CREATE TABLE jobs (

JOB\_ID NVARCHAR(10) PRIMARY KEY,

JOB\_TITLE NVARCHAR(100),

MIN\_SALARY INT,

MAX\_SALARY INT

);

INSERT INTO jobs (JOB\_ID, JOB\_TITLE, MIN\_SALARY, MAX\_SALARY)

VALUES

('AD\_PRES', 'President', 20080, 40000),

('AD\_VP', 'Administration Vice President', 15000, 30000),

('AD\_ASST', 'Administration Assistant', 3000, 6000),

('FI\_MGR', 'Finance Manager', 8200, 16000),

('FI\_ACCOUNT', 'Accountant', 4200, 9000),

('AC\_MGR', 'Accounting Manager', 8200, 16000),

('AC\_ACCOUNT', 'Public Accountant', 4200, 9000),

('SA\_MAN', 'Sales Manager', 10000, 20080),

('SA\_REP', 'Sales Representative', 6000, 12008),

('PU\_MAN', 'Purchasing Manager', 8000, 15000),

('PU\_CLERK', 'Purchasing Clerk', 2500, 5500),

('ST\_MAN', 'Stock Manager', 5500, 8500),

('ST\_CLERK', 'Stock Clerk', 2008, 5000),

('SH\_CLERK', 'Shipping Clerk', 2500, 5500),

('IT\_PROG', 'Programmer', 4000, 10000),

('MK\_MAN', 'Marketing Manager', 9000, 15000),

('MK\_REP', 'Marketing Representative', 4000, 9000),

('HR\_REP', 'Human Resources Representative', 4000, 9000),

('PR\_REP', 'Public Relations Representative', 4500, 10500);

CREATE TABLE job\_history (

EMPLOYEE\_ID INT,

START\_DATE DATE,

END\_DATE DATE,

JOB\_ID NVARCHAR(10),

DEPARTMENT\_ID INT,

PRIMARY KEY (EMPLOYEE\_ID, START\_DATE, JOB\_ID), -- Composite primary key

FOREIGN KEY (JOB\_ID) REFERENCES jobs(JOB\_ID) -- Assuming `jobs` table exists and is referenced

);

INSERT INTO job\_history (EMPLOYEE\_ID, START\_DATE, END\_DATE, JOB\_ID, DEPARTMENT\_ID)

VALUES

(102, '2001-01-13', '2006-07-24', 'IT\_PROG', 60),

(101, '1997-09-21', '2001-10-27', 'AC\_ACCOUNT', 110),

(101, '2001-10-28', '2005-03-15', 'AC\_MGR', 110),

(201, '2004-02-17', '2007-12-19', 'MK\_REP', 20),

(114, '2006-03-24', '2007-12-31', 'ST\_CLERK', 50),

(122, '2007-01-01', '2007-12-31', 'ST\_CLERK', 50),

(200, '1995-09-17', '2001-06-17', 'AD\_ASST', 90),

(176, '2006-03-24', '2006-12-31', 'SA\_REP', 80),

(176, '2007-01-01', '2007-12-31', 'SA\_MAN', 80),

(200, '2002-07-01', '2006-12-31', 'AC\_ACCOUNT', 90);

CREATE TABLE countries (

COUNTRY\_ID NVARCHAR(2) PRIMARY KEY,

COUNTRY\_NAME NVARCHAR(100),

REGION\_ID INT

);

INSERT INTO countries (COUNTRY\_ID, COUNTRY\_NAME, REGION\_ID)

VALUES

('AR', 'Argentina', 2),

('AU', 'Australia', 3),

('BE', 'Belgium', 1),

('BR', 'Brazil', 2),

('CA', 'Canada', 2),

('CH', 'Switzerland', 1),

('CN', 'China', 3),

('DE', 'Germany', 1),

('DK', 'Denmark', 1),

('EG', 'Egypt', 4),

('FR', 'France', 1),

('IL', 'Israel', 4),

('IN', 'India', 3),

('IT', 'Italy', 1),

('JP', 'Japan', 3),

('KW', 'Kuwait', 4),

('ML', 'Malaysia', 3),

('MX', 'Mexico', 2),

('NG', 'Nigeria', 4),

('NL', 'Netherlands', 1),

('SG', 'Singapore', 3),

('UK', 'United Kingdom', 1),

('US', 'United States of America', 2),

('ZM', 'Zambia', 4),

('ZW', 'Zimbabwe', 4);

CREATE TABLE employees (

EMPLOYEE\_ID INT PRIMARY KEY,

FIRST\_NAME VARCHAR(50),

LAST\_NAME VARCHAR(50),

EMAIL VARCHAR(100),

PHONE\_NUMBER VARCHAR(20),

HIRE\_DATE DATE,

JOB\_ID VARCHAR(10),

SALARY DECIMAL(10, 2),

COMMISSION\_PCT DECIMAL(2, 2),

MANAGER\_ID INT,

DEPARTMENT\_ID INT

);

