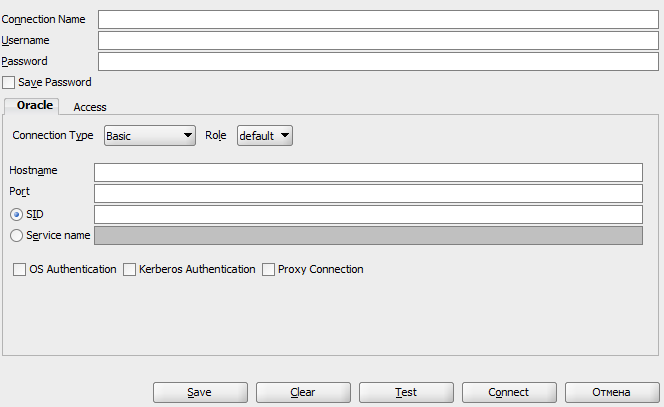
**1 лаба**

1. **Перечислите и поясните параметры подключения к серверу Oracle.**



* Имя подключения
* Имя пользователя
* Пароль
* Тип подключения
* Роль
* Имя хоста - Это имя или IP-адрес сервера Oracle Database
* Порт - номер порта, через который будет установлено соединение с сервером. По умолчанию порт 1521.
* СИД - идентифицирует конкретную экземпляр базы данных на сервере. Это строка, которая обычно предоставляется администратором базы данных.
* Название сервиса – более гибкий идентификатор бд

Соединение - объект который определяет необходимую информацию для подключения к определенное бд в качестве определенного пользователя.

1. **Опишите особенности выполнения DML операторов в Oracle.**

DML операторы:

* INSERT
* UPDATE
* DELETE
* SELECT

DML операции в Oracle выполняются в рамках транзакций.

Транзакция - Одна или несколько команд SQL, которые либо успешно выполняются как единое целое, либо отменяются как единое целое. Это означает, что изменения, внесенные в базу данных с использованием DML операторов, будут либо полностью выполнены, либо откатаны в случае возникновения ошибок. Для сохранения изменений необходимо использовать оператор COMMIT - фиксирует изменения и завершает текущую транзакцию. Для отмены изменения, сделанные в рамках текущей транзакции - оператор ROLLBACK. Оператор SAVEPOINT позволяет установить точку сохранения в транзакции. Это позволяет откатить изменения до определенной точки без отката всей транзакции.

Свойства транзакций:

* Атоммарность - операторы, либо выполнятся все, либо не выполнится ни один
* Согласованность - транзакция должна фиксировать новое согласованное состояние
* Изолированность - отсутствие взаимного влияния параллельных транзакций на результаты их выполнения
* Долговечность - изменения в БД, выполненные и зафиксированные транзакцией, могут быть отменены только с помощью новой транзакции

INSERT ALL – позволяет вставить несколько строк в одну или более таблиц, используя один оператор.

1. **Перечислите знакомые вам типы объектов, которые вы увидели в окне подключения.**

* **Таблицы**
* **Представления**
* **Индексы**
* **Хранимые процедуры**
* **Функции**
* **Триггеры**
* **Пользователи**
* **Дефолты**
* **XML Schemas**

1. **Перечислите известные вам типы данных Oracle.**

* CHAR(n): Фиксированной длины строки.
* VARCHAR2(n): Строка переменной длины.
* NUMBER(p, s): Число с фиксированной точностью и масштабом. p - общая точность, s - количество знаков после запятой.
* DATE: Тип данных для хранения даты и времени в формате ГГГГ-ММ-ДД ЧЧ:ММ:СС.
* TIMESTAMP: Расширенный тип данных для хранения даты и времени с большей точностью по сравнению с DATE.
* CLOB (Character Large Object): Для хранения больших объемов символьных данных, таких как текстовые документы.
* BLOB (Binary Large Object): Для хранения больших объемов бинарных данных, таких как изображения или видео.
* BINARY\_FLOAT и BINARY\_DOUBLE: Для хранения чисел с плавающей запятой одинарной и двойной точности соответственно.
* RAW(n): Для хранения данных в виде байтов. n - фиксированная длина в байтах.
* BOOLEAN: Логический тип данных, который может принимать значения TRUE, FALSE или NULL.
* ROWID: Уникальный идентификатор строки в таблице.
* XMLType: Для хранения и обработки данных в формате XML.

1. **Перечислите известные вам ограничения целостности.**

* Ограничение уникальности (UNIQUE): Гарантирует уникальность значений в столбце или группе столбцов.
* Ограничение первичного ключа (PRIMARY KEY): Гарантирует уникальность значений и предотвращает наличие NULL в указанных столбцах.
* Ограничение внешнего ключа (FOREIGN KEY): Устанавливает связь между столбцами в двух таблицах, гарантируя согласованность данных.
* Ограничение проверки (CHECK): Позволяет определить условие, которое должно быть истинным для всех записей в таблице.
* Ограничение NOT NULL: Обеспечивает, что значения в указанных столбцах не могут быть NULL.
* Ограничение уникальности по условию (UNIQUE WHERE): Позволяет создать уникальное ограничение, которое применяется только к определенным значениям.

СУБД ORACLE - объектно-реляционная система управления базами данных

JOIN - позволяет извлекать данные более чем из одной таблицы

INNER JOIN - возвращает только те строки, которые имеют соответствие в обеих таблицах, основываясь на условии соединения. каждая строка одной таблицы соединяется с каждой строкой другой ->выбираются строки, удовлетворяющие указанному условию -> выбираются столбцы, указанные в списке SELECT

LEFT JOIN возвращает все строки из левой таблицы и соответствующие строки из правой таблицы. Если нет соответствия, то значения столбцов из правой таблицы будут NULL.

Кросс Возвращает декартово произведение строк из обеих таблиц

FULL JOIN Возвращает все строки из обеих таблиц. Если нет соответствующих строк в одной из таблиц, возвращаются NULL значения.

Outer - строки, которые не имеют соответствия в одной из таблиц.

1. **лаба**

**1.Перечислите редакции поставки Oracle 12с.**

Express Edition: бесплатная редакция

Standard Edition One: начальная редакция для дома и малого бизнеса

Standard Edition: для малого и среднего бизнеса

Enterprise Edition: решения промышленного уровня

**2.Перечислите предопределенных пользователей БД Oracle.**

SYS – предопределенный привилегированный пользователь ранга администратора базы данных, который является владельцем ключевых ресурсов БД Oracle

SYSTEM – предопределенный привилегированный пользователь, которому принадлежат ключевые ресурсы БД Oracle

DBA – предопределенная роль, которая автоматически создаётся для каждой базы данных Oracle и содержит все системные привилегии, кроме SYSDBA и SYSOPER

SYSDBA и SYSOPER - специальные привилегии администратора, которые позволяют выполнять базовые задачи администрирования: запуск или остановка экземпляра БД; создание, удаление, открытие или монтирования базы и др.

Роль DBA не включает SYSDBA и SYSOPER. Привилегии могут быть указаны при подключении (connect) пользователя к БД

**3.Что такое табличное пространство?**

Табличное пространство – логическая структура хранения данных, контейнер сегментов. Оно объединяет один или несколько файлов данных для организации данных внутри базы данных.

Файлы данных - это физические файлы на диске, в которых хранятся реальные данные таблиц и индексов.

