

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**КЕМЕРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**  
**Институт цифры**

**ОТЧЕТ**  
**О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ 1**

**«Задания для повторения SQL»**

студента 1 курса, группы ПИМ-231

**Давыдова Игоря Павловича**

Направление 09.04.03 – «Прикладная информатика»

Руководитель:  
Доцент Завозкин С. Ю.

Работа защищена

« \_\_\_\_\_ »  
“ \_\_\_\_\_ ” \_\_\_\_\_ 2023г.

Кемерово 2023

## Задание

1. Для каждого служащего вычислите количество месяцев со дня начала работы до настоящего времени. Результаты отсортируйте по количеству отработанных месяцев. Количество месяцев округлите до ближайшего целого.
2. Получите список фамилий и заработной платы всех служащих отделов 31, 42, и 50, месячный заработок которых не лежит в интервале от 1000 до 2500. Назовите столбец "Employee Name", а столбец заработной платы – "MONTHLY SALARY".
3. Для каждого вида товара, заказанного, по крайней мере, три раза, выведите номер этого товара и количество заказов на него. Столбец с количеством заказов на товар озаглавьте "Times Ordered". Отсортируйте данные по номерам заказанных товаров.
4. Для каждого заказа с общим количеством заказанных товаров 100 или более выведите номер заказа и общее количество заказанных товаров в нем. (Если, например, заказ номер 99 содержит заказ на один товар в количестве 30, а на другой – в количестве 75, то общее количество заказанных товаров равно 105).
5. Для каждого заказчика, общая сумма заказа которого превышает 100 000, выведите наименование заказчика, заказанные им товары, их количество.
6. Выведите номер служащего, имя, фамилию и имя пользователя для всех служащих, заработная плата которых выше средней.
7. Выведите наименование и краткое описание каждого товара, который ни разу не был заказан в сентябре 1992 г.
8. Выведите фамилию каждого торгового представителя в регионах 1 и 2, наименования их клиентов и итоговые суммы заказов каждого клиента.
9. Для каждого торгового представителя, относящегося к указанному менеджеру, выдать сумму заказов.
10. Выдать фамилии и названия отделов всех сотрудников, которые оформляли заказы на суммы большие средней суммы заказов за все время.

## Ход работы

1. Для каждого служащего вычислите количество месяцев со дня начала работы до настоящего времени. Результаты отсортируйте по количеству отработанных месяцев. Количество месяцев округлите до ближайшего целого.

```
SELECT
    FIRST_NAME, LAST_NAME, round(months_between(SYSDATE,
START_DATE)) AS "Months"
FROM
    S_EMP
ORDER BY
    "Months"
```

FIRST_NAME VARCHAR2(25)	LAST_NAME VARCHAR2(25)	Months NUMBER
Elena	Maduro	379
Antoinette	Catchpole	379
Mai	Nguyen	380
Vikram	Patel	385
Chad	Newman	386
Midori	Nagayamn	387
Alexander	Markarian	388
Sylvie	Schwartz	388
Bela	Dancs	390
Marta	Havel	391

2. Получите список фамилий и заработной платы всех служащих отделов 31, 42, и 50, месячный заработок которых не лежит в интервале от 1000 до 2500. Назовите столбец "Employee Name", а столбец заработной платы – "MONTHLY SALARY".

```
SELECT
    LAST_NAME AS "Employee name", SALARY AS "Monthly salary",
DEPT_ID
FROM
    S_EMP
WHERE
    (DEPT_ID = 31 OR DEPT_ID = 42 OR DEPT_ID = 50) AND SALARY NOT
BETWEEN 1000 and 2500
```

Employee name VARCHAR2(25)	Monthly salary NUMBER(11, 2)	DEPT_ID NUMBER(7)
Patel	795	42

3. Для каждого вида товара, заказанного, по крайней мере, три раза, выведите номер этого товара и количество заказов на него. Столбец с количеством заказов на товар озаглавьте "Times Ordered". Отсортируйте данные по номерам заказанных товаров.

```
SELECT
    S_ITEM.PRODUCT_ID AS "Номер товара",
    COUNT(DISTINCT S_ITEM.ORD_ID) AS "Times Ordered"
FROM
    S_ITEM
GROUP BY
    S_ITEM.PRODUCT_ID
HAVING
    COUNT(DISTINCT S_ITEM.ORD_ID) >= 3
ORDER BY
    S_ITEM.PRODUCT_ID;
```

Номер товара NUMBER(7)	Times Ordered	
30321	4	
50536	3	
50530	3	
50418	3	
50417	3	
32779	3	
30421	3	
20512	3	
20510	3	
20201	3	
20108	3	
20106	3	

4. Для каждого заказа с общим количеством заказанных товаров 100 или более выведите номер заказа и общее количество заказанных товаров в нем. (Если, например, заказ номер 99 содержит заказ на один товар в количестве 30, а на другой – в количестве 75, то общее количество заказанных товаров равно 105).

```
SELECT
    S_ORD.ID AS "Номер заказа",
    SUM(S_ITEM.QUANTITY) AS "Количество товаров"
FROM
    S_ORD
INNER JOIN
    S_ITEM ON S_ORD.ID = S_ITEM.ORD_ID
```

```

GROUP BY
  S_ORD.ID
HAVING
  SUM(S_ITEM.QUANTITY) >= 100;

