**Вопросы по SQL**

1. За что отвечает символ звёздочка (\*) в SQL-запросе select?

За выбор всех столбцов из таблицы

1. Чем отличается оператор **union** от оператора **join**?

UNION объединяет строки по вертикали в одну результирующую.

JOIN объединяет строки **на основе соответствия значений в указанных столбцах**. Он создает результирующую таблицу **по горизонтали.**

1. Чем отличается **union** от **union all**?

UNION ALLне удаляет дубликаты после объединения запросов, а UNION автоматически удаляет повторения

1. Расставьте в хронологическом порядке операции над БД выполняемые в следующем запросе, написав номер операции над ней в таблице ниже:

select top 10 \*

from weather w (nolock) left join holidays h (nolock)

on w.date = h.date and h.event != 'Christmas\_holidays'

where h.type = 'Weekend'

group by month(w.date)

having datepart(year, h.date) % 4 = 0

order by h.weekday

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **6** | **1** | **2** | **3** | **4** | **5** | **7** |
| SELECT | FROM | JOIN | WHERE | GROUP BY | HAVING | ORDER BY |

1. Если имеются две таблицы с совпадающими названиями полей (например таблицы Weather с полем date и таблица Holidays с полем date), объединенный join-ом по некоторому полю != date , а нужно вывести результат только одной таблицы. Каким образом это указывается в запросе?

Для того, чтобы вывести результат только одной таблицы после объединенияJOIN-ом необходимо в SELECT запросе явно указать поля для каждой таблицы в виде weather.date.

SELECT weather.date

FROM weather

INNER JOIN holidays USING(weather\_id)

1. Какая конструкция будет (в среднем) выполняться быстрее:

select

Клиент,

case when Источник\_клиента = 'интернет-заявка'

then '1'

when Источник\_клиента = 'оффлайн'

then '2'

end "Категория клиента"

from Clients (nolock)

--=======================

select

Клиент,

1 as Категория клиента

from Clients (nolock)

where Источник\_клиента = 'интернет-заявка'

union all

select

Клиент,

2 as Категория клиента

from Clients (nolock)

where Источник\_клиента = 'оффлайн'

1 конструкция будет выполняться быстрее, это связано с тем, что она проходится по таблице всего один раз, и в нем сразу каждому клиенту присваивает категории клиентам.

1. Имеются 2 поля в формате datetime, но из-за неудачного проектирования запись идет таким образом, что в одном поле дата, но без времени, а в другом поле – время, но без даты, нужно получить одно поле в котором будет и дата и время:

|  |  |  |
| --- | --- | --- |
| date (datetime, но время не пишется – оно всегда статично) | Time (пишется время, но день всегда статичен) | Нужное поле – содержит и время, и дату |
| 2020-01-14 00:00:00:000 | 1970-01-01 01:38:00:000 | 2020-01-14 01:38:00:0000 |
| 2020-01-15 00:00:00:000 | 1970-01-01 12:56:00:000 | 2020-01-15 12:56:00:0000 |
| 2020-01-16 00:00:00:000 | 1970-01-01 11:58:04:000 | 2020-01-16 11:58:04:0000 |
| 2020-01-17 00:00:00:000 | 1970-01-01 08:38:07:000 | 2020-01-17 08:38:07:0000 |

SELECT DATE\_ADD(DATE(date), INTERVAL TIME\_TO\_SEC(time) SECOND) AS date\_time

FROM tab;

1. Что является более быстрым при прочих равных при select-запросе из двух объединенных таблиц – указание условий в блоке join или указание условий в блоке where?

Указание условий в блоке JOIN будет более быстрым, так как при объединении JOIN-ом будут отбираться сразу нужные значения, а при условии в блоке WHERE придется повторно проходиться по всей таблице и отбирать только нужные строки.

1. Чем отличается **raw\_number** от **rank**?

ROW\_NUMBER присваивает уникальный порядковый номер каждой строке, независимо от наличия одинаковых значений.

RANK присваивает одинаковый ранг строкам с одинаковыми значениями и пропускает следующие номера при наличии одинаковых рангов.

1. Какой командой можно вернуть все строки таблице, где поле Comment содержит хотя бы пробел или пустое выражение (но не NULL):

Пример таблицы

|  |  |
| --- | --- |
| Id | Comment |
| 1 | NULL |
| 2 | \_ |
| 3 | Ada\_\_\_ |
| 4 | Не известноыыю |
| 5 | Кто здесь?? |

SELECT Comment FROM tab

WHERE Comment LIKE '% %' OR Comment = ''

1. Напишите условие, которое вернет все строки, содержащие нижнее подчеркивание в поле **Comment**

SELECT \* FROM tab

WHERE Comment LIKE '%\\_%';

1. Перед вами таблица и запрос, формирующий, поле «Сумма» , которое должно содержать сумму значений полей «Проценты» и «Основной долг». При этом поля «Проценты» и «Основной долг» не заполнялись значениями, если оплат не поступало. Какая была допущена ошибка при формировании поля Сумма, на ваш взгляд и как её исправить?

