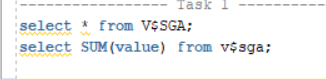
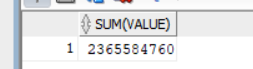
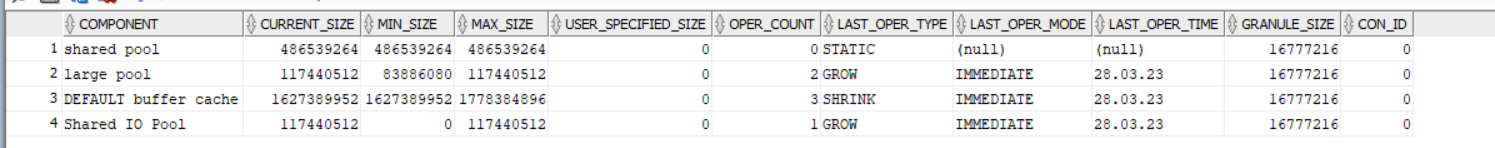
1. Определите общий размер области SGA.



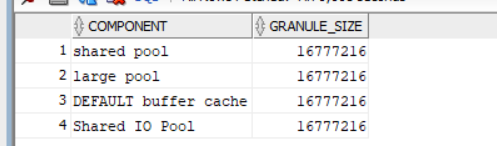


1. Определите текущие размеры основных пулов SGA.

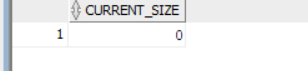
1. Определите размеры гранулы для каждого пула.



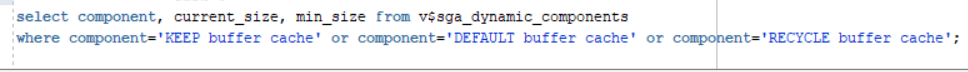


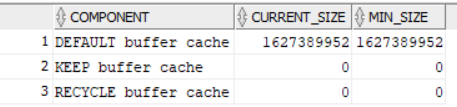
1. Определите объем доступной свободной памяти в SGA.



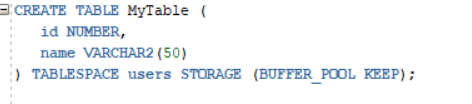


1. Определите максимальный и целевой размер области SGA.
2. Определите размеры пулов КЕЕP, DEFAULT и RECYCLE буферного кэша.

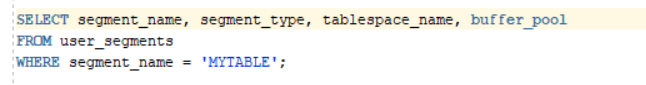
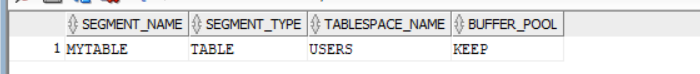




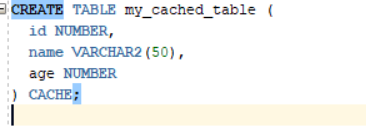
1. Создайте таблицу, которая будет помещаться в пул КЕЕP. Продемонстрируйте сегмент таблицы.

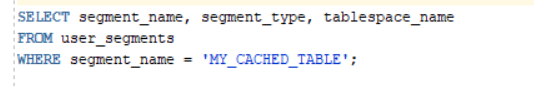




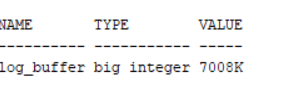
1. Создайте таблицу, которая будет кэшироваться в пуле default. Продемонстрируйте сегмент таблицы.



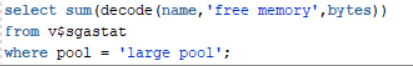


1. Найдите размер буфера журналов повтора.



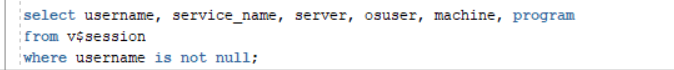


1. Найдите размер свободной памяти в большом пуле.





