



Вопросы и ответы на собеседование Java Junior Developer.

Заказывайте
Музторг

INTERVIEW

JAVA DEVELOPER

SQL

Ответы на вопросы на собеседование SQL (часть 2).

vasyl1889 18:34:00 3 Комментарии

• ЧТО ДЕЛАЕТ ФУНКЦИЯ EXISTS?

Аргументом функции EXISTS есть внутренний запрос. она возвращает истину
запрос возвращает один или более строк, и возвращает ложь если запрос верне



Реклама от Google

Не показывать это объявление

Почему это объявление?

Реляционный оператор PIVOT можно использовать для изменения возвращаемого табличного значения выражения в другой таблице. Оператор PIVOT разворачивает возвращаемое табличное значение выражения, преобразуя уникальные значения одного столбца выражения в несколько выходных столбцов, а также, в зависимости от необходимости, объединяет оставшиеся повторяющиеся значения столбца и отображает их в выходных данных.

- **ОПИШИТЕ РАЗНИЦУ ТИПОВ ДАННЫХ DATETIME И TIMESTAMP.**

DATETIME предназначен для хранения целого числа: YYYYMMDDHHMMSS. И это не зависит от временной зоны настроенной на сервере. Хранит: 8 байт. TIME хранит значение равное количеству секунд, прошедших с полуночи 1 января 1970 по усреднённому времени Гринвича. При получении из базы отображается с учётом часового пояса. Хранит: 4 байта.

- **ДЛЯ КАКИХ ЧИСЛОВЫХ ТИПОВ НЕДОПУСТИМО ИСПОЛЬЗОВАТЬ ОПЕРАЦИЮ СЛОЖЕНИЯ (ВЫЧИТАНИЯ), А ЗНАЧИТ И ФУНКЦИЯ SUM()?**

--	--	--	--

Заказывайте
Музторг

В качестве операндов операций сложения и вычитания допустимо любое корректное выражение любого типа данных числовой категории, кроме типа данных bit.

- **ЧТО ТАКОЕ ХРАНИМЫЕ ПРОЦЕДУРЫ?**



Реклама от Google

[Не показывать это объявление](#)[Почему это объявление?](#)

выходной результат, они могут как выполнять различные численные вычисления, так и выполнять стандартные операции с БД. Как и в процедурах других языков программирования, в них могут быть циклы и ветвления.

• ФУНКЦИИ РАНЖИРОВАНИЯ ЧТО ЭТО И КАКИЕ СУЩЕСТВУЕТ?

Ранжирующие функции возвращают ранжирующее значение для каждой строки секции. В зависимости от используемой функции значения некоторых строк совпадают. Ранжирующие функции являются недетерминированными.

Transact-SQL содержит следующие ранжирующие функции:

- RANK
- NTILE
- DENSE_RANK
- ROW_NUMBER

• МОЖЕТ ЛИ ЗНАЧЕНИЕ В СТОЛБЦЕ(АХ), НА КОТОРЫЙ НАЛОЖЕНО ОГРАНИЧЕНИЕ FOREIGN KEY, РАВНЯТЬСЯ NULL?

Может, если на данный столбец не наложено ограничение not null, например, при построении таблицы дерева файловой системы, где столбец foreign key – ссылка на эту же самую таблицу, на кортеж с информацией о родительской директории, для корневой директории файловой системы в столбце родительской директории – null.

• НАЗОВИТЕ ОСНОВНЫЕ СВОЙСТВА ТРАНЗАКЦИИ.

ACID – atomicity (атомарность), consistency (непротиворечивость), isolation (изолированность), durability (устойчивость).

