## Лабораторная работа 1

* Выборка данных из разных таблиц
* Описание структуры таблиц
* Выполнение арифметических вычислений и задание имён столбцов
  1. Команда SELECT содержит 4 ошибки. Укажите их.  
     SELECT employee\_id, last\_name,  
     sal x 12 ANNUAL SALARY  
     FROM employees;

select employee\_id, last\_name, salary\*12 "ANNUAL SALARY" from employees

|  |  |  |
| --- | --- | --- |
| **Name** | **Null?** | **Type** |
| DEPARTMENT\_ID | NOT NULL | NUMBER(4) |
| DEPARTMENT\_NAME | NOT NULL | VARCHAR(2) |
| MANAGER\_ID |  | NUMBER(6) |
| LOCATION\_ID |  | NUMBER(4) |

* 1. Покажите структуру таблицы DEPARTMENS desc departments
  2. . Выберите данные из неё. select \* from departments
  3. Покажите структуру таблицы EMPLOYEES. desc employees
  4. Составьте запрос для вывода имени каждого служащего, должности, даты найма и номера. Номер служащего должен быть первым.

select EMPLOYEE\_ID, LAST\_NAME, JOB\_ID, HIRE\_DATE from employees  
  
….

|  |
| --- |
| **JOB\_ID** |
| AC\_ACCOUNT |
| AC\_MGR |
| AD\_ASST |
| AD\_PRES |
| AD\_VP |
| IT\_PROG |
| MK\_MAN |
| MK\_REP |
| SA\_MAN |
| SA\_REP |
| ST\_CLERK |
| ST\_MAN |

* 1. Составьте запрос для вывода неповторяющихся должностей из таблицы EMPLOYEES.

SELECT DISTINCT JOB\_ID FROM EMPLOYEES

* 1. Выведите на экран имя, соединённое с идентификатором должности через запятую и пробел. Назовите новый столбец Employee and Title

SELECT LAST\_NAME||', '||JOB\_ID AS "Employee and Title" FROM EMPLOYEES

|  |
| --- |
| **Employee and Title** |
| King, AD\_PRES |
| Kochhar, AD\_VP |
| De Haan, AD\_VP |
| Hunold, IT\_PROG |
| Ernst, IT\_PROG |
| Lorentz, IT\_PROG |
| Mourgos, ST\_MAN |
| Rajs, ST\_CLERK |
| Davies, ST\_CLERK |
| Matos, ST\_CLERK |
| Vargas, ST\_CLERK |

* 1. Создайте запрос для вывода всех данных из таблицы EMPLOYEES. Разделите столбцы запятыми. Назовите столбец THE\_OUTPUT

SELECT EMPLOYEE\_ID||','||FIRST\_NAME||','||LAST\_NAME||','||EMAIL||','||PHONE\_NUMBER||','||HIRE\_DATE||','||JOB\_ID||','||SALARY||','||COMMISSION\_PCT||','||MANAGER\_ID||','||DEPARTMENT\_ID AS "THE\_OUTPUT" FROM EMPLOYEES

|  |
| --- |
| **THE\_OUTPUT** |
| 100,Steven,King,SKING,515.123.4567,AD\_PRES,17-JUN-87,24000,,90 |
| 101,Neena,Kochhar,NKOCHHAR,515.123.4568,AD\_VP,100,21-SEP-89,17000,,90 |
| 102,Lex,De Haan,LDEHAAN,515.123.4569,AD\_VP,100,13-JAN-93,17000,,90 |
| 103,Alexander,Hunold,AHUNOLD,590.423.4567,IT\_PROG,102,03-JAN-90,9000,,60 |
| 104,Bruce,Ernst,BERNST,590.423.4568,IT\_PROG,103,21-MAY-91,6000,,60 |
| 107,Diana,LorentzDLORENTZ,590.423.5567,IT\_PROG,103,07-FEB-99,4200,,60  ….. |

## Лабораторная работа 2

* Выборка данных и изменение последовательности вывода строк
* Ограничение количества возвращаемых строк с помощью предложения WHERE
* Сортировка строк с помощью предложения ORDER BY

1. Создайте запрос для вывода имени и заработной платы служащих, зарабатывающих более $12000. Сохраните команду SQL в файле lab2\_1.sql. выполните запрос.

|  |  |
| --- | --- |
| **LAST\_NAME** | **SALARY** |
| King | 24000 |
| Kochhar | 17000 |
| De Haan | 17000 |
| Hartstein | 13000 |

1. Создайте запрос для вывода фамилии и номера отдела служащего под номером 176

SELECT LAST\_NAME, DEPARTMENT\_ID FROM EMPLOYEES WHERE EMPLOYEE\_ID=176

|  |  |
| --- | --- |
| **LAST\_NAME** | **DEPARTMENT\_ID** |
| Taylor | 80 |

1. Измените файл lab2\_1.sql и выведите фамилии и оклады всех служащих, чей оклад не входит в диапазон от $5000 до $12000. сохраните команду SQL в файле lab2\_3.sql.

SELECT LAST\_NAME, SALARY FROM EMPLOYEES WHERE SALARY NOT BETWEEN 5000 AND 12000

|  |  |
| --- | --- |
| **LAST\_NAME** | **SALARY** |
| King | 24000 |
| Kochhar | 17000 |
| De Haan | 17000 |
| Lorentz | 4200 |
| Rajs | 3500 |
| Davies | 3100 |
| Matos | 2600 |
| Vargas | 2500 |
| Whalen | 4400 |
| Hartstein | 13000 |

|  |  |  |
| --- | --- | --- |
| **LAST\_NAME** | **JOB\_ID** | **HIRE\_DATE** |
| Matos | ST\_CLERK | 15-MAR-98 |
| Taylor | SA\_REP | 24-MAR-98 |

1. Выведите фамилию, идентификатор должности и дату начала работы всех служащих, нанятых в период с 20 февраля 1998 по 1 мая 1998 г. Отсортируйте данные в порядке возрастания даты найма.

SELECT LAST\_NAME, JOB\_ID, HIRE\_DATE FROM EMPLOYEES WHERE HIRE\_DATE BETWEEN '20.02.1998' AND '01.05.1998' ORDER BY HIRE\_DATE

1. Выведите фамилию и номер отдела всех служащих из отделов 20 и 50. Отсортируйте данные по фамилиям в алфавитном порядке.

|  |  |
| --- | --- |
| **LAST\_NAME** | **DEPARTMENT\_ID** |
| Davies | 50 |
| Fay | 20 |
| Hartstein | 20 |
| Matos | 50 |
| Mourgos | 50 |
| Rajs | 50 |
| Vargas | 50 |

SELECT LAST\_NAME, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID IN(20,50) ORDER BY LAST\_NAME

1. Измените файл lab2\_3.sql для вывода фамилий и окладов служащих отделов 20 и 50, зарабатывающих от $5000 до $12000. Назовите столбцы Employee и Mounthly Salary, соответственно. Вновь сохраните команду SQL в файле lab2\_6.sql. Выполните запрос.