Сегмент – область на диске, выделяемая под объекты.

Сегменты типизируются в зависимости от типа данных, хранящихся в них – сегменты таблиц, сегменты индексов, сегменты кластеров и т.д.(всего 10 типов).

**4.В каком соотношении находятся табличные пространства и файлы данных?**

Один файл данных может принадлежать только одному табличному пространству.

Одно табличное пространство может включать в себя один или несколько файлов данных.

**5.Перечислите известные вам параметры табличного пространства.**

* DATAFILE (или TEMPFILE): Указывает файл данных, связанный с табличным пространством. Этот параметр указывает физическое расположение на диске, где будут храниться данные.
* SIZE: Определяет начальный размер файла данных табличного пространства при его создании.
* AUTOEXTEND: Указывает, разрешено ли автоматическое расширение файла данных при достижении его максимального размера.
* NEXT: Задает размер увеличения файла данных, когда он расширяется.
* MAXSIZE: Определяет максимальный размер файла данных, до которого он может расширяться при использовании автоматического расширения.
* ONLINE (или OFFLINE): Управляет состоянием доступности табличного пространства. Если табличное пространство находится в режиме OFFLINE, данные недоступны для обработки.
* DEFAULT STORAGE: Позволяет определить параметры хранения по умолчанию для объектов, создаваемых в табличном пространстве.
* PERMANENT - хранение постоянных объектов БД
* TEMPORARY - хранение временных данных
* UNDO - хранение сегментов отката, используется всегда один

**6.Перечислите типы табличных пространств и их назначение.**

* DATA TABLESPACE (постоянное табличное пространство): Используется для хранения постоянных данных, таких как таблицы и индексы.
* TEMPORARY TABLESPACE (временное табличное пространство): Используется для временных данных, таких как временные таблицы, сортировки и другие временные операции.
* UNDO TABLESPACE (табличное пространство отката): Используется для хранения данных отмены (отката), которые необходимы для обеспечения целостности транзакций и восстановления после сбоев.
* INDEX TABLESPACE (табличное пространство для индексов): Используется для хранения индексов.
* SYSTEM TABLESPACE (системное табличное пространство): Содержит системные объекты и метаданные базы данных, такие как системные таблицы и системные индексы. Он также может содержать некоторые пользовательские объекты.
* READ ONLY TABLESPACE (только для чтения): Используется для создания табличных пространств, которые можно установить в режим только для чтения. Это может быть полезно для архивирования данных.
* BIGFILE TABLESPACE: Позволяет создавать большие файлы данных (больше 4 ГБ) для табличных пространств, что полезно для хранения крупных объемов данных.
* PERMANENT – для постоянных данных: таблицы, индексы, представления, функции, процедуры и т.д

**7.Что такое роль и для чего она применяется?**

Роль – это именованный набор привилегий. Роли используются для назначения определенных привилегий (прав доступа) к объектам базы данных. Вместо назначения прав доступа каждому пользователю отдельно, администратор может создать роли и присвоить их пользователям.

Привилегия - это право выполнять конкретный тип предложений SQL, или право доступа к объекту другого пользователя.

ORACLE имеет два вида привилегий: системные и объектные. Назначаются оператором GRANT. Отзываются оператором REVOKE

**8.Какие системные роли вы знаете? Объясните их назначение**

* DBA (Database Administrator): предоставляет полный доступ к базе данных и включает в себя все системные привилегии.
* CONNECT: предоставляет минимальные привилегии, необходимые для установления соединения с базой данных.
* RESOURCE: предоставляет привилегии, которые полезны для разработки и тестирования. Она включает в себя права на создание таблиц, процедур, триггеров и других объектов.
* SELECT\_CATALOG\_ROLE: предоставляет привилегии на выполнение запросов к системным представлениям (views), позволяя пользователю получить доступ к метаданным базы данных.
* EXECUTE\_CATALOG\_ROLE: предоставляет право выполнения некоторых системных процедур и функций, связанных с метаданными базы данных.
* DBFS\_ROLE: предоставляет доступ к функциональности Oracle Database File System (DBFS). Это позволяет работать с файловой системой, хранящейся в базе данных.
* AUDIT\_ADMIN, AUDIT\_VIEWER: Эти роли предназначены для управления аудитом в базе данных. Аудит – это наблюдение за выбранными действиями пользователей базы данных.
* RESOURCE: предоставляет привилегии для создания объектов, таких как таблицы, процедуры, пакеты и другие, что может быть полезно для пользователей, занимающихся разработкой.

**9.Что такое системная привилегия, какие системные привилегии вы знаете?**

Системные привилегии представляют собой права доступа, предоставляемые пользователям или ролям для выполнения конкретных действий в контексте базы данных. Эти привилегии управляют доступом к различным системным ресурсам и операциям в базе данных.

Существуют системные и объектные привилегии.

Системная – право на выполнение определенного действия (напр. DMLоператора), а объектная – право на выполнение действия над определенным объектом схемы (напр. над таблицей или представлением).

Системные (записываются с create any/drop any/alter any/… в начале):

* table
* procedure
* profile
* role
* tablespace
* user
* index
* session
* sql transaction
* pluggable database
* mining model
* sysdba

Объектные привилегии:

* select
* alter
* insert
* update
* delete
* read
* write
* execute

**10.Какие специальные системные привилегии администратора БД вы знаете? Как их назначить?**

Назначение с помощью grant/revoke от имени юзера с привилегией admin option или grant any privilege.

Системные привилегии – SYSDBA, SYSOPER

Возможности SYSDBA:

* STARTUP и SHUTDOWN операции
* ALTER/CREATE/DROP DATABASE
* ALTER DATABASE ARCHIVELOG
* ALTER DATABASE RECOVER
* Создать SPFILE
* Включает привилегию RESTRICTED SESSION
* Подключиться как пользователь SYS

Возможности SYSOPER:

* STARTUP и SHUTDOWN операции
* ALTER DATABASE
* ALTER DATABASE ARCHIVELOG
* Включает привилегию RESTRICTED SESSION
* Создать SPFILE
* ALTER DATABASE RECOVER (только полное восстановление; другие опции восстановления требуют SYSDBA)
* Можно выполнять базовые операции, но без просмотра данных пользователя

**11.Что такое профиль безопасности и для чего он нужен?**

Профиль устанавливает ограничение на использование ресурсов конкретным пользователем, а также управляет паролями. Может ограничивать кол-во открытых сеансов, продолжительность соединения, время бездействия, использование ЦП, кол-во попыток ввода пароля, время его жизни и т.д.

**12.Перечислите известные вам параметры профиля безопасности и поясните их.**

* CONNECT\_TIME. Указывает общее время (в минутах), в течение которого сеанс может оставаться подключенным к базе данных.
* SESSIONS\_PER\_USER. Указывает максимальное количество сеансов, которые могут быть параллельно открыты пользователем.
* IDLE\_TIME. Ограничивает время, в течение которого сеанс может оставаться бездействующим.
* FAILED\_LOGIN\_ATTEMPTS. Указывает количество последовательных попыток регистрации, которые пользователь может предпринять, прежде чем будет заблокирован.
* PASSWORD\_LIFE\_TIME. Устанавливает временной предел использования конкретного пароля. Если пароль не будет изменен в течение указанного времени, срок его действия истечет.
* PASSWORD\_LOCK\_TIME. Указывает количество дней, в течение которых пользователь будет заблокирован после достижения максимального числа неудачных попыток регистрации.
* PASSWORD\_REUSE\_TIME. Указывает количество дней, которые должны пройти, прежде чем можно будет снова использовать тот же пароль.