INSERT INTO employees (EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME, EMAIL, PHONE\_NUMBER, HIRE\_DATE, JOB\_ID, SALARY, COMMISSION\_PCT, MANAGER\_ID, DEPARTMENT\_ID) VALUES

(100, 'Steven', 'King', 'SKING', '515.123.4567', '2003-06-17', 'AD\_PRES', 24000.00, 0.00, 0, 90),

(101, 'Neena', 'Kochhar', 'NKOCHHAR', '515.123.4568', '2005-09-21', 'AD\_VP', 17000.00, 0.00, 100, 90),

(102, 'Lex', 'De Haan', 'LDEHAAN', '515.123.4569', '2001-01-13', 'AD\_VP', 17000.00, 0.00, 100, 90),

(103, 'Alexander', 'Hunold', 'AHUNOLD', '590.423.4567', '2006-01-03', 'IT\_PROG', 9000.00, 0.00, 102, 60),

(104, 'Bruce', 'Ernst', 'BERNST', '590.423.4568', '2007-05-21', 'IT\_PROG', 6000.00, 0.00, 103, 60),

(105, 'David', 'Austin', 'DAUSTIN', '590.423.4569', '2005-06-25', 'IT\_PROG', 4800.00, 0.00, 103, 60),

(106, 'Valli', 'Pataballa', 'VPATABAL', '590.423.4560', '2006-02-05', 'IT\_PROG', 4800.00, 0.00, 103, 60),

(107, 'Diana', 'Lorentz', 'DLORENTZ', '590.423.5567', '2007-02-07', 'IT\_PROG', 4200.00, 0.00, 103, 60),

(108, 'Nancy', 'Greenberg', 'NGREENBE', '515.124.4569', '2002-08-17', 'FI\_MGR', 12008.00, 0.00, 101, 100),

(109, 'Daniel', 'Faviet', 'DFAVIET', '515.124.4169', '2002-08-16', 'FI\_ACCOUNT', 9000.00, 0.00, 108, 100),

(110, 'John', 'Chen', 'JCHEN', '515.124.4269', '2005-09-28', 'FI\_ACCOUNT', 8200.00, 0.00, 108, 100),

(111, 'Ismael', 'Sciarra', 'ISCIARRA', '515.124.4369', '2005-09-30', 'FI\_ACCOUNT', 7700.00, 0.00, 108, 100),

(112, 'Jose Manuel', 'Urman', 'JMURMAN', '515.124.4469', '2006-03-07', 'FI\_ACCOUNT', 7800.00, 0.00, 108, 100),

(113, 'Luis', 'Popp', 'LPOPP', '515.124.4567', '2007-12-07', 'FI\_ACCOUNT', 6900.00, 0.00, 108, 100),

(114, 'Den', 'Raphaely', 'DRAPHEAL', '515.127.4561', '2002-12-07', 'PU\_MAN', 11000.00, 0.00, 100, 30),

(115, 'Alexander', 'Khoo', 'AKHOO', '515.127.4562', '2003-05-18', 'PU\_CLERK', 3100.00, 0.00, 114, 30),

(116, 'Shelli', 'Baida', 'SBAIDA', '515.127.4563', '2005-12-24', 'PU\_CLERK', 2900.00, 0.00, 114, 30),

(117, 'Sigal', 'Tobias', 'STOBIAS', '515.127.4564', '2005-07-24', 'PU\_CLERK', 2800.00, 0.00, 114, 30),

(118, 'Guy', 'Himuro', 'GHIMURO', '515.127.4565', '2006-11-15', 'PU\_CLERK', 2600.00, 0.00, 114, 30),

(119, 'Karen', 'Colmenares', 'KCOLMENA', '515.127.4566', '2007-08-10', 'PU\_CLERK', 2500.00, 0.00, 114, 30),

(120, 'Matthew', 'Weiss', 'MWEISS', '650.123.1234', '2004-07-18', 'ST\_MAN', 8000.00, 0.00, 100, 50),

(121, 'Adam', 'Fripp', 'AFRIPP', '650.123.2234', '2005-04-10', 'ST\_MAN', 8200.00, 0.00, 100, 50),

(122, 'Payam', 'Kaufling', 'PKAUFLIN', '650.123.3234', '2003-05-01', 'ST\_MAN', 7900.00, 0.00, 100, 50),

(123, 'Shanta', 'Vollman', 'SVOLLMAN', '650.123.4234', '2005-10-10', 'ST\_MAN', 6500.00, 0.00, 100, 50),

(124, 'Kevin', 'Mourgos', 'KMOURGOS', '650.123.5234', '2007-11-16', 'ST\_MAN', 5800.00, 0.00, 100, 50),

(125, 'Julia', 'Nayer', 'JNAYER', '650.124.1214', '2005-07-16', 'ST\_CLERK', 3200.00, 0.00, 120, 50),

(126, 'Irene', 'Mikkilineni', 'IMIKKILI', '650.124.1224', '2006-09-28', 'ST\_CLERK', 2700.00, 0.00, 120, 50),