- Свойство атомарности гарантирует неделимость набора операторов, которые изменяют данные в базе данных и являются частью транзакции. Это означает, что либо выполняются все изменения данных в транзакции, либо в случае любой ошибки все уже выполненные изменения отменяются.
- Согласованность гарантирует, что транзакция не даст возможности базе данных содержать несогласованные данные. Другими словами, трансформация данных в рамках одной транзакции переводит базу данных из одного согласованного состояния в другое согласованное состояние.
- Свойство изолированности разделяет все одновременно выполняющиеся транзакции. Другими словами, ни одна активная транзакция не может изменить данные, выполненные в параллельной, но не завершённой транзакции.

Реклама от Google

Не показывать это объявление

Почему это объявление?

происходит фиксация транзакции, означающая, что ее действие постоянно да-
сбое системы.

• КАК УДАЛИТЬ ПОВТОРЯЮЩИЕСЯ СТРОКИ С ИСПОЛЬЗОВАНИЕМ КЛЮЧЕВОГО СЛОВА DISTINCT?

SELECT DISTINCT columnsName FROM tableName;

где: columnsName – одно или несколько реальных имен столбцов, перечисленных через запятую; tableName – имя той таблицы, из которой выбираются эти столбцы. Если в предложение SELECT DISTINCT включить более одного столбца, то в результате уникальность любой строки будет определяться уникальностью соответствующей комбинации всех значений столбцов, включенных в предложение, на этой строке среди аналогичных комбинаций, соответствующих другим строкам.

ИНТЕРВАЛЬНОЕ ГОЛОДАНИЕ ДЛЯ ПОХУДЕНИЯ

7-ДНЕВНЫЙ ПЛАН ДЛЯ НАЧИНАЮЩИХ

ВОЗРАСТ:	Понедельник	Вторник	Среда	Четверг	Пятница	Суббота	Воскресенье
18-25	К	О	К	К	О	О	К
	Омлет	Омлет	Омлет	Омлет	Омлет	Омлет	Омлет
	Ужин	Ужин	Ужин	Ужин	Ужин	Ужин	Ужин
26-35	К	Омлет	Омлет	Омлет	Омлет	Омлет	Омлет
	К	Омлет	Омлет	Омлет	Омлет	Омлет	Омлет
	К	Ужин	Ужин	Ужин	Ужин	Ужин	Ужин
36-50	Омлет	О	Омлет	Омлет	О	Омлет	Омлет
	Омлет	О	Омлет	Омлет	О	Омлет	Омлет
	Ужин	Ужин	Ужин	Ужин	Ужин	Ужин	Ужин
50+	Омлет	О	Омлет	Омлет	О	Омлет	Омлет
	Омлет	Омлет	Омлет	Омлет	Омлет	Омлет	Омлет
	К	Ужин	Ужин	К	Ужин	Ужин	Ужин

Заказывайте

Музторг



Несмотря на то что значения null никогда не бывают равны друг другу (они считаются неизвестными), предложение DISTINCT, напротив, считает их дублик



Реклама от Google

[Не показывать это объявление](#)[Почему это объявление?](#)

- **КОГДА ПОЛНОЕ СКАНИРОВАНИЕ ТАБЛИЦЫ ВЫГОДНЕЕ ДОСТУП ИНДЕКСУ? ОПИШИТЕ ВКРАТЦЕ ОБЩИЕ ПРИНЦИПЫ, ОПТИМИЗАТОР ВЫБИРАЕТ ПРОИЗВОДИТЬ ЛИ ПО СКАНИРОВАНИЕ ТАБЛИЦЫ ИЛИ ДОСТУП ПО ИНДЕКСУ.**

Полное сканирование производится многоблочным чтением. Сканирование по индексу – одноблочным. Также, при доступе по индексу сначала идет сканирование индекса, а затем чтение блоков из таблицы. Число блоков, которые надо прочитать из таблицы зависит от фактора кластеризации.

Если суммарная стоимость всех необходимых одноблочных чтений больше стоимости полного сканирования многоблочным чтением, то полное сканирование выгодно. Оно выбирается оптимизатором. Таким образом, полное сканирование выбирается в случае слабой селективности предикатов запроса и/или слабой кластеризации данных, в случае очень маленьких таблиц.

