Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Кафедра электронных вычислительных машин

Лабораторная работа №4

«Реализация SQL-запросов на простую выборку данных»

Вариант № 2 (Автосалон)

Выполнил Проверила:

студент группы 150503: Игнатович А.О.

Гаврилов Ф.Н.

Минск 2024

1. **Цель работы**

В ходе выполнения лабораторной работы необходимо выполнить создание простых запросов на выборку данных на языке SQL.

1. **Порядок выполнения работы**

1) Получить у преподавателя задания по вашей собственной схеме данных, созданной в лабораторной работе №2 и реализованной в виде таблиц в СУБД в лабораторной работе №3. Создать запросы по заданиям (по одному запросу на каждое задание).

2) Правила выполнения заданий:

* для каждого задания создать реализацию в виде одного оператора выборки, в котором НЕЛЬЗЯ использовать подзапросы и группировку данных (это еще будет в другой лабораторной работе);
* при использовании соединений нескольких таблиц обратить внимание на условие задания и сделать выбор между внутренним и внешним соединениями и их вариантами реализации;
* перед запуском запроса на выполнение, изучить данные в используемых запросом таблицах, и, если требуется, добавить в вашу схему необходимые новые данные, чтобы результат выборки был контролируемым и не пустым;
* выполнить запрос и проанализировать его результат – если есть расхождения между ожидаемыми данными и результатом запроса, то есть повод задуматься о проверке правильности выполнения этого задания.

3) Оформить отчет.

1. **Выполнение работы**

SQL-скрипт приведен в файле «lab4.sql».

Результаты выполнения скрипта приведены в файле «lab4.log».

Для связывания таблиц используются INNER JOIN, в котором сопоставляются первичный и внешний ключи, LEFT/RIGHT OUTER JOIN, в которых сохраняются кортежи, которые были бы утрачены при использовании внутреннего соединения, соответственно из левой/правой таблиц, которые не имеют соответствие в другой таблице, CROSS JOIN, в котором мы получаем декартово произведение записей таблиц, FULL OUTER JOIN, в котором мы получаем все пары записей из таблиц, независимо от наличия соответствий.

Для задания условия выборки данных используется WHERE.

Для сортировки строк используется ORDER BY.

Для обработки строк данных используются скалярные функции ROUND,TRUNC,INITCAP,TRANSLATE,TRIM,CONCAT,LOWER,SUBSTR,UPPER,LENGTH,TO\_CHAR.

Для объединения множеств строк используются UNION,INTERSECT,EXCEPT. Выполняются следующие запросы:

1. Выводится модель автомобиля, название его производителя, страна производителя, категория автомобиля, VIN номер автомобиля, цвет автомобиля. Выбираются автомобили определенного цвета, а также с буквой ‘a’ в названии. Данные отсортированы в алфавитном порядке по названию производителя.
2. Выводится дата доставки заказа, дата оформления заказа, статус заказа, имя клиента, номер телефона клиента в формате 80xx. Выбираются заказы с определенной датой оформления и имена клиентов с буквой ‘а’. Данные отсортированы по возрастанию по дате доставки, причем NULL значения в конце.
3. Выводится время работы по должности, зарплата с округлением, дата назначения на должность, имя и фамилия работника. Выбираются должности с зарплатой более 600 единиц. Данные отсортированы по убыванию по дате назначения на должность.
4. Выводится имя и фамилия клиента, оценка отзыва, имя производителя, название модели и год производства автомобиля, на который оставлен соответствующий отзыв. Выбираются отзывы с оценками 8, 9 и 10 соответственно. Данные отсортированы по убыванию по значению оценки отзыва.
5. Выводится имя и фамилия работника, его опыт работы, название занимаемой должности, описание должности, время работы по должности, модель автомобиля, на обслуживание которого назначен работник. Выбираются записи с временем работы по дожности более 8 часов и опытом работы работника более 1 года. Данные отсортированы по убыванию по значению опыта работы.
6. Выводится имя и фамилия работника, название производителя и модель соответствующего автомобиля, на обслуживание которого назначен данный работник. Выбираются работники только на должности Автомеханик.
7. Выводится имя производителя и модель соответствующего автомобиля, страна производителя. Выбирается множество автомобилей с странами производителей соответственно Германия, Франция, Италия, черным цветом автомобилей и объединяется, включая дубликаты, с множеством автомобилей с странами производителей соответственно Япония, Южная Корея и белым цветом автомобилей.
8. Выводится имя и фамилия клиента, стоимость его заказа. Выбираются заказы с ценой более 10 000 единиц и статусом соответственно ПОДТВЕРЖДЕНО/В ОБРАБОТКЕ. Данные отсортированы по возрастанию по дате заказа, причем записи с датой заказа равной NULL находятся в начале.
9. Выводится имя и фамилия клиента, номер телефона в формате 80xx, улица и дом клиента, оценка его отзыва. Выбираются множество записей с оценкой более 5 и длиной имени клиента более 5 символов и пересекается, включая дубликаты, с множеством записей, где номер телефона клиента имеет код 44 либо 29.
10. Выводится название должности, имя и фамилия работника, время работы по его должности. Выбирается множество записей с временем работы более 9 часов и из него вычитается множество записей с зарплатой менее 1000 единиц.
11. Выводится цена заказа, модель и название производителя соответствующего автомобиля. Выбираются записи с страной производителя соответственно США, Германия, Великобритания. Данные отсортированы по убыванию по значению цены.
12. Выводится название производителя и модель соответствующего автомобиля, имя и фамилия клиента, цена соответствующего заказа. Выбираются записи с статусом заказа ЗАВЕРШЕНО. Данные отсортированы по возрастанию по значению цены.
13. Выводится цвет, имя производителя и модель соответствующего автомобиля. Выбираются автомобили, которые не были заказаны.
14. Выводится дата заказа, цена, название производителя и модель соответствующего автомобиля. Выбираются записи с датой доставки до 2023-12-31. Данные отсортированы по дате заказа.
15. Выводится имя производителя и модель соответствующего автомобиля. Выбираются автомобили, на которые не были оставлены отзывы.