(127, 'James', 'Landry', 'JLANDRY', '650.124.1334', '2007-01-14', 'ST\_CLERK', 2400.00, 0.00, 120, 50),

(128, 'Steven', 'Markle', 'SMARKLE', '650.124.1434', '2008-03-08', 'ST\_CLERK', 2200.00, 0.00, 120, 50),

(129, 'Laura', 'Bissot', 'LBISSOT', '650.124.5234', '2005-08-20', 'ST\_CLERK', 3300.00, 0.00, 121, 50),

(130, 'Mozhe', 'Atkinson', 'MATKINSO', '650.124.6234', '2005-10-30', 'ST\_CLERK', 2800.00, 0.00, 121, 50),

(131, 'James', 'Marlow', 'JAMRLOW', '650.124.7234', '2005-02-16', 'ST\_CLERK', 2500.00, 0.00, 121, 50),

(132, 'TJ', 'Olson', 'TJOLSON', '650.124.8234', '2007-04-10', 'ST\_CLERK', 2100.00, 0.00, 121, 50),

(133, 'Jason', 'Mallin', 'JMALLIN', '650.127.1934', '2004-06-14', 'ST\_CLERK', 3300.00, 0.00, 122, 50),

(134, 'Michael', 'Rogers', 'MROGERS', '650.127.1834', '2006-08-26', 'ST\_CLERK', 2800.00, 0.00, 122, 50),

(135, 'Irene', 'Zabers', 'IZABERS', '650.127.1734', '2007-01-30', 'ST\_CLERK', 2300.00, 0.00, 122, 50);

CREATE TABLE locations (

LOCATION\_ID INT PRIMARY KEY,

STREET\_ADDRESS VARCHAR(100),

POSTAL\_CODE VARCHAR(20),

CITY VARCHAR(50),

STATE\_PROVINCE VARCHAR(50),

COUNTRY\_ID CHAR(2)

);

INSERT INTO locations (LOCATION\_ID, STREET\_ADDRESS, POSTAL\_CODE, CITY, STATE\_PROVINCE, COUNTRY\_ID) VALUES

(1000, '1297 Via Cola di Rie', '989', 'Roma', NULL, 'IT'),

(1100, '93091 Calle della Testa', '10934', 'Venice', NULL, 'IT'),

(1200, '2017 Shinjuku-ku', '1689', 'Tokyo', 'Tokyo Prefecture', 'JP'),

(1300, '9450 Kamiya-cho', '6823', 'Hiroshima', NULL, 'JP'),

(1400, '2014 Jabberwocky Rd', '26192', 'Southlake', 'Texas', 'US'),

(1500, '2011 Interiors Blvd', '99236', 'South San Francisco', 'California', 'US'),

(1600, '2007 Zagora St', '50090', 'South Brunswick', 'New Jersey', 'US'),

(1700, '2004 Charade Rd', '98199', 'Seattle', 'Washington', 'US'),

(1800, '147 Spadina Ave', 'M5V 2L7', 'Toronto', 'Ontario', 'CA'),

(1900, '6092 Boxwood St', 'YSW 9T2', 'Whitehorse', 'Yukon', 'CA'),

(2000, '40-5-12 Laogianggen', '190518', 'Beijing', NULL, 'CN'),

(2100, '1298 Vileparle (E)', '490231', 'Bombay', 'Maharashtra', 'IN'),

(2200, '12-98 Victoria Street', '2901', 'Sydney', 'New South Wales', 'AU'),

(2300, '198 Clementi North', '540198', 'Singapore', NULL, 'SG'),

(2400, '8204 Arthur St', NULL, 'London', NULL, 'UK'),

(2500, 'Magdalen Centre, The Oxford Science Park', 'OX9 9ZB', 'Oxford', 'Oxford', 'UK'),

(2600, '9702 Chester Road', '9629850293', 'Stretford', 'Manchester', 'UK'),

(2700, 'Schwanthalerstr. 7031', '80925', 'Munich', 'Bavaria', 'DE'),

(2800, 'Rua Frei Caneca 1360', '01307-002', 'Sao Paulo', 'Sao Paulo', 'BR'),

(2900, '20 Rue des Corps-Saints', '1730', 'Geneva', 'Geneve', 'CH'),

(3000, 'Murtenstrasse 921', '3095', 'Bern', 'BE', 'CH'),

(3100, 'Pieter Breughelstraat 837', '3029SK', 'Utrecht', 'Utrecht', 'NL'),

(3200, 'Mariano Escobedo 9991', '11932', 'Mexico City', 'Distrito Federal', 'MX');

CREATE TABLE departments (

DEPARTMENT\_ID INT PRIMARY KEY,

DEPARTMENT\_NAME VARCHAR(50),

MANAGER\_ID INT,

LOCATION\_ID INT

);

