINCT** job\_id) > 1



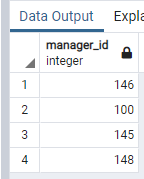
48. Таблица Employees. Получить список manager\_id у которых количество подчиненных больше 5 и сумма всех зарплат его подчиненных больше 50000

**SELECT** manager\_id

**FROM** employees

**GROUP** **BY** manager\_id

**HAVING** **COUNT** (\*) > 5 **AND** **SUM** (salary) > 50000



49. Таблица Employees. Получить список manager\_id у которых средняя зарплата всех его подчиненных находится в промежутке от 6000 до 9000 которые не получают бонусы (commission\_pct пустой)

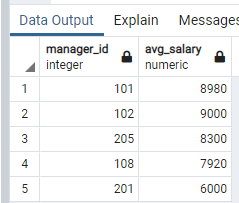
**SELECT** manager\_id, **AVG** (salary) avg\_salary

**FROM** employees

**WHERE** commission\_pct **IS** NULL

**GROUP** **BY** manager\_id

**HAVING** **AVG** (salary) **BETWEEN** 6000 **AND** 9000

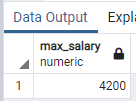


50. Таблица Employees. Получить максимальную зарплату из всех сотрудников job\_id которых заканчивается на слово 'CLERK'

**SELECT** **MAX** (salary) max\_salary

**FROM** employees

**WHERE** job\_id **LIKE** '%CLERK'



51. Таблица Employees. Получить максимальную зарплату среди всех средних зарплат по департаменту

**Не смог(**

52. Таблица Employees. Получить количество сотрудников с одинаковым количеством букв в имени. При этом показать только тех у кого длина имени больше 5 и количество сотрудников с таким именем больше 20. Сортировать по длинне имени

**SELECT** **LENGTH** (first\_name), **COUNT** (\*)

**FROM** employees

**GROUP** **BY** **LENGTH** (first\_name)

**HAVING** **LENGTH** (first\_name) > 5 **AND** **COUNT** (\*) > 20

**ORDER** **BY** **LENGTH** (first\_name);

**SELECT** **LENGTH** (first\_name), **COUNT** (\*)

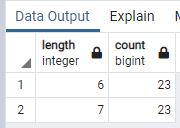
**FROM** employees

**WHERE** **LENGTH** (first\_name) > 5

**GROUP** **BY** **LENGTH** (first\_name)

**HAVING** **COUNT** (\*) > 20

**ORDER** **BY** **LENGTH** (first\_name)



**Displaying Data from Multiple Tables Using Joins**

53. Таблица Employees, Departaments, Locations, Countries, Regions. Получить список регионов и количество сотрудников в каждом регионе

**SELECT** region\_name, **COUNT** (\*)

**FROM** employees e

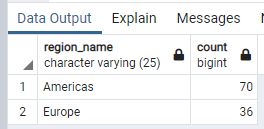
**JOIN** departments d **ON** (e.department\_id = d.department\_id)

**JOIN** locations l **ON** (d.location\_id = l.location\_id)

**JOIN** countries c **ON** (l.country\_id = c.country\_id)

**JOIN** regions r **ON** (c.region\_id = r.region\_id)

**GROUP** **BY** region\_name;



54. Таблица Employees, Departaments, Locations, Countries, Regions. Получить детальную информацию о каждом сотруднике:

First\_name, Last\_name, Departament, Job, Street, Country, Region

**SELECT** First\_name,

Last\_name,

Department\_name,

Job\_id,

street\_address,

Country\_name,

Region\_name

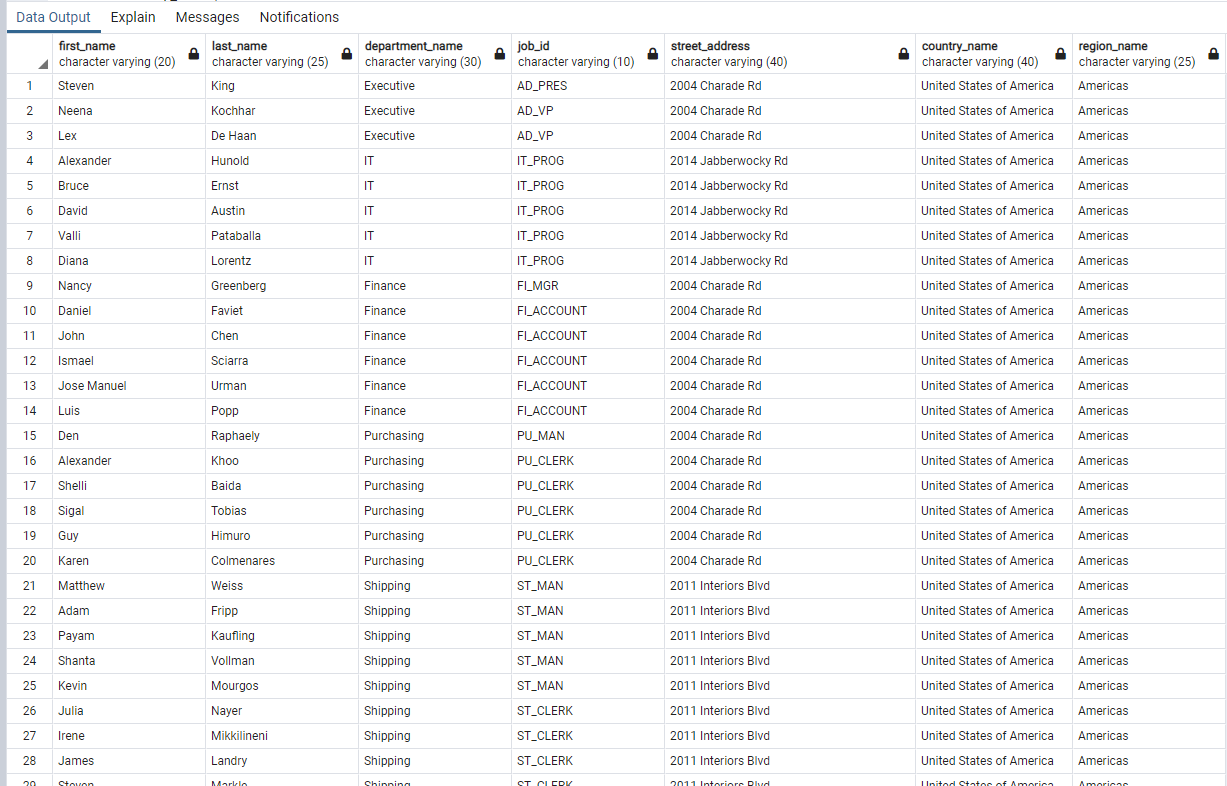
**FROM** employees e

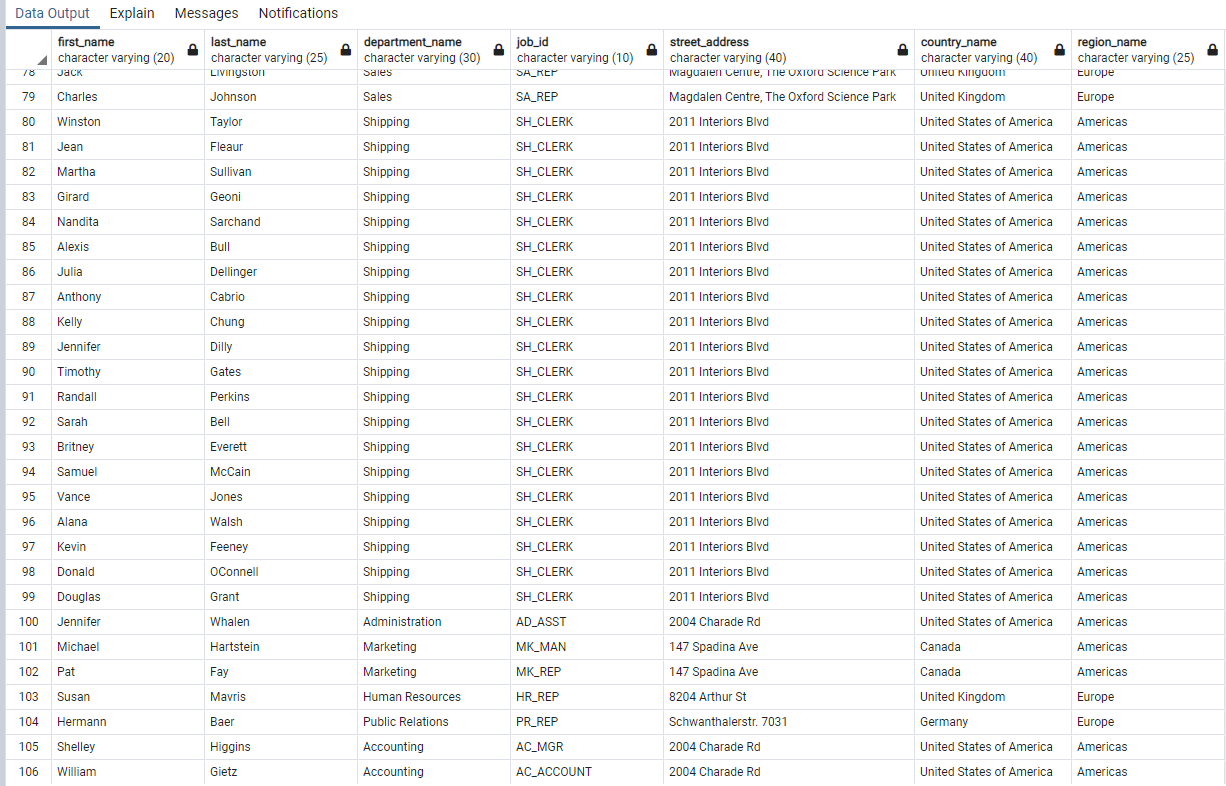
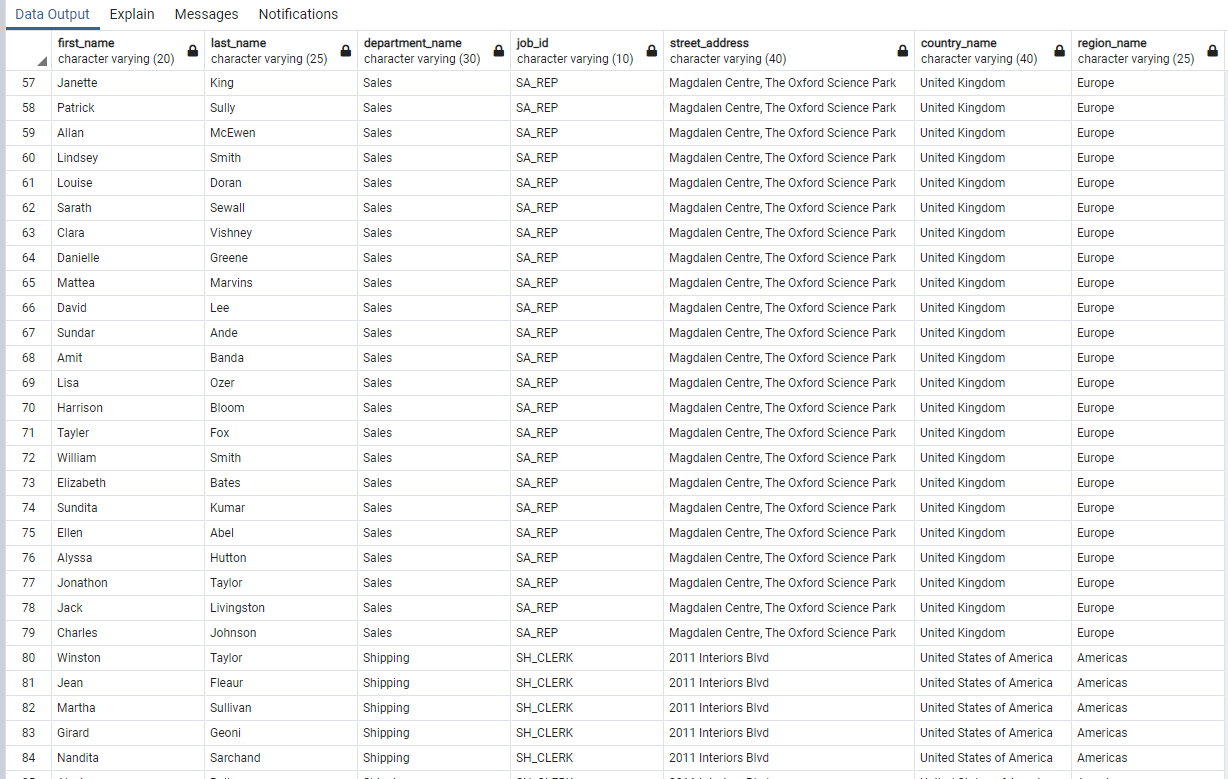
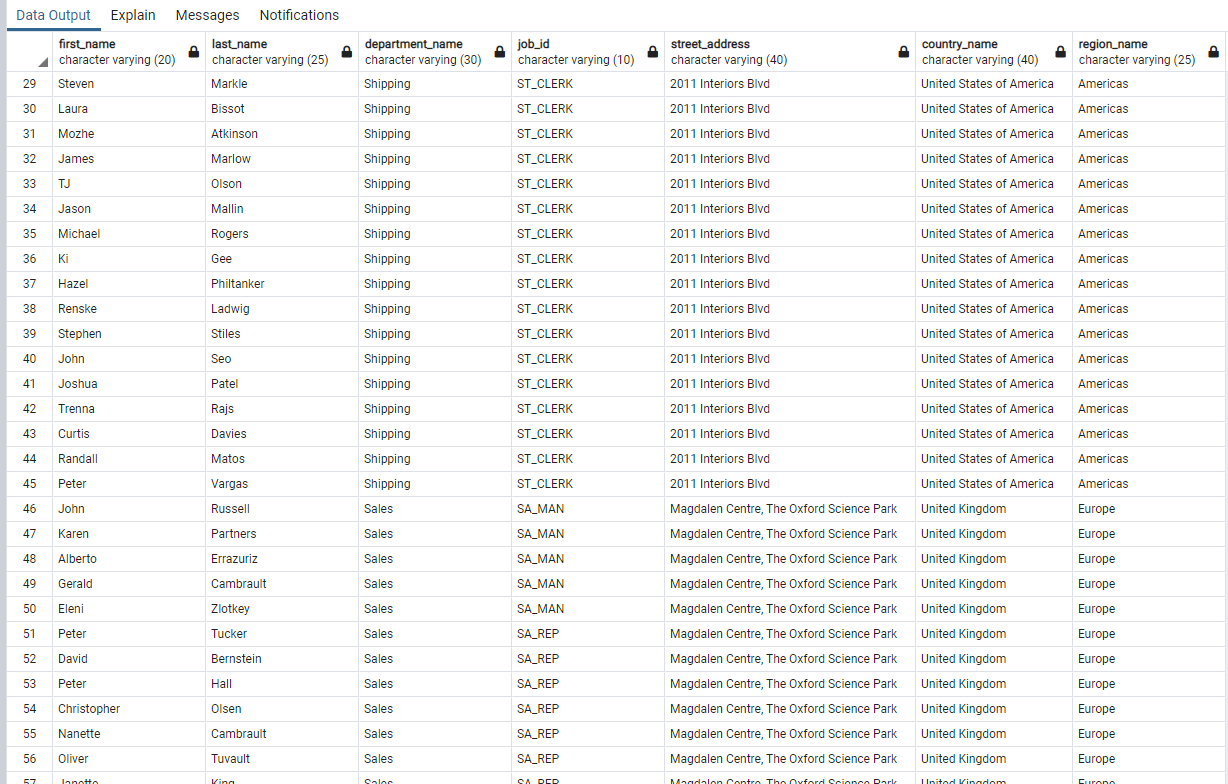
**JOIN** departments d **ON** (e.department\_id = d.department\_id)