Далее представлены запросы и результаты их выполнения на рисунках 3.1 – 3.15 соответственно.

SELECT

LOWER(m.title) AS title,

c.model,

UPPER(m.country) AS country,

LOWER(c.category) AS category,

c.manufactureYear,

UPPER(c.VIN) AS VIN,

INITCAP (c.colour) AS colour

FROM

cars AS c

INNER JOIN manufacturers as m ON m.ID = c.manufacturerID

WHERE

c.colour IN ('GREEN', 'GREY', 'BLUE', 'PINK')

AND c.model LIKE '%a%'

ORDER BY

m.title DESC;

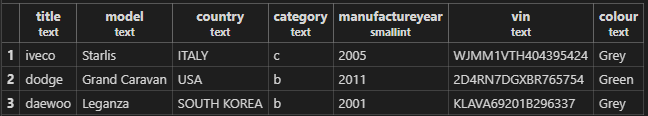


Рисунок 3.2 – Результат выполнения запроса 1

SELECT

TO\_CHAR (o.deliverydate, 'YYYY-Month, Day') AS delivery\_date,

TO\_CHAR (o.dateoforder, 'YYYY-Month, Day') AS date\_of\_order,

INITCAP (o.orderstatus) AS order\_status,

clients.firstname AS first\_name,

CONCAT (

'80',

SUBSTRING(

clients.phonenumber

FROM

5

)

) AS phone\_number

FROM

orders AS o

INNER JOIN clients ON o.clientID = clients.ID

WHERE

o.dateoforder IS NOT NULL

AND clients.firstname LIKE '%a%'

ORDER BY

o.deliverydate ASC NULLS LAST;

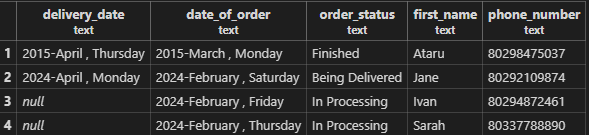


Рисунок 3.2 – Результат выполнения запроса 2

SELECT

CONCAT (

TO\_CHAR (p.endhour - p.starthour, 'HH24'),

' hours'

) AS time\_to\_work,

CONCAT (TRANSLATE(TRUNC(CAST(p.salary AS numeric), 0)::text, '?', ''), ' $') AS salary,

TO\_CHAR(w.assignmentdate, 'YYYY-MONTH DAY') AS assignment\_date,

CONCAT(e.firstname,' ',e.secondname) AS employee

FROM

positions AS p

INNER JOIN works AS w ON p.ID = w.positionID

INNER JOIN employees AS e ON e.ID = w.employeeID

WHERE

p.salary > 600::money

ORDER BY

w.assignmentdate DESC;



Рисунок 3.3 – Результат выполнения запроса 3

SELECT

CONCAT(c.firstname,' ',c.secondname) AS full\_name,

r.rating,

CONCAT(m.title, ' ', cars.model, ', ', TO\_CHAR(cars.manufactureYear, '9999')) AS car