**13.Объясните назначение профиля с именем DEFAULT.**

Это профиль безопасности, создающийся по умолчанию в Oracle. Имеет внутри себя ограничения, однако они очень слабые, ведь почти на все пределы ресурсов устанавливается ограничение UNLIMITED. Если пользователю не будет явно назначен профиль безопасности, то будет назначен профиль default с почти безграничными возможностями на использование ресурсов.

**14.Что такое пользователь базы данных?**

Пользователь – это человек, который подключается к пользовательскому аккаунту, устанавливая сессию к экземпляру базы данных.

**15.Перечислите известные вам параметры пользователя БД Oracle.**

* имя пользователя
* наличие пароля и сам пароль, при необходимости
* табличное пространство по умолчанию
* временное табличное пространство по умолчанию
* лимит (квота) на табличное пространство
* профиль безопасности
* статус (open, locked, expired)

**16.Что такое квота?**

Квота – размер дискового пространства, которое может использовать пользователь в выданном ему табличном пространстве.

**17.Перечислите все известные вам представления словаря БД Oracle.**

Словарь – совокупность таблиц и представлений с основной администраторской информацией о базе данных, созданных и обслуживаемых самим Oracle. Представления словаря, с помощью которых и просматривается информация, содержат пользователей, объекты базы данных, уровни привилегий пользователей и прочую информацию.

Основные представления словаря:

• DBA\_USERS – все пользовательские аккаунты с хешем паролей

• DBA\_ROLES – все роли базы данных

• DBA\_SYS\_PRIVS – системные привилегии

• DBA\_TAB\_PRIVS – объектные привилегии

• USER\_ROLE\_PRIVS – роли текущего пользователя

• USER\_SYS\_PRIVS – системные привилегии текущего пользователя

\* Для изменения пароля какого-либо уже существующего пользователя

нужно:

1) При необходимости выйти с командной строки Докера (Ctrl+D) – это

нужно, чтобы завершить сессию.

2) Удалить пользователя каскадно в скрипте.

3) Выполнить строку: alter session set "\_ORACLE\_SCRIPT"= true;

4) Создать заново этого пользователя.

5) Дать привилегии добавленному пользователю.

6) Открыть опять командную строку Докера, ввести sqlplus, ввести

существующий пароль, который мы указали в создании пользователя.

**3 лаба**

1. **Что такое словарь данных Oracle? Для чего он необходим?**

**Словарь Oracle** - набор таблиц и связанных с ними представлений, который предоставляет возможность отследить внутреннюю структуру базы данных и деятельность СУБД Oracle

1. **Проклассифицируйте представления словаря данных Oracle.**

USER Объекты, принадлежащие пользователю

ALL Объекты, к которым пользователь имеет доступ

DBA Все объекты базы данных (для администратора БД)

V$ Производительность сервера

1. **Что значит Oracle 12c имеет клиент-серверную архитектуру?**

В этой архитектуре управление данными и обработка запросов происходят на сервере базы данных, в то время как клиентские приложения служат для взаимодействия с пользователем и отправки запросов на сервер.

1. **Что такое экземпляр (инстанс) сервера СУБД?**

экземпляр — это программное обеспечение, которое работает на сервере и предоставляет доступ к информации, содержащейся в базе данных Oracle. Экземпляр исполняется на конкретном сервере либо компьютере, в то самое время как база данных хранится на дисках, подключённых к этому серверу.

Экземпляр может являться частью только одной базы данных. При этом с одной базой данных бывает ассоциировано несколько экземпляров.

Включает:

запущенный сервер (программа) СУБД Oracle

общая (глобальная) область памяти (SGA – system global area) и др. системные области памяти

фоновые процессы, предназначенные для управления файлами базы данных

1. **Перечислите все этапы старта и останова инстанса Oracle 12c. Поясните их назначение.**

**Запуск**

**STARTUP NOMOUNT**:

**Запуск экземпляра** подразумевает выполнение следующих задач:

1) Поиск файла параметров (хранит параметры экземпляра)

2) Выделение SGA; часть оперативной памяти, разделяемой всеми процессами одного экземпляра базы данных. Pga Программная Глобальная Область - это такая область памяти, в которой хранятся данные и управляющая информация о серверных процессах Oracle.

3) Запуск фоновых процессов.

**STARTUP MOUNT:**

Ассоциация базы данных с предварительно запущенным экземпляром:

Определение местоположения управляющих файлов (файлы, содержащие имена (местоположение) основных физических файлов), которые указаны в файле параметров:

**STARTUP OPEN:** Открытие базы данных подразумевает выполнение следующих задач:

• Открытие оперативных файлов данных;

• Открытие оперативных журнальных файлов.

**Остановка**

**SHUTDOWN NORMAL:** Запрещено создавать новые сессии. Ожидается завершение работы всех пользователей. Никаких восстановительных работ при следующем старте не проводится;

**SHUTDOWN TRANSACTIONAL**: Запрещено создавать новые сессии. Запрещено запускать новые транзакции. Сервер дожидается завершения уже начатых транзакций и отключает пользователей, не имеющих активных транзакций. Никаких восстановительных работ при следующем старте не проводится.

**SHUTDOWN IMMEDIATE:** Запрещено создавать новые сессии. Запрещено запускать новые транзакции. Все незафиксированные транзакции откатываются. Никаких восстановительных работ при следующем старте не проводится.

**SHUTDOWN ABORT:** Все действия прекращаются. Все транзакции не фиксируются и не откатываются. Пользователей отсоединяют от БД. При следующем старте будет выполнено возможное восстановление.

1. **Где указывается, что при старте операционной системы будет запущен инстанс Oracle 12c?**

В службе OracleServerИмяЭкземпляра должен стоять автоматический запуск.

SELECT instance\_name FROM v$instance

1. **Поясните назначение опции Oracle 12c Multitenant.**

**Oracle Multitenant** - технология, позволяющая запустить несколько независимых баз данных в рамках одного экземпляра.

Каждая база данных имеет свой набор табличных пространств и набор схем, но при этом у них общая SGA и один набор серверных процессов.

Базы данных изолированы, друг о друге ничего не знают, не конфликтуют между собой.

1. **Поясните назначение баз данных CDB, PDB, PDB$SEED.**

В случае мультиарендной (multitenant) архитектуры создается одна контейнерная база данных (Container Database, **CDB**), которую обслуживает один экземпляр Oracle.

А все вновь создаваемые БД (они называются подключаемыми – Pluggable Database, **PDB**) помещаются в эту контейнерную БД. При этом для обслуживания множества независимых БД используются один набор процессов и одна область оперативной памяти.

Cтарый словарь БД разделяется на 2 части. Общая для всех PDB часть словаря хранится в CDB, а в каждой PDB хранится информация словаря, специфичная для данной PDB. Мультиарендная БД имеет один набор журнальных файлов (redo logs) и один набор управляющих файлов, общих для всех PDB в контейнерной БД.