**JOIN** locations l **ON** (d.location\_id = l.location\_id)

**JOIN** countries c **ON** (l.country\_id = c.country\_id)

**JOIN** regions r **ON** (c.region\_id = r.region\_id)





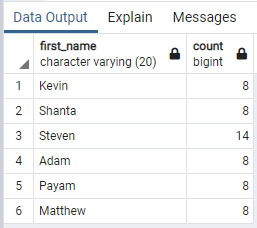
55. Таблица Employees. Показать всех менеджеров которые имеют в подчинении больше 6ти сотрудников

**SELECT** man.first\_name, **COUNT** (\*)

**FROM** employees emp **JOIN** employees man **ON** (emp.manager\_id = man.employee\_id)

**GROUP** **BY** man.first\_name

**HAVING** **COUNT** (\*) > 6



56. Таблица Employees. Показать всех сотрудников которые ни кому не подчиняются

**SELECT** emp.first\_name

**FROM** employees emp

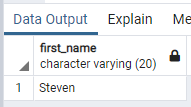
**LEFT** **JOIN** employees man **ON** (emp.manager\_id = man.employee\_id)

**WHERE** man.FIRST\_NAME **IS** NULL;

**SELECT** first\_name

**FROM** employees

**WHERE** manager\_id **IS** NULL



57. Таблица Employees, Job\_history. В таблице Employee хранятся все сотрудники. В таблице Job\_history хранятся сотрудники которые покинули компанию. Получить репорт о всех сотрудниках и о его статусе в компании (Работает или покинул компанию с датой ухода)

Пример:

first\_name | status

Jennifer | Left the company at 31 of December, 2006

Clara | Currently Working

**Не смог(**

58. Таблица Employees, Departaments, Locations, Countries, Regions. Получить список сотрудников которые живут в Europe (region\_name)

**SELECT** first\_name

**FROM** employees

**JOIN** departments **USING** (department\_id)

**JOIN** locations **USING** (location\_id)

**JOIN** countries **USING** (country\_id)

**JOIN** regions **USING** (region\_id)

**WHERE** region\_name = 'Europe';

**SELECT** first\_name

**FROM** employees e

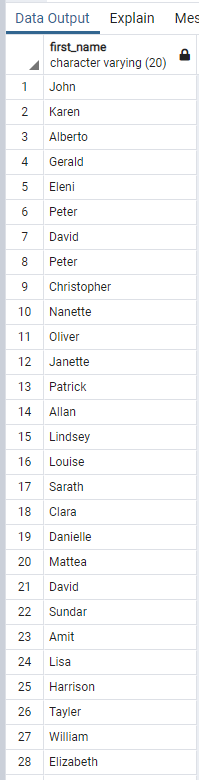
**JOIN** departments d **ON** (e.department\_id = d.department\_id)

**JOIN** locations l **ON** (d.location\_id = l.location\_id)

**JOIN** countries c **ON** (l.country\_id = c.country\_id)

**JOIN** regions r **ON** (c.region\_id = r.region\_id)

**WHERE** region\_name = 'Europe'



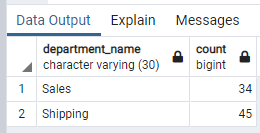
59. Таблица Employees, Departaments. Показать все департаменты в которых работают больше 30ти сотрудников

**SELECT** department\_name, **COUNT** (\*)

**FROM** employees e **JOIN** departments d **ON** (e.department\_id = d.department\_id)

**GROUP** **BY** department\_name

**HAVING** **COUNT** (\*) > 30



60. Таблица Employees, Departaments. Показать всех сотрудников которые не состоят ни в одном департаменте

**SELECT** first\_name

**FROM** employees e

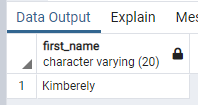
**LEFT** **JOIN** departments d **ON** (e.department\_id = d.department\_id)

**WHERE** d.department\_name **IS** NULL;

**SELECT** first\_name

**FROM** employees

**WHERE** department\_id **IS** NULL



61. Таблица Employees, Departaments. Показать все департаменты в которых нет ни одного сотрудника

**SELECT** department\_name

**FROM** employees e

**RIGHT** **JOIN** departments d **ON** (e.department\_id = d.department\_id)

**WHERE** first\_name **IS** NULL

**SELECT** department\_name

**FROM** employees e

**RIGHT** **JOIN** departments d **ON** (e.department\_id = d.department\_id)

**WHERE** first\_name **IS** NULL



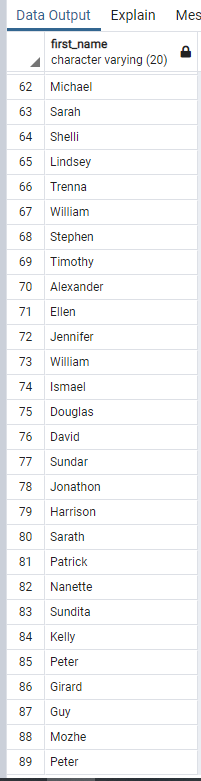
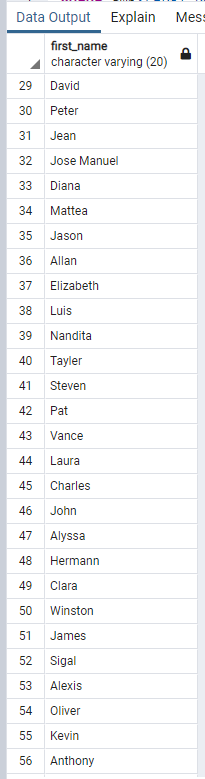
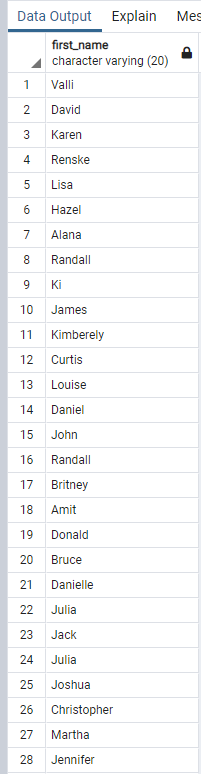
62. Таблица Employees. Показать всех сотрудников у которых нет ни кого в подчинении

**SELECT** man.first\_name

**FROM** employees emp

**RIGHT** **JOIN** employees man **ON** (emp.manager\_id = man.employee\_id)

**WHERE** emp.FIRST\_NAME **IS** NULL



63. Таблица Employees, Jobs, Departaments. Показать сотрудников в формате: First\_name, Job\_title, Department\_name.

