

Вопрос	Варианты ответа	
Для чего используется команда ROLLBACK?	Для начала транзакции *Для завершения транзакции *Для отката совершённых изменений Для приостановки транзакции на определённое время	
При какой блокировке перед записью изменений перепроверяется значение выделенного атрибута?	Пессимистическая блокировка *Оптимистическая блокировка Реалистическая блокировка Идеалистическая блокировка	
Какой командой в PostgreSQL можно посмотреть текущий уровень изоляции	*SHOW transaction_isolation; SHOW transaction; SHOW isolation;	
Какая функция не может модифицировать базу данных и всегда возвращает одинаковые результаты для одних и тех же аргументов?	VOLATILE STABLE *IMMUTABLE MUTABLE	
Когда наступает согласованность?	*Когда транзакция переводит базу данных из одного корректного состояния в другое корректное состояние Когда транзакция в базе данных не изменяет состояние Когда транзакция переводит базу данных из некорректного состояния в корректное Когда транзакция в базе данных изменяет состояние	
Выберите, на какие типы делят транзакции:	*Распределённые, локальные Глобальные, локальные Раздельные, распределённые Глобальные, раздельные	
Атомарность транзакции означает, что:	После фиксации транзакции изменения становятся постоянными *Транзакция либо выполняется полностью, либо не выполняется Транзакция переводит базу данных из одного согласованного состояния в другое Результаты транзакции становятся доступны для других транзакций только после ее фиксации	
Функция экспорта снимка, возвращающая id снимка (например, 000003A1-1)	*pg_export_snapshot pg_stat_file pg_reload_conf pg_switch_xlog	

Фантомное чтение означает, что:	<p>Если строка читается в момент времени T1, а затем перечитывается в момент времени T2, то за этот период она может измениться; строка может исчезнуть, может быть обновлена и так далее</p> <p>Допускается чтение незафиксированных данных; при этом нарушается как целостность данных, так и требования внешнего ключа, а требования уникальности игнорируются</p> <p>*Если выполнить запрос в момент времени T1, а затем выполнить его повторно в момент времени T2, в базе данных могут появиться дополнительные строки, влияющие на результаты; при этом прочитанные данные не изменились, но критериям запроса стало удовлетворять больше данных, чем прежде</p>
<p>В данном примере</p> <pre>CREATE FUNCTION get_amount(id integer) RETURNS numeric AS \$\$ SELECT amount FROM accounts a WHERE a.id = get_amount.id; \$\$ VOLATILE LANGUAGE sql;</pre> <p>знак \$\$ (долларовые кавычки):</p>	<p>Комментирует строки кода, заключенные в долларовые кавычки</p> <p>*Означает тело функции в виде строкового литерала с долларовыми кавычками</p> <p>Является именем переменной</p> <p>Является синтаксической ошибкой</p>
Какой лимит на количество команд SQL в одной транзакции?	<p>10</p> <p>2^10 (1024)</p> <p>*2^32 (4 миллиарда)</p> <p>2^5000 (очень много)</p>
Какой командой происходит установка требуемого уровня изоляции?	<p>*SET TRANSACTION</p> <p>GET TRANSACTION</p> <p>PUT TRANSACTION</p> <p>невозможно установить вручную</p>
Дайте определение MVCC?	<p>Метод управления параллелизмом при доступе к данным БД, который гарантирует сериализуемость. Один из механизмов СУБД для обеспечения параллельного доступа к базам данных. Предшественник Snapshot Isolation.</p> <p>Метод управления параллелизмом при доступе к данным БД, основанный на двухфазных блокировках.</p> <p>*Один из механизмов СУБД для обеспечения параллельного доступа к базам данных посредством многоверсионности. Является расширением над Snapshot Isolation.</p>
Какой является VOLATILE-функция?	<p>Временной функцией</p> <p>Стабильной функцией</p> <p>*Изменчивой функцией</p> <p>Постоянной функцией</p>

Какой вид имеет транзакция, применённая в запросе? SELECT last_name, first_name FROM users WHERE age > 20;	Явная транзакция *Неявная транзакция Ограничивающая транзакция Выборочная транзакция В запросе транзакция не используется
Какой тип блокировки исключает возможность модификации изменяемых данных сторонними транзакциями?	*Пессимистическая блокировка Оптимистическая блокировка Реалистическая блокировка Идеалистическая блокировка
Каким уровнем изоляции не допускается аномалия потерянного обновления?	Read Uncommitted Read Committed Repeatable Read Serializable *Ни на одном
С помощью какой функции можно приостановить выполнение запроса на определённое время в PostgreSQL?	*pg_sleep volatile sleep wait
Какие существуют модели одновременного конкурентного доступа?	Реалистический одновременный конкурентный доступ *Оптимистический одновременный конкурентный доступ Перфекционистский одновременный конкурентный доступ Все варианты верны *Пессимистический одновременный конкурентный доступ
Какие существуют глобально разл реализации изолированности?	*Версионирование Коммитирование Все варианты верны Ни один из предложенных *Блокирование
На каких уровнях допускается фантомное чтение?	*Read Uncommitted Write Uncommitted *Read Committed *Repeatable Read Write Committed

Какие аномалии допускаются на уровне изоляции Serializable?	Потерянные изменения Грязное чтение Неповторяющееся чтение Фантомное чтение *Ни один из предложенных
Какими свойствами обладают транзакции?	Атомарность, Изолированность Изолированность, Долговечность Долговечность, Изолированность, Атомарность *Согласованность, Долговечность, Изолированность, Атомарность Атомарность, Согласованность
Выберите правильные характеристики уровней изоляции пользователей	уровень Read committed всегда обеспечивает параллельную работу транзакций уровень Read uncommitted предотвращает феномен потерянных изменений и феномен грязного чтения *уровень Repeatable read обеспечивает уровень изоляции выше, чем у Read committed и Read uncommitted *уровень Serializable обеспечивает самую высокую изоляцию транзакций ничего из выше перечисленного
Что означает уровень изоляции SERIALIZABLE	*Чтение запрещено до завершения транзакции. Повторное чтение строки возвратит первоначально считанные данные, несмотря на любые обновления, произведенные другими пользователями до завершения транзакции. Завершенное чтение, при котором отсутствует черновое, "грязное" чтение. Тем не менее в процессе работы одной транзакции другая может быть успешно завершена и сделанные ею изменения зафиксированы. Незавершенное чтение, или допустимо черновое чтение. Низший уровень изоляции, соответствующий уровню 0. Ни один из выше перечисленных

Укажите режим изоляции транзакций (ISOLATION LEVEL), который позволяет осуществлять чтение строк, измененных незафиксированными транзакциями.	REPEATABLE READ READ COMMITTED *READ UNCOMMITTED SERIALIZABLE Здесь нет такого
Какое свойство транзакции обеспечивает, что в результате выполнения транзакции база данных не будет содержать несогласованных данных?	а) атомарность (Atomicity) *б) согласованность (Consistency) в) изолированность (Isolation) г) долговечность (Durability) д) точность (Accuracy)
На каких уровнях изоляции может возникнуть особое условие "Неповторяемое чтение"?	*a) Read uncommitted *б) Read committed в) Repeatable read г) Serializable д) Ни на каком из перечисленных
Какая аномалия характеризуется чтением данных, полученных в результате действия транзакции, которая после этого откатится?	а) Потерянные изменения *б) Грязное чтение в) Неповторяющееся чтение г) Фантомное чтение д) Никакая из перечисленных
Если характеристика изменчивости не задаётся явно в команде CREATE FUNCTION, то по умолчанию подразумевается?	*a) VOLATILE б) STABLE в) IMMUTABLE г) MUTABLE д) UNSTABLE
Что такое isolation?	а) Свойство, гарантирующее, что никакая транзакция не будет зафиксирована в системе частично б) Свойство, позволяющее базе оставаться консистентной до выполнения операции и после в) Свойство, позволяющее независимо от проблем на нижних уровнях остаться сохраненными изменения, сделанные успешно завершённой транзакцией, после возвращения системы в работу *г) Свойство, позволяющее параллельным транзакциям не оказывать влияния на результат транзакции