INSERT INTO departments (DEPARTMENT\_ID, DEPARTMENT\_NAME, MANAGER\_ID, LOCATION\_ID) VALUES

(10, 'Administration', 200, 1700),

(20, 'Marketing', 201, 1800),

(30, 'Purchasing', 114, 1700),

(40, 'Human Resources', 203, 2400),

(50, 'Shipping', 121, 1500),

(60, 'IT', 103, 1400),

(70, 'Public Relations', 204, 2700),

(80, 'Sales', 145, 2500),

(90, 'Executive', 100, 1700),

(100, 'Finance', 108, 1700),

(110, 'Accounting', 205, 1700),

(120, 'Treasury', 0, 1700),

(130, 'Corporate Tax', 0, 1700),

(140, 'Control And Credit', 0, 1700),

(150, 'Shareholder Services', 0, 1700),

(160, 'Benefits', 0, 1700),

(170, 'Manufacturing', 0, 1700),

(180, 'Construction', 0, 1700),

(190, 'Contracting', 0, 1700),

(200, 'Operations', 0, 1700),

(210, 'IT Support', 0, 1700),

(220, 'NOC', 0, 1700),

(230, 'IT Helpdesk', 0, 1700),

(240, 'Government Sales', 0, 1700),

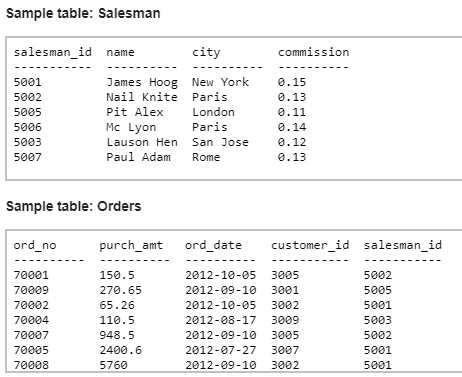
(250, 'Retail Sales', 0, 1700),

(260, 'Recruiting', 0, 1700),

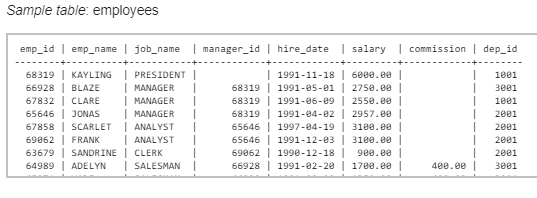
(270, 'Payroll', 0, 1700);

**Subqueries**

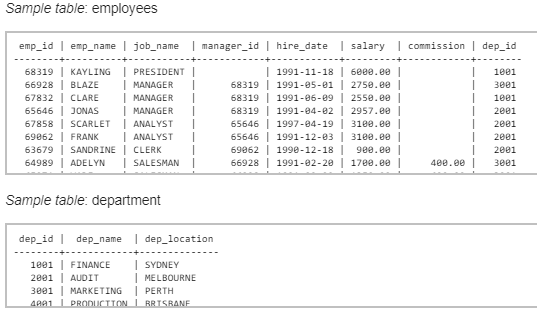
1. Write a query to display all the orders from the orders table issued by the salesman 'Paul Adam'.



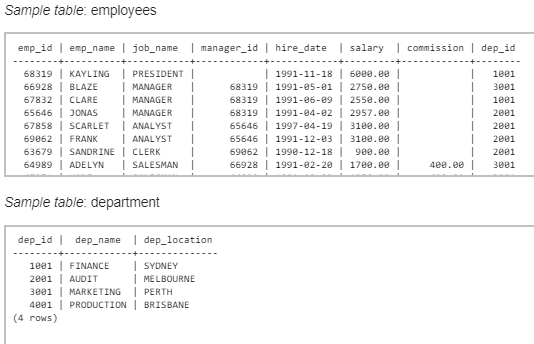
1. Write a query in SQL to display all the details of managers.



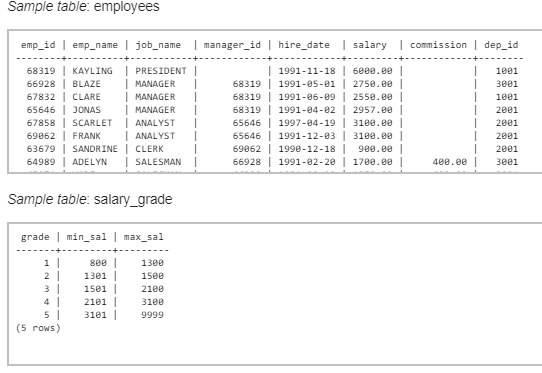
1. Write a query in SQL to list the employee ID, name, salary, department name of all the 'MANAGERS' and 'ANALYST' working in SYDNEY, PERTH with an exp more than 5 years without receiving the commission and display the list in ascending order of location.