Пример:

First\_name | Job\_title | Department\_name

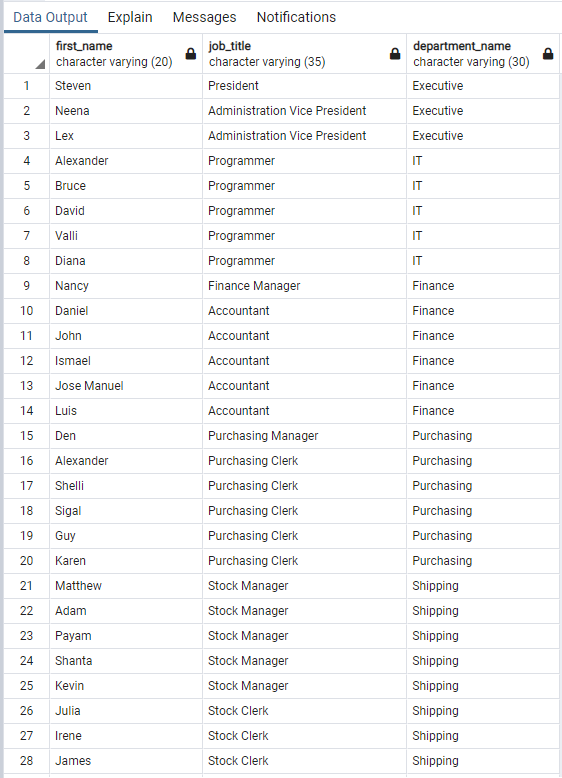
Donald | Shipping | Clerk Shipping

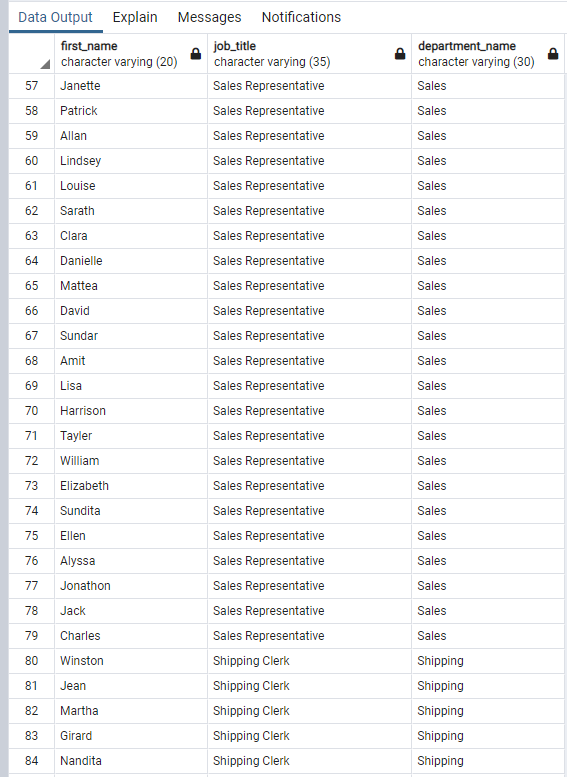
**SELECT** first\_name, job\_title, department\_name

**FROM** employees e

**JOIN** jobs j **ON** (e.job\_id = j.job\_id)

**JOIN** departments d **ON** (d.department\_id = e.department\_id)





64. Таблица Employees. Получить список сотрудников менеджеры которых устроились на работу в 2005ом году но при это сами эти работники устроились на работу до 2005 года

**SELECT** emp.\*

**FROM** employees emp **JOIN** employees man **ON** (emp.manager\_id = man.employee\_id)

**WHERE** TO\_CHAR (man.hire\_date, 'YYYY') = '2005'

**AND** emp.hire\_date < **TO\_DATE** ('01012005', 'DDMMYYYY')



**Скорей всего не правильно(**

65. Таблица Employees. Получить список сотрудников менеджеры которых устроились на работу в январе месяце любого года и длина job\_title этих сотрудников больше 15ти символов

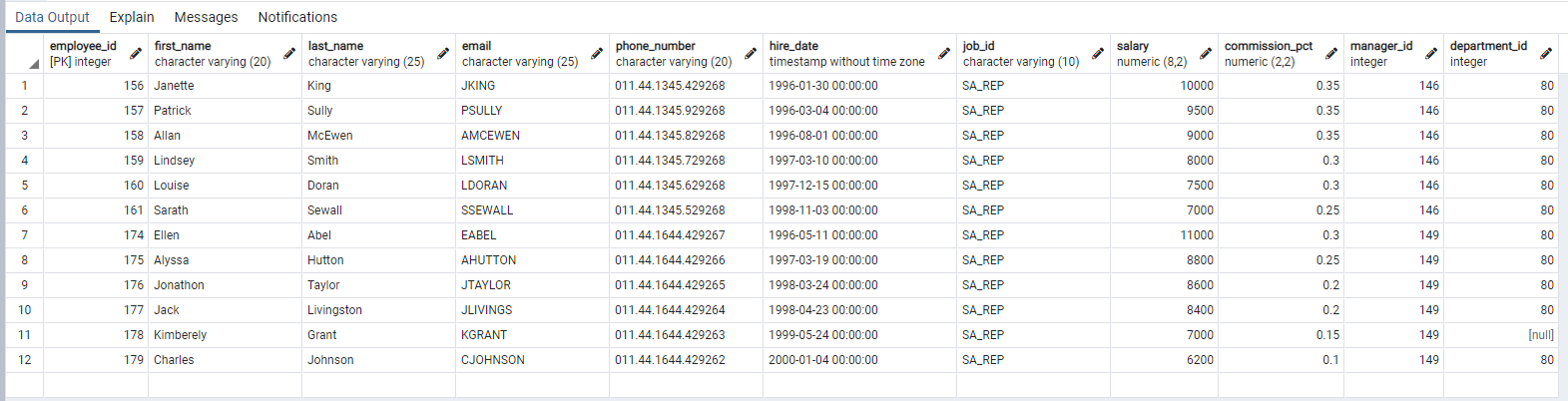
**SELECT** emp.\*

**FROM** employees emp

**JOIN** employees man **ON** (emp.manager\_id = man.employee\_id)

**JOIN** jobs j **ON** (emp.job\_id = j.job\_id)

**WHERE** TO\_CHAR (man.hire\_date, 'MM') = '01' **AND** **LENGTH** (j.job\_title) > 15



**Using Subqueries to Solve Queries**

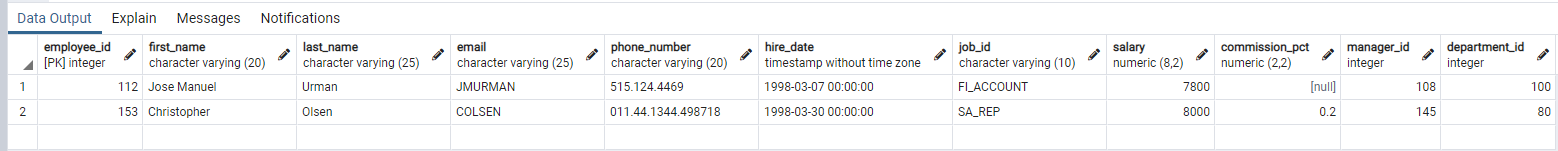
66. Таблица Employees. Получить список сотрудников с самым длинным именем.

**SELECT** \*

**FROM** employees

**WHERE** **LENGTH** (first\_name) =

(**SELECT** **MAX** (**LENGTH** (first\_name)) **FROM** employees)