На каком уровне стандарта допускается допускается потерянное обновление?	а) READ COMMITTED б) READ UNCOMMITTED в) REPEATABLE READ г) SERIALIZABLE * д) ни на одном уровне
Какой уровень изоляции настроен по умолчанию?	*а) READ COMMITTED б) READ UNCOMMITTED в) REPEATABLE READ г) SERIALIZABLE д) ANTISERIALIZABLE
Самый высокий уровень изоляции?	а) READ COMMITTED б) READ UNCOMMITTED в) REPEATABLE READ *г) SERIALIZABLE д) ANTISERIALIZABLE
Что понимается под целостностью БД?	*а) Правильность и непротиворечивость его содержимого б) Противоречивость его содержимого в) Неправильность его содержимого г) Чтение, удаление, вставка и модификация содержимого БД д) Обработка или выдача правильных данных
Блокировки могут устанавливаться на:	*а) трех уровнях б) четырех уровнях в) пяти уровнях
Грязное чтение означает, что:	*а) допускается чтение незафиксированных данных; при этом нарушается как целостность данных, так и требования внешнего ключа, а требования уникальности игнорируются б) если строка читается в момент времени T1, а затем перечитывается в момент времени T2, то за этот период она может измениться; строка может исчезнуть, может быть обновлена и так далее в) если выполнить запрос в момент времени T1, а затем выполнить его повторно в момент времени T2, в базе данных могут появиться дополнительные строки, влияющие на результаты; при этом прочитанные данные не изменились, но критериям запроса стало удовлетворять больше данных, чем прежде

Выберите правильные характеристики уровней изоляции пользователей	<p>*а) уровень Serializable это самый высокий уровень изоляции, но транзакции-писатели, обращающиеся к одним и тем же ресурсам, не могут выполняться параллельно</p> <p>б) уровень Repeatable read обеспечивает доступ к промежуточным результатам других транзакций</p> <p>*в) уровень Read uncommitted позволяет читать незафиксированные изменения других транзакций</p> <p>г) уровень Read committed обеспечивает отсутствие всех феноменов</p>
Что делает команда COMMIT	<p>Начинает транзакцию;</p> <p>Заканчивает транзакцию;</p> <p>Возвращает БД в исходное состояние;</p> <p>*Фиксирует транзакцию в БД;</p> <p>Все перечисленное;</p>
Какие аномалии допустимы на уровне изоляции Read Uncommitted	<p>Потерянные изменения, грязное чтение, неповторяющееся чтение, фантомное чтение, другие аномалии;</p> <p>*Грязное чтение, неповторяющееся чтение, фантомное чтение, другие аномалии;</p> <p>Неповторяющееся чтение, фантомное чтение;</p> <p>Только фантомное чтение;</p> <p>Только другие аномалии;</p> <p>Потерянные изменения, неповторяющееся чтение, фантомное чтение, другие аномалии;</p>
Какая аномалия допускается на уровне Read Uncommitted, но не допускается на уровне Read Committed?	<p>*Грязное чтение;</p> <p>Фантомное чтение;</p> <p>Никакие;</p> <p>Потерянные изменения;</p> <p>Неповторяющееся чтение;</p>
Какие изоляции являются известными?	<p>*Read Uncommitted;</p> <p>Read Snapshot;</p> <p>*Read Committed;</p> <p>Lost Snapshot;</p> <p>Repeatable Snapshot;</p>

Какая модель одновременного конкурентного доступа предполагает, что транзакция вряд ли модифицирует данные, которые изменяются в другой транзакции в то же самое время?	*Оптимистический конкурентный контроль Частичный конкурентный контроль Строгий конкурентный контроль Пессимистический конкурентный контроль
Какой уровень изоляции имеет самую быструю скорость выполнения транзакций?	Snapshot Serializable *Read Uncommitted Repeatable Read
Из каких двух фаз состоит двухфазная блокировка (2PL)?	фаза блокировки *фаза расширения *фаза сжатия фаза проверки
Что такое Deadlock?	Удаление строки таблицы навсегда Ситуация, когда невозможно совершить больше, чем одну транзакцию *Ситуация, в которой два процесса борются за изменение одного и того же ресурса, но ни одна транзакция не имеет приоритета над другой, в следствие чего возникает тупик Ситуация, когда транзакция достигла лимита возможных совершаемых ей операций
Как называется операция, при которой все изменения данных, осуществленные в процессе выполнения транзакции не могут быть потеряны?	Согласованность Изолированность Атомарность Автоматизированность *Долговременность
Отметьте, какие особые условия НЕ возможны на уровне изоляции Read Committed.	*«Грязное» чтение Фантомное чтение Неповторяемое чтение Аномалия сериализации Ничего из выше перечисленного

<p>двухфазная блокировка</p>	<p>*Метод, определяющий стратегию управления блокировками для обеспечения строгой сериализации. Делит транзакцию на момент, где блокировкам разрешается получать блокировки с начала транзакции до ее окончания, и фазой фиксации или отката, так как в конце транзакции все полученные блокировки освобождаются.</p> <p>Метод, при котором каждая транзакция работает с согласованным снимком данных на определенный момент времени, в который попадают только те изменения, которые были зафиксированы до момента создания снимка</p> <p>Блокировка, в котором в качестве ресурсов выступают отношения и другие объекты</p> <p>Блокировка, в которой для ключа или индекса указывается значение или диапазон значений, на которые распространяется блокировка</p>
<p>Какая функция не может модифицировать базу данных и гарантированно возвращает одинаковый результат, получая одинаковые аргументы, для всех строк в одном операторе?</p>	<p>MUTABLE VOLATILE PG_SLEEP *STABLE IMMUTABLE</p>
<p>Распределённой базой данных называют</p>	<p>*систему взаимодействующих между собой баз данных, которую пользователь информационной системы воспринимает как одну базу данных</p> <p>систему взаимодействующих между собой баз данных, которую пользователь информационной системы не воспринимает как одну базу данных</p> <p>нет правильного ответа</p> <p>систему взаимодействующих между собой баз данных, которую пользователь информационной системы воспринимает как одну или несколько баз данных</p> <p>систему взаимодействующих между собой баз данных, которую пользователь информационной системы не может воспринять</p>

<p>Уровень изоляции Repeatable read</p>	<p>Уровень, позволяющий предотвратить феномен повторяющегося чтения. При этом не видно в исполняющейся транзакции T2 измененные и удаленные записи другой транзакции T1.</p> <p>*Уровень, позволяющий предотвратить феномен неповторяющегося чтения. При этом не видно в исполняющейся транзакции T2 измененные и удаленные записи другой транзакции T1.</p> <p>Уровень, позволяющий предотвратить феномен неповторяющегося чтения. При этом не видно в неисполняющейся транзакции T2 записи другой транзакции T1.</p> <p>все варианты верны</p> <p>нет правильных вариантов</p>
<p>Как написать код корректно</p>	<p>*Не писать код, который приведет двоякому смыслу. (можно использовать ограничение целостности) ALTER TABLE accounts ADD CHECK amount >= 0;</p> <p>*Использовать один SQL-оператор (общие табличные выражения (CTE) или INSERT ON CONFLICT) Пользовательские блокировки (SELECT FOR UPDATE, LOCK TABLE)</p> <p>Использовать несколько SQL- операторов(общие табличные выражения (CTE) или INSERT ON CONFLICT) Пользовательские блокировки (SELECT FOR UPDATE, LOCK TABLE)</p> <p>Не писать код, который приведет двоякому смыслу. (нельзя использовать ограничение целостности) ALTER TABLE accounts ADD amount <= 0;</p> <p>нет правильного ответа</p>

<p>Особенности выполнения в PostgreSQL</p>	<p>*PostgreSQL позволяет определять функции, а у функций есть понятие категории изменчивости.</p> <p>PostgreSQL позволяет определять функции, а у функций нет понятия категории изменчивости.</p> <p>PostgreSQL позволяет не определять функции, у которых есть понятие категории изменчивости.</p> <p>PostgreSQL позволяет определять функции, у которых нет понятия категории изменчивости.</p> <p>нет правильного ответа</p>
<p>Что такое транзакции?</p>	<p>*Группа последовательных операций с базой данных, которая представляет собой логическую единицу работы с данными</p> <p>Логически завершённая банковская операция, в процессе осуществления которой происходит перевод определённой суммы денег с одного счёта на другой</p> <p>Атомарные операции с базой данных, производимые в единицу времени</p> <p>Сложные SELECT-запросы к транзакционным базам данных</p> <p>Логически объединённые операции с базой данных и запросы к ней</p>
<p>Какие аномалии предотвращает уровень изоляции Repeatable Read?</p>	<p>*Грязное чтение</p> <p>*Неповторяющееся чтение</p> <p>*Потерянные изменения</p> <p>Фантомное чтение</p> <p>Другие аномалии</p>