**PDB$SEED** - шаблон для создания новых, пустых PDB. Readonly. Неотключаемый.

1. **Что такое общий пользователь Oracle 12c Multitenant?**

▪ создаётся CDB$ROOT и виден в каждой PDB

▪ может подключаться к тем PDB, где у него есть привилегия “Create Session”

▪ Может владеть объектами в root и в PDB

▪ может администрировать всю CDB-базу данных, если у него есть привилегия DBA

▪ обычно имеет имя, начинающееся на C##

**Какие табличные пространства автоматически создаются при инсталляции СУБД Oracle 12c?**

* SYSTEM используется для управления БД, содержит словарь базы данных, стандартные пакеты процедур.
* SYSAUX – вспомогательное табличное пространство.
* TEMP – временное табличное пространство по умолчанию.
* UNDOTBS1 – табличное пространство отката.
* USERS – хранение пользовательских объектов и данных.
* EXAMPLE – демонстрационные схемы.

1. **Какие 2 пользователя обязательно создаются при инсталляции СУБД Oracle 12c?**

SYS – предопределенный привилегированный пользователь ранга администратора базы данных, который является владельцем ключевых ресурсов БД Oracle

SYSTEM – предопределенный привилегированный пользователь, которому принадлежат ключевые ресурсы БД Oracle

1. **Какие табличные пространства дублируются в каждой PDB?**

Которые создаются по умолчанию

1. **Что такое консолидация баз данных? В каких случаях прибегают к консолидации баз данных?**

Консолидация баз данных - это процесс объединения нескольких баз данных в одну для упрощения управления данными. В Oracle, это может включать в себя слияние нескольких баз данных в одну, уменьшение количества физических серверов, на которых размещаются базы данных, или перенос данных с устаревших платформ на более современные.

Множество подключаемых баз данных в одной контейнерной базе данных совместно используют ее память и фоновые процессы. Благодаря этому на одной серверной платформе можно использовать намного больше подключаемых баз данных, чем при развертывании выделенных одноэкземплярных баз данных. Эти преимущества похожи на те, что предоставляет консолидация схем.

1. **Какие преимущества дает Oracle 12c Multitenant?**

▪ Быстрое, легко автоматизируемое развертывание новых баз, включая клонирование.

▪ Эффективная консолидация множества баз на одном сервере - экономия оперативной памяти.

▪ Полноценная изоляция PDB внутри CDB.

▪ Управление многими базами данных как одной. За счет консолидации имеющихся подключаемых баз данных администраторы могут управлять многими базами как одной. Например, такие задачи, как резервное копирование и аварийное восстановление, можно выполнять на уровне контейнерной базы данных.

**4 лаба**

1. **Из каких файлов состоит база данных Oracle?**

**Файлы параметров**: Содержат конфигурационные параметры для базы данных Oracle, определяющие ее поведение и настройки. Они влияют на аспекты, такие как размеры памяти, параметры сети, и другие настройки, используемые при старте базы данных.

**Управляющие файлы:** Содержат метаданные о структуре базы данных и служат для отслеживания изменений в базе данных. Необходимы для успешного запуска и восстановления базы данных.

**Файлы паролей**: Содержат хеши паролей пользователей базы данных. Используются для аутентификации пользователей при подключении к базе данных.

**Файлы сообщений**: Записывают сообщения, предупреждения и ошибки, возникающие в базе данных.

**Журналы повтора**: Записывают последовательность изменений данных в базе данных, обеспечивая целостность данных и возможность восстановления после сбоев.

**Архивы**: Содержат скопированные журналы повтора (редо-логи), которые могут использоваться для восстановления базы данных до определенного момента в прошлом, поддерживая точку восстановления.

**Файлы табличных пространств:** Содержат данные таблиц, индексов и других объектов базы данных. Логические структуры для управления распределением данных.

**Файлы аудита:** Эти файлы фиксируют попытки доступа, изменения в данных и другие важные события для обеспечения безопасности базы данных.

1. **Какие табличные пространства создаются при инсталляции Oracle 12с? Поясните их назначение.**

* SYSTEM используется для управления БД, содержит словарь базы
* данных, стандартные пакеты процедур.
* SYSAUX – вспомогательное табличное пространство.
* TEMP – временное табличное пространство по умолчанию.
* UNDOTBS1 – табличное пространство отката.
* USERS – хранение пользовательских объектов и данных.
* EXAMPLE – демонстрационные схемы.

1. **Что означает свойство табличных пространств smallfile/bigfile?**

Определяют тип файла, используемого для хранения данных в данном табличном пространстве

Smallfile – 1022 файлов, по 222 блоков: для длительного использования.

Bigfile – 1 файл, 128TB(блок 32К) или 32TB(блок 8К): для хранения одного большого файла данных.

1. **Что означает свойство табличных пространств logging/nologging?**

Свойства табличных пространств "logging" и "nologging" в Oracle определяют, будут ли операции загрузки данных в данное табличное пространство фиксироваться в журнале повтора (redo log).

Logging генерирует данные REDO во время обновления индекса / таблицы, вставки и удаления.

NOLOGGING останавливает создание данных REDO во время обновления, вставки и удаления индекса / таблицы. Вы получите лучшую производительность, но вы не сможете восстановить данные.

1. **Что означает свойство табличных пространств offline/online?**

Обычно мы переводим табличные пространства в автономный режим в целях обслуживания. Вы можете перевести онлайн-табличное пространство в автономный режим, чтобы оно было временно недоступно для общего использования. Остальная часть базы данных остается открытой и доступна пользователям для доступа к данным

1. **Каким образом можно выяснить наименование применяемого инстансом Oracle 12c имя UNDO-табличного пространства?**

SELECT tablespace\_name

FROM dba\_tablespaces

WHERE contents = 'UNDO';

Или в файле параметров с ключ словом .undo\_tablespace.

1. **Для чего необходимы журналы повтора?**

Журналы повторного выполнения - дисковые ресурсы, в которых фиксируются изменения вносимых пользователями в базу данных;

* журнал представляет собой файл операционной системы;
* как минимум должно быть два файла;
* журналы применяются при восстановлении базы данных.

1. **Поясните термин «мультиплексирование журналов повтора».**

Мультиплексирование журналов повтора – поддержка несколько копий

каждого журнала. Мультиплексирование файлов журналов повторного

выполнения посредством помещения двух или более копий этих файлов на

разные диски гарантирует, что вы не потеряете изменений данных, которые не

были записаны в файлы данных.

1. **Поясните термин «группа журналов повтора».**

Группа журналов повтора: при заполнении файлов группы запись происходит в файлы другой группы.

1. **Какие параметры регламентируют максимальное количество групп журналов повтора и максимальное количество файлов в группе?**

MAXLOGFILES - максимальное количество групп журналов повтора

MAXLOGMEMBERS - максимальное количество файлов в группе

1. **Где эти параметры находятся? Каким образом их можно их посмотреть? Каким образом изменить?**

Указывается в управляющих файлах

SHOW PARAMETER MAXLOGFILES;

SHOW PARAMETER MAXLOGMEMBERS;

ALTER SYSTEM SET MAXLOGFILES = новое\_значение SCOPE = SPFILE;

ALTER SYSTEM SET MAXLOGMEMBERS = новое\_значение SCOPE = SPFILE;

1. **Каким образом можно определить группу повтора, в настоящий момент используемую инстансом Oracle 12c?**

SELECT \* FROM V$LOG Поле STATUS.

1. **Расшифруйте аббревиатуру SCN. Что это такое?**

SCN – System change number – системный номер изменений в базе данных.