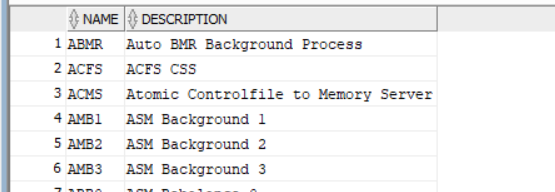
1. Определите режимы текущих соединений с инстансом (dedicated, shared).





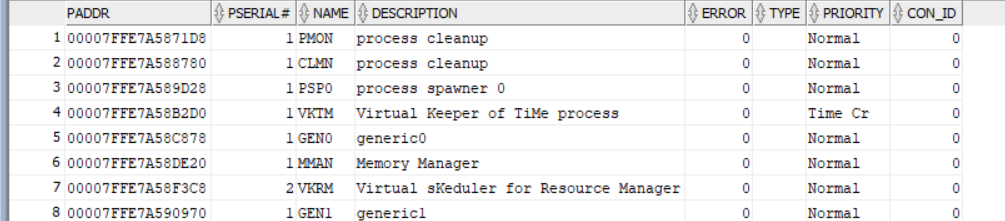
1. Получите полный список работающих в настоящее время фоновых процессов.





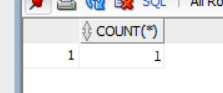
1. Получите список работающих в настоящее время серверных процессов.





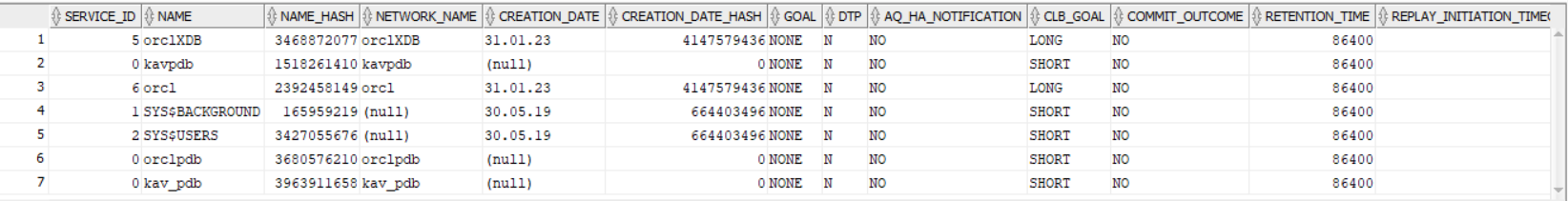
1. Определите, сколько процессов DBWn работает в настоящий момент.





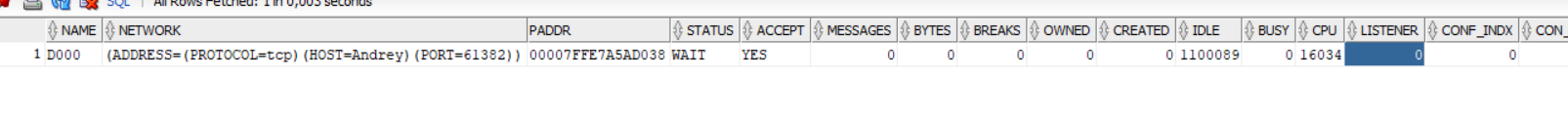
1. Определите сервисы (точки подключения экземпляра).





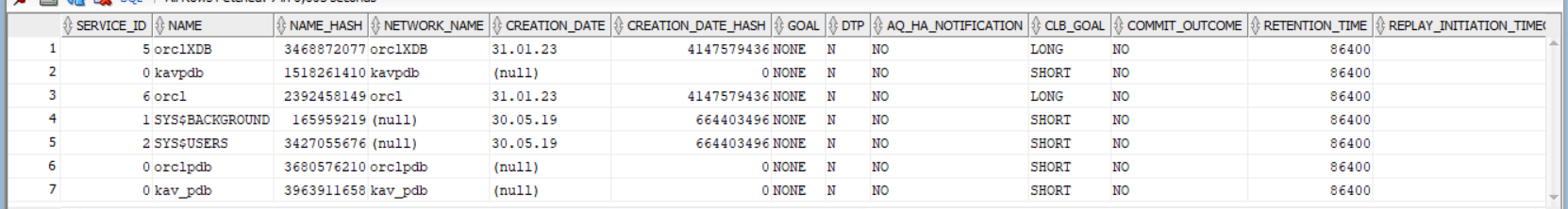
1. Получите известные вам параметры диспетчеров.