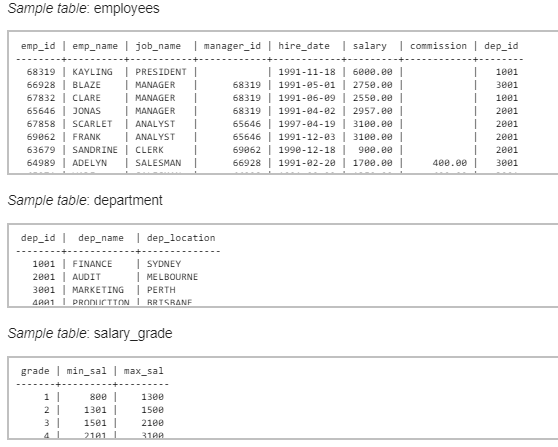
1. Write a query in SQL to display the employee ID, name, salary, department name, location, department ID, job name of all the employees working at SYDNEY or working in the FINANCE department with an annual salary above 28000, but the monthly salary should not be 3000 or 2800 and who does not works as a MANAGER and whose ID containing a digit of '3' or '7' in 3rd position. List the result in ascending order of department ID and descending order of job name.



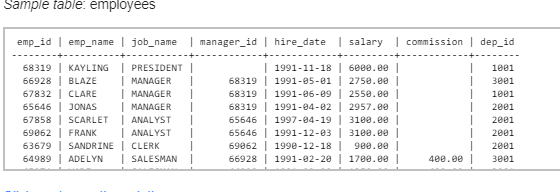
1. Write a query in SQL to list all the employees of grade 2 and 3.



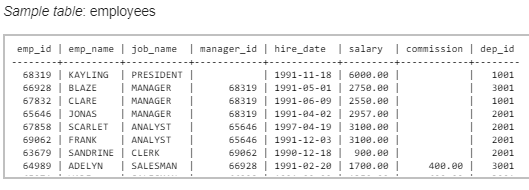
1. Write a query in SQL to list the employees of grade 3 and 4 working in the department of FINANCE or AUDIT and whose salary is more than the salary of ADELYN and experience is more than FRANK. List the result in the ascending order of experience.



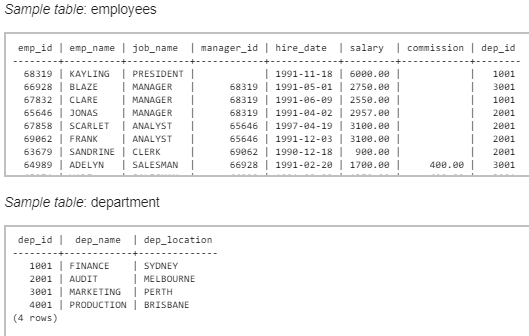
1. Write a query in SQL to list the employees whose designation is same as the designation of SANDRINE or ADELYN.



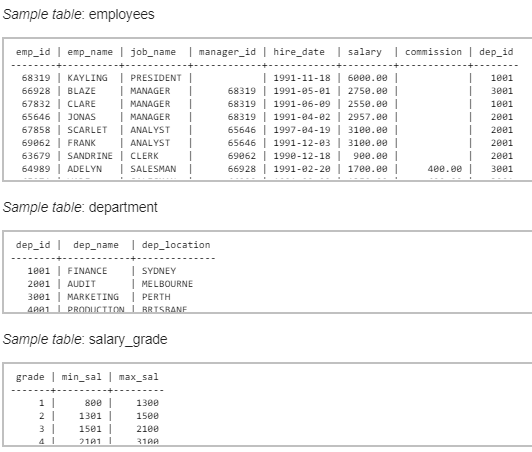
1. Write a query in SQL to list any job of department ID 1001 those that are not found in department ID 2001.



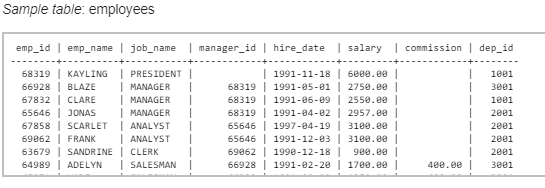
1. Write a query in SQL to find the highest paid employees in the department MARKETING.



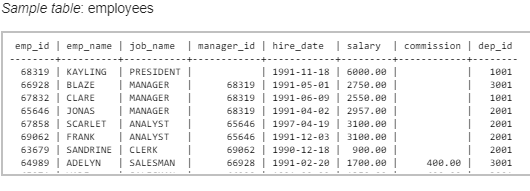
1. Write a query in SQL to list the employees of grade 3 who have been hired in most recently and belongs to PERTH.



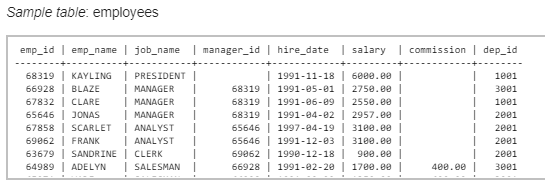
1. Write a query in SQL to list the employees who are senior to most recently hired employee working under KAYLING.



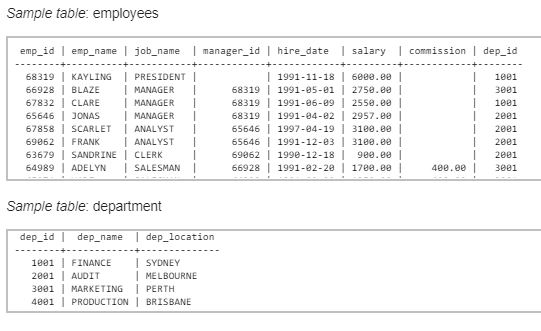
1. Write a query in SQL to list the employees who joined in 1991 in a designation same as the most senior person of the year 1991.