SELECT LAST\_NAME "Employee", SALARY "Mounthly Salary" FROM EMPLOYEES WHERE department\_id in(20,50) AND SALARY BETWEEN 5000 AND 12000

|  |  |
| --- | --- |
| **EMPLOYEE** | **Mounthly Salary** |
| Mourgos | 5800 |
| Fay | 6000 |

1. Выведите фамилию и дату найма всех служащих, нанятых в 1994 г.

SELECT LAST\_NAME,JOB\_ID FROM EMPLOYEES WHERE MANAGER\_ID IS NULL

|  |  |
| --- | --- |
| **LAST\_NAME** | **HIRE\_DATE** |
| Higgins | 07-JUN-94 |
| Gietz | 07-JUN-94 |

1. Выведите фамилии и должности всех служащих, не имеющих менеджера.

SELECT LAST\_NAME, JOB\_ID FROM EMPLOYEES WHERE MANAGER\_ID IS NULL

|  |  |
| --- | --- |
| **LAST\_NAME** | **JOB\_ID** |
| King | AD\_PRES |

1. Выведите фамилию, и комиссионные всех служащих, зарабатывающих комиссионные. Отсортируйте данные в порядке убывания окладов и комиссионных.

SELECT LAST\_NAME, SALARY, COMMISSION\_PCT FROM EMPLOYEES WHERE COMMISSION\_PCT IS NOT NULL ORDER BY SALARY DESC, COMMISSION\_PCT DESC

|  |  |  |
| --- | --- | --- |
| **LAST\_NAME** | **SALARY** | **COMMISSION\_PCT** |
| Abel | 11000 | .3 |
| Zlotkey | 10500 | .2 |
| Taylor | 8600 | .2 |
| Grant | 7000 | .15 |

1. Выведите все фамилии служащих, в которых третья буква – *а.*

SELECT LAST\_NAME FROM EMPLOYEES WHERE LAST\_NAME LIKE '\_\_a%

|  |
| --- |
| **LAST\_NAME** |
| Grant |
| Whalen |

1. Выведите фамилии всех служащих, в которых есть буквы «*а*» и «*е*».

SELECT LAST\_NAME FROM EMPLOYEES WHERE LAST\_NAME LIKE '%a%' and LAST\_NAME LIKE '%e%'

|  |
| --- |
| **LAST\_NAME** |
| De Haan |
| Davies |
| Whalen |
| Hartstein |

1. запросите фамилии, должности и оклады всех служащих, работающих торговыми представителями (SA\_REP) или клерками на складе (ST\_CLERK) и с окладом, не равным $2500, $3500 и $7000.

SELECT LAST\_NAME, JOB\_ID, SALARY FROM EMPLOYEES WHERE JOB\_ID IN ('SA\_REP', 'ST\_CLERK') AND SALARY NOT IN(2500,3500,7000)

|  |  |  |
| --- | --- | --- |
| **LAST\_NAME** | **JOB\_ID** | **SALARY** |
| Davies | ST\_CLERK | 3100 |
| Matos | ST\_CLERK | 2600 |
| Abel | SA\_REP | 11000 |
| Taylor | SA\_REP | 8600 |

1. Измените lab2\_6.sql так, чтобы получить фамилии оклады и комиссионные всех служащих, у которых сумма комиссионных превышает 20%. Выполните запрос ещё раз. Сохраните запрос в файле lab2\_13.sql.

SELECT LAST\_NAME "Employee", SALARY "Mounthly Salary", COMMISSION\_PCT FROM EMPLOYEES WHERE COMMISSION\_PCT>.2

|  |  |  |
| --- | --- | --- |
| **Employee** | **Monthly Salary** | **COMMISSION\_PCT** |
| Zlotkey | 10500 | .2 |
| Taylor | 8600 | .2 |

## Лабораторная работа 3

* Составление запроса для вывода текущей даты.
* Составление запросов, требующих использования числовых, символьных функций и функций для работы с датами.
* Вычисление продолжительности работы служащего в месяцах и годах.

1. Напишите запрос для вывода текущей даты. Назовите столбец Date.

SELECT SYSDATE as "Date" FROM DUAL

|  |
| --- |
| **DATE** |
| 08-MAR-01 |

1. Выведите номер служащего, его фамилию, оклад и новый оклад, повышенный на 15% и округлённый до целого. Назовите столбец New Salary. Сохраните команду SQL в текстовом файле lab3\_2sql.
2. Выполните запрос из файла lab3\_2sql.

|  |  |  |  |
| --- | --- | --- | --- |
| **EMPLOYEE\_ID** | **LAST\_NAME** | **SALARY** | **New Salary** |
| 100 | King | 24000 | 27600 |
| 101 | Kochhar | 17000 | 19550 |
| 102 | De Haan | 17000 | 19550 |
| 103 | Hunold | 9000 | 10350 |
| 104 | Ernst | 6000 | 6900 |

………….

20 rows selected

SELECT EMPLOYEE\_ID, LAST\_NAME, SALARY, ROUND(SALARY\*1.15,0) AS "New Salary" FROM EMPLOYEES

1. Измените lab3\_2.sql. добавьте ещё один столбец, который будет содержать результат вычитания старого оклада из нового. Назовите столбец Increase. Сохраните изменённый запрос в lab3\_4.sql. Выполните запрос ещё раз.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EMPLOYEE\_ID** | **LAST\_NAME** | **SALARY** | **New Salary** | **Increase** |
| 100 | King | 24000 | 27600 | 3600 |
| 101 | Kochhar | 17000 | 19550 | 2550 |
| 102 | De Haan | 17000 | 19550 | 2550 |
| 103 | Hunold | 9000 | 10350 | 1350 |
| 104 | Ernst | 6000 | 6900 | 900 |

…………

20 rows selected

SELECT EMPLOYEE\_ID, LAST\_NAME, SALARY, ROUND(SALARY\*1.15,0) AS "New Salary", ROUND(SALARY\*1.15,0)-SALARY AS "Increase" FROM EMPLOYEES

1. Выведите фамилии служащих (первая буква каждой фамилии должна быть заглавной, а остальные - строчными) и длину каждой фамилии для тех служащих, фамилия которых начинается с символа J, A или М. Присвойте соответствующие заголовки столбцам.

SELECT INITCAP(LAST\_NAME) as "Name", LENGTH(LAST\_NAME) as "Lenght" FROM EMPLOYEES WHERE SUBSTR(LAST\_NAME,1,1) IN ('J','A','M')

|  |  |
| --- | --- |
| **Name** | **Lenght** |
| Abel | 4 |
| Matos | 5 |
| Mourgos | 7 |

## Лабораторная работа 4

* Составление запросов, требующих использования числовых, символьных функций и функций для работы с датами.
* Использование конкатенации с функциями.
* Составление запросов, нечувствительных к регистру символов, для проверки полезности символьных функций.
* Вычисление продолжительности работы служащего в месяцах и годах.
* Определение даты аттестации служащего.