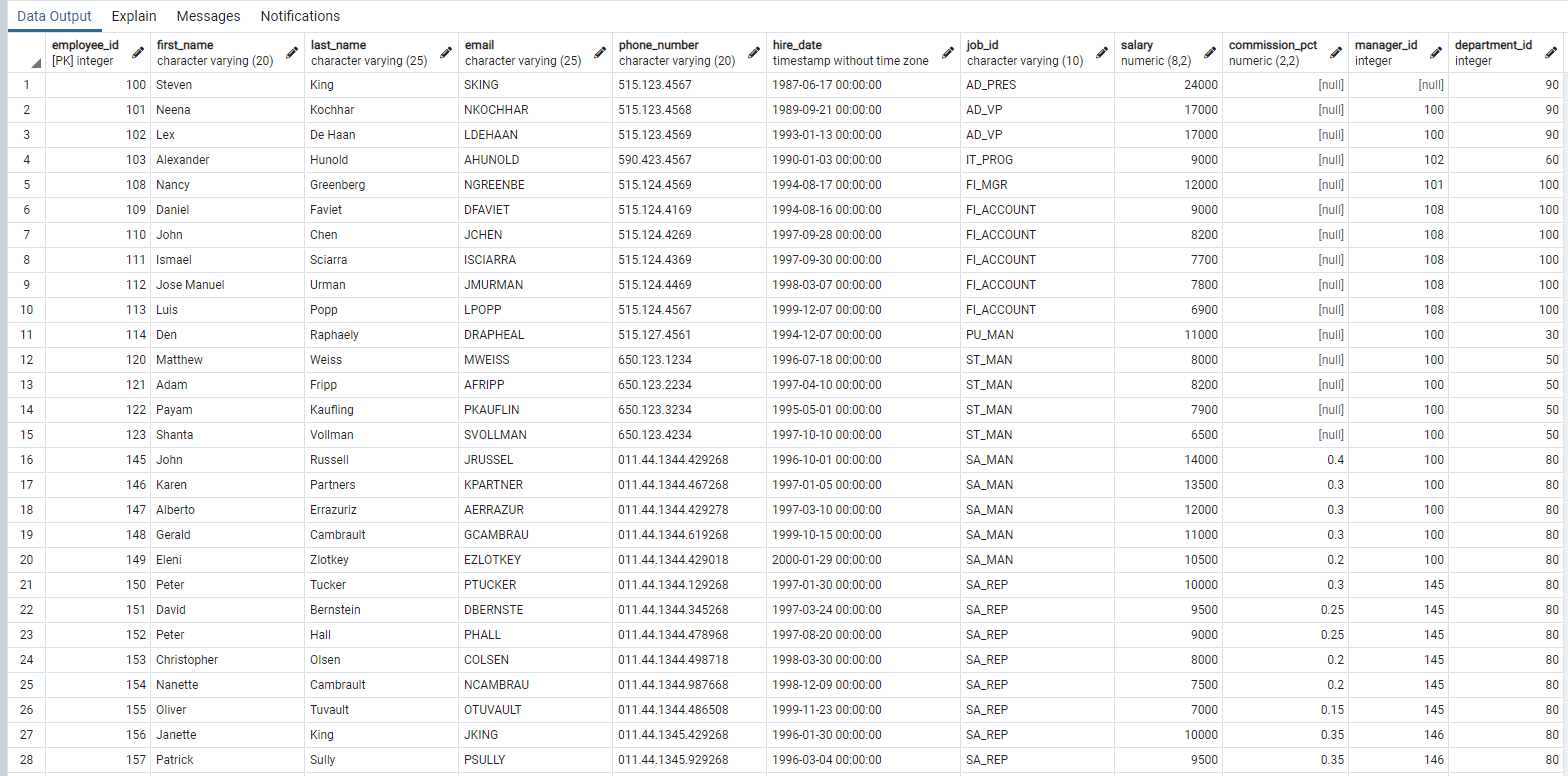


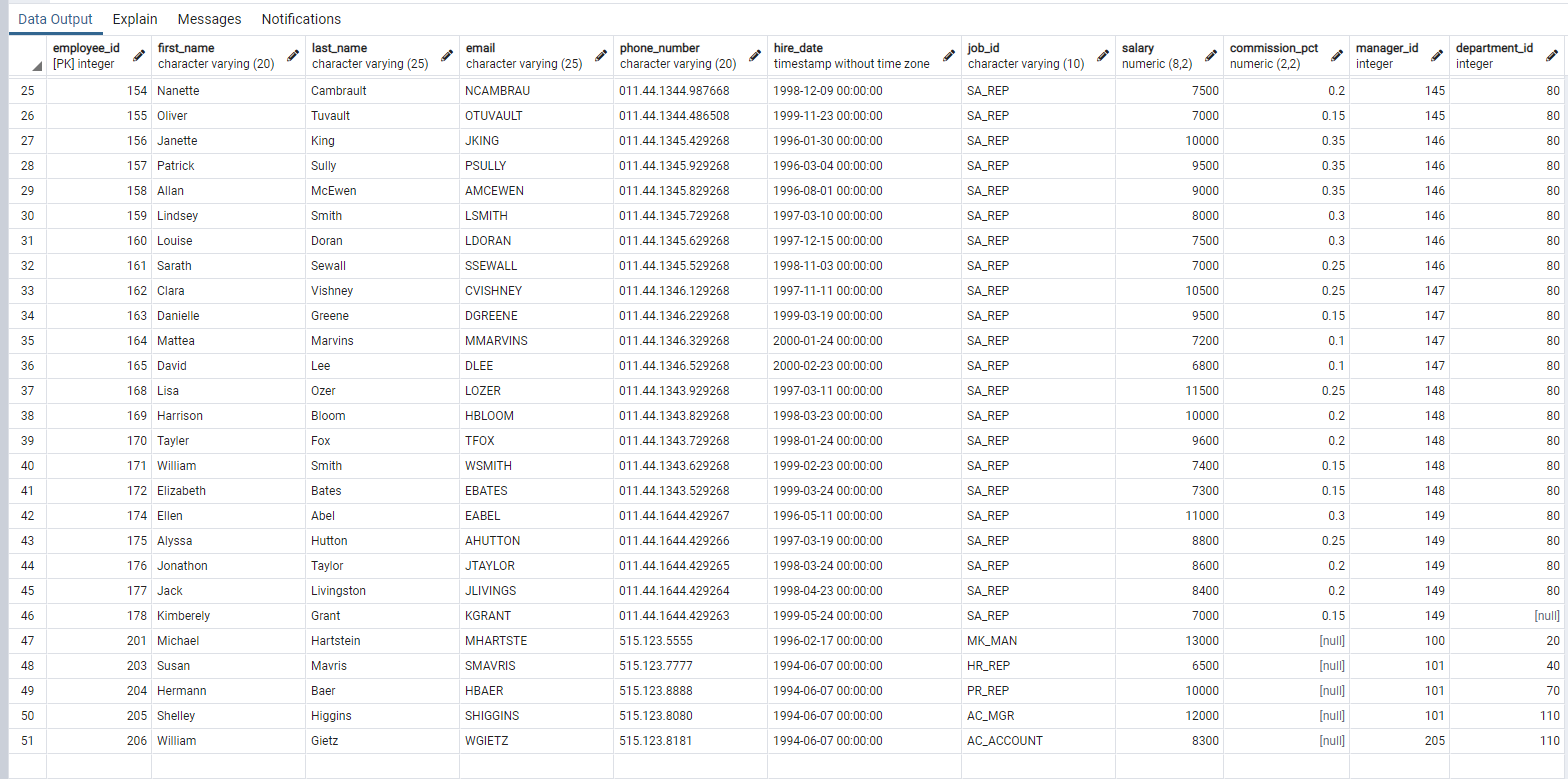
67. Таблица Employees. Получить список сотрудников с зарплатой большей средней зарплаты всех сотрудников.

**SELECT** \*

**FROM** employees

**WHERE** salary > (**SELECT** **AVG** (salary) **FROM** employees)





68. Таблица Employees, Departments, Locations. Получить город в котором сотрудники в сумме зарабатывают меньше всех.

**Не смог(**

69. Таблица Employees. Получить список сотрудников у которых менеджер получает зарплату больше 15000.