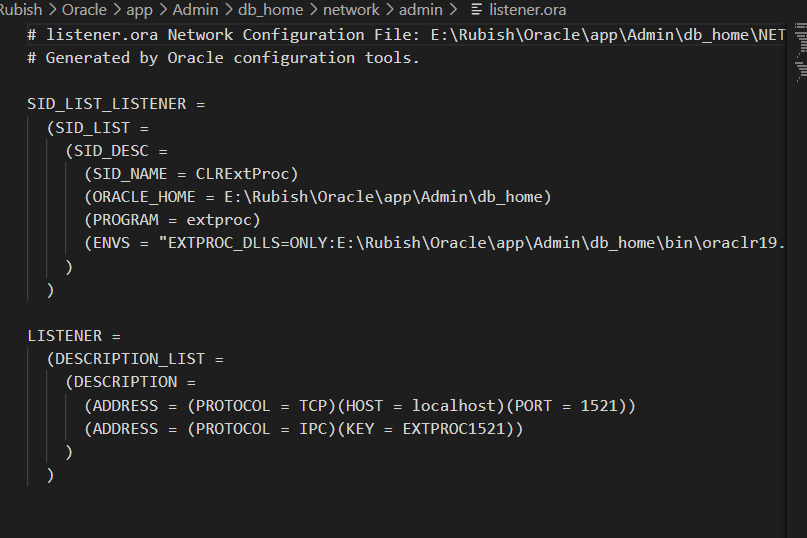
1. Укажите в списке Windows-сервисов сервис, реализующий процесс LISTENER.



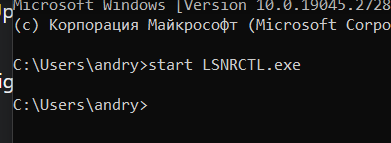


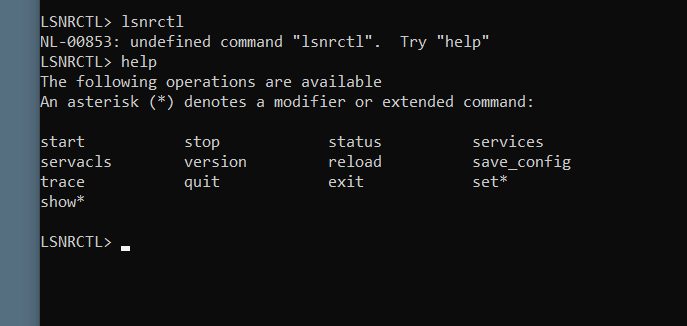
1. Продемонстрируйте и поясните содержимое файла LISTENER.ORA.