1. Для каждого служащего выведите фамилию и вычислите количество месяцев со дня найма до настоящего времени, округлённое до ближайшего целого. Назовите столбец MONTH\_WORKED. Результаты отсортируйте по количеству отработанных месяцев. Округлите число до ближайшего целого.

|  |  |
| --- | --- |
| **LAST\_NAME** | **MONTH\_WORKED** |
| Zlotkey | 13 |
| Mourgos | 16 |
| Grant | 22 |
| Lorentz | 25 |
| Vargas | 32 |
| Taylor | 36 |
| Matos | 36 |
| Fay | 43 |

……………..

SELECT LAST\_NAME, ROUND(MONTHS\_BETWEEN(SYSDATE,HIRE\_DATE)) MONTH\_WORKED FROM EMPLOYEES ORDER BY MONTH\_WORKED

20 rows selected

1. Получите по каждому служащему отчёт в следующем виде:  
   *<фамилия> зарабатывает <оклад> в месяц, но желает <утроенный оклад>*. Назовите столбец Dream Salaries

|  |
| --- |
| **Dream Salaries** |
| King earns $24,000.00 monthly but wants $72,000.00 |
| Kochhar earns $17,000.00 monthly but wants $51,000.00 |
| De Haan earns $17,000.00 monthly but wants $51,000.00 |
| Hunold earns $9,000.00 monthly but wants $27,000.00 |
| Ernst earns $6,000.00 monthly but wants $18,000.00 |
| Lorentz earns $4,200.00 monthly but wants $12,600.00 |
| Mourgos earns $5,800.00 monthly but wants $17,400.00 |
| Rajs earns $3,500.00 monthly but wants $10,500.00 |
| Davies earns $3,100.00 monthly but wants $9,300.00 |

……………

SELECT LAST\_NAME||' earns '||TO\_CHAR(SALARY, '$99,999.00')||' monthly but wants '||TO\_CHAR(SALARY\*3, '$99,999.00') AS "Dream Salaries" FROM EMPLOYEES

20 rows selected

1. Напишите запрос для вывода фамилий и окладов всех служащих. Назовите выходной столбец SALARY. Длина столбца SALARY – 15 символов с заполнением символом $.

|  |  |
| --- | --- |
| **LAST\_NAME** | **SALARY** |
| King | $$$$$$$$$$24000 |
| Kochhar | $$$$$$$$$$17000 |
| De Haan | $$$$$$$$$$17000 |
| Hunold | $$$$$$$$$$$9000 |
| Ernst | $$$$$$$$$$$6000 |
| Lorentz | $$$$$$$$$$$4200 |
| Mourgos | $$$$$$$$$$$5800 |
| Rajs | $$$$$$$$$$$3500 |
| Davies | $$$$$$$$$$$3100 |
| Matos | $$$$$$$$$$$2600 |

………………

SELECT LAST\_NAME, LPAD(SALARY,15,'$') SALARY FROM EMPLOYEES

20 rows selected

1. Для каждого служащего выведите фамилию, дату найма и дату пересмотра зарплаты, которая приходится на первый понедельник после 6 месяцев работы. Назовите столбец REVIEW. Формат даты при выводе имеет вид: “Monday, the Thirty-First of MM, YYYY”.

|  |  |  |
| --- | --- | --- |
| **LAST\_NAME** | **HIRE\_DATE** | **REVIEW** |
| King | 17-JUN-87 | Monday, the Thirty-First of December, 1987 |
| Kochhar | 21-SEP-89 | Monday, the Thirty-Sixth of March, 1990 |
| De Haan | 13-JAN-93 | Monday, the Nineteenth of July, 1993 |
| Hunold | 03-JAN-90 | Monday, the Ninth of July, 1990 |

…………………

SELECT LAST\_NAME, HIRE\_DATE, TO\_CHAR(NEXT\_DAY(ADD\_MONTHS(HIRE\_DATE,6), 1), 'Day, "the" Ddspth "of" Month, YYYY') REVIEW FROM EMPLOYEES

20 rows selected

1. По каждому служащему выведите фамилию, дату найма и день недели, когда он был нанят на работу. Назовите последний столбец DAY. Отсортируйте результаты по датам.

|  |  |  |
| --- | --- | --- |
| **LAST\_NAME** | **HIRE\_DATE** | **DAY** |
| Grant | 24-MAY-99 | MONDAY |
| Ernst | 21-MAY-91 | TUESDAY |
| Mourgos | 16-NOV-99 | TUESDAY |
| Taylor | 24-MAR-98 | TUESDAY |
| Rajs | 17-OCT-95 | TUESDAY |
| Gietz | 07-JUN-94 | TUESDAY |
| Higgins | 07-JUN-94 | TUESDAY |
| King | 17-JUN-87 | WEDNESDAY |
| De Haan | 13-JAN-93 | WEDNESDAY |

………………

SELECT LAST\_NAME, HIRE\_DATE, TO\_CHAR(HIRE\_DATE, 'DAY') DAY FROM EMPLOYEES ORDER BY TO\_CHAR(HIRE\_DATE, 'D')

20 rows selected

1. Напишите запрос для вывода фамилии и суммы комиссионных каждого служащего. Если служащий не зарабатывает комиссионных, укажите в столбце «No Commission». Назовите столбец COMM.

|  |  |
| --- | --- |
| **LAST\_NAME** | **COMM** |
| King | No Commission |
| Kochhar | No Commission |
| De Haan | No Commission |
| Hunold | No Commission |
| Ernst | No Commission |
| Lorentz | No Commission |
| Mourgos | No Commission |
| Rajs | No Commission |
| Davies | No Commission |
| Matos | No Commission |
| Vargas | No Commission |
| Zlotkey | .2 |
| Abel | .3 |

…………….

SELECT LAST\_NAME, NVL(TO\_CHAR(COMMISSION\_PCT), 'No Commission') FROM EMPLOYEES

20 rows selected

1. Используя функцию DECODE, напишите запрос для отображения должности сотрудника и её разряда (grade). Разряд каждого типа должности JOB\_ID приведён в таблице.

|  |  |
| --- | --- |
| Должность | Разряд |
| *AD\_PRES* | A |
| *ST\_MAN* | B |
| *IT\_PROG* | C |
| *SA\_REP* | D |
| *ST\_CLERK* | E |
| Другая | 0 |

|  |  |
| --- | --- |
| **JOB\_ID** | **G** |
| AD\_PRES | A |
| AD\_VP | 0 |
| AD\_VP | 0 |
| IT\_PROG | C |
| IT\_PROG | C |
| IT\_PROG | C |
| ST\_MAN | B |
| ST\_CLERK | E |

…………………..

20 rows selected

SELECT job\_id,

DECODE(job\_id, 'AD\_PRES', 'A',

'ST\_MAN', 'B',

'IT\_PROG', 'C',

'SA\_REP', 'D',

'ST\_CLERK', 'E',

0) G

FROM employees;

1. Перепешите команду из предыдущего задания, используя синтаксис выражения CASE.