Select «Проценты» + «Основной долг» as «Сумма» from Tab

|  |  |  |
| --- | --- | --- |
| Проценты | Основной долг | Сумма |
| 10 | 8 000 | 8 010 |
| 12 | 9 000 | 9 012 |
| NULL | 11 000 | NULL |
| NULL | 80 000 | NULL |

Если значение одного из полей равно NULL при сложении, результат будет равен NULL. Чтобы это исправить, необходимо использовать функцию COALESCE(столбец, 0), которая будет выдавать первое не NULL значение.

SELECT COALESCE(«Проценты», 0) + COALESCE(«Основной долг», 0) AS «Сумма» FROM Tab;

1. Напишите запрос, возвращающий последнее действие клиента в хронологическом порядке на веб-странице в виде таблицы «Клиент такой-то» - «Переход на вкладку Займы»

Tab

|  |  |  |
| --- | --- | --- |
| Client | Date\_time | Action |
| 312 | 2020-01-01 14:22:01 | Авторизация |
| 312 | 2020-01-01 14:24:01 | Переход на вкладку Займы |
| 312 | 2020-01-01 14:25:01 | Переход на вкладку Займы |
| 312 | 2020-01-01 14:28:01 | Досрочное гашение тела |
| 312 | 2020-01-01 14:29:01 | Выход из личного кабинета |
| 5967 | 2020-01-01 14:10:01 | Авторизация |
| 5967 | 2020-01-01 14:12:01 | Переход на вкладку Займы |
| 5967 | 2020-01-01 14:15:01 | Досрочное гашение тела |
| 899 | 2020-01-01 14:40:01 | Авторизация |

SELECT Client, Action

FROM tab

WHERE Date\_time IN (SELECT MAX(Date\_time)

FROM tab

GROUP BY Client)

1. Имеется таблица

Tab

|  |  |  |
| --- | --- | --- |
| ID\_CLIENT | ID\_DOGOVOR | Interest\_rate (Процентная ставка по договору) |
| 1 | 1 | 1 |
| 2 | 2 | 1.2 |
| 1 | 3 | 1.2 |
| 1 | 4 | 1.5 |
| 2 | 5 | 2 |
| 3 | 6 | 2 |

Напишите запрос, который выведет список клиентов (без повторений), у которых процентная ставка больше 1.3%

SELECT DISTINCT(ID\_CLIENT)

FROM tab

WHERE Interest\_rate > 1.3

1. Напишите запрос, подсчитывающий по таблице из предыдущего задания, сколько у каждого из клиентов договоров

SELECT ID\_CLIENT, COUNT(ID\_DOGOVOR)

FROM tab

GROUP BY ID\_CLIENT

1. Имеются две таблицы: Dogovor (список договоров клиентов, уникальный ключ ID – номер договора) и Clients (содержит персональные данные клиента, уникальный ключ – ID- идентификатор клиента):

Dogovor

|  |  |  |
| --- | --- | --- |
| ID | ID\_CLIENT | Interest\_rate (Процентная ставка по договору) |
| 1 | 1 | 1 |
| 2 | 2 | 1.2 |
| 3 | 1 | 1.2 |
| 4 | 1 | 1.5 |
| 5 | 2 | 2 |
| 6 | 3 | 2 |

Clients

|  |  |  |
| --- | --- | --- |
| ID | Second\_name (фамилия) | Age (возраст) |
| 1 | Иванов | 23 |
| 2 | Кукина | 42 |
| 3 | Костицин | 30 |

Напишите запрос, который вернет номера **договоров** тех клиентов, которым больше 25 лет

SELECT dogovor.id

FROM dogovor

INNER JOIN clients ON dogovor.id\_client = clients.id

WHERE age > 25

1. Имеются две таблицы, смысл таблиц не имеет значения

Tab1

|  |  |  |
| --- | --- | --- |
| **Document\_number** | **Date** | **Expiration\_days** |
| 1 | 2019-01-01 | 7 |
| 2 | 2019-01-01 | 8 |
| 3 | 2020-03-03 | 9 |

Tab2

|  |  |
| --- | --- |
| **Document\_number** | **Date** |
| 1 | 2019-01-01 |
| 2 | 2019-01-01 |

Имеется запрос по **left join** таблицы Tab2 к таблице Tab1 по полю **Date** (есть в обеих таблицах), благодаря которому возвращаются столбцыобеих таблиц (все):

Запрос написан в виде:

Select

Tab1.Date

,Tab1.Document\_number as “Первый документ”

,Tab2.Document\_number as “Второй документ”