FROM clients AS c

RIGHT OUTER JOIN reviews as r ON c.ID = r.clientid

INNER JOIN cars ON cars.ID = r.carID

INNER JOIN manufacturers AS m ON cars.manufacturerid = m.ID

WHERE

r.rating IN (8,9,10)

ORDER BY

r.rating DESC;

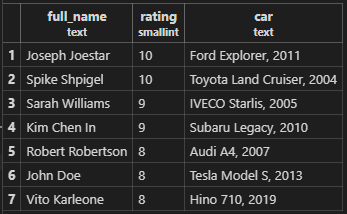


Рисунок 3.4 – Результат выполнения запроса 4

SELECT

CONCAT(e.firstname,' ',e.secondname) AS employee,

e.experience,

p.title,

TRIM(p.positiondescription) AS position\_description,

CONCAT (

TO\_CHAR (p.endhour - p.starthour, 'HH24'),

' hours'

) AS time\_to\_work,

c.model

FROM positions AS p

LEFT OUTER JOIN works AS w ON p.ID = w.positionID

INNER JOIN employees AS e ON w.employeeID = e.ID

INNER JOIN services AS s ON s.employeeID = e.ID

INNER JOIN cars AS c ON s.carID = c.ID

WHERE

(p.endhour - p.starthour) > INTERVAL '8 hours' AND

e.experience > 1

ORDER BY

e.experience DESC;

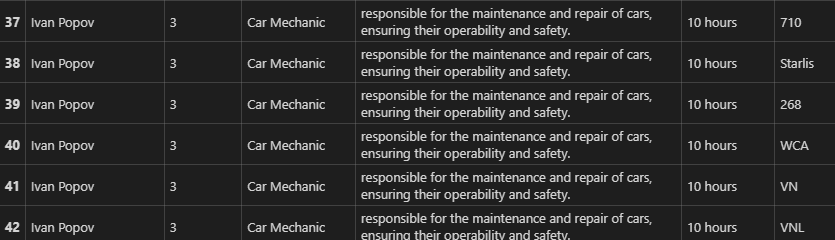
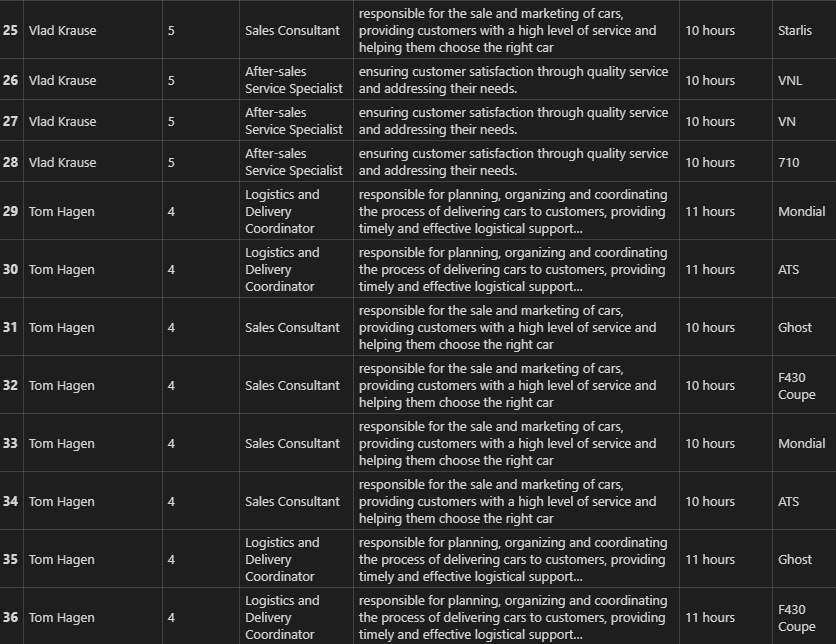
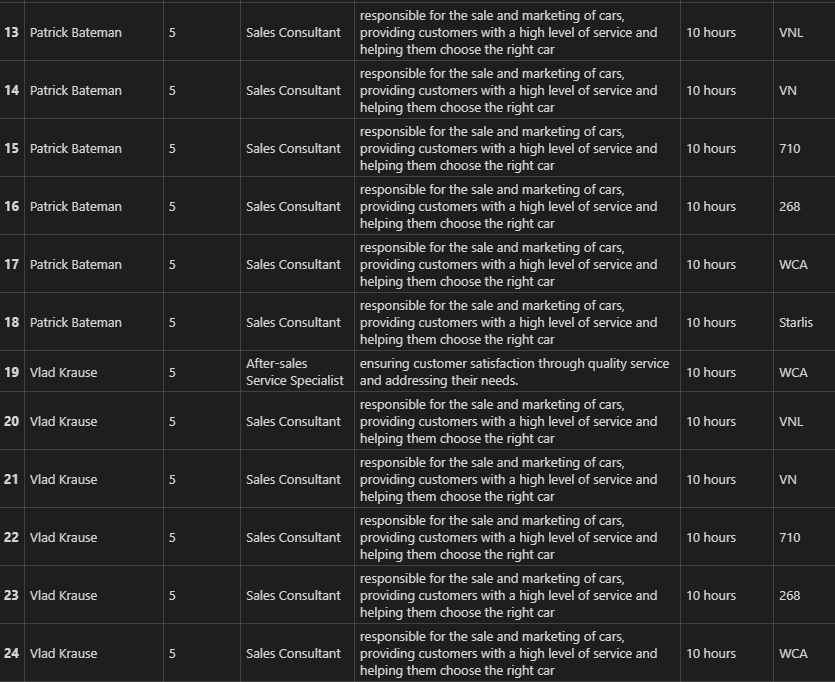