SELECT job\_id,

CASE job\_id WHEN 'AD\_PRES' THEN 'A'

WHEN 'ST\_MAN' THEN 'B'

WHEN 'IT\_PROG' THEN 'C'

WHEN 'SA\_REP' THEN 'D'

WHEN 'ST\_CLERK' THEN 'E'

ELSE '0' END G

FROM employees;

# Лабораторная работа 5

* Соединение таблиц с использованием эквисоединения.
* Выполнение внешних соединений и соединений таблицы с собой
* Включение дополнительных условий.

1. Напишите запрос для вывода фамилии, номера отдела и названия отдела дл всех служащих.

|  |  |  |
| --- | --- | --- |
| **LAST\_NAME** | **DEPARTMENT\_ID** | **DEPARTMENT\_NAME** |
| Whalen | 10 | Administration |
| Hartstein | 20 | Marketing |
| Fay | 20 | Marketing |
| Mourgos | 50 | Shipping |
| Rajs | 50 | Shipping |
| Davies | 50 | Shipping |
| Matos | 50 | Shipping |
| Vargas | 50 | Shipping |

……………..

SELECT LAST\_NAME, EMPLOYEES.DEPARTMENT\_ID, DEPARTMENT\_NAME FROM employees, departments WHERE EMPLOYEES.DEPARTMENT\_ID=DEPARTMENTS.DEPARTMENT\_ID

19 rows selected

1. Выведите список всех должностей в отдел 80 (должности в списке не должны повторяться) и местоположение отдела.

|  |  |
| --- | --- |
| **JOB\_ID** | **LOCATION\_ID** |
| SA\_MAN | 2500 |
| SA\_REP | 2500 |

SELECT DISTINCT(JOB\_ID), LOCATION\_ID FROM employees, departments WHERE EMPLOYEES.DEPARTMENT\_ID=DEPARTMENTS.DEPARTMENT\_ID AND DEPARTMENTS.DEPARTMENT\_ID=80

1. Напишите запрос для вывода фамилий, названия отдела, идентификатора местоположения отдела и города, в котором он находится, для всех служащих, зарабатывающих комиссионные.

|  |  |  |  |
| --- | --- | --- | --- |
| **LAST\_NAME** | **DEPARTMENT\_NAME** | **LOCATION\_ID** | **CITY** |
| Zlotkey | Sales | 2500 | Oxford |
| Abel | Sales | 2500 | Oxford |
| Taylor | Sales | 2500 | Oxford |

SELECT LAST\_NAME, DEPARTMENT\_NAME, DEPARTMENTS.LOCATION\_ID, CITY FROM employees, departments, locations WHERE EMPLOYEES.DEPARTMENT\_ID=DEPARTMENTS.DEPARTMENT\_ID AND DEPARTMENTS.LOCATION\_ID=locations.location\_id AND COMMISSION\_PCT IS NOT NULL

1. Выведите фамилии всех служащих, содержащтих букву «а» (в строчном регистре), с названиями отделов. Сохраните свою команду SQL в файле lab4\_4.sql.

|  |  |
| --- | --- |
| **LAST\_NAME** | **DEPARTMENT\_NAME** |
| Whalen | Administration |
| Hartstein | Marketing |
| Fay | Marketing |
| Rajs | Shipping |
| Davies | Shipping |
| Matos | Shipping |
| Vargas | Shipping |
| Taylor | Sales |
| Kochhar | Executive |
| De Haan | Executive |

10 rows selected.

SELECT LAST\_NAME, DEPARTMENT\_NAME FROM EMPLOYEES, departments WHERE EMPLOYEES.DEPARTMENT\_ID=DEPARTMENTS.DEPARTMENT\_ID AND LAST\_NAME LIKE '%a%'

SELECT LAST\_NAME, DEPARTMENT\_NAME FROM EMPLOYEES JOIN departments USING (department\_id) WHERE LAST\_NAME LIKE '%a%'

1. Напишите запрос для вывода фамилии, должности, номера отдела и названия отдела всех служащих, работающих в городе Toronto.

|  |  |  |  |
| --- | --- | --- | --- |
| **LAST\_NAME** | **JOB\_ID** | **DEPARTMENT\_ID** | **DEPARTMENT\_NAME** |
| Hartstein | MK\_MAN | 20 | Marketing |
| Fay | MK\_REP | 20 | Marketing |

SELECT LAST\_NAME, JOB\_ID, DEPARTMENT\_ID, DEPARTMENT\_NAME FROM EMPLOYEES JOIN departments USING (department\_id) JOIN locations USING (location\_id) WHERE CITY='Toronto'

1. Выведите фамилии и номера всех служащих вместе с фамилиями и номерами их менеджеров. Назовите столбцы EMPLOYEE, Emp#, Manager и Mgr#. Сохраните свою команду SQL в файле lab4\_6.sql.

|  |  |  |  |
| --- | --- | --- | --- |
| **Employee** | **EMP#** | **Manager** | **Mgr#** |
| Kochhar | 101 | King | 100 |
| De Haan | 102 | King | 100 |
| Mourgos | 124 | King | 100 |
| Zlotkey | 149 | King | 100 |

…………….

SELECT e1.LAST\_NAME AS "Emloyee", e1.EMPLOYEE\_ID AS "Emp#", e2.LAST\_NAME AS "Manager", e2.employee\_id AS "Mgr#" FROM EMPLOYEES e1 JOIN EMPLOYEES e2 ON e1.manager\_id=e2.employee\_id

1. Измените файл lab4\_6.sql так, чтобы получить фамилии всех служащих, включая Кинга, который не имеет менеджера. Упорядочьте результат по возрастанию номера служащего. Сохраните изменённый запрос в текстовом файле lab4\_7.sql. Выполните запрос.

|  |  |  |  |
| --- | --- | --- | --- |
| **Employee** | **EMP#** | **Manager** | **Mgr#** |
| King | 100 |  |  |
| Kochhar | 101 | King | 100 |
| De Haan | 102 | King | 100 |
| Hunolds | 103 | De Haan | 102 |
| Ernst | 104 | Hunold | 103 |
| Lorentz | 107 | Hunold | 103 |

…………………………….