В чем особенности уровня изоляции Serializable?	<p>*Это самый высокий уровень изолированности</p> <p>*Транзакции на этом уровне изоляции полностью изолируются друг от друга, каждая выполняется так, как будто параллельных транзакций не существует</p> <p>*Уровень Serializable должен предотвращать вообще все аномалии</p> <p>Этот уровень не поддерживается некоторыми из популярных СУБД</p> <p>Все ответы верные</p>
Какие аномалии предотвращает уровень изоляции Read Committed?	<p>*Грязное чтение</p> <p>Неповторяющееся чтение</p> <p>*Потерянные изменения</p> <p>Фантомное чтение</p> <p>Другие аномалии</p>
Выберите правильный(е) ответ(ы) относительно модели MVCC	<p>*Операции чтения не блокируются операциями записи</p> <p>Операции чтения блокируются операциями записи</p> <p>*Операции записи не блокируются операциями чтения</p> <p>Операции записи блокируются операциями чтения</p> <p>Нет правильного ответа</p>

<p>Уровень изоляции Read Uncommitted</p>	<p>Уровень, позволяющий предотвратить феномен повторяющегося чтения. При этом не видно в исполняющейся транзакции T2 измененные и удаленные записи другой транзакции T1</p> <p>Уровень, имеющий самую лучшую согласованность данных, но самую низкую скорость выполнения транзакций. Каждая транзакция видит незафиксированные изменения другой транзакции (феномен грязного чтения)</p> <p>На этом уровня параллельно исполняющиеся транзакции видят только зафиксированные изменения из других транзакций. Таким образом, данный уровень обеспечивает защиту от грязного чтения</p> <p>*Уровень, имеющий самую плохую согласованность данных, но самую высокую скорость выполнения транзакций. Каждая транзакция видит незафиксированные изменения другой транзакции (феномен грязного чтения)</p> <p>Уровень, позволяющий предотвратить феномен неповторяющегося чтения. При этом не видно в исполняющейся транзакции T2 измененные и удаленные записи другой транзакции T1</p> <p>Нет правильного ответа</p>
<p>На каких уровнях изоляций допускается аномалия неповторяющееся чтение?</p>	<p>*Read Uncommitted</p> <p>*Read Committed</p> <p>Repeatable Read</p> <p>Serializable</p>

Выберите правильное утверждение о сериализации транзакций	<p>Результат успешной фиксации группы транзакций, выполняющихся последовательно, совпадает с результатом каждого из возможных вариантов упорядочения этих транзакций, если бы они выполнялись параллельно</p> <p>*Результат успешной фиксации группы транзакций, выполняющихся параллельно, не совпадает с результатом ни одного из возможных вариантов упорядочения этих транзакций, если бы они выполнялись последовательно</p> <p>Результат успешной фиксации группы транзакций, выполняющихся параллельно, совпадает с результатом каждого возможных вариантов упорядочения этих транзакций, если бы они выполнялись последовательно</p> <p>Результат успешной фиксации группы транзакций, выполняющихся параллельно, частично совпадает с результатами возможных вариантов упорядочения этих транзакций, если бы они выполнялись последовательно</p>
Какие проблемы решают транзакции?	<p>*Проблема конкуренции</p> <p>Проблема сегментации</p> <p>*Проблема целостности</p> <p>Проблема частого использования</p> <p>Все выше перечисленные</p>
Какие существуют уровни изоляции транзакций?	<p>Read committed</p> <p>Repeatable read</p> <p>Read uncommitted</p> <p>Serializable</p> <p>*Все выше перечисленные</p>
Какой уровень изоляции является самым устойчивым к аномалиям?	<p>Read committed</p> <p>Repeatable read</p> <p>Read uncommitted</p> <p>*Serializable</p> <p>Ни один</p>

Какая ситуация описывает взаимную блокировку транзакций?	<p>1) Случаи, когда две транзакции могут в ходе их обработки пытаться получить доступ к разным частям базы данных в разное время, препятствуя совершению операции.</p> <p>2) Случаи, когда две транзакции могут в ходе их обработки пытаться получить доступ к разным частям базы данных в одно и то же время, препятствуя совершению операции.</p> <p>3) Случаи, когда две транзакции могут в ходе их обработки пытаться получить доступ к одной и той же части базы данных в разное время, препятствуя совершению операции.</p> <p>*4) Случаи, когда две транзакции могут в ходе их обработки пытаться получить доступ к одной и той же части базы данных в одно и то же время, препятствуя совершению операции.</p> <p>5) Нет верного ответа</p>
Какие существуют модели одновременного конкурентного доступа?	<p>Прагматический</p> <p>*Пессимистический</p> <p>Агностический</p> <p>*Оптимистический</p> <p>Все вышеперечисленное</p>
Какие существуют особые условия изоляции транзакций?	<p>*Грязное чтение (Dirty Reads)</p> <p>Повторяемое чтение (Repeatable Reads)</p> <p>*Фантомное чтение (Phantom Reads)</p> <p>*Потерянное обновление (Lost Update)</p> <p>Все вышеперечисленное</p>
На каких уровнях изоляции не допускается аномалия фантомного чтения (Phantom Read)?	<p>Read committed</p> <p>Repeatable read</p> <p>Read uncommitted</p> <p>*Serializable</p> <p>Все вышеперечисленное</p>
Вставьте правильное слово в утверждение: ____ может произойти в случаях, когда две транзакции в ходе обработки пытаются получить доступ к одной и той же части базы данных в одно и то же время.	<p>Откат (Rollback)</p> <p>Прогон (Rollforward)</p> <p>*Взаимная блокировка (Deadlock)</p> <p>Commit</p> <p>Ничего из вышеперечисленного</p>

Сколько бит выделяется на идентификаторы команд в PostgreSQL	1 бит 2 бит 8 бит *32 бит
Какое значение уровня изоляции по умолчанию для PostgreSQL	Никакого Serializable *Read Committed Repeatable Read
Выберите протокол изоляции на основе снимков	Photo isolation *Snapshot Isolation Cinema isolation Снимки не выполняются
Какими способами можно определить функцию в PostgreSQL?	*CREATE FUNCTION имя RETURNS TABLE (имя_столбца тип_столбца [, ...]) *CREATE FUNCTION имя RETURNS тип_результата DELETE FUNCTION имя [RETURNS тип_результата RETURNS TABLE (имя_столбца тип_столбца [, ...])] CREATE FUNCTION [RETURNS тип_результата RETURNS TABLE (имя_столбца тип_столбца [, ...])] имя функции
Какое требование ACID гарантирует, что во время выполнения транзакции никакие параллельные транзакции не должны оказывать влияния на результат?	*Isolation; Durability; Atomicity; Consistency
Какой уровень изоляции не имплементирован в PostgreSQL?	Read Committed; *Read Uncommitted; Repeatable Read; Serializable
Аномалия "неповторяющееся чтение" возникает, когда:	При повторном чтении одна и та же выборка дает разные множества строк. Производится чтение данных, полученных в результате действия транзакции, которая после этого откатится. *В течение одной транзакции при повторном чтении данные оказываются перезаписанными. При обновлении поля двумя транзакциями одно из изменений теряется.

На каких уровнях изоляции может возникнуть аномалия "неповторяющееся чтение"?	*Read Committed; *Read Uncommitted; Repeatable Read; Serializable
С помощью какой команды начинается явная транзакция?	START *BEGIN START_COMMIT BEGIN_COMMIT такой команды нет
Какое особое условие возможно на уровне изоляции READ UNCOMMITTED и невозможно на уровне READ COMMITTED и в PostgreSQL?	Dirty Reads Non-repeatable read Phantom Reads Все из перечисленных *Никакое из перечисленных
Как называется особое условие изоляции при котором транзакции разрешено читать незафиксированную строку, которая была изменена другой транзакцией?	*Dirty Reads Non-repeatable read Phantom Reads Serialization anomaly Lost Update
Какая аномалия возможна на уровне изоляции READ COMMITTED и невозможно на уровне REPEATABLE READ?	Dirty Reads *Non-repeatable read Serialization anomaly Все из перечисленных Никакая из перечисленных
Какие утверждения являются верными в отношении транзакций?	*Транзакции обладают свойствами: Согласованность, Долговечность, Изолированность, Атомарность Транзакции обладают свойствами: Согласованность, Атомарность *Транзакции - группа последовательных операций с базой данных, которая представляет собой логическую единицу работы с данными Транзакции - сложные SELECT запросы к транзакционным базам данных *Транзакция может иметь два исхода: фиксация результат и отмена результата
Отметьте, какие особые условия возможны на уровне изоляции Serializable.	Dirty Read None-Repeatable Reads Phantom Reads Lost Update *Ничего из выше перечисленного