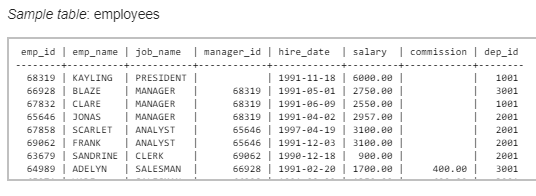
1. Write a query in SQL to list the employees in department 1001 whose salary is more than the average salary of employees in department 2001



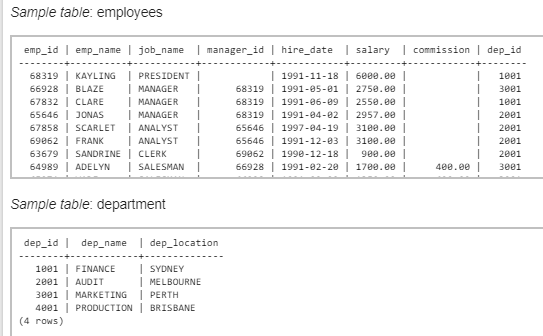
1. Write a query in SQL to list the details of the departments where maximum number of employees are working.



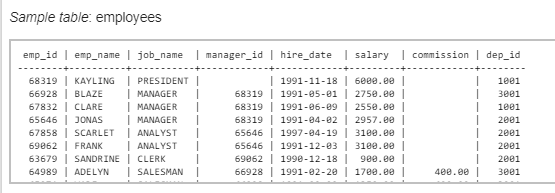
1. Write a query in SQL to display the employees whose manager name is JONAS.



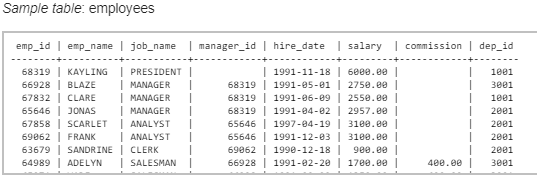
1. Write a query in SQL to list the name, job name, department name, location for those who are working as a manager.



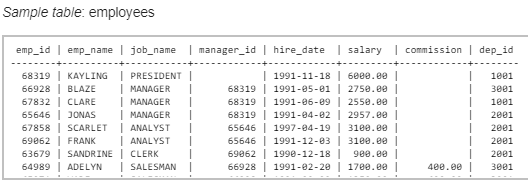
1. Write a query in SQL to list the name of the employees who are getting the highest salary of each department.



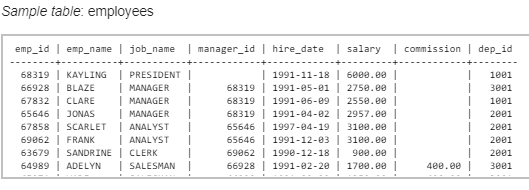
1. Write a query in SQL to list the employees whose salary is equal or more to the average of maximum and minimum salary.



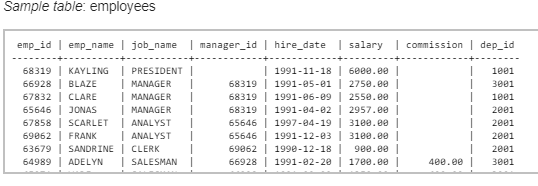
1. Write a query in SQL to list the employees who are SALESMAN and gathered an experience which month portion is more than 10



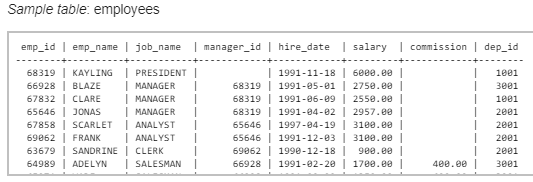
1. Write a query in SQL to list the employees whose salary is less than the salary of his manager but more than the salary of any other manager.



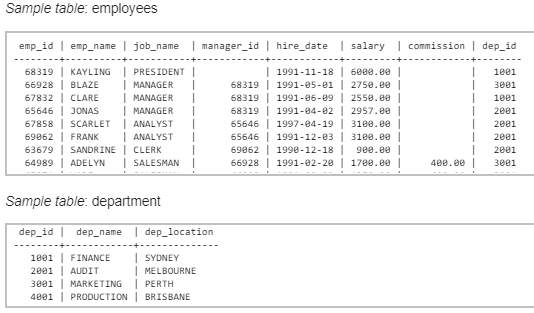
1. Write a query in SQL to find out the least 5 earners of the company.