1. **Каким образом можно проследить последовательность SCN в журналах повтора?**

SELECT \* FROM V$LOG Поле FIRST\_CHANGE.

1. **Что такое архивы Oracle 12c?**

Архивы представляют собой скопированные и сохраненные версии журналов повтора. Архивы создаются с целью обеспечения точек восстановления и восстановления данных до конкретного момента времени.

1. **Каким образом можно проследить последовательность SCN в архивах и журналах повтора?**

SELECT NAME, FIRST\_CHANGE#, NEXT\_CHANGE#

FROM V$ARCHIVED\_LOG;

SELECT GROUP#, FIRST\_CHANGE#, NEXT\_CHANGE#

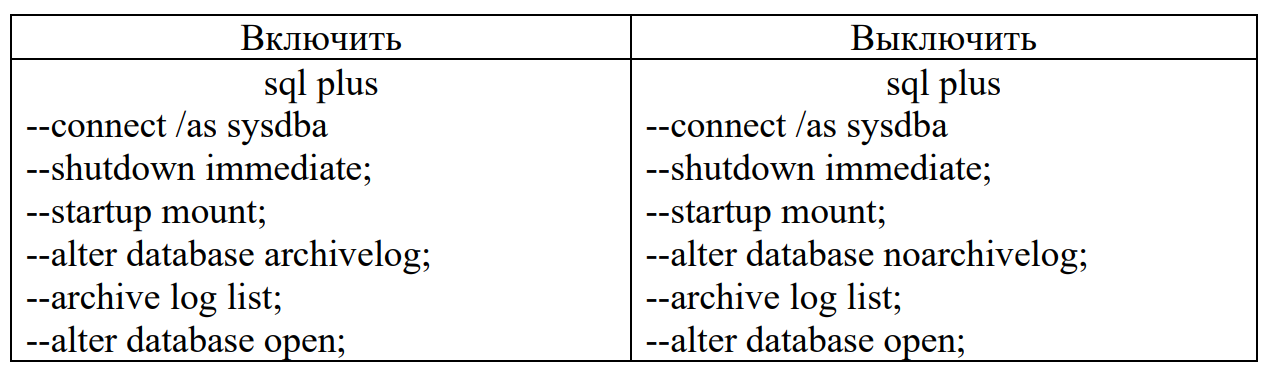
FROM V$LOG;

1. **Как выяснить, выполняется ли архивирование инстансом или нет?**

SELECT GROUP#, FIRST\_CHANGE#, NEXT\_CHANGE#

FROM V$LOG;

1. **Как включить архивирование и как выключить?**

****

1. **Как определить номер последнего архива?**

SELECT MAX(SEQUENCE#) AS LAST\_ARCHIVE\_LOG

FROM V$ARCHIVED\_LOG;

1. **Как определить местоположение архивных файлов?**

SELECT NAME AS ARCHIVE\_LOG\_FILE

FROM V$ARCHIVED\_LOG

1. **Что такое управляющие файлы Oracle 12c?**

Control files – файлы, содержащие имена (местоположение) основных физических файлов базы данных и некоторых параметров. Используются для поиска других файлов операционной системы.

1. **Поясните, почему требуется мультиплексирование управляющих файлов.**

Мультиплексирование управляющих файлов (Multiplexing of Control Files) в Oracle - это практика создания нескольких копий управляющих файлов базы данных и распределение их по разным физическим устройствам или директориям.

* Предотвращает потерю метаданных и обеспечивает функционирование базы данных при сбоях или повреждениях.
* Гарантирует наличие актуальных копий управляющих файлов, минимизируя риски потери данных и улучшая восстановление базы данных.
* Распределяет операции записи по разным устройствам, улучшая производительность и уменьшая нагрузку.
* Позволяет системе преждевременно выявлять несоответствия между копиями, предупреждая об аномалиях.

1. **Где задано количество и местоположение управляющих файлов?**

Местоположение управляющих файлов экземпляр получает из файла параметров. По умолчанию для надежности создается 2 управляющих файла. Можно создать больше. Обычно их размещают на разных дисковых носителях (для надежности).

$ORACLE\_HOME \oradata\orcl – расположение;

1. **Как получить содержимое управляющих файлов?**

show parameter control – содержимое

1. **Каким образом можно уменьшить/увеличить количество управляющих файлов.**

Если надо изменить управляющий файл, то следует создать сценарий,

откорректировать его и выполнить:

Остановить Oracle (shutdown transactional или immediate);

Скопировать один из управляющих файлов;

Изменить параметр CONTROL\_FILES в файле параметров;

Стартовать Oracle (startup open).

Добавление нового управляющего файла:

ALTER DATABASE ADD CONTROLFILE TO '/path/to/new/controlfile.ctl';

Уменьшение количества управляющих файлов:

Определите новое количество управляющих файлов с помощью параметра CONTROL\_FILES в файле параметров базы данных (init.ora или spfile.ora). Этот параметр должен содержать полные пути к управляющим файлам.

ALTER SYSTEM SET CONTROL\_FILES = '/path/to/new/controlfile1.ctl', '/path/to/new/controlfile2.ctl' SCOPE=SPFILE;

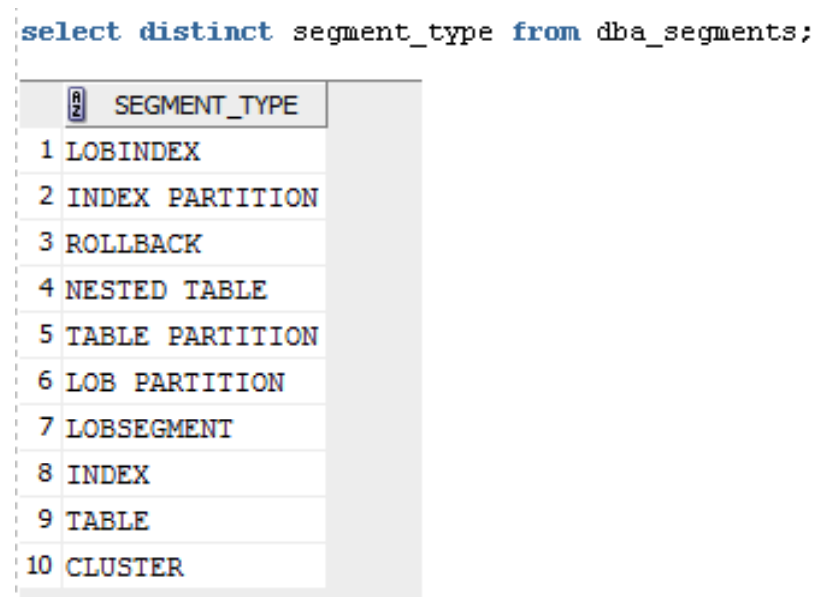
После изменения параметра, перезапустите базу данных для применения изменений.

**5 лаба**

1. **Что такое сегмент табличного пространства?**

Сегмент – область на диске, выделяемая под объекты. Сегменты типизируются в зависимости от типа данных, хранящихся в них – сегменты таблиц, сегменты индексов, сегменты кластеров и т.д.