В какой последовательности появлялись методы блокировки транзакций	MVCC, 2PL, Snapshot Isolation, ARP *2PL, Snapshot Isolation, MVCC STP, Snapshot Isolation, MVVC Snapshot Isolation, ISO, CVVM, 2PL MVCC, 2PL, Snapshot Isoaltion
Характеристика изменчивости функции, при которой функция может делать все, что угодно, в том числе модифицировать базу данных	STABLE *VOLATILE IMMUTABLE PRIVATE SERIALIZIBLE
Выберите команду(-ы), указывающую(-ие) на то, что транзакция является явной	INSERT UPDATE DELETE *BEGIN *COMMIT *ROLLBACK
Выберите правильное(-ые) утверждение(-ия)	*На уровне изоляции Serializable невозможны следующие особые условия: Dirty Reads, None-Repeatable Reads, Phantom Reads, Lost Update. *На уровне изоляции Read Uncommitted возможны все особые условия (кроме PostgreSQL): Dirty Reads, None-Repeatable Reads, Phantom Reads, Lost Update. На уровне изоляции Read Committed возможно «Грязное» чтение. На уровне изоляции Serializable возможно Фантомное чтение. На уровне изоляции Repeatable Read невозможна аномалия сериализации (Lost Update).
Какое(-ие) особое(-ые) условие(-ия) невозможно(-ы) в PostgreSQL в отличие от других структурированных языков запросов	*Dirty Reads на уровне изоляции Read Uncommitted *Phantom Reads на уровне изоляции Repeatable Read Phantom Reads на уровне изоляции Read Committed Lost Update на уровне изоляции Repeatable Read None-Repeatable Reads на уровне изоляции Read Committed

Выберите правильное(-ые) утверждение(-ия)	<p>*Особенность сериализации транзакций результат успешной фиксации группы транзакций, выполняющихся параллельно, не совпадает с результатом ни одного из возможных вариантов упорядочения этих транзакций, если бы они выполнялись последовательно.</p> <p>Особенность сериализации транзакций результат успешной фиксации группы транзакций, выполняющихся параллельно, полностью совпадает с результатом всех возможных вариантов упорядочения этих транзакций, если бы они выполнялись последовательно.</p> <p>*Взаимная блокировка может произойти в случаях, когда две транзакции в ходе обработки пытаются получить доступ к одной и той же части базы данных в одно и то же время.</p> <p>Взаимная блокировка может произойти в случаях, когда две транзакции в ходе обработки пытаются получить доступ к разным частям базы данных в одно и то же время.</p>
Сколько исходов может иметь транзакция?	<p>1</p> <p>*2</p> <p>3</p> <p>4</p> <p>бесконечность</p>
Какой уровень изоляции транзакций является самым медленным?	<p>Скорость всех уровней одинакова</p> <p>Read uncommitted</p> <p>Read committed</p> <p>Repeatable read</p> <p>*Serializable</p>
Какой уровень изоляции допускает аномалию "Грязное чтение" в postgresSQL?	<p>*Никакой</p> <p>Read uncommitted</p> <p>Read committed</p> <p>Repeatable read</p> <p>Serializable</p>
Какая функция может делать все, что угодно, в том числе модифицировать базу данных и возвращать различные результаты при нескольких вызовах с одинаковыми аргументами?	<p>*VOLATILE</p> <p>STABLE</p> <p>IMMUTABLE</p> <p>MUTABLE</p> <p>PG_SLEEP</p>

Определение неявной транзакции	<p>*Задаёт любую отдельную инструкцию INSERT, UPDATE или DELETE как единицу транзакции Обычно это группа инструкций языка SQL, начало и конец которой обозначаются такими инструкциями, как BEGIN, COMMIT и ROLLBACK</p> <p>Задаёт любую отдельную инструкцию BEGIN, COMMIT или ROLLBACK как единицу транзакции Обычно это группа инструкций языка SQL, начало и конец которой обозначаются такими инструкциями, как INSERT, UPDATE и BEGIN</p> <p>Нет верного определения</p>
На каких уровнях изоляции возможна аномалия Phantom Reads (Фантомное чтение) в PostgreSQL?	<p>*READ UNCOMMITTED</p> <p>*READ COMMITTED</p> <p>REPEATABLE READ</p> <p>SERIALIZABLE</p> <p>Все вышеперечисленные</p>
Какой командой в PostgreSQL можно посмотреть уровень изоляции по умолчанию?	<p>SHOW transaction_isolation;</p> <p>*SHOW default_transaction_isolation;</p> <p>SHOW native_transaction_isolation;</p> <p>SHOW transaction_isolation_default;</p> <p>SHOW default_transaction;</p>
Продолжите предложение: Постоянная функция (IMMUTABLE) ...	<p>Может делать всё, что угодно, в том числе, модифицировать базу данных</p> <p>Не может модифицировать базу данных и гарантированно всегда возвращает одинаковые результаты для разных аргументов</p> <p>Не может модифицировать базу данных и гарантированно возвращает одинаковый результат, получая одинаковые аргументы, для всех строк в одном операторе</p> <p>*Не может модифицировать базу данных и гарантированно всегда возвращает одинаковые результаты для одних и тех же аргументов</p> <p>Нет верного продолжения</p>
Какой командой начинается транзакция?	<p>start;</p> <p>begin;*</p> <p>commit;</p> <p>transaction;</p>
Какой уровень изоляции не допускает аномалию "Грязное чтение"(Dirty reads), но допускает аномалию "неповторяемое чтение" (None repeatable reads)?	<p>Serializable</p> <p>Repeatable read</p> <p>Read uncommitted</p> <p>Read committed*</p>

Какой уровень изоляции отсусвет в СУБД PostgreSQL?	Serializable Repeatable read Read uncommitted* Read committed
Какой командой можно получить снимок данных в активной транзакции?	SELECT snapshot(); SELECT pg_current_xact_id(); SET TRANSACTION SNAPSHOT(); SELECT pg_current_snapshot());*
Команда создания точки сохранения	COMMIT BACK ROLLBACK * SAVEPOINT BEGIN
Какие аномалии допустимы на уровне Repeatable read в Postgres?	Потерянные изменения Грязное чтение Неповторяющееся чтение *Другие аномалии Фантомное чтение
Какая аномалия допускается на уровне Read Uncommitted, но не допускается на уровне Read Committed в Postgres?	Потерянные изменения Грязное чтение Неповторяющееся чтение *Никакая Фантомное чтение

На каком уровне изоляции не допускаются "другие аномалии"?	Read Uncommitted Read Committed Repeatable Read *Serializable Repeatable Unread
Что такое журнал транзакций?	*это часть БД, в которую поступают данные обо всех изменениях всех объектов БД Это записи в БД для документирования переводов Это файлы, содержащие системную информацию о работе бд Это физический документ, нужный для написания отчетов по работе бд
Установка уровня изоляции	*set transaction isolation level ... set isolation level... set ... isolation level set isolation level of transaction
Аномалия потерянное обновление	*при одновременном изменении одного блока данных разными транзакциями теряются одно из изменений при изменении поля изменение дублировалось при одновременном изменении одного блока данных разными транзакциями одна из них меняла не тот блок такой аномалии не существует
Для чего нужна взаимная блокировка транзакций	*В случае если они одновременно попытаются получить доступ к одному полю в БД Для разгрузки бд Для отладки функций Для осуществления безопасности на уровне изоляции
Какой исход может иметь транзакция?	*Изменения данных, успешно зафиксированы в базе данных Изменения данных, которые не записаны в базу данных *Транзакция отменяется, и отменяются все изменения, выполненные в ее рамках Транзакция отменяется, все изменения сохраняются в базу данных Все вышеперечисленное
Какое значение по умолчанию для изоляции используется в MySQL?	Read uncommitted Read committed *Repeatable read Serializable Repeatable write