- **ИМЕЕТ ЛИ СМЫСЛ ИНДЕКСИРОВАТЬ ПОЛЯ ТАБЛИЦЫ, ИМЕЮЩИЕ ТИП BOOLEAN ИЛИ ПОДОБНЫЕ ИМ – С НЕБОЛЬШИМ КОЛИЧЕСТВОМ ВОЗМОЖНЫХ ЗНАЧЕНИЙ?**

Индекс по логическим полям имеет смысл, только в случае, если значения ИСТИНА и ЛОЖЬ распределены примерно равномерно по таблице. Предельный случай – когда все значения ИСТИНА или все значения ЛОЖЬ. В остальных случаях индексирование не имеет смысла.

- **ЧТО ТАКОЕ АГРЕГАТНАЯ ФУНКЦИЯ? ПРИВЕДИТЕ ПРИМЕР АГРЕГАТНЫХ ФУНКЦИЙ В SQL.**



Реклама от Google

Не показывать это объявление

Почему это объявление?

Индивидуальный пошив муж
обуви в Москве от 9600 руб.
дней. Натуральная кожа.

YARD

От

Агрегатная функция – это функция, которая возвращает одиночное значение в основании множества записей.

Вот список некоторых агрегатных функций SQL:

- COUNT(*) – Возвращает количество строк источника записей
- COUNT – Возвращает количество значений в указанном столбце
- SUM – Возвращает сумму значений в указанном столбце
- AVG – Возвращает среднее значение в указанном столбце
- MIN – Возвращает минимальное значение в указанном столбце
- MAX – Возвращает максимальное значение в указанном столбце

• ДАЙТЕ ОПРЕДЕЛЕНИЕ ТРЕТЬЕЙ НОРМАЛЬНОЙ ФОРМЕ БД.

Определение третьей нормальной форме БД.

- Любое поле любой записи хранит только одно значение. (1NF) Например, поле хранится список идентификаторов, разделённых запятыми, то это нарушает данное определение.
- Выполняется условие 1NF и любое неключевое поле полностью зависит от 1NF (2NF) Например, у нас есть запись с полями (Идентификатор, Название CD-Диска, Название группы), где ключом является поле «Идентификатор». При этом очевидно, что поле «Название группы» зависит не только от «Идентификатора», но и от поля «Название CD-Диска». Поэтому такая БД не находится во второй нормальной форме.
- Выполняется условие 2NF и нет неключевых полей зависящих от значения 1NF неключевых полей. Например у нас в записи хранятся код региона и его название. Понятно, что название региона зависит от кода, и наоборот, поэтому такая БД будет находиться в третьей нормальной форме.

• ЧТО ТАКОЕ ДЕНОРМАЛИЗАЦИЯ БД? ДЛЯ ЧЕГО ОНА НУЖНА?

Денормализация – это процесс осознанного приведения базы данных к виду, в котором она не будет соответствовать правилам нормализации. Обычно это необходимо для повышения производительности и скорости извлечения данных.



Реклама от Google

Не показывать это объявление

Почему это объявление?

рассмотреть возможность проведения денормализации.

--	--	--

Возможное решение следующее: вынести результаты выборки в отдельную таблицу. Это позволит увеличить скорость выполнения запросов, но также означает появление необходимости в постоянном обслуживании этой новой таблицы. Прежде чем приступать к денормализации, необходимо убедиться, что ожидаемые результаты оправдывают издержки, с которыми придется столкнуться.

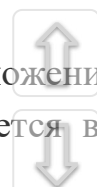
• ЧТО ТАКОЕ ТРИГГЕР?

Триггер – это SQL процедура, которая срабатывает при каком-нибудь событии (INSERT, DELETE или UPDATE). Триггеры хранятся и управляются СУБД. Триггеры используются для поддержания ссылочной целостности данных в одинаковой таблице, реагируя на события изменения этих данных. Триггер не может быть вызван вручную, СУБД автоматически вызывает его после модификации данных в соответствующей таблице. В этом и есть его отличие от хранимых процедур, которые нужно выполнять вручную вызовом CALL. Также триггер может вызывать другие процедуры.