1. **Перечислите типы сегментов?**

****

1. **Как получить все типы сегментов?**

SELECT SEGMENT\_NAME, SEGMENT\_TYPE

FROM DBA\_SEGMENTS;

1. **Удаляется ли (или сокращается) сегмент таблицы при удалении (DELETE) всех строк таблицы?**

Сразу после создания таблицы сегмент не создается. Сегмент хранит только данные, поэтому он создается только при добавлении данных.

Операция DELETE удаляет строки из таблицы, но сегмент остается в базе данных, занимая пространство.

1. **Что происходит с сегментом таблицы XXX\_T1 при удалении ее оператором drop table XXX\_T1?**

При удалении таблицы изменяется имя сегмента, и информация об удалении записывается в словарь базы данных

1. **Поясните назначение представление USER\_RECYCLEBIN.**

Представление USER\_RECYCLEBIN предоставляет информацию об объектах базы данных, которые были удалены с использованием операции DROP и временно помещены в корзину (recycle bin). Корзина — это механизм в Oracle, позволяющий восстанавливать удаленные объекты, а также освобождать пространство после окончательного удаления из корзины.

Таблица и ее содержимое (сегмент) могут быть восстановлены с

помощью механизма RECYCLEBIN

1. **Что происходит с сегментом таблицы XXX\_T1 при удалении ее оператором drop table XXX\_T1 purge?**

Происходит окончательное удаление таблицы и ее сегмента из базы данных, без возможности восстановления.

1. **Что такое экстент табличного пространства?**

Экстент – непрерывный фрагмент дисковой памяти. Является единицей выделения вторичной памяти (выделяется целым числом экстентов). Когда экстент заполняется выделяется следующий. Размер экстента варьируется от одного блока до 2 Гб.

1. **Поясните назначение опции UNIFORM при создании табличного пространства.**

Oracle стремится равномерно распределить блоки данных по всем файлам внутри табличного пространства, чтобы обеспечить балансировку нагрузки и оптимизировать производительность.

1. **Что такое блок данных табличного пространства?**

Блок данных табличного пространства (Data Block) в Oracle - это минимальная единица хранения данных в базе данных. Блок данных представляет собой фиксированный объем памяти, в который упакованы фактические данные таблицы.

1. **Где и как задается его размер?**

Размер блока данных задается при создании табличного пространства

Устанавливается в файле параметров экземпляра при создании БД

1. **Как выяснить размер блока?**

SELECT TABLESPACE\_NAME, BLOCK\_SIZE

FROM DBA\_TABLESPACES;

**ROWID** представляет собой уникальный идентификатор строки в таблице Oracle.

Он содержит информацию о файле данных, блоке данных и местоположении строки внутри блока. **ROWID** может быть использован для быстрого доступа к конкретной строке в таблице, так как он является фактическим адресом строки внутри базы данных. Пример **ROWID**: **AAAWP2AAEAAAADlAAA**.

ля хранения данных ROWID требуется 80 бит (10 байт). Идентификаторы ROWID состоят из четырех компонентов: номера объекта (32 бита), относительного номера файла (10 бит), номера блока (22 бита) и номера строки (16 бит). Эти идентификаторы отображаются как 18-символьные последовательности, указывающие местонахождение данных в БД, причем каждый символ представлен в формате base-64, состоящем из символов A-Z, a-z, 0-9, + и /. Первые шесть символов – это номер объекта данных, следующие три – относительный номер файла, следующие шесть – номер блока, последние три – номер строки.

**ROWSCN** (системный номер изменения строки) представляет собой число, которое увеличивается при каждом изменении строки в таблице. Это используется для контроля версий данных и предоставляет механизм определения времени, когда была внесена последняя модификация в строку. Когда строка обновляется, ей присваивается новый **ROWSCN**. **ROWSCN** можно использовать, например, для определения, когда данные были изменены, и выполнения действий на основе времени изменения данных.

**6 лаба**

1. **Что такое файл параметров?**

* **Файлы параметров**: Содержат конфигурационные параметры для базы данных Oracle, определяющие ее поведение и настройки. Они влияют на аспекты, такие как размеры памяти, параметры сети, и другие настройки, используемые при старте базы данных.
* **Управляющие файлы:** Содержат метаданные о структуре базы данных и служат для отслеживания изменений в базе данных. Необходимы для успешного запуска и восстановления базы данных.
* **Файлы паролей**: Содержат хеши паролей пользователей базы данных. Используются для аутентификации пользователей при подключении к базе данных.
* **Файлы сообщений**: Записывают сообщения, предупреждения и ошибки, возникающие в базе данных.

1. **Как выяснить его местоположение?**

SELECT name, value FROM V$PARAMETER

WHERE name IN ('spfile', 'pfile');

1. **В чем разница между SPFILE и PFILE?**

SPFILE - постоянно находящийся на сервере бинарный файл, который может быть изменен только с помощью команды "ALTER SYSTEM SET".

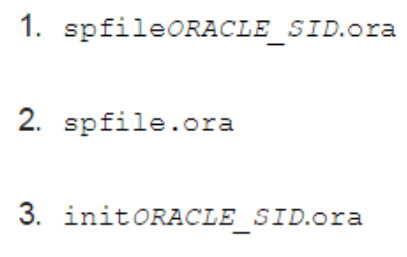
PFILE - статичный, пользовательский текстовый файл, который редактируется стандартными текстовыми редакторами

1. **Какая возможность появляется (приведите пример) при наличии бинарного формата файла параметров?**

нет необходимости хранить локальную копию файла параметров для запуска экземпляра с удаленной машины.

1. **В какой последовательности инстанс ищет файлы параметров?**

В зависимости от названия

****

1. **Каким образом можно получить PFILE из SPFILE?**

CREATE SPFILE= filename FROM PFILE= filename

CREATE PFILE=filename FROM SPFILE= filename

А если имеется ввиду получить их местоположение …

1. **Где находится файл паролей инстанса?**

SELECT \* FROM V$Passwordfile\_info;

1. **Каким образом можно выяснить местоположение файлов диагностики и сообщений?**

SELECT \* FROM V$DIAG\_INFO;

1. **Что находится в файле LOG.XML?**

Данный файл содержит все изменения, произведённые над базой данных

**7 лаба**

1. **Расшифруйте аббревиатуру SGA.**

Shared Global Area - Системная Глобальная область.

* группа областей разделяемой памяти
* содержат данные и управляющую информацию для одного экземпляра Oracle

1. **Перечислите основные пулы памяти SGA, поясните их назначение.**

Основные пулы области SGA:

* **Java pool** : Представляет пространство «кучи» для создания объектов Java.
* **Large pool** : область памяти SGA, применяемая для хранения больших фрагментов памяти.
* **Shared pool**: Разделяемый пул. Содержит библиотечный кэш для хранения, разобранного SQL и PL/SQL кода, готового к использованию всеми пользователями.
* **Streams pool** : Поддерживает средство Oracle Streams (средство для репликации данных между базами данных)
* **Буферный пул** (буферный кэш) – область SGA, которая содержит образы блоков, считанные из файлов данных или созданные динамически, чтобы реализовать модель согласованного чтения
* **Буфер журнала повторного выполнения** предназначен для временного циклического хранения данных журнала повтора. Позволяет ускорить работу сервера за счет буферизации
* **Фиксированная область.** Хранит переменные, указывающие на другие области памяти, значения параметров; размером фиксированной области SGA управлять нельзя.