Когда возникает аномалия грязного чтения?	<p>*Когда транзакция читает еще не зафиксированные изменения, сделанные другой транзакцией</p> <p>Когда транзакция читает одну и ту же строку два раза, а в промежутке между чтениями вторая транзакция изменяет (или удаляет) эту строку и фиксирует изменения</p> <p>Когда две транзакции читают одну и ту же строку таблицы, затем одна транзакция обновляет эту строку, после чего вторая транзакция обновляет эту же строку, не учитывая изменений, сделанных первой транзакцией</p> <p>Когда одна транзакция два раза читает набор строк по одинаковому условию, а в промежутке между чтениями другая транзакция добавляет строки, удовлетворяющие этому условию, и фиксирует изменения</p> <p>Когда одна транзакция читает уже зафиксированную строку второй транзакции и изменяет ее</p>
Когда происходит взаимная блокировка транзакций?	<p>*Когда две транзакции в ходе обработки пытаются получить доступ к одной и той же части базы данных в одно и то же время</p> <p>Когда две транзакции одновременно пытаются получить доступ к разным частям базы данных</p> <p>Когда две транзакции в разное время пытаются получить доступ к одной и той же части базы данных</p> <p>Во всех перечисленных случаях</p> <p>Когда две транзакции завершаются в одно и то же время</p>
Какие существуют виды транзакций?	<p>Явная транзакция</p> <p>Неявная транзакция</p> <p>*Обе существуют</p> <p>Обе не существуют</p>
Отметьте существующие уровни изоляции	<p>*Read Committed</p> <p>*Repeatable Read</p> <p>Look Committed</p> <p>*Serializable</p>
Какой уровень поддерживает грязное чтение?	<p>Read Committed</p> <p>Repeatable Read</p> <p>*Read Uncommitted</p> <p>Serializable</p>
Полноценная транзакция...	<p>начинается с BEGIN, объявления уровня изоляции (если нужно) и заканчивается ROLLBACK</p> <p>*начинается с BEGIN, объявления уровня изоляции (если нужно) и заканчивается COMMIT</p> <p>начинается с BEGIN и заканчивается COMMIT</p> <p>начинается с BEGIN и заканчивается ROLLBACK</p>

<p>Существует сервис по считыванию подарочных купонов. Злоумышленник пытается активировать один и тот же купон несколько раз. Какой уровень изоляции транзакций не позволит это сделать?</p>	<p>*Read Committed Readable Repeat Read Uncommitted Никакой, это делается с помощью дополнительного ПО</p>
<p>Аномалия фантомного чтения устраняется уровнем изоляции...</p>	<p>Serizalizable Read Unreadable Read Committed *Synchronized</p>
<p>Какая аномалия устраняется на любом уровне изоляции?</p>	<p>Грязное чтение *Потерянные изменения Фантомное чтение Неповторяющееся чтение</p>
<p>Транзакция - это...</p>	<p>*Совокупность операций над базой данных, которые вместе образуют логически целостную процедуру, и могут быть либо выполнены все вместе, либо не будет выполнена ни одна из них. Операция над чьим-либо счётом, в результате которой на счёте произошли изменения. Совокупность операций над базой данных, которые вместе образуют логически целостную процедуру, и могут быть выполнены частично. Минимальная логически осмысленная операция, которая имеет смысл и может быть совершена только полностью.</p>
<p>Если команде set transaction не предшествует begin или start transaction она сделает:</p>	<p>Крашнет сервер *Выдаст предупреждение Вызовет аномалию и с большой вероятностью выполнит неточную операцию Ничего не выведет и ничего не сделает</p>

Выберите определение аномалии "потерянное обновление":	<p>Чтение данных, полученных в результате действия транзакции, которая после этого откатится.</p> <p>*При обновлении поля двумя транзакциями одно из изменений теряется.</p> <p>В течение одной транзакции при повторном чтении данные оказываются перезаписанными.</p> <p>В течение одной транзакции при повторном чтении одна и та же выборка дает разные множества строк.</p>
В некоторых случаях, две транзакции могут в ходе их обработки пытаться получить доступ к одной и той же части базы данных в одно и то же время, таким образом, что это будет препятствовать их совершению. Как называется это явление?	<p>Read Lock</p> <p>Write Lock</p> <p>*Deadlock</p> <p>Update Lock</p>
Целостность БД это:	<p>*А.соответствие имеющейся в базе данных информации её внутренней логике, структуре и всем явно заданным правилам</p> <p>В.Структурированность и возможность хранить данные</p> <p>С.соответствие имеющейся в базе данных информации её внешней логике</p> <p>Д.все вместе</p>
Какая аномалия возможна при уровне изоляции Repeatable Read	<p>*Аномалия сериализации</p> <p>Неповторяемое чтение</p> <p>Грязное чтение</p>
Уровень Serializable:	<p>Предотвращает все аномалии</p> <p>Сериализует транзакции</p> <p>Чаще выдает ошибки, чем другие уровни доступа</p> <p>*Все вместе</p>

Функция STABLE:	*А.не может модифицировать базу данных и гарантированно возвращает одинаковый результат, получая одинаковые аргументы, для всех строк в одном операторе В.может модифицировать базу данных и гарантированно возвращает одинаковый результат, получая одинаковые аргументы, для всех строк в одном операторе С.не может модифицировать базу данных и гарантированно возвращает одинаковый результат, получая одинаковые аргументы, не для всех строк в одном операторе D.не может модифицировать базу данных и не гарантированно возвращает одинаковый результат, получая одинаковые аргументы, для всех строк в одном операторе
Сколько бит выделяется на идентификаторы транзакций в PostgreSQL?	*32 бита 16 бит 8 бит 4 бита 32 байта
Что возвращает функция экспорта снимка? (Команда pg_export_snapshot)	*id снимка Предупреждение Ничего не возвращает Транзакцию со снимком данных Уровень изоляции
Snapshot Isolation - это...	Скриншот Протокол изоляции на основе транзакций *Протокол изоляции на основе снимков Расширение над MVCC Изоляция снимка
Какие аномалии допускаются на уровне изоляции Repeatable Read?	*Фантомное чтение Неповторяющееся чтение Грязное чтение Потерянные изменения *Другие аномалии
Что такое MVCC?	Паттерн проектирования *Механизм СУБД для обеспечения параллельного доступа к БД Методология разработки Средство создания транзакций

Какого уровня изоляции не существует?	Read Uncommitted Serializable *Transaction revert Repeatable Read
Какую аномалию уровень изоляции Read Uncommitted обрабатывает?	Грязное чтение Неповторяющееся чтение Фантомное чтение *Потерянное обновление
Какой категории изменчивости функции не существует?	VOLATILE STABLE *SIMPLE IMMUTABLE
С помощью какой команды можно начать транзакцию?	UPDATE COMMIT *BEGIN START
Какие особые условия существуют при работе транзакций?	*"Грязное" чтение (Dirty Reads) *Неповторяемое чтение (None-Repeatable Reads) *Фантомное чтение (Phantom Reads) *Потерянное обновление (Lost Update) Непоследовательная запись (Inconsistent entry)
Какие аномалии допускаются на уровне изоляции Read Committed?	Грязное чтение *Неповторяемое чтение *Фантомное чтение Потерянные изменения
Какая категория изменчивости стоит по умолчанию?	*VOLATILE STABLE IMMUTABLE Нет правильного ответа
Отметьте все свойства транзакций	*атомарность (Atomicity) *согласованность (Consistency) *изолированность (Isolation) *долговечность (Durability) продолжительность (Continuance)

Сколько существует различных подходов к реализации изолированности транзакций?	1 *2 3 4 5
Отметьте аномалии, которые предотвращает уровень изоляции транзакций Read Uncommitted	Грязное чтение Неповторяющееся чтение *Потерянные изменения Фантомное чтение Другие аномалии
Как устронить несогласованное чтение (уровень read comited)?	*Использовать только один оператор select Использовать минимум два оператора select Не использовать оператор select Использовать pg_sleep() Ни один из ответов не является верным
Что такое неявная транзакция?	обычно это группа инструкций языка SQL, начало и конец которой обозначаются такими инструкциями, как BEGIN, COMMIT и ROLLBACK. обычно это группа инструкций языка SQL, начало и конец которой обозначаются такими инструкциями, как BEGIN и ROLLBACK. задает любую отдельную инструкцию INSERT или UPDATE как единицу транзакции. *задает любую отдельную инструкцию INSERT, UPDATE или DELETE как единицу транзакции. обычно это группа инструкций языка SQL, начало и конец которой обозначаются такими инструкциями, как COMMIT и ROLLBACK.