From

Tab1

Left join

Tab2

On Tab1.Date = Tab2.Date

**Подумайте, сколько строк будет в результате его выполнения:**

2

4

5

6

**Нарисуйте таблицу с результатом**

|  |  |  |
| --- | --- | --- |
| Date | Первый документ | Второй документ |
| 2019-01-01 | 1 | 1 |
| 2019-01-01 | 1 | 2 |
| 2019-01-01 | 2 | 1 |
| 2019-01-01 | 2 | 2 |
| 2020-03-03 | 3 | NULL |

**Будут ли пустые значения (NULL) хоть в каком-нибудь поле?**

да

**Сколько будет строк, если left join заменить на inner join**

2

3

4

5

1. Имеется таблица по клиентам и договорам, заключенным с ними, напишите команду (часть запроса), которая бы проставляла порядковый номер договора по каждому клиенту

Пример таблицы и поля, в котором проставляется порядковый номер договора по клиенту

|  |  |  |
| --- | --- | --- |
| ID\_CLIENT | ID\_DOGOVOR | Порядковый номер договора клиента |
| 1 | 1 | 1 |
| 2 | 2 | 1 |
| 1 | 3 | 2 |
| 1 | 4 | 3 |
| 2 | 5 | 2 |
| 3 | 6 | 1 |

ROW\_NUMBER() OVER (PARTITION BY ID\_CLIENT ORDER BY ID\_DOGOVOR) AS "Порядковый номер договора клиента"

1. Напишите конструцию (часть запроса), которая считает число дней просрочки клиента по следующей таблице (а если клиент погасил ранее, чем дата по договору, должен проставляться нолик – нет просрочки)

|  |  |  |  |
| --- | --- | --- | --- |
| Договор | Fact\_expiration (фактическая дата закрытия договора) | Plan\_expiration (Планируемая дата закрытия договора) | Формируемое поле по числу дней просрочки |
| 1 | 2018-07-11 | 2018-08-10 | 0 |
| 2 | 2019-02-15 | 2019-01-25 | 20 |
| 3 | 2019-07-15 | 2019-07-14 | 1 |

UPDATE dogovor

SET pole = IF(DATEDIFF(fact, plan) > 0, DATEDIFF(fact, plan), 0);

1. Имеется таблица по списку договоров клиентов, в ней есть поле **D\_DATEINPUT** с датой заключения договора в формате “**2019-05-01 18:12:15**” типа **datetime**

Напишите команду для отделения от даты месяца (в каком виде будет возвращаться месяц – May или «05» или «5» - не имеет значения)

SELECT MONTHNAME(D\_DATEINPUT) AS month

FROM dogovor

1. Преобразуйте запись в формате строки ’30-01-2020’ в дату

SELECT DATE('2019-05-01 18:12:15')

1. Какие существуют функции, позволяющиеся представить запись формата datetime (например, 20181031 00:00:00:012)

* в строку нужного вида (например ’31 октября 2018’)

SELECT DATE\_FORMAT(STR\_TO\_DATE('20181031 00:00:00.012', '%Y%m%d %H:%i:%s.%f'), '%d %M %Y')

* в запись того же формата datetime, но в нужном виде (например 31.10.18)

SELECT DATE\_FORMAT(STR\_TO\_DATE('20181031 00:00:00.012', '%Y%m%d %H:%i:%s.%f'), '%d %M %Y')

1. За что отвечает команда GO в SQL?

Команда GO указывает **конец текущего пакета инструкций** и начало нового.

1. С помощью какой команды можно завести переменную?

Переменные можно объявлять с помощью оператора **SET** или **DECLARE**

1. Какие бывают типы данных в SQL?

INT, DECIMAL, FLOAT, REAL, VARCHAR, DATE, TIME,

1. Соотнесите команды SQL с их смысловой нагрузкой:

|  |  |
| --- | --- |
| Drop (if exists) | Удалить таблицу из базы данных (полностью, как объект, а не просто стереть)) |
| Update | Изменить строки в таблицы в соответствии с некоторым условием |
| Insert | Добавить записи к уже существующей таблице |
| Select … into… from … | Создать новую таблицу с одновременным помещением туда записей |
| Create table … | Создать пустую таблицу, дать наименования полям и назначить типы данных для полей |
| Alter table … add… | Добавить столбцы к существующей таблицу |
| Alter table … alter column … | Изменить типы данных полей или добавить новые столбцы |
| Truncate … | Очистить таблицу, но не удалять её |

1. Как сделать так, чтобы команды в SQL-скрипте выполнялись пошагово, а не одновременно? (например при создании таблицы, добавлении полей и изменения их типов)

Необходимо разделять команды “;” и они сами будут выполняться последовательно друг за другом.

1. Вернет ли ошибку нижеприведенный запрос или отработает без ошибок:

declare @name varchar = 'Nikolay'

select \* from Clients\_offline where Name = @name

GO

select \* from Clients\_online where Name = @name

Отработает с ошибкой, так как во втором SELECT запросе не будет доступна переменная @name, она была определена только для предыдущей процедуры с использованием declare.