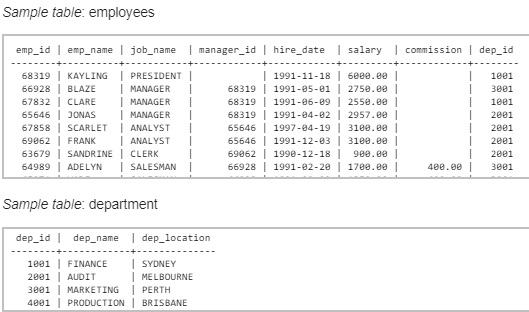
1. Write a query in SQL to list the managers who are not working under the PRESIDENT.



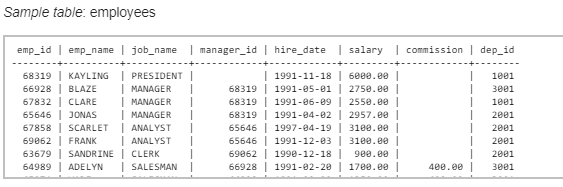
1. Write a query in SQL to list the name of the department where number of employees is equal to the number of characters in the department name.



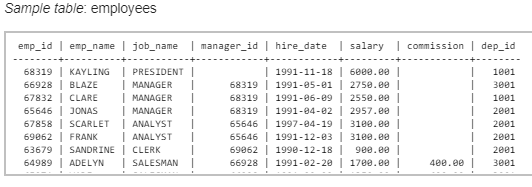
1. Write a query in SQL to list the name of the departments where more than average number of employees are working.



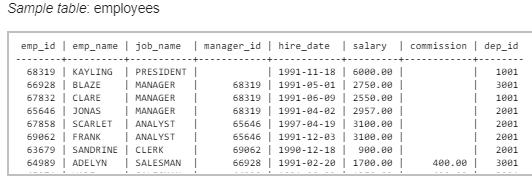
1. Write a query in SQL to list the employees who are working as managers, using co-related subquery.



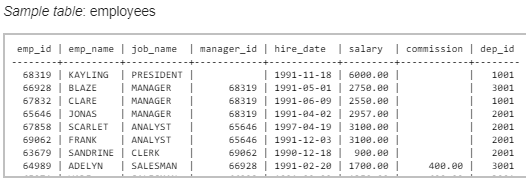
1. Write a query in SQL to list the name of the employees for their manager JONAS and also the name of the manager of JONAS.



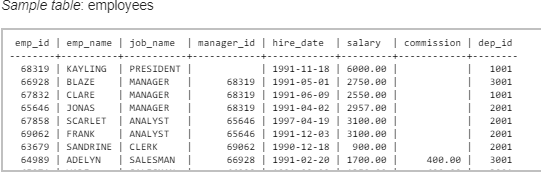
1. Write a query in SQL to find all the employees who earn the minimum salary for a designation and arrange the list in ascending order on salary.



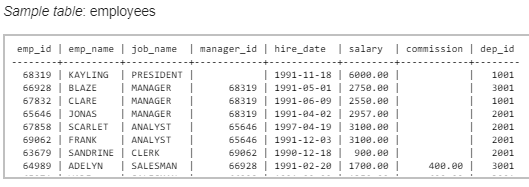
1. Write a query in SQL to find the most recently hired emps in each department order by hire\_date.



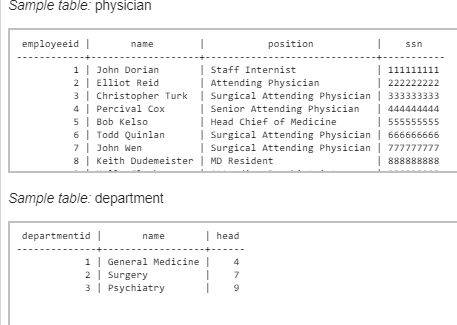
1. Write a query in SQL to list the name,salary, and department id for each employee who earns a salary greater than the average salary for their department and list the result in ascending order on department id.



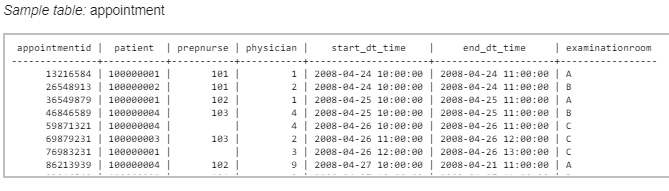
1. Write a query in SQL to list the name, designation, and salary of the employees who does not work in the department 1001 but works in same designation and salary as the employees in department 3001.



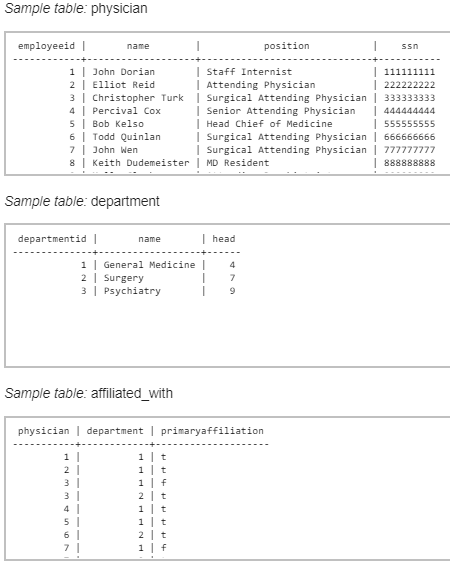
1. Write a query in SQL to obtain the name of the physicians who are the head of each department.