SELECT e1.LAST\_NAME AS "Emloyee", e1.EMPLOYEE\_ID AS "Emp#", e2.LAST\_NAME AS "Manager", e2.employee\_id AS "Mgr#" FROM EMPLOYEES e1 LEFT JOIN EMPLOYEES e2 ON e1.manager\_id=e2.employee\_id ORDER BY e1.employee\_id

1. Создайте запрос для вывода номера отдела, фамилии служащего и фамилий всех служащих, работающих в одном отделе с данным служащим. Дайте столбцам соответствующие имена.

|  |  |  |
| --- | --- | --- |
| **DEPARTMENT** | **EMPLOYEE** | **COLLEAGUE** |
| 20 | Fay | Hartstein |
| 20 | Hartstein | Fay |
| 50 | Davies | Matos |
| 50 | Davies | Mourgos |
| 50 | Davies | Rajs |
| 50 | Davies | Vargas |

…………………

SELECT e1.department\_id DEPARTMENT, e1.last\_name EMPLOYEE, e2.last\_name COLLEAGUE FROM employees e1 JOIN employees e2 ON e1.department\_id=e2.department\_id where e1.employee\_id != e2.employee\_id

1. Покажите структуру таблицы JOB\_GRADES. Создайте запрос для вывода фамилии, должности, названия отдела, оклада и категории (GRADE\_LEVEL) всех служащих.

|  |  |  |
| --- | --- | --- |
| **Name** | **Null?** | **Type** |
| GRADE\_LEVEL |  | VARCHAR2(3) |
| LOWEST\_SAL |  | NUMBER |
| HIGHEST\_SAL |  | NUMBER |

DESC JOB\_GRADES

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **LAST\_NAME** | **JOB\_ID** | **DEPARTMENT\_NAME** | **SALARY** | **GRA** |
| Matos | ST\_CLERK | Shipping | 2600 | A |
| Vargas | ST\_CLERK | Shipping | 2500 | A |
| Lorentz | IT\_PROG | IT | 4200 | B |
| Mourgos | ST\_MAN | Shipping | 5800 | B |
| Rajs | ST\_CLERK | Shipping | 3500 | B |
| Davies | ST\_CLERK | Shipping | 3100 | B |
| Whalen | AD\_ASST | Administration | 4400 | B |

SELECT LAST\_NAME, JOB\_ID, DEPARTMENT\_NAME, SALARY, GRADE\_LEVEL FROM EMPLOYEES JOIN departments

USING (department\_id), JOB\_GRADES WHERE SALARY BETWEEN lowest\_sal AND highest\_sal

1. Создайте запрос для вывода фамилий и дат найма всех служащих, нанятых после Davies.

|  |  |
| --- | --- |
| **LAST\_NAME** | **HIRE\_DATE** |
| Lorentz | 07-FEB-99 |
| Mourgos | 16-NOV-99 |
| Matos | 15\_MAR-98 |
| Vargas | 09-JUL-98 |
| Zlotkey | 29-JAN-00 |
| Taylor | 24-MAR-98 |
| Grant | 24-MAY-99 |
| Fay | 17-AUG-97 |

SELECT LAST\_NAME, HIRE\_DATE FROM EMPLOYEES WHERE HIRE\_DATE > (SELECT HIRE\_DATE FROM EMPLOYEES WHERE LAST\_NAME='Davies')

1. По всем служащим нанятым раньше своих менеджеров, выведите фамилии и даты найма самих служащих, а также фамилии и даты найма их менеджеров. Назовите столбцы Employee, Emp, Manager и Manager Hired.

|  |  |  |  |
| --- | --- | --- | --- |
| **Employee** | **Emp Hired** | **Manager** | **Mgr hired** |
| Whalen | 17-SEP-87 | Kochhar | 21-SEP-89 |
| Hunold | 03-JAN-90 | De Haan | 13-JAN-93 |
| Rajs | 17-OCT-95 | Mourgos | 16-NOV-99 |
| Davies | 29-JAN-97 | Mourgos | 16-NOV-99 |
| Matos | 15-MAR-98 | Mourgos | 16-NOV-99 |
| Vargas | 09-JUL-98 | Mourgos | 16-NOV-99 |
| Abel | 11-MAY-96 | Zlotkey | 29-JAN-00 |
| Taylor | 24-MAR-98 | Zlotkey | 29-JAN-00 |
| Grant | 24-MAY-99 | Zlotkey | 29-JAN-00 |

SELECT e1.LAST\_NAME AS "Emloyee", e1.HIRE\_DATE AS "Emp Hired", e2.LAST\_NAME AS "Manager", e2.HIRE\_DATE AS "Mgr hired" FROM EMPLOYEES e1 JOIN EMPLOYEES e2 ON e1.manager\_id=e2.employee\_id WHERE e1.HIRE\_DATE < e2.HIRE\_DATE

1. Выведите номера, наименования и местоположение всех отделов, а также количество работающих в них сотрудников. Обеспечьте вывод отделов в которых нет сотрудников.

|  |  |  |  |
| --- | --- | --- | --- |
| **DEPARTMENT\_ID** | **DEPARTMENT\_NAME** | **LOCATION\_ID** | **COUNT(E.EMPLOYEE\_ID)** |
| 10 | Administration | 1700 | 1 |
| 20 | Marketing | 1800 | 2 |
| 50 | Shipping | 1500 | 5 |
| 60 | IT | 1400 | 3 |
| 80 | Sales | 2500 | 3 |
| 90 | Executive | 1700 | 3 |
| 110 | Accounting | 1700 | 2 |
| 190 | Contracting | 1700 | 0 |

SELECT DEPARTMENT\_ID, DEPARTMENT\_NAME, LOCATION\_ID, NVL(CEMP,0) AS "COUNT(E.EMPLOYEE\_ID)" FROM departments LEFT JOIN (SELECT department\_id, count(employee\_id) CEMP FROM employees group by department\_id) using (department\_id)

1. Какие должности имеются в отделах Administration и Executive, а также сколько сотрудников занимают эти должности? Выведите первыми наиболее часто встречаемые должности.

|  |  |
| --- | --- |
| **JOB\_ID** | **FREQUENCY** |
| AD\_VP | 2 |
| AD\_ASST | 1 |
| AD\_PRES | 1 |

SELECT job\_id, count(job\_id) FREQUENCY FROM employees JOIN DEPARTMENTS USING (department\_id) where department\_name in ('Administration', 'Executive') group by job\_id order by FREQUENCY DESC

1. Покажите сотрудников, менеджеры которых имеют оклад выше $15000. выведите следующие сведения: фамилию сотрудника, фамилию менеджера, его оклад и уровень оклада (GRADE\_LEVEL).

|  |  |  |  |
| --- | --- | --- | --- |
| **LAST\_NAME** | **MANAGER** | **SALARY** | **GRA** |
| Kochhar | King | 24000 | E |
| De Haan | King | 24000 | E |
| Mourgos | King | 24000 | E |
| Zlotkey | King | 24000 | E |
| Hartstein | King | 24000 | E |
| Whalen | Kochhar | 17000 | E |
| Higgins | Kochhar | 17000 | E |
| Hunold | De Haan | 17000 | E |

SELECT e1.LAST\_NAME, e2.LAST\_NAME MANAGER, e2.salary, GRADE\_LEVEL GRA

FROM EMPLOYEES e1 JOIN EMPLOYEES e2 ON e1.manager\_id=e2.employee\_id, JOB\_GRADES

WHERE e2.SALARY BETWEEN lowest\_sal AND highest\_sal

AND e2.SALARY>15000

## Лабораторная работа 6

* Использование групповых функций
* Вывод данных по группам с помощью предложения GROUP BY
* Включение и исключение групп с помощью предложения HAVING