Триггер также может содержать вызовы INSERT, DELETE и UPDATE внутри себя, тем самым вызывая другой триггер. Такие триггеры называются вложенными (nested).

• ЧТО ТАКОЕ КУРСОРЫ В БАЗАХ ДАННЫХ?

Курсор – это объект базы данных, который позволяет приложениям работать с записями "по-одной", а не сразу с множеством, как это делается в обычных SQL-запросах.



Реклама от Google

Не показывать это объявление

Почему это объявление?

- Открыть курсор (OPEN)
- Получить запись из курсора (FETCH)
- Обработать запись
- Закрыть курсор (CLOSE)

• КАКИЕ КОМПРОМИССЫ ПРЕДЛАГАЕТ ИСПОЛЬЗОВАНИЕ ИНДЕКСОВ?

Некоторые из них:

- Более быстрые выборки, но более медленные изменения. (При изменении тратится время на перестройку индекса).
- Для хранения индексов необходимо дополнительное дисковое пространство.

• ЧТО ДЕЛАЕТ SQL ОПЕРАЦИЯ MERGE?

Операция MERGE официально появилась в стандарте ANSI SQL:2008.

Она позволяет одновременно вставлять или изменять записи таблицы со заданным критерием. При выполнении критерия строки изменяются, иначе – вставляют новые. Можно заменить последовательным вызовом INSERT и UPDATE. В некоторых случаях похожая операция называется UPSERT.

• В ЧЕМ РАЗЛИЧИЕ МЕЖДУ ВЫРАЖЕНИЯМИ HAVING И WHERE?

WHERE – это ограничивающее выражение. Оно выполняется до того, как будет получен результат операции.

HAVING – фильтрующее выражение. Оно применяется к результату операции. Выполняется уже после того, как этот результат будет получен, в отличие от WHERE. Выражения WHERE используются вместе с операциями SELECT, UPDATE, DELETE, а HAVING только с SELECT и предложением GROUP BY.

Например, WHERE нельзя использовать таким образом:

```
1 | SELECT name, SUM(salary) FROM Employees WHERE SUM(salary) > 1000 GROUP BY name
```

В данном случае больше подходит HAVING:

```
1 | SELECT name, SUM(salary) FROM Employees GROUP BY name HAVING SUM(salary) > 1000
```

То есть, использовать WHERE в запросах с агрегатными функциями нельзя, для этого был введен HAVING.

• ЧТО ТАКОЕ ЦЕЛОСТНОСТЬ ДАННЫХ? ОБЪЯСНИТЕ, ЧТО ТАКОЕ ОГРАНИЧЕНИЯ.

Реклама от Google

Не показывать это объявление

Почему это объявление?

найти другими способами. целостность данных поддерживается с пол ограничений.

В SQL стандарта ANSI есть 4 основных ограничения: PRIMARY KEY, CHECK, UNIQUE, FOREIGN KEY. Они не являются обязательными для таблицы.

- PRIMARY KEY – набор полей (1 или более), значения которых образуют уника комбинацию и используются для однозначной идентификации записи в таблице. Таблица может быть создано только одно такое ограничение. Данное ограничение используется для обеспечения целостности сущности, которая описана таблицей.
- CHECK используется для ограничения множества значений, которые могут помещены в данный столбец. Это ограничение используется для обеспечения целостности предметной области, которую описывают таблицы в базе.
- Ограничение UNIQUE обеспечивает отсутствие дубликатов в столбце или в столбцах. Разница между PRIMARY KEY и UNIQUE описана в primary и unique constraints.
- Ограничение FOREIGN KEY защищает от действий, которые могут нарушить связь между таблицами. FOREIGN KEY в одной таблице указывает на PRIMARY KEY другой. Поэтому данное ограничение нацелено на то, чтобы не было записей FOREIGN KEY, которым не отвечают записи PRIMARY KEY. Таким образом, FOREIGN KEY поддерживает ссылочную целостность данных.