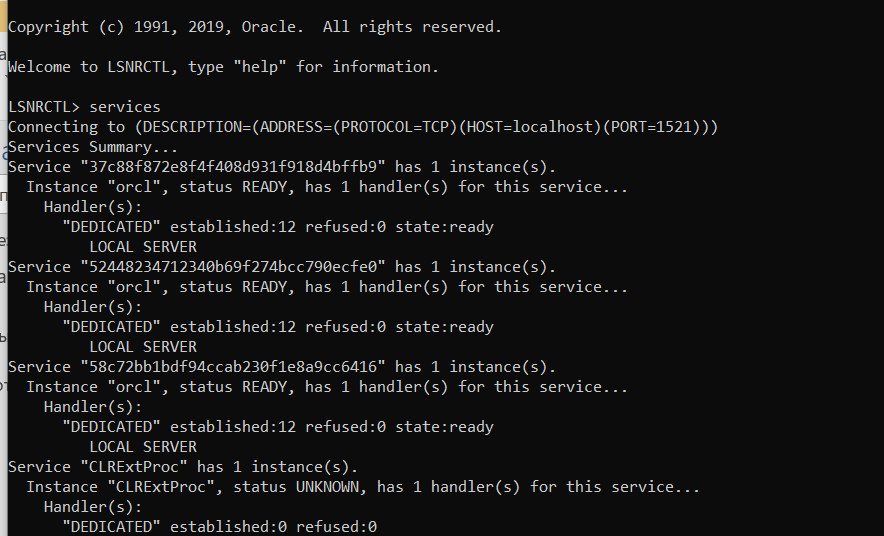


1. Запустите утилиту lsnrctl и поясните ее основные команды.





1. Получите список служб инстанса, обслуживаемых процессом LISTENER.



1. Расшифруйте аббревиатуру SGA.
2. Перечислите основные пулы памяти SGA, поясните их назначение.
3. Поясните параметры SGA\_MAX\_SIZE и SGA\_TARGET.
4. Поясните назначение буферного кэша инстанса.
5. Поясните назначение пулов КЕЕP, DEFAULT и RECYCLE буферного кэша.
6. Поясните принцип вытеснения блоков буферного кэша (LRU).
7. Поясните принцип вытеснения блоков таблицы, созданной оператором CREATE TABLE … CACHE.
8. Как изменить размеры пулов?
9. Какие пулы допускают изменение размеров?

ALTER SYSTEM SET DB\_RECYCLE\_CACHE\_SIZE = 2G;

ALTER SYSTEM SET DB\_CACHE\_SIZE = 10G;

ALTER SYSTEM SET DB\_KEEP\_CACHE\_SIZE = 4G;

1. Поясните назначение процесса LISTENER.
2. Поясните назначение утилиты lsnrctl.
3. Что такое сервис?
4. Какие сервисы создаются автоматически при инсталляции инстанса?
5. Поясните принцип работы dedicated-соединения и shared-соединения.
6. Поясните назначение файла LISTENER.ORA.
7. Перечислите основные фоновые процессы, перечислите их назначение.
8. Что такое серверный процесс? Как просмотреть серверные процессы?
9. SGA - это аббревиатура от "System Global Area". SGA - это область памяти в базе данных Oracle, которая содержит общие данные и структуры данных, используемые всеми процессами, работающими в базе данных.
10. Основные пулы памяти в SGA Oracle:

* Buffer Cache: это пул памяти, который содержит кэшированные блоки данных из таблиц и индексов, чтобы уменьшить количество операций ввода-вывода. Когда данные запрашиваются, Oracle сначала ищет их в буферном кэше, и только если они там отсутствуют, данные загружаются с диска в память.
* Shared Pool: это пул памяти, который содержит общие структуры данных и SQL-запросы, которые выполняются в базе данных. В него также входят данные сессий и информация о блокировках.
* Large Pool: это пул памяти, который используется для выполнения сложных операций, таких как копирование данных между экземплярами базы данных, обработка параллельных запросов и управление памятью для соединений, использующих протоколы, отличные от TCP/IP.
* Java Pool: это пул памяти, который используется для выполнения программ на языке Java в базе данных Oracle. Он содержит общие структуры данных для работы с Java-объектами и кэш для загруженных классов Java.

Каждый из этих пулов памяти в SGA имеет свое назначение и играет важную роль в работе базы данных Oracle.

3. Поясните параметры SGA\_MAX\_SIZE и SGA\_TARGET

SGA\_MAX\_SIZE и SGA\_TARGET - это два параметра, используемые для управления размером System Global Area (SGA) в базе данных Oracle.

SGA\_MAX\_SIZE - это максимальный размер SGA, который может быть выделен базе данных. Он определяет верхний предел для размера SGA и может быть установлен только при остановке базы данных. Если значение параметра SGA\_MAX\_SIZE превышает доступный объем памяти на сервере, то база данных не будет запущена.