1. Напишите запрос для вывода самого высокого, самого низкого и среднего оклада по всем служащим, а также суммы всех окладов. Назовите столбцы Maximum, Minimum, Average и Sum. Округлите суммы до ближайшего целого значения. Сохраните свой запрос в файле lab5\_6.sql.

|  |  |  |  |
| --- | --- | --- | --- |
| **Maximum** | **Minimum** | **Sum** | **Average** |
| 24000 | 2500 | 175500 | 8775 |

SELECT MAX(SALARY) AS "Maximum", MIN(SALARY) AS "Minimum", ROUND(AVG(SALARY))AS "Average", SUM(SALARY) AS "Sum" FROM employees

1. Измените запрос в lab5\_6.sql. так, чтобы получить самый низкий, самый высокий и средний оклады, а также сумму окладов по каждой должности. Сохраните изменённый запрос в файле lab5\_5.sql. Выполните запрос из lab5\_5.sql.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **JOB\_ID** | **MAXIMUM** | **MINIMUM** | **SUM** | **AVERAGE** |
| AC\_ACCOUNT | 8300 | 8300 | 8300 | 8300 |
| AC\_MGR | 12000 | 12000 | 12000 | 12000 |
| AD\_ASST | 4400 | 4400 | 4400 | 4400 |
| AD\_PRES | 24000 | 24000 | 24000 | 24000 |
| AD\_VP | 17000 | 17000 | 34000 | 17000 |

SELECT JOB\_ID, MAX(SALARY) AS "Maximum", MIN(SALARY) AS "Minimum", ROUND(AVG(SALARY))AS "Average", SUM(SALARY) AS "Sum" FROM employees GROUP BY JOB\_ID

…………….

1. Напишите запрос для вывода должности и количества служащих, занимающих каждую должность.

|  |  |
| --- | --- |
| **JOB\_ID** | **COUNT(\*)** |
| AD\_ACCOUNT | 1 |
| AC\_MGR | 1 |
| AD\_ASST | 1 |
| AD\_PRES | 1 |
| AD\_VP | 2 |
| IT\_PROG | 3 |
| MK\_MAN | 1 |
| MK\_REP | 1 |
| SA\_MAN | 1 |
| SA\_REP | 3 |
| ST\_CLERK | 4 |
| ST\_MAN | 1 |

SELECT JOB\_ID, COUNT(EMPLOYEE\_ID) FROM EMPLOYEES GROUP BY JOB\_ID

1. Получите количество служащих, имеющих подчинённых, без их перечисления. Назовите столбец Number of Managers. Используйте столбец MANAGER\_ID для определения числа менеджеров.

|  |
| --- |
| **Number of Managers** |
| 8 |

SELECT COUNT(DISTINCT MANAGER\_ID) FROM EMPLOYEES

1. Напишите запрос для вывода разности между самым высоким и самым низким окладами. Назовите столбец DIFFERENCE

|  |
| --- |
| **DIFFERENCE** |
| 21500 |

SELECT (MAX(SALARY)-MIN(SALARY)) DIFFERENCE FROM employees

1. Напишите запрос для вывода номера каждого менеджера, имеющего подчинённых, и заработную плату самого низкооплачиваемого из его подчинённых. Исключите менеджеров для которых неизвестны их менеджеры. Исключите все группы, где минимальный оклад составляет менее $6000. Отсортируйте выходные строки в порядке убывания оклада.

|  |  |
| --- | --- |
| **MANAGER\_ID** | **MIN(SALARY)** |
| 102 | 9000 |
| 205 | 8300 |
| 149 | 7000 |

SELECT MANAGER\_ID, MIN(SALARY) FROM EMPLOYEES GROUP BY MANAGER\_ID

HAVING MANAGER\_ID IS NOT NULL AND MIN(SALARY)>6000

ORDER BY MIN(SALARY) DESC

1. Напишите запрос для вывода названия отдела, местоположения отдела, количества служащих и среднего оклада по этому отделу. Назовите столбцы Name, Location, Number of People и Salary. Округлите средний оклад до ближайшего целого значения.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Location** | **Number of People** | **Salary** |
| Accounting | 1700 | 2 | 10150 |
| Administration | 1700 | 1 | 4400 |
| Executive | 1700 | 3 | 19333.33 |
| IT | 1400 | 3 | 6400 |
| Marketing | 1800 | 2 | 9500 |
| Sales | 2500 | 3 | 10033.33 |
| Shipping | 1500 | 5 | 3500 |

SELECT DEPARTMENT\_NAME AS "Name", LOCATION\_ID AS "Location", NVL(COUNTEMP,0) AS "Number of People", SAL AS "Salary"

FROM departments LEFT JOIN

(SELECT department\_id, count(employee\_id) COUNTEMP, ROUND(AVG(SALARY)) SAL

FROM employees group by department\_id) using (department\_id)

1. Напишите запрос для вывода общего количества служащих и количества служащих, нанятых в 1995, 1996, 1997 и 1998 годах. Дайте соответствующие заголовки столбцам.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TOTAL** | **1995** | **1996** | **1997** | **1998** |
| 20 | 1 | 2 | 2 | 3 |

SELECT COUNT (\*) TOTAL,

SUM (DECODE (TO\_CHAR (hire\_date, 'YYYY'), 1995, 1)) "1995",

SUM (DECODE (TO\_CHAR (hire\_date, 'YYYY'), 1996, 1)) "1996",

SUM (DECODE (TO\_CHAR (hire\_date, 'YYYY'), 1997, 1)) "1997",

SUM (DECODE (TO\_CHAR (hire\_date, 'YYYY'), 1998, 1)) "1998"

FROM employees;

1. Напишите матричный запрос для вывода всех должностей и суммы заработной платы служащих, работающих в этой должности в отделах 20, 50, 80 и 90. Последний столбец должен содержать сумму заработной платы служащих этих отделов, занимающих каждую конкретную должность. Дайте столбцам соответствующие заголовки.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Job** | **Dept20** | **Dept50** | **Dept80** | **Dept90** | **Total** |
| AC\_ACCOUNT |  |  |  |  | 8300 |
| AC\_MGR |  |  |  |  | 12000 |
| AD\_ASST |  |  |  |  | 4400 |
| AD\_PRES |  |  |  | 24000 | 24000 |
| AD\_VP |  |  |  | 34000 | 34000 |
| IT\_PROG |  |  |  |  | 19200 |

SELECT job\_id "Job",

nvl(SUM(DECODE(department\_id,20,salary)),0) "Dept20",

nvl(SUM(DECODE(department\_id,50,salary)),0) "Dept50",

nvl(SUM(DECODE(department\_id,80,salary)),0) "Dept80",

nvl(SUM(DECODE(department\_id,90,salary)),0) "Dept90",

nvl(SUM(DECODE(department\_id,20,salary)),0)+

nvl(SUM(decode(department\_id,50,salary)),0)+

nvl(SUM(decode(department\_id,80,salary)),0)+

nvl(SUM(DECODE(department\_id,90,salary)),0) "Total"

FROM employees group by job\_id;

……………………………

## Лабораторная работа 7

* + Создание подзапросов для выборки данных по неизвестным критериям
  + Использование подзапросов для выявления значений, существующих в одном наборе данных и отсутствующих в другом

1. Создайте запрос для вывода фамилии и даты найма каждого служащего, работающего в одном отделе с Zlotkey. Исключите Zlotkey из выходных данных.