1. **Поясните параметры SGA\_MAX\_SIZE и SGA\_TARGET.**

SGA\_MAX\_SIZE задает максимальный размер SGA для времени жизни экземпляра.

SGA\_TARGET – указывает текущий (возможный) размер памяти

1. **Поясните назначение буферного кэша инстанса.**

Буферный кэш базы данных состоит из буферов памяти, которые Oracle использует для хранения данных, прочитанных серверным процессом из файлов данных на диске в ответ на запросы пользователей. Доступ к буферному кэшу осуществляется намного быстрее, чем чтение данных из дискового хранилища.

1. **Поясните назначение пулов КЕЕP, DEFAULT и RECYCLE буферного кэша.**

* КЕЕP: Постоянно хранит блоки данных в памяти. Для предотвращения их удаления из буферного кэша им можно назначить постоянный буферный пул при создании таблицы.
* DEFAULT: Содержит все данные и объекты, которые не назначены в постоянный и повторно используемый буферные пулы
* RECYCLE: Удаляет данные из кэша немедленно после использования. Повторно используемый буферный пул удаляет объект из кэша сразу по завершении транзакции.

1. **Поясните принцип вытеснения блоков буферного кэша (LRU).**

В первый раз, когда пользовательскому процессу Oracle требуется определенный фрагмент данных, он выполняет поиск данных в буферном кэше базы данных. Если процесс находит данные уже в кэше (попадание в кэш), он может считывать данные непосредственно из памяти. Если процесс не может найти данные в кэше (промах кэша), он должен скопировать блок данных из файла данных на диске в буфер в кэше, прежде чем обращаться к данным.

Least recently used (LRU): в первую очередь, вытесняется неиспользованный дольше всех. Этот алгоритм требует отслеживания того, что и когда использовалось. Общая реализация этого метода требует сохранения «бита возраста» для строк кэша и за счет этого происходит отслеживание наименее использованных строк .

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

**CACHE** – помещение таблицы в конец LRU-списка (для малых таблиц) обычно в default pool.

Если вы последовательно выполните один и тот же запрос два или более раз, вы обратите внимание, что второй запросы выполняются быстрее, чем первый. Дело в том, что начальный запрос перемещает данные таблицы в буфера данных в памяти, где они остаются, пока не будут вытолкнуты другими данными из других таблиц, используемых для ответа на другие запросы. Чтобы сохранить конкретную таблицу в памяти, вы должны закрепить ее в кэше. Если вы обнаружили, что сервер выталкивает основные таблицы из памяти, вы можете закрепить их в памяти, используя параметр CACHE оператора CREATE/ALTER TABLE: ALTER TABLE [TABLENAME] CACHE; Table altered. Этот параметр гарантирует, что данные из таблицы после полного ее сканирования находятся в списке самых недавно использованных (most recently used - MRU) данных, а не в списке самых давно использованных (least recently used - LRU) данных, в результате чего они будут сохранены в памяти для последующего использования

1. **Как изменить размеры пулов?**

Память различным пулам в SGA выделяется блоками, которые называются гранулами. Одна гранула (granule) - это область памяти размером 4, 8 или 16 Мбайт. Гранула является наименьшей единицей выделения памяти.

ALTER SYSTEM SET SGA\_TARGET = …;

1. **Какие пулы допускают изменение размеров?**

• буферный кэш базы данных; - ALTER SYSTEM SET DB\_CACHE\_SIZE = …;

• разделяемый пул; - ALTER SYSTEM SET SHARED\_POOL\_SIZE = …;

• большой пул; - ALTER SYSTEM SET LARGE\_POOL\_SIZE = …;

• пул Java; - ALTER SYSTEM SET JAVA\_POOL\_SIZE = …;

• пул потоков. - ALTER SYSTEM SET STREAMS\_POOL\_SIZE = …;

1. **Поясните назначение процесса LISTENER.**

**Listener** – это программа-сервер, прослушивающая TCP-порт, принимающая запросы на соединение с Oracle экземпляром от программ-клиентов.

1. **Поясните назначение утилиты lsnrctl.**

Она предоставляет возможность управления слушателем, включая его запуск, остановку, перезапуск, а также выполнение различных операций с настройками и управление соединениями.

1. **start**: Запускает слушателя.
2. **stop**: Останавливает слушателя.
3. **status**: Отображает текущий статус слушателя (запущен, остановлен).
4. **reload**: Перезагружает конфигурацию слушателя без его остановки.
5. **save\_config**: Сохраняет текущую конфигурацию слушателя.
6. **trace**: Включает или отключает трассировку для слушателя.
7. **set**: Устанавливает параметры конфигурации слушателя (например, **set password**).
8. **unset**: Сбрасывает параметры конфигурации слушателя.
9. **services**: Отображает список сервисов, зарегистрированных в слушателе.
10. **version**: Отображает версию Oracle Net Listener.
11. **reload\_admin**: Перезагружает файл (listener.ora).
12. **save\_config**: Сохраняет текущую конфигурацию слушателя в файле.
13. **quit** или **exit**: Выходит из утилиты **lsnrctl**.
14. **Что такое сервис?**

Экземпляр может иметь несколько точек подключения

Точки подключения называются сервисами и имеют символические имена.

Сервис предоставляет клиентам доступ к базе данных и определяет параметры подключения, такие как адрес сервера базы данных, порт, имя базы данных и другие настройки.

1. **Какие сервисы создаются автоматически при инсталляции инстанса?**

a. **SYS$USERS:** При создании службы SYS$USERS автоматически используется SID базы данных в качестве имени службы.

b. **Служба с именем инстанса:** Этот сервис предоставляет пользовательский доступ к базе данных и может быть использован для более точного определения, к какой базе данных будет выполнено подключение.

1. **Поясните принцип работы dedicated-соединения и shared-соединения.**

**Dedicated** - (выделенный), обслуживает только один пользовательский процесс (каждому пользовательскому подключению создается выделенный процесс)

**Shared** - (распределённый), который обслуживает несколько пользовательских процессов (ранее известный как MTS - multi-threded-server), это когда имеем пул процессов, который асинхронно раздаем пользователям. Уменьшается количество серверных процессов.

По умолчанию всегда сконфигурирован dedicated сервер

1. **Поясните назначение файла LISTENER.ORA.**

Этот конфигурационный файл отвечает за связь Листенера с СУБД. Для нас важнейшим моментом является хранимая в нем строка подключения, которая содержит такие параметры подключения, как системный идентификатор (SID) и порт, на который будут приниматься запросы для данного SID. Этот файл очень важен для нас – получив к нему доступ с возможностью внесения модификаций, мы сможем обойти такие ограничения безопасности, как пароль на службу Листенера и протоколирование событий.

1. **Перечислите основные фоновые процессы, перечислите их назначение.**

**Фоновые процессы** – запускаются вместе с базой данных и выполняют разнообразные задачи обслуживания

Фоновые (background) процессы – специальная группа процессов для обеспечения производительности и поддержки работы большого числа пользователей

**LREG** - Listener Registration Process – периодическая регистрация сервисов в процессе Listener

**DBWn (Database Writer Process):** Записывает изменения из кэша базы данных на диск. Он делает это по мере необходимости, используя алгоритм LRU для определения, какие блоки записывать.

**CKPT (Checkpoint Process):** Создает контрольные точки в базе данных. Контрольные точки фиксируют текущее состояние базы данных, что помогает восстановлению после сбоев. CKPT записывает информацию о контрольной точке, сигнализирует другим процессам о необходимости сброса буферов и обновляет соответствующие управляющие файлы.