SGA\_TARGET - это значение, которое указывает базе данных на желаемый размер SGA в байтах. SGA\_TARGET может быть изменен при работающей базе данных и параметр позволяет Oracle автоматически управлять памятью, выделяемой под SGA. Если SGA\_TARGET установлен, Oracle будет пытаться динамически увеличивать или уменьшать размер SGA, чтобы удовлетворять текущим потребностям в памяти.

Если параметры SGA\_MAX\_SIZE и SGA\_TARGET не установлены, то Oracle будет использовать значения по умолчанию. При этом SGA\_TARGET будет равен SGA\_MAX\_SIZE и размер SGA будет определяться автоматически на основе конфигурации сервера и потребностей базы данных.

В целом, управление размером SGA является важным аспектом производительности базы данных Oracle, поэтому необходимо тщательно настраивать параметры SGA\_MAX\_SIZE и SGA\_TARGET, учитывая объем доступной памяти на сервере и потребности в памяти приложений.

4. Поясните назначение буферного кэша инстанса.

Буферный кэш инстанса (Instance Buffer Cache) в базе данных Oracle - это область оперативной памяти, используемая для хранения копий данных из таблиц и индексов, которые были прочитаны из дискового хранилища.

Назначение буферного кэша инстанса заключается в ускорении доступа к данным и снижении количества операций ввода-вывода с диска. Когда пользователь или приложение запрашивает данные из таблицы или индекса, Oracle ищет их в буферном кэше инстанса, и только в случае, если данные отсутствуют в кэше, происходит операция чтения данных с диска в кэш. Если данные уже есть в кэше, то они могут быть сразу же предоставлены пользователю или приложению, без необходимости выполнения операции ввода-вывода с диска, что уменьшает нагрузку на дисковую систему и сокращает время ответа на запросы.

Буферный кэш инстанса настраивается с помощью параметров базы данных Oracle, таких как DB\_CACHE\_SIZE, который определяет размер кэша в байтах, и DB\_BLOCK\_SIZE, который определяет размер блока данных в кэше. При настройке параметров кэша необходимо учитывать объем доступной памяти на сервере и потребности в памяти приложений, чтобы максимально эффективно использовать буферный кэш инстанса и обеспечить высокую производительность базы данных.

5. Поясните назначение пулов КЕЕP, DEFAULT и RECYCLE буферного кэша.

В базе данных Oracle буферный кэш инстанса может быть разделен на три различных пула памяти: KEEP, DEFAULT и RECYCLE. Каждый из пулов имеет свою уникальную функциональность и назначение:

1. Пул КЕЕР (KEEP) - используется для хранения данных, которые должны оставаться в кэше независимо от того, сколько времени они не используются. Этот пул используется для хранения данных, которые используются редко, но имеют высокую стоимость обновления и занимают большой объем памяти. Например, это могут быть крупные таблицы, индексы, справочники, или служебные таблицы, используемые для поддержки работы базы данных.
2. Пул DEFAULT - используется для хранения данных, которые часто запрашиваются, но не имеют высокой стоимости обновления. В этом пуле хранятся данные, которые могут быть удалены из кэша при необходимости, чтобы освободить место для новых данных.
3. Пул RECYCLE - используется для хранения данных, которые быстро устаревают или могут быть удалены без потери ценности. В этом пуле хранятся данные, которые могут быть перезаписаны, когда в кэше не останется свободного места для новых данных.

Каждый пул имеет свой размер, который можно настраивать отдельно с помощью параметров базы данных Oracle, таких как KEEP, RECYCLE и DEFAULT\_POOL\_SIZE. При правильной настройке пулов буферного кэша можно значительно улучшить производительность базы данных, оптимизировав использование памяти и уменьшив нагрузку на дисковую систему.

6. Поясните принцип вытеснения блоков буферного кэша (LRU).

Принцип вытеснения блоков буферного кэша, используемый в базе данных Oracle, называется LRU (Least Recently Used), или "алгоритм замещения наименее недавно использованных блоков". Он используется для определения, какие блоки данных нужно удалить из кэша, чтобы освободить место для новых блоков.