SELECT LAST\_NAME, HIRE\_DATE FROM EMPLOYEES

WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE LAST\_NAME='Zlotkey')

AND LAST\_NAME !='Zlotkey'

1. Создайте запрос для вывода номеров и фамилий всех служащих, оклад которых выше среднего. Отсортируйте данные в порядке увеличения окладов.

SELECT EMPLOYEE\_ID, LAST\_NAME, SALARY FROM EMPLOYEES

WHERE SALARY > (SELECT AVG(SALARY) FROM EMPLOYEES) ORDER BY SALARY

1. Создайте запрос для вывода номеров и фамилий всех служащих, работающих в одном отделе с любым служащим, фамилия которого содержит букву «u». Сохраните свой запрос в тестовом файле lab6\_3.sql. Выполните запрос.

SELECT EMPLOYEE\_ID, LAST\_NAME FROM EMPLOYEES

WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE LAST\_NAME LIKE '%u%')

1. Создайте запрос для вывода фамилии, номера отдела и должности каждого служащего, идентификатор местоположения отдела которого равен 1700.

SELECT LAST\_NAME, DEPARTMENT\_ID, JOB\_ID FROM EMPLOYEES

WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM DEPARTMENTS WHERE LOCATION\_ID='1700')

1. Получите список фамилий и окладов всех служащих, подчинённых Кингу.



SELECT LAST\_NAME, SALARY FROM EMPLOYEES

WHERE MANAGER\_ID IN (SELECT EMPLOYEE\_ID FROM EMPLOYEES WHERE LAST\_NAME='King')

1. Получите номер отдела, фамилию и должность для каждого служащего, работающего в администрации (department\_name = ‘Executive’).

SELECT DEPARTMENT\_ID, LAST\_NAME, JOB\_ID FROM EMPLOYEES

WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM DEPARTMENTS WHERE DEPARTMENT\_NAME='Executive')

1. Измените lab6\_3.sql для вывода номеров, фамилий и окладов всех служащих, которые работают в одном отделе с любым служащим с буквой «u» в фамилии и оклады которых превышают средний. Сохраните новый запрос в файле lab6\_7.sql. Выполните его.



SELECT EMPLOYEE\_ID, LAST\_NAME, SALARY FROM EMPLOYEES

WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE LAST\_NAME LIKE '%u%')

AND SALARY > (SELECT AVG(SALARY) FROM employees)

1. Покажите номер отдела с наивысшей средней заработной платой и наименьший оклад работающего в нём сотрудника.

SELECT DEPARTMENT\_ID, MIN(SALARY) FROM EMPLOYEES WHERE DEPARTMENT\_ID IN

(SELECT department\_id FROM (SELECT DEPARTMENT\_ID , AVG(SALARY) S FROM employees GROUP BY DEPARTMENT\_ID )

WHERE S=(SELECT MAX(AVG(salary))FROM employees GROUP BY DEPARTMENT\_ID))

GROUP BY DEPARTMENT\_ID

1. Выведите номера , наименования и местоположения отделов, в которых не работают торговые представители (job\_id = ‘SA\_REP’).

## SELECT DEPARTMENT\_ID, DEPARTMENT\_NAME, MANAGER\_ID, LOCATION\_ID FROM DEPARTMENTS

## WHERE DEPARTMENT\_ID NOT IN

## (SELECT DISTINCT NVL(DEPARTMENT\_ID, 0) FROM EMPLOYEES WHERE JOB\_ID ='SA\_REP')

## Лабораторная работа 8

* + Создание запросов, в которых используются операторы SET.
  + Альтернативные методы соединения.

1. Используя оператор SET, выведите номера отделов (department\_id), в которых нет служащих с идентификатором должности (job\_id) ST\_CLERK. 

SELECT DEPARTMENT\_ID FROM DEPARTMENTS

MINUS SELECT DISTINCT DEPARTMENT\_ID FROM EMPLOYEES WHERE JOB\_ID ='ST\_CLERK'

1. Используя оператор SET, выведите идентификаторы и наименования стран, в которых не располагаются отделы компании. 

SELECT COUNTRY\_ID, COUNTRY\_NAME FROM COUNTRIES WHERE COUNTRY\_ID IN

(SELECT COUNTRY\_ID FROM COUNTRIES

MINUS SELECT DISTINCT COUNTRY\_ID FROM LOCATIONS JOIN DEPARTMENTS USING (LOCATION\_ID))

1. Используя оператор SET, выведите список должностей отделов 10, 50 и 20 в таком же порядке отделов. Выведите столбцы job\_id и department\_id. 

SELECT DISTINCT JOB\_ID, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID=10

UNION ALL SELECT DISTINCT JOB\_ID, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID=50

UNION ALL SELECT DISTINCT JOB\_ID, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID=20

1. Выведите номер сотрудника и идентификатор его должности, если его текущая должность совпадает с той, которую он уже занимал, работая в компании. 

SELECT EMPLOYEE\_ID, JOB\_ID FROM EMPLOYEES

INTERSECT SELECT EMPLOYEE\_ID, JOB\_ID FROM JOB\_HISTORY

1. Напишите составной запрос, который выводит следующее:
   * Номера и отделы всех сотрудников из таблицы EMPLOYEES, независимо от того, относятся ли они к какому-то отделу или нет;
   * Номера и наименования всех отделов из таблицы DEPARTMENTS, независимо от того, есть ли в них сотрудники или нет.

SELECT EMPLOYEE\_ID, DEPARTMENT\_ID, TO\_CHAR (NULL) DEPARTMENT\_NAME FROM EMPLOYEES

UNION SELECT TO\_NUMBER (NULL), DEPARTMENT\_ID, DEPARTMENT\_NAME FROM DEPARTMENTS

# Лабораторная работа 9

* + Создание многостолбцовых подзапросов
  + Написание связанных подзапросов
  + Использование оператора EXIST
  + Использование скалярных подзапросов

1. Создайте запрос для вывода фамилии, номера отдела и оклада всех служащих, чей номер отдела и оклад совпадают с номером отдела и окладом любого служащего, зарабатывающего комиссионные. 

SELECT LAST\_NAME, DEPARTMENT\_ID, SALARY FROM EMPLOYEES

WHERE (DEPARTMENT\_ID, SALARY) IN

(SELECT DEPARTMENT\_ID, SALARY FROM EMPLOYEES WHERE COMMISSION\_PCT > 0)

1. Выведите фамилию, название отдела и оклад всех служащих, чей оклад и комиссионные совпадают с окладом и комиссионными любого служащего, работающего в отделе, идентификатор местоположения которого Location\_ID = 1700. 