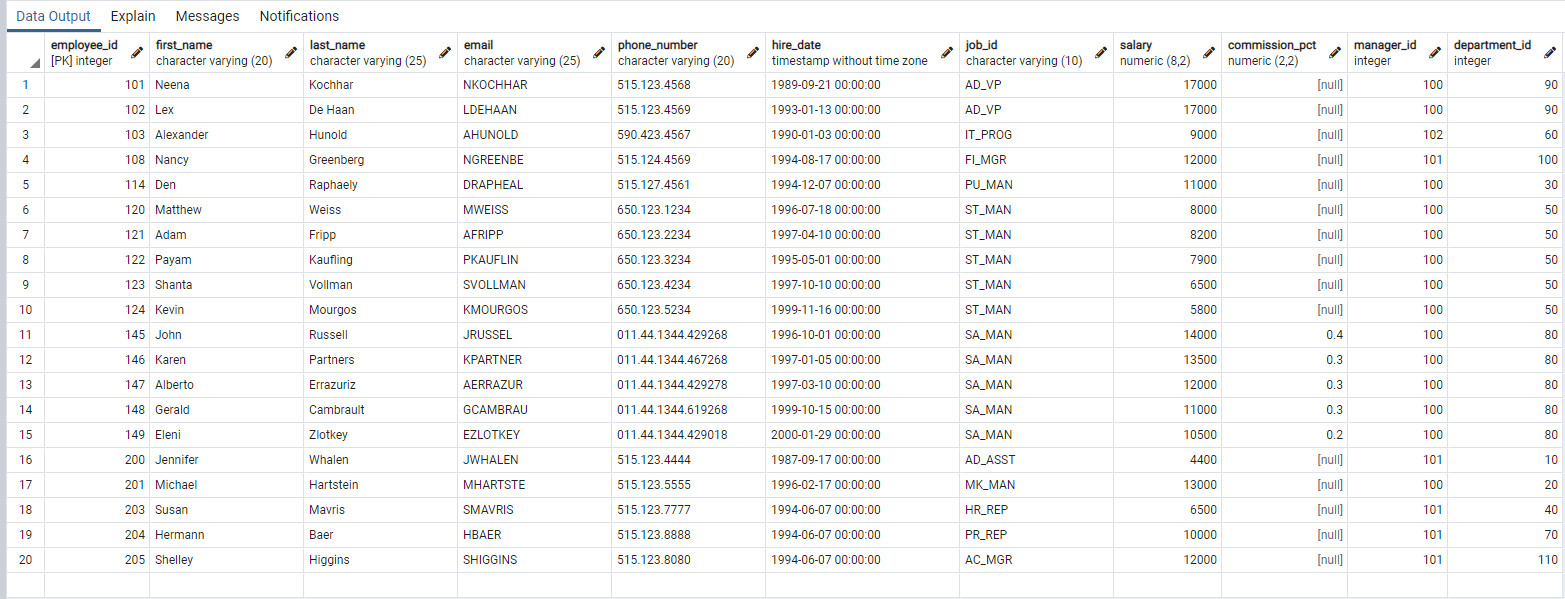
**SELECT** \*

**FROM** employees

**WHERE** manager\_id **IN** (**SELECT** employee\_id

**FROM** employees

**WHERE** salary > 15000)



70. Таблица Employees, Departaments. Показать все департаменты в которых нет ни одного сотрудника