Суть алгоритма заключается в том, что каждый блок данных в кэше имеет свой временной штамп, который показывает, когда этот блок был последний раз использован. Когда кэш заполнен, и необходимо освободить место для новых блоков, блоки данных, которые не были использованы дольше всего, будут удалены из кэша. Таким образом, блоки данных, которые были недавно использованы, сохранятся в кэше, а те, которые давно не использовались, будут удалены.

Однако, существуют дополнительные параметры, которые могут повлиять на принцип работы LRU-вытеснения в Oracle. Например, блоки из пула KEEP могут быть сохранены в кэше независимо от того, как часто они используются, и блоки из пула RECYCLE могут быть удалены даже если они были недавно использованы, если в кэше не будет достаточно места для хранения новых блоков. Кроме того, при использовании механизма предварительного чтения (prefetching) блоки данных могут быть помечены как "горячие" и сохраняться в кэше, даже если они давно не использовались, чтобы ускорить доступ к этим блокам в будущем.

Таким образом, принцип LRU-вытеснения является основным для управления кэшем в базе данных Oracle, но в зависимости от настроек пулов кэша, параметров и других факторов, он может быть дополнен или изменен для оптимизации производительности базы данных.

7. Поясните принцип вытеснения блоков таблицы, созданной оператором CREATE TABLE … CACHE.

Оператор **CREATE TABLE ... CACHE** в Oracle используется для создания таблицы, которая будет кэшироваться в буферном кэше инстанса, в отличие от обычной таблицы, которая будет кэшироваться только при необходимости.

При использовании оператора **CREATE TABLE ... CACHE** для таблицы применяется специальный принцип вытеснения блоков, называемый LRU-качества (LRU-Warm), который отличается от стандартного принципа вытеснения блоков (LRU).

Суть LRU-качества заключается в том, что при кэшировании блоков таблицы, они помечаются как "теплые" (warm). Когда нужно освободить место в кэше, блоки таблицы с наименьшим "качеством" (то есть те, которые не были использованы дольше всего), удаляются из кэша, но не полностью вытесняются. Вместо этого, они помечаются как "холодные" (cold) и остаются в кэше, но в другом списке, где они не конкурируют за место с другими блоками. Когда блоки таблицы снова запрашиваются, они перемещаются из списка "холодных" в список "теплых" и начинают снова участвовать в конкуренции за место в кэше.

Таким образом, блоки таблицы, созданной оператором **CREATE TABLE ... CACHE**, имеют более высокий приоритет в кэше, чем другие блоки, и остаются доступными в кэше дольше, даже если они не используются. Это может улучшить производительность при частом доступе к таблице, но может также привести к уменьшению доступной памяти в кэше для других блоков данных, поэтому не рекомендуется использовать этот прием для всех таблиц в базе данных.

8. Как изменить размеры пулов?

Изменение размеров пулов буферного кэша можно произвести с помощью параметров Oracle, используя SQL\*Plus, Enterprise Manager или командную строку.

Для изменения размеров пулов КЕЕР, DEFAULT и RECYCLE необходимо выполнить следующие шаги:

1. Определить текущий размер пулов, используя запрос к динамическому представлению V$BUFFER\_POOL\_STATISTICS:

SQLCopy code

SELECT name, bytes, substr(pool,1,6) pool, sets, gethitratio, status FROM v$buffer\_pool\_statistics;

Этот запрос выведет текущий размер пулов, количество блоков в каждом пуле, попадания и промахи кэша, а также текущий статус пула (активен/неактивен).

1. Изменить размер пулов, используя команду ALTER SYSTEM. Например, для увеличения размера пула KEEP можно выполнить следующую команду:

SQLCopy code

ALTER SYSTEM SET DB\_KEEP\_CACHE\_SIZE = 4G;

Эта команда установит размер пула KEEP в 4 гигабайта. Аналогично можно изменить размеры других пулов, используя соответствующие параметры, например:

SQLCopy code

ALTER SYSTEM SET DB\_RECYCLE\_CACHE\_SIZE = 2G; ALTER SYSTEM SET DB\_CACHE\_SIZE = 10G;

1. После изменения размеров пулов необходимо перезапустить инстанс Oracle, чтобы изменения вступили в силу.