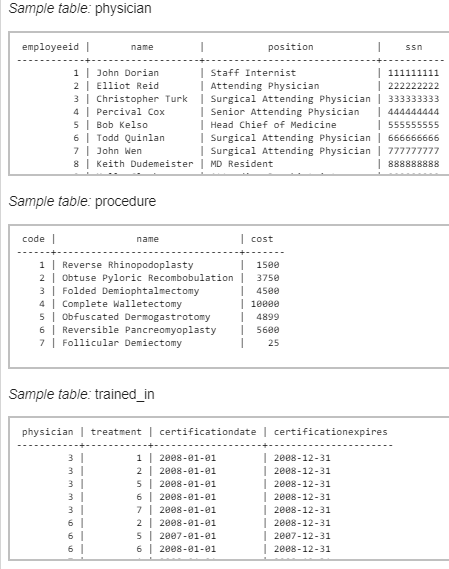
1. Write a query in SQL to count the number of patients who taken appointment with at least one physician.



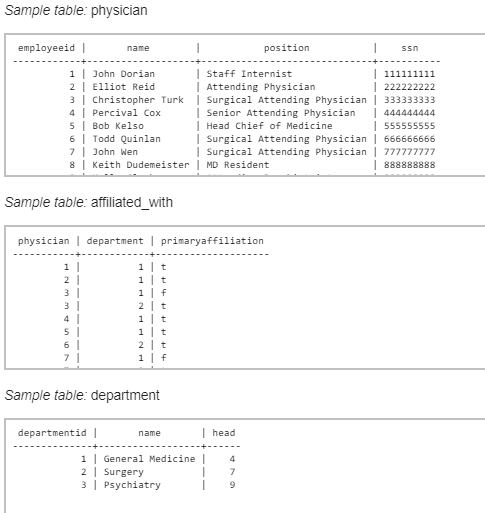
1. Write a query in SQL to obtain the name of the physician and the departments they are affiliated with.



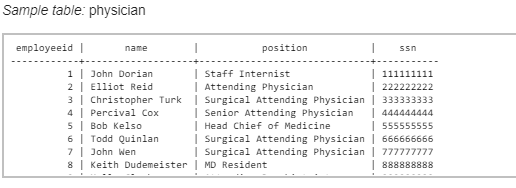
1. Write a query in SQL to obtain the name of the physicians who are trained for a special treatement.



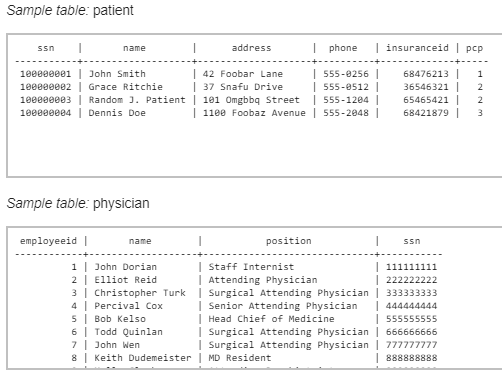
1. Write a query in SQL to obtain the name of the physicians with department who are yet to be affiliated.



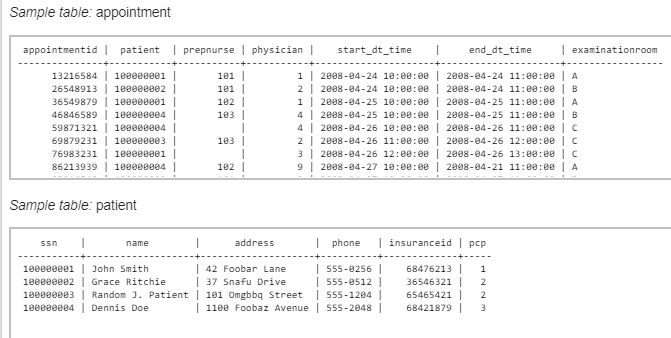
1. Write a query in SQL to obtain the name of the physicians who are not a specialized physician.



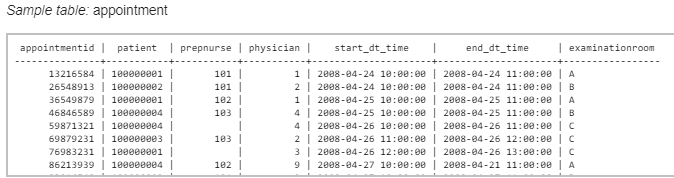
1. Write a query in SQL to obtain the name of the patients with their physicians by whom they got their preliminary treatement.



1. Write a query in SQL to find the name of the patients and the number of physicians they have taken appointment.



1. Write a query in SQL to count number of unique patients who got an appointment for examination room C.



1. Write a query in SQL to find the name of the patients and the number of the room where they have to go for their treatment.