Рисунок 3.5 – Результат выполнения запроса 5

SELECT

CONCAT(e.firstname,' ',e.secondname) AS employee,

CONCAT(m.title, ' ', c.model) AS car

FROM employees AS e

CROSS JOIN cars AS c

JOIN positions AS p ON p.title = 'Car Mechanic'

JOIN services AS s ON e.ID = s.employeeID AND c.ID = s.carID

JOIN manufacturers AS m ON c.manufacturerID = m.ID;



Рисунок 3.6 – Результат выполнения запроса 6

SELECT

CONCAT(m.title, ' ', c.model) AS car,

UPPER(m.country) AS country

FROM cars AS c

INNER JOIN manufacturers AS m ON c.manufacturerID = m.ID

WHERE

m.country IN ('Germany','France','Italy') AND c.colour LIKE 'BLACK'

UNION ALL

SELECT

CONCAT(m.title, ' ', c.model) AS car,

UPPER(m.country) AS country

FROM cars AS c

INNER JOIN manufacturers AS m ON c.manufacturerID = m.ID

WHERE

m.country IN ('Japan','South Korea') AND c.colour LIKE 'WHITE';



Рисунок 3.7 – Результат выполнения запроса 7

SELECT

CONCAT(UPPER(c.firstname),' ',UPPER(c.secondname)) AS client,

CONCAT (TRANSLATE(o.price::text, '?', ''), ' $') AS price

FROM clients AS c

FULL OUTER JOIN orders AS o ON o.clientID = c.ID

WHERE o.price > 10000::money AND o.orderstatus IN ('CONFIRMED', 'IN PROCESSING')

ORDER BY o.dateoforder NULLS FIRST;

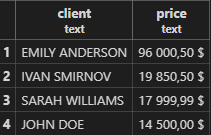


Рисунок 3.8 – Результат выполнения запроса 8

SELECT

CONCAT(UPPER(c.firstname),' ',UPPER(c.secondname)) AS client,

CONCAT ('80', SUBSTRING( c.phonenumber FROM 5)) AS phone\_number,

CONCAT (c.street, ' ', c.house) AS client\_address,

r.rating

FROM clients AS c

INNER JOIN reviews AS r ON r.clientID = c.ID

WHERE r.rating > 5 AND LENGTH(c.firstname) > 5

INTERSECT ALL

SELECT

CONCAT(UPPER(c.firstname),' ',UPPER(c.secondname)) AS client,

CONCAT ('80', SUBSTRING(c.phonenumber FROM 5)) AS phone\_number,

CONCAT (c.street, ' ', c.house) AS client\_address,

r.rating

FROM clients AS c

INNER JOIN reviews AS r ON r.clientID = c.ID

WHERE SUBSTRING(c.phonenumber FROM 5 FOR 2) IN ('29','44');

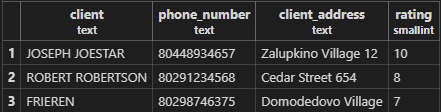


Рисунок 3.9 – Результат выполнения запроса 9

SELECT

p.title,

CONCAT(LOWER(e.firstname),' ',LOWER(e.secondname)) AS employee,

CONCAT (

TO\_CHAR (p.endhour - p.starthour, 'HH24'),

' hours'