SQLCopy code

SHUTDOWN IMMEDIATE; STARTUP;

1. Проверить, что изменения вступили в силу, используя запрос к динамическому представлению V$BUFFER\_POOL\_STATISTICS.

SQLCopy code

SELECT name, bytes, substr(pool,1,6) pool, sets, gethitratio, status FROM v$buffer\_pool\_statistics;

Этот запрос выведет новые значения размеров пулов и соответствующие изменения в статистике кэша.

9. Какие пулы допускают изменение размеров?

В Oracle существует несколько пулов, которые можно изменять по их размеру:

1. DEFAULT Buffer Pool – это основной пул буферного кэша и он может быть изменен с помощью параметра DB\_CACHE\_SIZE.
2. KEEP Buffer Pool – этот пул используется для хранения часто используемых блоков данных и может быть изменен с помощью параметра DB\_KEEP\_CACHE\_SIZE.
3. RECYCLE Buffer Pool – этот пул используется для хранения данных, которые не используются очень часто, и может быть изменен с помощью параметра DB\_RECYCLE\_CACHE\_SIZE.
4. 2ND LEVEL CACHE – этот пул используется для кеширования SQL-операторов и может быть изменен с помощью параметра SHARED\_POOL\_SIZE.

Помимо этих пулов, также существуют другие пулы в SGA, которые имеют фиксированный размер и не могут быть изменены во время выполнения инстанса Oracle.

10. Поясните назначение процесса LISTENER.

Процесс LISTENER в Oracle представляет собой специальный сетевой процесс, который слушает и обрабатывает входящие соединения от клиентских приложений и передает их на нужный экземпляр базы данных для обработки.

При запуске Oracle-сервера, LISTENER стартует автоматически и начинает слушать определенный порт на сервере. Когда клиентское приложение пытается подключиться к Oracle-серверу, оно отправляет запрос на этот порт, где LISTENER его принимает. Если клиентский запрос содержит правильные учетные данные и имена базы данных, LISTENER направляет его на соответствующий экземпляр базы данных. Если экземпляр базы данных существует, он принимает соединение и начинает обработку запроса.

LISTENER может настраиваться для обеспечения дополнительной безопасности и эффективности работы сети. Например, можно определить несколько портов прослушивания, настроить аутентификацию клиентов или добавить дополнительные параметры безопасности.

Обычно, в Oracle-архитектуре LISTENER является важным компонентом для обеспечения соединения между клиентскими приложениями и экземплярами базы данных.

11. Поясните назначение утилиты lsnrctl.

Утилита lsnrctl (Listener Control Utility) в Oracle используется для управления процессом LISTENER. С ее помощью администраторы могут контролировать, настраивать и проверять состояние процесса LISTENER, а также изменять его параметры.

Некоторые из основных задач, которые может выполнять утилита lsnrctl:

1. Запуск и остановка процесса LISTENER.
2. Просмотр текущих настроек и параметров процесса LISTENER.
3. Добавление или удаление баз данных для прослушивания.
4. Настройка портов прослушивания.
5. Проверка журнала активности LISTENER.

Утилита lsnrctl имеет командный интерфейс и может использоваться как локально на сервере, так и удаленно через сеть. Она также предоставляет возможность автоматизировать некоторые задачи управления процессом LISTENER с помощью скриптов или файлов настроек.

12. Что такое сервис? э

В контексте Oracle, сервис представляет собой логическое имя или идентификатор, который связывает клиентские приложения с конкретной экземпляром базы данных. Сервисы позволяют администраторам управлять доступом к базе данных, контролировать производительность и упрощать администрирование.

Когда клиентское приложение пытается подключиться к Oracle-базе данных, оно должно указать имя сервиса, которое соответствует конкретному экземпляру базы данных или группе экземпляров. Каждый экземпляр базы данных может иметь несколько сервисов, каждый из которых может обслуживать различные приложения и запросы.