Самый высший уровень изоляции транзакций это....	Read committed Read uncommitted *Serializable Repeatable read Unrepeatable read
Что такое 2PL?	протокол изоляции на основе снимков *двухфазная блокировка уровень изоляции управление параллельным доступом посредством многоверсионности аномалия
Какие категории изменчивости функций существуют?	*STABLE *VOLATILE *IMMUTABLE HARD RANDOM
Система взаимодействующих между собой баз данных, которую пользоваель воспринимает как одну БД называют:	*распределенной базой данных централизованной базой данных совмещенной базой данных синхронизированными базами данных
Какой уровень изоляции по умолчанию стоит в MySQL?	*Repeatable Read Read Committed Read Uncommitted Serializable
Какие аномалии предотвращает уровень изоляции Serializable?	*все перечисленные Грязное чтение Потерянные изменения Фантомное чтение
Как в английской терминологии называется термин "Взаимная блокировка транзакций"?	*deadlock deadlocks deadline deadtransaction
Согласно теореме CAP, в любой реализации распределённых вычислений возможно обеспечить не более двух из трёх свойств. Какие эти 3 свойства?	*Согласованность данных *Доступность *Устойчивость к разделению Долговечность Постоянность доступа

Существующими подходами к реализации уровней изоляции транзакций являются:	<ul style="list-style-type: none"> *Блокирование *Версионирование Копирование Соответствие Ветвление
Какие аномалии присутствуют на уровне изоляции Repeatable Read?	<ul style="list-style-type: none"> *Фантомное чтение *Другие аномалии Потерянные изменения Грязное чтение Никакие
Какие существуют категории изменчивости функций в PostgreSQL?	<ul style="list-style-type: none"> *Изменчивая функция *Стабильная функция *Постоянная функция Временная функция Периодическая функция
Транзакция это	<ul style="list-style-type: none"> *совокупность операций над базой данных блок информации в базе данных изменения данных, успешно зафиксированы в базе данных зарезервированное слово, или символ, который используется в SQL выражениях
Чем сильнее уровень изоляции, тем	<ul style="list-style-type: none"> *меньше производительность базы данных больше памяти требуется для работы базой данных меньше памяти требуется для работы с базой данных больше производительность базы данных
Какая из блокировок транзакций появилась ранее остальных?	<ul style="list-style-type: none"> *2PL - двухфазная блокировка Snapshot Isolation - протокол изоляции на основе снимков MVCC - управление параллельным доступом посредством многоверсионности Read-Copy-Update - механизм синхронизации в многопоточных системах.

Особенность сериализации транзакции.	<p>*Результат успешной фиксации группы транзакций, выполняющихся параллельно, не совпадает с результатом ни одного из возможных вариантов упорядочения этих транзакций, если бы они выполнялись последовательно.</p> <p>Результат успешной фиксации группы транзакций, выполняющихся параллельно, совпадает с результатом одного из возможных вариантов упорядочения этих транзакций, если бы они выполнялись последовательно.</p> <p>Результат успешной фиксации группы транзакций оказывается несогласованным при всевозможных вариантах исполнения этих транзакций по очереди.</p> <p>Транзакция повторно выполняет запрос, возвращающий набор строк для некоторого условия, и обнаруживает, что набор строк, удовлетворяющих условию, изменился из-за транзакции, завершившейся за это время.</p>
Виды транзакций	<p>*Неявная транзакция</p> <p>*Явная транзакция</p> <p>Мнимая транзакция</p> <p>Прямая транзакция</p> <p>Закрытая транзакция</p>
Какие аномалии предотвращает уровень изоляции Repeatable Read в PostgreSQL?	<p>*Грязное чтение</p> <p>*Неповторяемое чтение</p> <p>*Фантомное чтение</p> <p>Аномалия сериализации</p>
На каких уровнях изоляции допускается аномалия фантомного чтения?	<p>*Read Uncommitted</p> <p>*Read Committed</p> <p>*Repeatable Read</p> <p>Serializable</p> <p>На всех уровнях</p>

Взаимная блокировка транзакций это	*случай, при котором транзакции обращаются к одному участку базы данных в одно и то же время, таким образом, что будет препятствовать их завершению случай, при котором транзакции обращаются к разным участкам базы данных в одно и то же время, таким образом, что будет препятствовать их завершению случай, при котором транзакции обращаются к одному участку базы данных в одно и то же время, таким образом, что не будет препятствовать их завершению случай, при котором транзакции обращаются к одному участку базы данных в разное время, таким образом, что будет препятствовать их завершению
Предпосылки для появления транзакций	*Проблема конкуренции *Проблема целостности Проблема изменчивости Проблема сохранн Ни один из предложенных
Какой уровень транзакций имеет самую плохую согласованность, но самую высокую скорость выполнения транзакций?	*Serializable Read Uncommitted Read Committed Repeatable Read Ни один из предложенных
Какой уровень транзакций должен предотвращать вообще все аномалии?	*Serializable Read Uncommitted Read Committed Repeatable Read Ни один из предложенных

<p>Выберите определение категории изменчивости функции volatile</p>	<p>* Изменчивая функция может делать всё, что угодно, в том числе, модифицировать базу данных. Она может возвращать различные результаты при нескольких вызовах с одинаковыми аргументами. Оптимизатор не делает никаких предположений о поведении таких функций. В запросе, использующем изменчивую функцию, она будет вычисляться заново для каждой строки, когда потребуется её результат.</p> <p>Стабильная функция не может модифицировать базу данных и гарантированно возвращает одинаковый результат, получая одинаковые аргументы, для всех строк в одном операторе. Эта характеристика позволяет оптимизатору заменить множество вызовов этой функции одним. В частности, выражение, содержащее такую функцию, можно безопасно использовать в условии поиска по индексу.</p> <p>Постоянная функция не может модифицировать базу данных и гарантированно всегда возвращает одинаковые результаты для одних и тех же аргументов. Эта характеристика позволяет оптимизатору предварительно вычислить функцию, когда она вызывается в запросе с постоянными аргументами.</p> <p>Подходят все определения</p> <p>Ни одно из описанных определений не подходит</p>
<p>Какое свойство транзакций гарантирует, что никакая транзакция не будет зафиксирована в системе частично?</p>	<p>Изолированность</p> <p>Согласованность</p> <p>*Атомарность</p> <p>Такого свойства не существует</p> <p>Долговечность</p>
<p>Какой аномалии не может возникнуть при параллельном выполнении транзакций?</p>	<p>"Грязное" чтение</p> <p>Фантомное чтение</p> <p>*Аномалия арнольда киари</p> <p>Неповторяющееся чтение</p> <p>Потерянное обновление</p>
<p>На каком уровне изоляции допускается аномалия "грязное" чтение?</p>	<p>Read committed</p> <p>Repeatable read</p> <p>*Read uncommitted</p> <p>Serializable</p> <p>Никакой из перечисленных</p>

Что применяется для обеспечения сериализации?	Восстановление объектов Удаление объектов Чтение объектов * "Захват" объектов * "Освобождение" объектов
Какой инструкцией обозначается начало транзакции?	ROLLBACK INSERT COMMIT *BEGIN UPDATE
Какие могут быть аномалии на уровне изоляции Read Uncommitted	Потерянные изменения *Грязное чтение *Неповторяющееся чтение *Фантомное чтение *Другие аномалии
На каком уровне изоляции допускается аномалия "Потерянное обновление"	Read committed Repeatable read Read uncommitted Serializable *Ни на одном уровне
Какие могут быть аномалии на уровне изоляции Repeatable Read	Потерянные изменения Грязное чтение Неповторяющееся чтение *Фантомное чтение *Другие аномалии
Явная транзакция:	Задаёт любую отдельную инструкцию INSERT, UPDATE или DELETE как единицу транзакции * Группа инструкций языка SQL, начало и конец которой обозначаются такими инструкциями как BEGIN, COMMIT и ROLLBACK Логически завершённая банковская операция, в процессе осуществления которой происходит перевод определённой суммы денег с одного счёта на другой. Оба определения верны Нет правильного ответа

Какой уровень изоляции имеет наибольший риск блокировки?	Read Committed Read Uncommitted Repeatable Read * Serializable Нет правильного ответа
Какой уровень изоляции в SQL Server устанавливается по умолчанию?	* Read Committed Read Uncommitted Repeatable Read Serializable Нет правильного ответа
Студент создает транзакцию внутри транзакции. Его внешняя транзакция откатывает данные, а внутренняя транзакция фиксирует данные. Каким будет результат?	Обе транзакции будут зафиксированы Будет зафиксирована только внешняя транзакция * Обе транзакции совершат откат Будет зафиксирована только внутренняя транзакция Завершится сеанс работы в SQL Server
Укажите свойства транзакций	*Атомарность *Согласованность *Изолированность *Долговечность У транзакций нет свойств
Какие особые условия могут быть на уровне изоляции Read committed?	Грязное чтение *Неповторяемое чтение *Фантомное чтение *Аномалия сериализации Никаких особых условий быть не может