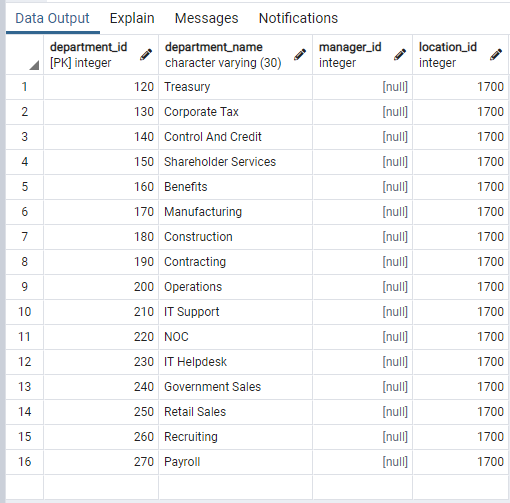
**SELECT** \*

**FROM** departments

**WHERE** department\_id **NOT** **IN** (**SELECT** department\_id

**FROM** employees

**WHERE** department\_id **IS** **NOT** NULL)



71. Таблица Employees. Показать всех сотрудников которые не являются менеджерами

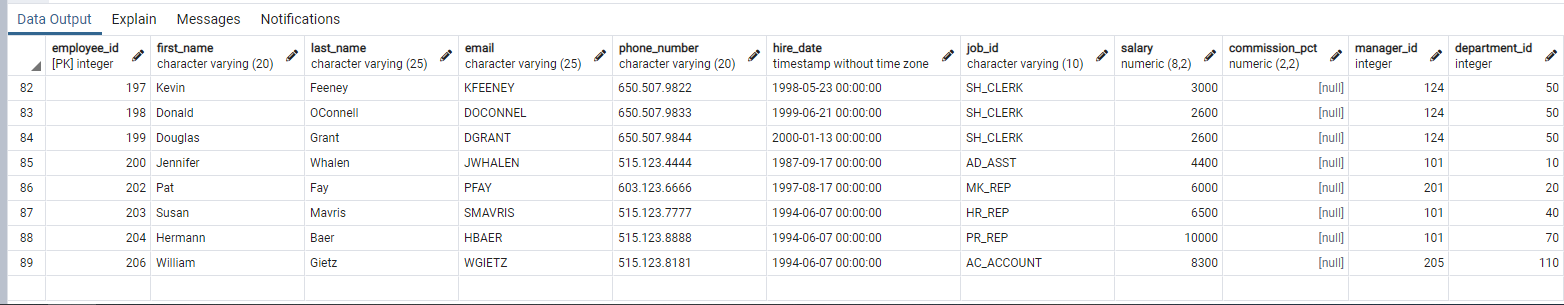
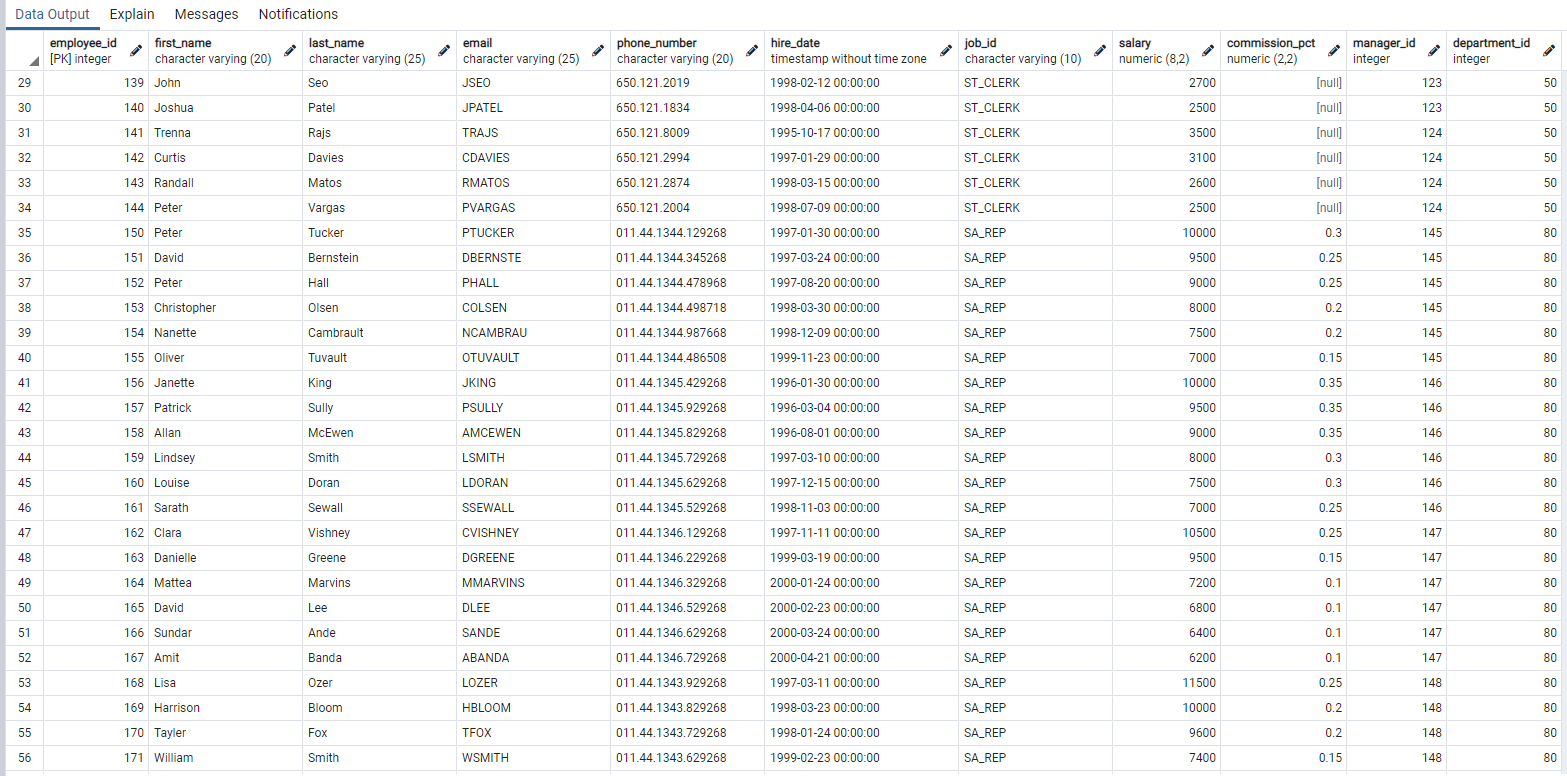
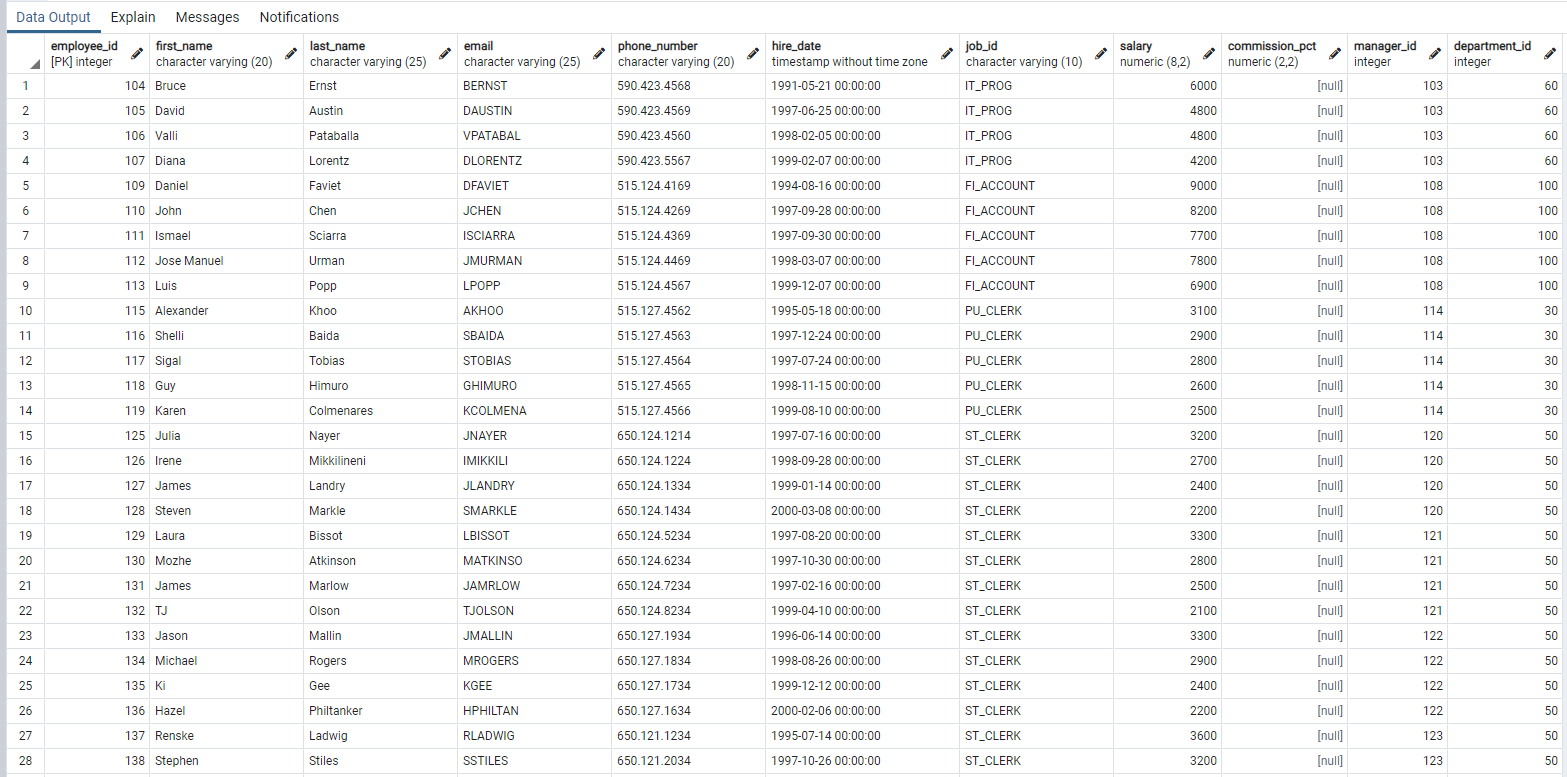
**SELECT** \*

**FROM** employees

**WHERE** employee\_id **NOT** **IN** (**SELECT** manager\_id

**FROM** employees

**WHERE** manager\_id **IS** **NOT** NULL)



72. Таблица Employees. Показать всех менеджеров которые имеют в подчинении больше 6ти сотрудников

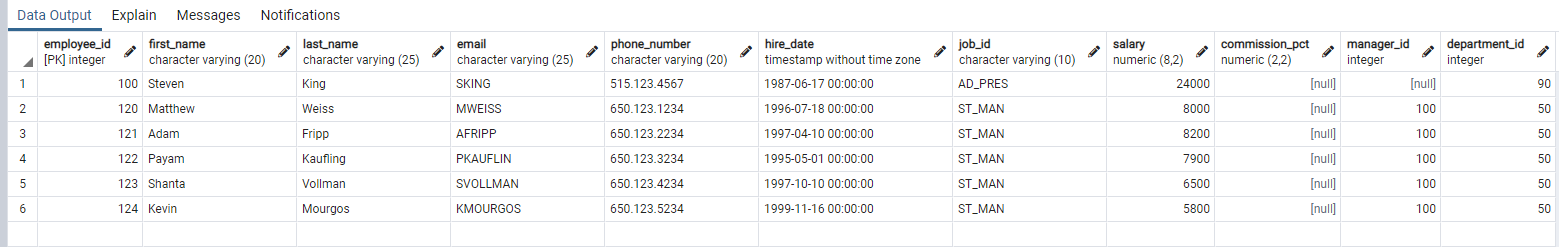
**SELECT** \*

**FROM** employees e

**WHERE** (**SELECT** **COUNT** (\*)

**FROM** employees

**WHERE** manager\_id = e.employee\_id) > 6



73. Таблица Employees, Departaments. Показать сотрудников которые работают в департаменте IT

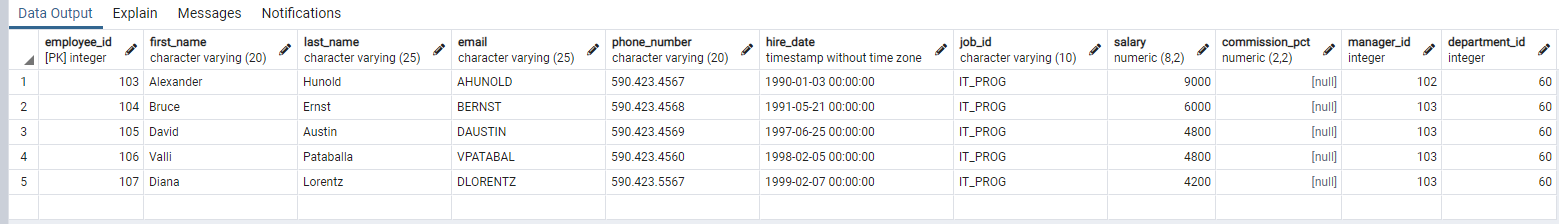
**SELECT** \*

**FROM** employees

**WHERE** department\_id = (**SELECT** department\_id

**FROM** departments

**WHERE** department\_name = 'IT')



74. Таблица Employees, Jobs, Departaments. Показать сотрудников в формате: First\_name, Job\_title, Department\_name.