• В ЧЕМ ОТЛИЧИЕ МЕЖДУ КЛАСТЕРНЫМИ ИНДЕКСАМИ И НЕКЛАСТЕРНЫМИ?

Некластерные индексы создаются СУБД по умолчанию. Данные физически расположены в произвольном порядке, но логически упорядочены согласно индексу. Такой тип индексов подходит для таблиц, где часто изменяются значения.

При кластерном индексировании данные физически упорядочены, что существенно повышает скорость выборки данных (но только в случае последовательного доступа к данным). Для одной таблицы может быть создан только один кластерный индекс.

• КАКИЕ ОТЛИЧИЯ МЕЖДУ ОГРАНИЧЕНИЯМИ PRIMARY И UNIQUE?

Ограничения primary и unique призваны обеспечить уникальность значений столбца, на котором они определены. Но по умолчанию ограничение primary создает кластерный индекс на столбце, а unique – некластерный. Другим отличием является то, что primary не разрешает NULL записей, в то время как unique разрешает одну NULL запись.



РАССКАЗАТЬ ДРУЗЬЯМ:





Реклама от Google

[Не показывать это объявление](#)[Почему это объявление?](#)

ПРЕДЫДУЩЕЕ

СЛЕДУЮЩЕЕ



ПОХОЖИЕ СТАТЬИ



Ответы на вопросы на собеседование Apache Ant (часть 1).



Список вопросов и ответов на собеседование Java Junior (продолжение).



Ответы на вопросы на собеседование JSP, Servlets, . (часть 4).

3 КОМЕНТАРИ :



АЛЕКСАНДР 6 ДЕКАБРЯ 2016 Г., 15:04

"В том время как unique разрешает только одну NULL запись" - это неверно. NULL не берется во внимание, то есть не считается повторением. NULL-ов может быть сколько угодно.

[Ответить](#)

Ответы



UNKNOWN 9 ЯНВАРЯ 2017 Г., 19:34

Вы не правы.

В unique может присутствовать одно значение NULL, при этом может быть создан кластерный уникальный индекс и иметь поле с ограничением unique = NULL или добавлен unique констрейнт или и кластерный уникальный индекс и unique констрейнт одновременно.

А вот с primary key такой фокус не пройдет.

[Ответить](#)



Реклама от Google

[Не показывать это объявление](#)[Почему это объявление?](#)

значения ИСТИНА и ЛОЖЬ распределены примерно равномерно по таблице. Предельный случай - две трети и одна треть." -- и какой смысл в таком индексе? В данном случае скан таблицы намного выгоднее, чем индексный поиск. А вот в случае, когда распределение сильно не равномерное (например, из 10 000 записей лишь одна имеет значение ИСТИНА или ЛОЖЬ), тогда индекс поможет, но только при поиске редкого значения.

[Ответить](#)

Подпись комментария:

egutorov@gmail.com ▼

[Выйти](#)[Публикация](#)[Просмотр](#)☐ [Отправлять уведомления](#)

ПОПУЛЯРНЫЕ ПУБЛИКАЦИИ

[Ответы на вопросы на собеседование ООП.](#)[Список вопросов и ответов на собеседование Java Junior.](#)[Ответы на вопросы на собеседование Java core \(часть 1\).](#)

ОСНОВНЫЕ КАТЕГОРИИ





Реклама от Google

Не показывать это объявление

Почему это объявление?

КОНТАКТНАЯ ФОРМА

Имя

Электронная почта *

Сообщение *

Отправить

© 2015-2021 jsehelper.blogspot.com Все права защищены.

Blogger Technologies.





Реклама от Google

Не показывать это объявление

Почему это объявление?