Когда возникает аномалия неповторяющегося чтения?	<p>*Когда в течении одной транзакции при повторном чтении данные оказываются перезаписанными</p> <p>Когда в течении одной транзакции при повторном чтении данные совпадают</p> <p>Когда при повторном чтении одна и та же выборка дает разные множества строк</p> <p>При чтении данных, полученных в результате действия транзакции, которая после этого откатится</p> <p>При обновлении поля двумя транзакциями одно из изменений теряется</p>
Что означает долговечность в свойствах транзакций ACID?	<p>1)Данные, сохраненные в результате транзакции, сохраняются надолго</p> <p>2)Изменения, сохраненные в результате транзакции, не могут быть изменены другими транзакциями</p> <p>*3)Изменения, сохраненные в результате транзакции, остаются сохраненными после возвращения системы в работу в результате сбоя</p> <p>4)Другие транзакции не должны оказывать влияния на результат данной транзакции</p> <p>5)Никакая транзакция не будет сохранена в системе частично</p>
Сколько уровней изоляции существует согласно стандарту SQL?	<p>1)1</p> <p>2)2</p> <p>3)3</p> <p>*4)4</p> <p>5)5</p>
Что НЕ могут делать остальные пользователи при write lock?	<p>1)Считать информацию</p> <p>2)Записать информацию</p> <p>*3)Считать и записать информацию</p> <p>4)Ничего не могут</p> <p>5)Write lock не ограничивает других пользователей</p>
На каких уровнях изоляции возможно появление артефакта "неповторяющееся чтение"?	<p>*1)Read Uncommitted</p> <p>*2)Read Committed</p> <p>3)Repeatable Read</p> <p>4)Serializable</p> <p>5)Ни на одном</p>
Какие исходы могут быть у транзакции:	<p>*1)изменения зафиксированы в базе данных</p> <p>2)изменения зафиксированы частично: только несколько строк из всей выборки</p> <p>*3)транзакция отменяется и отменяются все изменения</p> <p>4)транзакция отменяется, но изменения сохраняются в базе данных</p> <p>5)транзакция успешна, но данные были сохранены с ошибкой</p>

На каких из перечисленных уровней доступна реализация аномальной сериализации	<ul style="list-style-type: none"> *1)Read uncommitted *2)Read committed *3)Repeatable read 4)Serializable 5)ни на одном из вышеперечисленных
На каких из приведённых уровнях доступна реализация фантомного чтения?	<ul style="list-style-type: none"> *1)Read uncommitted *2)Read committed *3)Repeatable read 4)Serializable 5)ни на одном из вышеперечисленных
Что из нижеперечисленных вариантов может описать понятие Deadlock ?	<ul style="list-style-type: none"> *1)Это ситуация, при которой две транзакции блокируют друг друга, и ни одна из них не может продолжать свое выполнение. 2)Это ситуация при которой одна транзакция не может быть выполнена по внутренней ошибке базы данных 3)Это термин, описывающий блокировку отката транзакции при вызванных внутренними ошибках базы данных 4)Это термин, описывающий блокировку доступа пользователю при попытке удаленного взлома сервера базы данных 5)Это ситуация в которой теряется соединение с базой данных при многочисленных одинаковых запросов
В чем заключается проблема целостности	<ul style="list-style-type: none"> 1)Аппаратные устройства работают исправно 2)*Все операторы SQL работают исправно 3)Все сотрудники отдела работают исправно 4)Механизм имеет уязвимости 5)Механизм не имеет уязвимостей
Какая команда требуется для выбора нужного уровня изоляции?	<ul style="list-style-type: none"> 1)SELECT * FROM 2)ALTER TABLE 3)*SET TRANSACTION 4)ISOLATION LEVEL 5)UNCOMMITTED
С какой команды начинается транзакция?	<ul style="list-style-type: none"> 1)START 2)UPDATE 3)*BEGIN 4)GO 5)SET

Особенности сериализации транзакций?	<p>1)Результат успешной фиксации группы транзакций, выполняющихся последовательно, совпадает с результатом ни одного из возможных вариантов упорядочения этих транзакций</p> <p>2)*Результат успешной фиксации группы транзакций, выполняющихся параллельно, не совпадает с результатом ни одного из возможных вариантов упорядочения этих транзакций</p> <p>3)Результат успешной фиксации группы транзакций, выполняющихся последовательно, не совпадает с результатом ни одного из возможных вариантов упорядочения этих транзакций</p> <p>4)результат провала фиксации</p> <p>5)Результат успешной фиксации одной транзакции</p>
Назовите команды для начала и завершения транзакций:	<p>1.* BEGIN и COMMIT</p> <p>2. CUMIN и BAGGIT</p> <p>3. INSERT и UPDATE</p> <p>4. SAVEPOINT и ROLLBACK TO</p> <p>5. START и END</p>
Назовите два глобально различных подхода к реализации изолированности:	<p>1.* Версионирование (snapshot) и Блокирование (lock)</p> <p>2. Сериализуемость и цикличность</p> <p>3. Атомарность и нецелостность</p> <p>4. Блокирование и разблокирование</p> <p>5. Обновляемость и актуальность</p>
Какой уровень предотвращает все аномалии?	<p>1.*Serializable</p> <p>2. Read uncommitted</p> <p>3. Read committed</p> <p>4. Repeatable read</p> <p>5. Ни один</p>
На каких из приведённых уровнях можно реализовать грязное чтение?	<p>*1)Read uncommitted</p> <p>2)Read committed</p> <p>3)Repeatable read</p> <p>4)Serializable</p> <p>5)ни на одном из вышеперечисленных</p>
Что такое транзакция?	<p>*1. это совокупность операций над базой данных, которые вместе образуют логически целостную процедуру, и могут быть либо выполнены все вместе, либо не будет выполнена ни одна из них. 2. это операция, состоящая в переводе денежных средств с одного счёта на другой; сделка купли-продажи. 3. это операция по выдаче наличных денег или предоставление другого сервиса через банкомат. 4. это группа логически объединённых последовательных операций по работе с данными, обрабатываемая или отменяемая целиком. 5. минимальная логически осмысленная операция, которая имеет смысл и может быть совершена только полностью.</p>

Какие есть особенности в PostgreSQL?	1. команды определены в стандарте SQL 2. PostgreSQL поддерживает расширения функционала и уровней доступа 3. нет особенностей *4. команды определены в стандарте SQL, за исключение режима транзакции DEFERRABLE и формы SET TRANSACTION SNAPSHOT, которые являются расширениями PostgreSQL 5. PostgreSQL позволяет запускать автоматический хостинг для баз данных, чтобы проверять изменение, добавление, удаление данных
Что такое 2PL?	1. второй игрок 2. двухместный лот 3. двухфазное расположение *4. двухфазная блокировка 5. двухпротокольная блокировка
Какие категории изменчивости бывают у функций?	1. CHANGABLE, STABLE И UNSTABLE 2. STABLE, UNSTABLE *3. VOLATILE, STABLE и IMMUTABLE 4. MUTABLE, STABLE, VOLATICE 5. MUTABLE, IMMUTABLE, STABLE
Выберите правильный жизненный цикл транзакции:	*1) Последовательность "безопасных" операций -> Установка точки сохранения -> Последовательность "рискованных" операций -> Откат транзакции до точки сохранения -> Фиксация результатов "безопасных" операций" 2) Создание базы данных -> Начальное состояние -> Начало транзакции -> Последовательность "безопасных" операций" -> Сохранение, установка точки восстановления -> Выполнение операций редактирования базы данных -> Фиксация результатов изменения -> Продолжение работы с базами данных 3) У транзакции отсутствует жизненный цикл 4) Нет верного варианта ответа 5) Создание базы данных -> Начальное состояние -> Создание транзакции -> Начало транзакции -> Выполнение операций редактирования/изменения -> Фиксация, сохранение текущих параметров -> Продолжение выполнений операций редактирования/изменения -> Проверка условий -> Фиксация результатов
Из-за чего может снизиться производительность БД?	1) Из-за перегруженности созданных таблиц 2) Частые обновления значений внутри созданных таблиц 3) Миграции уже созданных баз данных в текущую сессию *4) Из-за уровня изоляции, транзакциям приходится ждать друг друга 5) Все из перечисленного

Когда происходит фантомное чтение?	<p>1) Когда транзакция обращается к данным, и данные были изменены, и эта модификация не была зафиксирована в базе данных</p> <p>*2) Когда транзакция не выполняется независимо, такое как первая транзакция, которая модифицирует данные в таблице, эта модификация включает в себя все строки данных в таблице</p> <p>3) До завершения текущей транзакции, когда альтернативная транзакция обращается к тем же данным, затем, между двумя считанными данными в первой транзакции, вследствие модификации второй транзакции, данные, считанные первой транзакцией, могут отличаться</p> <p>4) 2 и 3 варианты совместно</p> <p>5) Нет верного ответа</p>
Какой из утверждений про уровень изоляции SERIALIZABLE является верным?	<p>1) Устанавливает разделяемые блокировки на все считываемые данные и удерживает эти блокировки до тех пор, пока транзакция не будет подтверждена или отменена</p> <p>2) Представляет собой простую форму изоляции между транзакциями, поскольку он вообще не изолирует операции чтения от других транзакций</p> <p>*3) Является самым строгим, так как не допускает возникновения всех четырех проблем параллельного одновременного конкурентного доступа</p> <p>*4) Доступен только в пессимистической модели одновременного конкурентного доступа</p> <p>5) Нет верного ответа</p>
Как называют систему взаимодействующих между собой баз данных, которую пользователь информационной системы воспринимает как одну базу данных?	<p>1) Реляционная БД</p> <p>*2) Распределенная БД</p> <p>3) Нереляционная БД</p> <p>4) Централизованная БД</p> <p>5) Транзакция</p>
Какой уровень изоляции соответствует описанию: Транзакции могут выполняться только одна за одной; Медленная эффективность выполнения; В классическом представлении избавляет от эффекта чтения фантомов.	<p>1) Read uncommitted</p> <p>2) Read committed</p> <p>3) Repeatable read</p> <p>*4) Serializable</p> <p>5) SFTP</p>
На каких двух уровнях изоляции не допускается неповторяющееся чтение?	<p>1) Read uncommitted, Read committed</p> <p>2) Read committed, Repeatable read</p> <p>*3) Repeatable read, Serializable</p> <p>4) Serializable, Read uncommitted</p> <p>5) Read uncommitted, Repeatable read</p>