SELECT LAST\_NAME, DEPARTMENT\_NAME, SALARY FROM EMPLOYEES JOIN DEPARTMENTS USING(DEPARTMENT\_ID)

WHERE (SALARY, NVL(COMMISSION\_PCT,0)) IN

(SELECT SALARY, NVL(COMMISSION\_PCT,0) FROM EMPLOYEES JOIN DEPARTMENTS USING(DEPARTMENT\_ID) WHERE Location\_ID = 1700)

1. Создайте запрос для вывода фамилии, даты найма и оклада всех служащих, которые получают такой же оклад и такие же комиссионные, как Kochhar. Не выводите данные о сотруднике Kochhar.

SELECT LAST\_NAME, HIRE\_DATE, SALARY FROM EMPLOYEES

WHERE (SALARY, NVL(COMMISSION\_PCT,0)) IN

(SELECT SALARY, NVL(COMMISSION\_PCT,0) FROM EMPLOYEES WHERE LAST\_NAME='Kochhar')

AND LAST\_NAME!='Kochhar'

1. Выведите фамилию, должность и оклад всех служащих, оклад которых превышает оклад каждого клерка торгового менеджера (JOB\_ID = ‘SA\_MAN’). Отсортируйте результаты по убыванию окладов.

SELECT LAST\_NAME, JOB\_ID, SALARY FROM EMPLOYEES

WHERE SALARY>(SELECT MAX(SALARY) FROM EMPLOYEES WHERE JOB\_ID='SA\_MAN')

ORDER BY SALARY DESC

1. Выведите номера, фамилии и отделы служащих, живущих в городах, названия которых начинаются с буквы Т. 

SELECT EMPLOYEE\_ID, LAST\_NAME, DEPARTMENT\_ID FROM EMPLOYEES

WHERE DEPARTMENT\_ID IN

(SELECT DEPARTMENT\_ID FROM DEPARTMENTS JOIN LOCATIONS USING (LOCATION\_ID) WHERE CITY LIKE 'T%')

1. Напишите запрос для нахождения всех сотрудников, которые зарабатывают больше среднего оклада по их отделу. Выведите фамилию, оклад, номер отдела и средний оклад по отделу. Отсортируйте результаты по средней зарплате. Используйте псевдонимы для выбираемых столбцов.

SELECT LAST\_NAME ENAME, SALARY, DEPARTMENT\_ID DEPTNO, DEPT\_AVG FROM EMPLOYEES

JOIN (SELECT DEPARTMENT\_ID, AVG(SALARY) DEPT\_AVG FROM EMPLOYEES GROUP BY DEPARTMENT\_ID) USING (DEPARTMENT\_ID)

WHERE SALARY>DEPT\_AVG

ORDER BY DEPT\_AVG

1. Найдите всех сотрудников, не являющихся руководителями. Выполните это с помощью оператора NOT EXIST. 

SELECT LAST\_NAME FROM EMPLOYEES e

WHERE NOT EXISTS(SELECT NULL FROM EMPLOYEES WHERE manager\_id = e.employee\_id)

1. Может ли это же быть сделано с помощью оператора NOT IN?

SELECT LAST\_NAME FROM EMPLOYEES

WHERE EMPLOYEE\_ID NOT IN(SELECT NVL(MANAGER\_ID,0) FROM EMPLOYEES)

1. Выведите фамилии сотрудников, зарабатывающих меньше среднего оклада по их отделу. 

SELECT LAST\_NAME FROM EMPLOYEES

JOIN (SELECT DEPARTMENT\_ID, AVG(SALARY) DEPT\_AVG FROM EMPLOYEES GROUP BY DEPARTMENT\_ID) USING (DEPARTMENT\_ID)

WHERE SALARY<DEPT\_AVG

1. Выведите фамилии сотрудников, у которых есть коллеги по отделу, которые были приняты на работу позже, но имеют более высокий оклад. 

SELECT LAST\_NAME FROM EMPLOYEES E

WHERE EXISTS (SELECT NULL FROM EMPLOYEES

WHERE SALARY>E.SALARY AND DEPARTMENT\_ID=E.DEPARTMENT\_ID AND HIRE\_DATE>E.HIRE\_DATE)

1. Выведите номера, фамилии и наименования отделов всех сотрудников. Используйте скалярный подзапрос в команде SELECT для вывода наименований отделов. 

## SELECT EMPLOYEE\_ID, LAST\_NAME, (SELECT DEPARTMENT\_NAME FROM DEPARTMENTS WHERE DEPARTMENT\_ID=E.DEPARTMENT\_ID) DEPARTMENT FROM EMPLOYEES E

## Лабораторная работа 10

* + Отличие иерархических запросов от неиерархических
  + Обход дерева
  + Получение отчёта с отступом с помощью псевдостолбца LEVEL
  + Отсечение структур дерева
  + Сортировка выходных данных

1. Напишите отчёт, в котором отражена структура отдела, которым руководит Mourgos. Выведите фамилии, оклады и номер отдела сотрудников. 

SELECT LAST\_NAME, SALARY, DEPARTMENT\_ID FROM employees

START WITH last\_name = 'Mourgos'

CONNECT BY manager\_id = PRIOR employee\_id

1. Создайте отчёт, который показывает иерархию менеджеров, которым подчиняется сотрудник Lorentz. Выведите сначала менеджера, перед которым непосредственно отчитывается Lorentz. 

SELECT LAST\_NAME FROM EMPLOYEES WHERE LAST\_NAME!='Lorentz'

START WITH last\_name = 'Lorentz'

CONNECT BY PRIOR manager\_id = employee\_id

1. Создайте отчёт с отступом, в котором отражается иерархия управления, начиная с сотрудника по фамилии Kochhar. Выведите фамилии, номера менеджеров и номера отделов сотрудников. Назовите столбцы как показано в примере выходных результатов.

SELECT LPAD(last\_name, LENGTH(last\_name)+(LEVEL\*2)-2,'\_') NAME, MANAGER\_ID MGR, DEPARTMENT\_ID DEPTNO

FROM employees

START WITH last\_name='Kochhar'

CONNECT BY PRIOR employee\_id=manager\_id 

1. Создайте отчёт, отражающий иерархию управления компанией. Начните с сотрудника самого высокого уровня и исключите из выходных данных всех служащих с идентификатором должности IT\_PROG, а также сотрудника De Haan и всех, кто перед ним отчитывается.

SELECT LAST\_NAME, EMPLOYEE\_ID, MANAGER\_ID

FROM employees WHERE JOB\_ID!='IT\_PROG' AND LAST\_NAME!='De Haan'

START WITH manager\_id IS NULL

CONNECT BY PRIOR employee\_id=manager\_id AND LAST\_NAME!='De Haan' 