) AS work\_time

FROM positions AS p

INNER JOIN works AS w ON p.ID = w.positionID

INNER JOIN employees AS e ON e.ID = w.employeeID

WHERE

(p.endhour - p.starthour) > INTERVAL '9 hours'

EXCEPT

SELECT

p.title,

CONCAT(LOWER(e.firstname),' ',LOWER(e.secondname)) AS employee,

CONCAT (

TO\_CHAR (p.endhour - p.starthour, 'HH24'),

' hours'

) AS work\_time

FROM positions AS p

INNER JOIN works AS w ON p.ID = w.positionID

INNER JOIN employees AS e ON e.ID = w.employeeID

WHERE

p.salary < 1000::money;

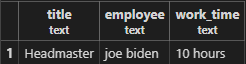


Рисунок 3.10 – Результат выполнения запроса 10

SELECT

CONCAT (TRANSLATE(ROUND(CAST(o.price AS numeric), 0)::text, '?', ''), ' $') AS price,

CONCAT(m.title, ' ', c.model) AS car

FROM orders AS o

LEFT OUTER JOIN cars AS c ON o.carID = c.ID

INNER JOIN manufacturers AS m ON c.manufacturerID = m.ID

WHERE m.country IN ('USA','Germany','UK')

ORDER BY o.price DESC;

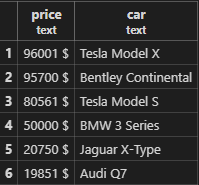


Рисунок 3.11 – Результат выполнения запроса 11

SELECT

CONCAT(m.title, ' ', c.model) AS car,

CONCAT(clients.firstname, ' ', clients.secondname) AS client,

CONCAT (TRANSLATE(ROUND(CAST(o.price AS numeric), 0)::text, '?', ''), ' $') AS price

FROM clients

RIGHT OUTER JOIN orders AS o ON clients.ID = o.clientID

INNER JOIN cars AS c ON c.ID = o.carID

INNER JOIN manufacturers AS m ON m.ID = c.manufacturerID

WHERE o.orderstatus LIKE 'FINISHED'

ORDER BY o.price ASC;

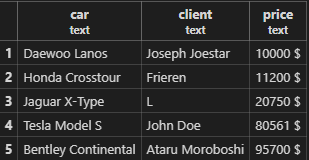


Рисунок 3.12 – Результат выполнения запроса 12

SELECT

CONCAT(INITCAP(c.colour), ' ', m.title, ' ', c.model) AS car

FROM cars AS c

LEFT OUTER JOIN orders AS o ON o.carID = c.ID

INNER JOIN manufacturers AS m ON m.ID = c.manufacturerID

WHERE o.carID IS NULL;

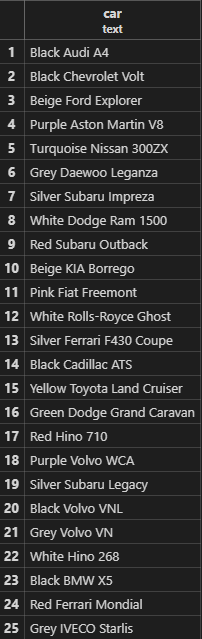


Рисунок 3.13 – Результат выполнения запроса 13

SELECT

o.dateoforder,

CONCAT (TRANSLATE(ROUND(CAST(o.price AS numeric), 0)::text, '?', ''), ' $') AS price,

CONCAT(m.title, ' ', c.model) AS car

FROM cars AS c

RIGHT OUTER JOIN orders AS o ON o.carID = c.ID

INNER JOIN manufacturers AS m ON c.manufacturerID = m.ID

WHERE o.deliverydate < '2023-12-31'

ORDER BY o.dateoforder;

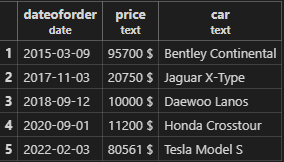


Рисунок 3.14 – Результат выполнения запроса 14

SELECT

CONCAT(m.title, ' ', c.model) AS car

FROM cars c

LEFT JOIN reviews r ON r.carID = c.ID

INNER JOIN manufacturers AS m ON c.manufacturerID = m.ID

WHERE r.carID IS NULL;

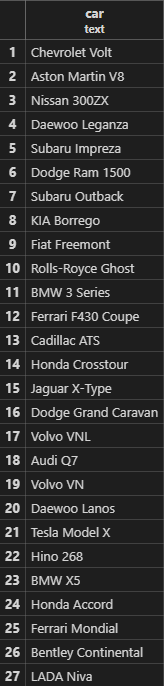


Рисунок 3.15 – Результат выполнения запроса 15