```

Номер заказа NUMBER(7)	Количество товаров NUMBER
97	1050
99	165
100	3100
102	322
104	114
106	192
107	332
108	321
109	2343

5. Для каждого заказчика, общая сумма заказа которого превышает 100 000, выведите наименование заказчика, заказанные им товары, их количество.

```

SELECT C.NAME AS "Customer Name", I.NAME AS "Product Name",
SUM(OI.QUANTITY) AS "Quantity Ordered"
FROM S_CUSTOMER C
INNER JOIN S_ORD O ON C.ID = O.CUSTOMER_ID
INNER JOIN S_ITEM OI ON O.ID = OI.ORD_ID
INNER JOIN S_PRODUCT I ON OI.PRODUCT_ID = I.ID
GROUP BY C.NAME, I.NAME
HAVING SUM(O.TOTAL) > 100000;

```

Customer Name VARCHAR2(50)	Product Name VARCHAR2(50)	Quantity Ordered NUMBER
Womansport	Himalaya Bicycle	600
Kuhns Sports	Junior Soccer Ball	50
Kuhns Sports	Grand Prix Bicycle	75
Hamada Sport	Ace Ski Boot	50
Hamada Sport	Winfield Bat	42
Big Johns Sports Emporium	Bunny Boot	150
Big Johns Sports Emporium	Ace Ski Boot	600
Big Johns Sports Emporium	Winfield Bat	50

6. Выведите номер служащего, имя, фамилию и имя пользователя для всех служащих, заработная плата которых выше средней.

```
SELECT E.ID AS "Employee ID", E.FIRST_NAME, E.LAST_NAME,
E.USERID
FROM S_EMP E
WHERE E.SALARY > (SELECT AVG(SALARY) FROM S_EMP);
```

Employee ID NUMBER(7)	FIRST_NAME VARCHAR2(25)	LAST_NAME VARCHAR2(25)	USERID VARCHAR2(8)
1	Carmen	Velasquez	cvelasqu
2	LaDoris	Ngao	Ingao
3	Midori	Nagayamn	mnagayam
4	Mark	Quick-To-See	mquickto
5	Audry	Ropeburn	aropebur
7	Robertn	Menchu	rmenchu

7. Выведите наименование и краткое описание каждого товара, который ни разу не был заказан в сентябре 1992 г.

```
SELECT P.NAME AS "Product Name", P.SHORT_DESC AS "Product
Description"
FROM S_PRODUCT P
WHERE P.ID NOT IN (SELECT DISTINCT OI.PRODUCT_ID
FROM S_ITEM OI
INNER JOIN S_ORD O ON OI.ORD_ID = O.ID
WHERE EXTRACT(YEAR FROM O.DATE_ORDERED) = 1992
AND EXTRACT(MONTH FROM O.DATE_ORDERED) = 9);
```

Product Name VARCHAR2(50)	Product Description VARCHAR2(255)
Pro Ski Boot	Advanced ski boot
Safe-T Helmet	Bicycle helmet
Prostar 20 Pound Weight	Twenty pound weight
Prostar 50 Pound Weight	Fifty pound weight

8. Выведите фамилию каждого торгового представителя в регионах 1 и 2, наименования их клиентов и итоговые суммы заказов каждого клиента.

```
SELECT E.LAST_NAME AS "Sales Rep Last Name", C.NAME AS "Customer
Name", SUM(O.TOTAL) AS "Total Order Amount"
FROM S_EMP E
INNER JOIN S_CUSTOMER C ON E.ID = C.SALES_REP_ID
```

```

INNER JOIN S_ORD O ON C.ID = O.CUSTOMER_ID
WHERE C.REGION_ID IN (1, 2)
GROUP BY E.LAST_NAME, C.NAME;

```

Sales Rep Last Name VARCHAR2(25)	Customer Name VARCHAR2(50)	Total Order Amount NUMBER
Magee	Womansport	603870
Magee	Unisports	84000
Magee	Big Johns Sports Emporium	1020935
Markarian	Futbol Sonora	16184

9. Для каждого торгового представителя, относящегося к указанному менеджеру, выдать сумму заказов.

```

SELECT E.MANAGER_ID, SUM(O.TOTAL) AS "Total Sales Amount"
FROM S_EMP E
INNER JOIN S_ORD O ON E.ID = O.SALES_REP_ID
WHERE E.MANAGER_ID = 8
GROUP BY E.MANAGER_ID;

```

MANAGER_ID NUMBER(7)	Total Sales Amount NUMBER
8	190785

10. Выдать фамилии и названия отделов всех сотрудников, которые оформляли заказы на суммы большие средней суммы заказов за все время.

```

SELECT E.LAST_NAME AS "Employee Last Name", D.NAME AS "Department Name"
FROM S_EMP E
INNER JOIN S_ORD O ON E.ID = O.SALES_REP_ID
INNER JOIN S_DEPT D ON E.DEPT_ID = D.ID
GROUP BY E.LAST_NAME, D.NAME
HAVING SUM(O.TOTAL) > (SELECT AVG(TOTAL) FROM S_ORD);

```

Employee Last Name VARCHAR2(25)	Department Name VARCHAR2(25)
Markarian	Operations
Magee	Sales

## **Заключение**

В ходе данной работы были рассмотрены и выполнены различные SQL-задания с использованием учебной базы данных, предоставленной в описании. Задания включали в себя запросы, направленные на извлечение информации из базы данных, фильтрацию данных, агрегацию и объединение таблиц.