Какая аномалия описывается ниже: чтение данных, добавленных или изменённых транзакцией, которая впоследствии не подтвердится (откатится).	1) Потерянное обновление *2) Грязное чтение 3) Неповторяющееся чтение 4) Фантомное чтение 5) Фантомная боль
Когда требуется транзакция?	*1) В случае предельно важных изменений в записях БД 2) Когда необходимо выполнить ветвление для параллельной разработки 3) Ни один из перечисленных *4) Когда требуется контроль точности выполнения операций 5) Для упрощенного доступа к логическим операциям в БД
Для чего нужны отложенные ограничения?	*1) Для более гибкой работы с транзакциями 2) Для быстрого форматирования таблиц в БД 3) Для выполнений команд без предупреждений об ошибках 4) Ни один из вариантов 5) Все варианты
Какой командой можно включить уровень изоляции RCSI (READ COMMITTED SNAPSHOT) при монопольном доступе к базе?	1) ALTER DATABASE SET ALLOW_SNAPSHOT_ISOLATION ON 2) ALTER DATABASE SET READ_COMMITTED_SNAPSHOT ON WITH ROLLBACK AFTER X SECONDS *3) ALTER DATABASE SET READ_COMMITTED_SNAPSHOT ON 4) ALTER DATABASE SET ON 5) Правильного ответа нет
Как разрешить ситуацию с взаимной блокировкой?	1) Отменить изменения обеих транзакций, дабы предотвратить взаимную блокировку 2) Продолжить внесение изменений, игнорируя сообщения о блокировках 3) Вызвать обработчик исключений *4) Выполнить откат одной из транзакций для подтверждения изменений, затем выполнить изменения в откатанной ветке транзакций 5) Ничего из перечисленного
Какие исходы может иметь транзакция?	*Изменение данных *Отмена транзакций, отмена изменений Транзакция не имеет исходов Исход только один, только отмена изменений Исход только один, только изменение данных

Что такое COMMIT в SQL?	*COMMIT - оператор управления транзакциями языка SQL для успешного завершения транзакции COMMIT - это аналог commit из github в SQL COMMIT - подтверждение запросов в PostgreSQL COMMIT - подтверждение запросов в MariaDB COMMIT - подтверждение запросов в MySQL
Какие проблемы возможны при параллельном выполнении транзакций?	*Появление "грязного" чтения Появление чтения "усреднённого" уровня *Появление "фантомного" чтения *Появление "потерянного" обновления *Появление "неповторяющегося" чтения
Что такое deadlocks?	*Deadlocks - взаимная блокировка транзакций Deadlocks - односторонняя блокировка транзакций Deadlocks - безответная блокировка транзакций Deadlocks - невзаимная блокировка транзакций Deadlocks - трёхсторонняя блокировка транзакций
Какое из перечисленных свойств не характерно транзакциям?	Атомарность Согласованность Изолированность *Полиморфизм Долговечность
Какой из уровней изоляций может иметь такое особое условие как "Грязное чтение"?	*Read uncommitted Read committed Repeatable read Serializable
Какой из уровней должен предотвращать вообще все аномалии?	Read uncommitted Read committed Repeatable read *Serializable
Какие из следующих аномалий свойственны такому уровню изоляции, как Read committed?	Потерянные изменения Грязное чтение *Неповторяющееся чтение *Фантомное чтение
С помощью какой команды фиксируются и записываются все действия внутри транзакции?	UPDATE *COMMIT SET ROLLBACK

Самым низким уровнем изоляции транзакции является:	*Read uncommitted Read committed Repeatable read Serializable
Протокол изоляции на основе снимков это:	Screenshot Seclusion Image Insulation *Snapshot Isolation Picture Segregation
Взаимная блокировка транзакций это:	crossblock mutualdisable *deadlocks reciprocaloff
1	BEGIN START *ROLLBACK UPDATE Нет такой команды
Какой уровень имеет самую высокую скорость выполнения транзакций?	*Read Uncommitted Read committed Repeatable read Serializable Все уровни имеют одинаковую скорость
Что такое Snapshot Isolation?	Двухфазная блокировка Управление параллельным доступом посредством многоверсионности *Протокол изоляции на основе снимков Аномалия Уровень изоляции

Выберите верное утверждение	<p>Особенность сериализации транзакций результат успешной фиксации группы транзакций, выполняющихся параллельно, полностью совпадает с результатом всех возможных вариантов упорядочения этих транзакций, если бы они выполнялись последовательно.</p> <p>*Особенность сериализации транзакций результат успешной фиксации группы транзакций, выполняющихся параллельно, не совпадает с результатом ни одного из возможных вариантов упорядочения этих транзакций, если бы они выполнялись последовательно.</p> <p>Взаимная блокировка может произойти в случаях, когда две транзакции в ходе обработки пытаются получить доступ к разным частям базы данных в одно и то же время.</p> <p>*Взаимная блокировка может произойти в случаях, когда две транзакции в ходе обработки пытаются получить доступ к одной и той же части базы данных в одно и то же время.</p> <p>*На уровне изоляции Read Uncommitted возможны все особые условия (кроме PostgreSQL): Dirty Reads, None-Repeatable Reads, Phantom Reads, Lost Update.</p>
Что предполагает модель пессимистического одновременного конкурентного доступа?	<p>*1)предполагает, что между двумя или большим количеством процессов в любое время может возникнуть конфликт и поэтому блокирует ресурсы, как только они потребуются в течение периода транзакции.</p> <p>2) предполагает, что модель может включать одновременно половину процессов</p> <p>3) предполагает, что модель может наполовину заблокировать ресурсы, при возникновении конфликта</p> <p>4) предполагает, что в модели не будут возникать конфликты</p> <p>5) предполагает, что ресурсы не будут блокироваться при возникновении конфликта</p>
Что такое режим транзакции Deferrable?	<p>1)Режим чтения</p> <p>*2)Режим отложенных ограничений</p> <p>3)Режим записи</p> <p>4)Режим грязного чтения</p> <p>5)Режим фантомного чтения</p>
На каких уровнях допускается неповторяющееся чтение?	<p>*1)Read Uncommitted</p> <p>2) Repeatable Read</p> <p>*3)Read Committed</p> <p>4)Serializable</p> <p>5)Все вышеперечисленное</p>
Что такое:"Это ситуация, при которой две транзакции блокируют друг друга, и ни одна из них не может продолжать свое выполнение."?	<p>1)Deadlift</p> <p>2)Deadbydaylight</p> <p>*3)Deadlock</p> <p>4) Deadlog</p> <p>5)Deadinside</p>

Каким сокращением обозначаются все свойства?	*ACID AQIT ASID ACIT OCID
Уровни изоляции транзакций	*Read uncommitted, Read committed, Repeatable read, Serializable Read commit, Write uncommitted, Repeatable, Serial Update read , Read commit, Repeatable read, Serializable Rollback read, Read committed, Start read, Serializable Read uncommitted, Phantom read, None-Repeatable read, Serializable
Назовите все аномалии?	*Потерянное обновление, грязное чтение, неповторяющееся чтение, фантомное чтение Найденное обновление, чистое чтение, повторяющее чтение, явное чтение Потерянное обновление, чистое чтение, повторяющее чтение, фантомное чтение Найденное обновление, грязное чтение, неповторяющееся чтение, явное чтение Потерянное обновление, грязное чтение, повторяющее чтение, фантомное чтение
Какой является функция VOLATILE?	*Изменчивой функцией Неизменчивой функцией Читаемой функцией Нечитаемой функцией Грязной функцией