Примеры типов сервисов в Oracle:

1. Сервис базы данных (Database Service) - это сервис, который обслуживает все подключения к базе данных, обычно используется для обработки транзакций.
2. Сервис приложения (Application Service) - это сервис, который связывает конкретное приложение с экземпляром базы данных и может использоваться для оптимизации производительности приложения.
3. Сервис резервирования (Reservation Service) - это сервис, который используется для резервирования ресурсов базы данных для конкретных задач.

Администраторы могут управлять сервисами в Oracle с помощью утилиты srvctl или SQL-команд. Они могут создавать, изменять или удалять сервисы, а также назначать им определенные параметры и настройки.

1. При установке Oracle-инстанса автоматически создаются следующие сервисы:

* сервис базы данных: это основной сервис, который обрабатывает все запросы к базе данных;
* сервис управления: это сервис, который обеспечивает возможность удаленного управления базой данных, например, через Oracle Enterprise Manager или SQL Developer;
* сервис для администрирования базы данных: это сервис, который предоставляет доступ к инструментам администрирования, таким как SQL\*Plus или Oracle Enterprise Manager.

1. Dedicated-соединение означает, что каждое клиентское подключение имеет выделенный серверный процесс на стороне сервера. Это означает, что для каждого клиента создается отдельный процесс, который полностью управляет его соединением с базой данных. Dedicated-соединения обеспечивают наилучшую производительность для приложений с высокой нагрузкой, но требуют больше системных ресурсов, так как каждое подключение использует свой собственный процесс.

Shared-соединение означает, что несколько клиентов могут использовать один и тот же серверный процесс. Это означает, что один процесс может обслуживать несколько соединений одновременно. Shared-соединения используются в ситуациях, когда число клиентов невелико, и каждый клиент обращается к базе данных только время от времени. Shared-соединения более экономичны с точки зрения использования ресурсов, но могут ухудшить производительность в случае большой нагрузки.

1. Файл listener.ora - это файл конфигурации, который используется для настройки Listener (слушателя) в Oracle. Он определяет параметры, такие как адрес слушателя, порт, на котором слушатель ожидает входящие соединения, и сервисы, которые могут быть обслужены этим слушателем. Файл listener.ora располагается на сервере базы данных в каталоге $ORACLE\_HOME/network/admin/.

16. Перечислите основные фоновые процессы, перечислите их назначение. 17. Что такое серверный процесс? Как просмотреть серверные процессы?

1. Основные фоновые процессы Oracle:

* DBWn (Database Writer) - записывает изменения из буферного кэша на диск;
* LGWR (Log Writer) - записывает данные из лог-буфера на диск;
* PMON (Process Monitor) - мониторит процессы клиентов и фоновые процессы, восстанавливает сеансы клиентов после сбоев;
* SMON (System Monitor) - мониторит состояние инстанса, выполняет сборку мусора, освобождает неиспользуемые ресурсы;
* CKPT (Checkpoint) - управляет процессом контрольной точки, записывает информацию о состоянии БД в файл контрольной точки;
* MMAN (Memory Manager) - управляет SGA, выделяет и освобождает память;
* RECO (Recovery Process) - отвечает за автоматическое восстановление транзакций после сбоев.

1. Серверный процесс (server process) - это процесс, который обрабатывает запросы от клиентских приложений. Он устанавливает соединение с клиентом и выполняет SQL-запросы от его имени. Каждый серверный процесс имеет свой идентификатор (PID) и управляется фоновым процессом PMON.

Для просмотра серверных процессов можно использовать утилиту SQL\*Plus и выполнить запрос к представлению V$PROCESS:

sqlCopy code

SELECT spid, pid, program FROM v$process WHERE background = 0;

Этот запрос вернет список всех активных серверных процессов с их идентификаторами и программами, которые они выполняют. Также можно использовать утилиту Enterprise Manager или другие утилиты управления БД для просмотра и мониторинга серверных процессов.

Начало формы