Пример:

First\_name | Job\_title | Department\_name

Donald | Shipping | Clerk Shipping

**SELECT** first\_name,

(**SELECT** job\_title

**FROM** jobs

**WHERE** job\_id = e.job\_id)

job\_title,

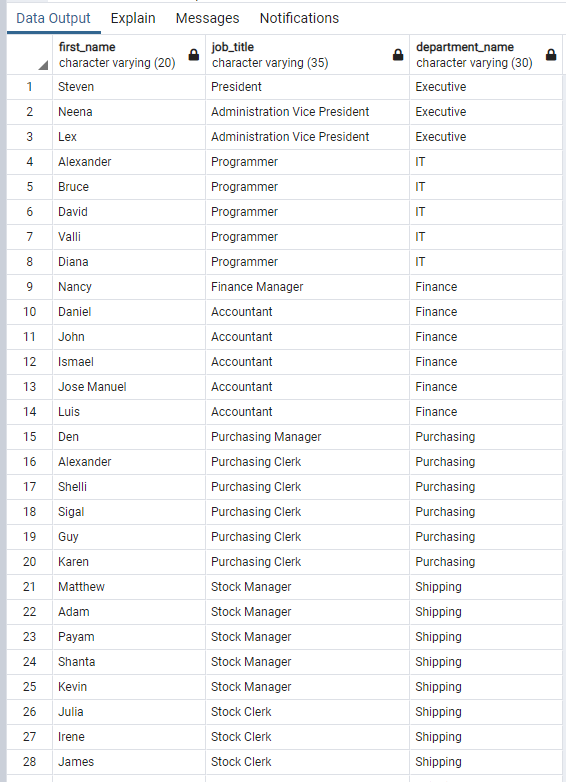
(**SELECT** department\_name

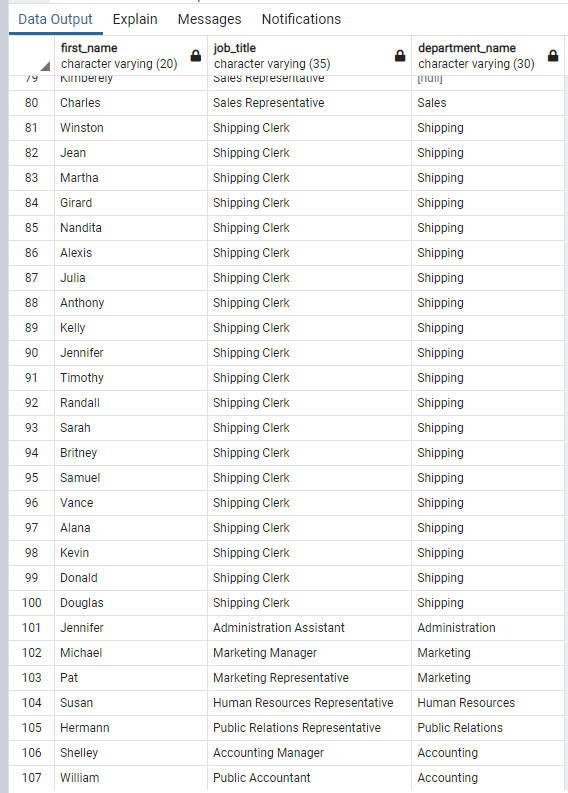
**FROM** departments

**WHERE** department\_id = e.department\_id)

department\_name

**FROM** employees e





75. Таблица Employees. Получить список сотрудников менеджеры которых устроились на работу в 2005ом году но при это сами эти работники устроились на работу до 2005 года

**SELECT** \*

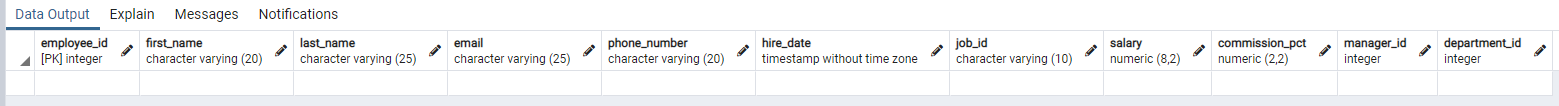
**FROM** employees

**WHERE** manager\_id **IN** (**SELECT** employee\_id

**FROM** employees

**WHERE** TO\_CHAR (hire\_date, 'YYYY') = '2005')

**AND** hire\_date < **TO\_DATE** ('01012005', 'DDMMYYYY')



**Скорей всего не правильно(**

76. Таблица Employees. Получить список сотрудников менеджеры которых устроились на работу в январе месяце любого года и длина job\_title этих сотрудников больше 15ти символов

**SELECT** \*

**FROM** employees e

**WHERE** manager\_id **IN** (**SELECT** employee\_id

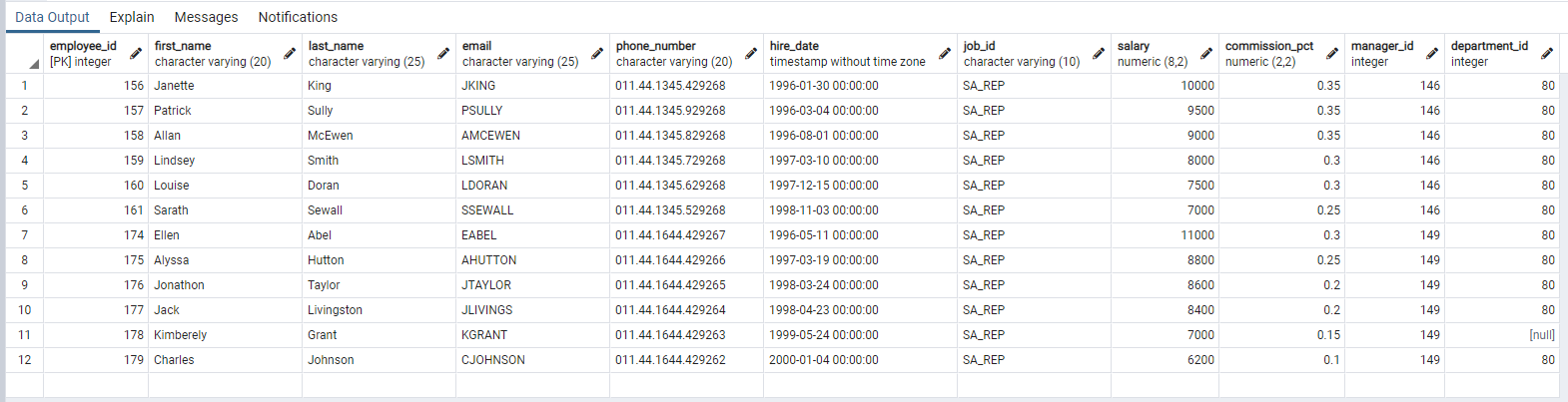
**FROM** employees

**WHERE** TO\_CHAR (hire\_date, 'MM') = '01')

**AND** (**SELECT** **LENGTH** (job\_title)

**FROM** jobs

**WHERE** job\_id = e.job\_id) > 15



SELECT \* FROM employees WHERE salary = (SELECT MAX(salary) FROM employees AS max WHERE max.department\_id = employees.department\_id)

SELECT CONCAT(employee\_id, ' "', first\_name, ' ', last\_name, '" ', salary, 'fm$99,999.00', ' ', manager\_id, ' ', department\_id) FROM employees WHERE salary = (SELECT MAX(salary) FROM employees AS max WHERE max.department\_id = employees.department\_id)