**LGWR (Log Writer Process):** Записывает изменения базы данных в журналы повтора до их фиксации DBWR. Это важно для обеспечения целостности данных и ускорения восстановления базы данных в случае сбоев.

**ARCn (Archiver Process):** Копирует журналы повтора после их переключения, обеспечивая сохранность данных для возможного восстановления.

**PMON (Process Monitor Process):** Отслеживает и управляет фоновыми процессами базы данных. Он восстанавливает работу процессов после сбоев, откатывает незафиксированные транзакции и освобождает ресурсы.

**SMON (System Monitor Process):** Занимается управлением и очисткой данных, восстановлением после сбоев и выполнением различных задач по управлению пространством и ресурсами базы данных.

**RECO (Recovery Process):** Решает проблемы, связанные с распределенными транзакциями. Он координирует фиксацию или откат транзакций и автоматически восстанавливает или откатывает транзакции в случае сбоев.

**FBDA -** Flashback Data Archiever – архивирование ретроспективных данных

1. **Что такое серверный процесс? Как просмотреть серверные процессы?**

**Серверные процессы** – процессы, выполняющиеся на основании клиентского запроса

Серверный процесс:

* Получает и выполняет SQL-операторы
* Читает файлы данных
* Осуществляет поиск в кеше

Назначение — отвечать на получаемые SQL-запросы

SELECT \* FROM V$PROCESS;

**8 лаба**

1. **Поясните принцип установления соединения с сервером Oracle по сети.**
2. **Объясните назначение файлов SQLNET.ORA, TNSNAMES.ORA, LISTENER.ORA.**

* LISTENER.ORA используется для настройки и конфигурации службы Oracle Net Listener, которая прослушивает входящие сетевые соединения и маршрутизирует их к соответствующим экземплярам баз данных.
* TNSNAMES.ORA содержит сведения о конфигурации сетевых подключений к базам данных Oracle. Он определяет сетевые алиасы для баз данных, указывая на имя хоста, порт и службу базы данных.
* SQLNET.ORA содержит параметры конфигурации сетевого уровня Oracle, определяющие параметры безопасности и параметры сетевых соединений. Он используется для настройки аутентификации клиента и сервера, установки параметров шифрования и управления механизмами безопасности.

1. **Какие виды соединений вы знаете? Кратко охарактеризуйте каждое из них.**

Простое подключение – **Basic**

* Должны быть установлены Oracle Net Services
* Поддержка протокола TCP/IP – на сервере и клиенте
* Нельзя использовать расширенные сетевые функциональные возможности Oracle

Локальное именование – **TNS**

* Требует файла Разрешения Имен на клиентской стороне
* Поддерживает все протоколы Сети Oracle
* Поддерживает продвинутые опции соединения, такие как:
  + Преодоление отказа во время соединения
  + Маршрутизация источника
  + Выравнивание нагрузки

**LDAP**-соединение

* Метод именования с помощью службы каталогов
* Требуется наличие специального LDAP-сервера

**Local/bequeath-**соединение

* Только на сервере
* Можно соединяться с помощью sqlplus или sqldeveloper без указания параметров соединения
* только с выделенным сервером
* Listener не задействован
* Соединение со стандартным сервисом SYS$USERS

1. **Что такое строка подключения?**

Строка подключения содержит в себе имя пользователя, пароль, адрес,

включающий название протокола, адрес хоста и порт, а также имя сервиса.

1. **Что такое дескриптор подключения?**

Дескриптор соединения – объединенная спецификация двух обязательных компонентов подключения к базе данных:

* Имени службы базы данных
* Местоположения адреса базы данных

1. **Расшифруйте аббревиатуру TNS.**

Протокол TNS (Transparent Network Substrate) — уровень связи, используемый базами данных Oracle. Имя службы TNS — это имя, с которым экземпляр базы данных Oracle представлен в сети. Имя службы TNS назначается при настройке подключений к базе данных Oracle.

1. **Объясните назначение утилиты Oracle Net Manager.**

Инструмент Oracle Net Manager может выполняться на клиентах и

серверах, и он позволяет конфигурировать различные методы именования и

слушателей. С помощью этого средства можно конфигурировать дескрипторы

соединений в локальных файлах tnsnames.ora или в централизованном OID, а

также легко добавлять и изменять методы подключения.

1. **Перечислите все этапы запуска и останова экземпляра Oracle, поясните каждый этап.**

**Запуск**

**STARTUP NOMOUNT:** Запуск экземпляра Oracle без монтирования

базы данных. Доступ пользователей запрещен. Используется для создания БД

и пересоздания управляющих файлов;

**STARTUP MOUNT:** Запустить экземпляр, монтировать БД, но не

запускать ее. Считан файл параметров. Происходит обращение к

управляющим файлам. Проверяется состояние файлов БД. Доступ

пользователей запрещен. Из состояния STARTUP NOMOUNT переводится

ALTER DATABASE MOUNT;

**3. STARTUP OPEN:** Запустить экземпляр. Монтировать и открыть БД.

Могут присоединяться пользователи. Из состояния STARTUP MOUNT

переводится ALTER DATABASE OPEN.

**Остановка**

**SHUTDOWN NORMAL:** Запрещено создавать новые сессии. Ожидается завершение работы всех пользователей. Самый безопасный и долгий способ останова. Никаких восстановительных работ при следующем старте не проводится;

**SHUTDOWN TRANSACTIONAL:** Запрещено создавать новые сессии. Запрещено запускать новые транзакции. Сервер дожидается завершения уже начатых транзакций и отключает пользователей, не имеющих активных транзакций. Применяется в случаях, когда нет возможности применить NORMAL. Никаких восстановительных работ при следующем старте не проводится.

**SHUTDOWN IMMEDIATE:** Запрещено создавать новые сессии. Запрещено запускать новые транзакции. Все незафиксированные транзакции откатываются. Применяется в случаях, когда нет возможности ждать. Никаких восстановительных работ при следующем старте не проводится.

**SHUTDOWN ABORT:** Применяется в крайних случаях, когда остальные режимы останова не приводят к результату. Все действия прекращаются. Все транзакции не фиксируются и не откатываются. Пользователей отсоединяют от БД. При следующем старте будет выполнено возможное восстановление.

1. **Какое имя группы пользователей Windows использует Oracle для администраторов?**

ora\_dba

**9 лаба**

1. **Что такое последовательность?**

Последовательность – это объект базы данных, который генерирует целые числа в соответствии с правилами, установленными во время его создания.

1. **Перечислите основные параметры последовательности.**

* schema —схема, в которой создается последовательность. Если schema опущена, Oracle создает последовательность в схеме пользователя.
* sequence — имя создаваемой последовательности
* start with —первое генерируемое ею значение.
* increment by n — определяет приращение последовательности при каждой ссылке на виртуальный столбец NEXVAL. Если значение не указано явно, по умолчанию устанавливается 1.
* minvalue — определяет минимальное значение, создаваемое последовательностью. Если оно не указано, Oracle применяет значение по умолчанию NOMINVALUE
* nominvalue — указывает, что минимальное значение равно 1, если последовательность возрастает, или -1026, если последовательность убывает
* maxvalue — определяет максимальное значение, создаваемое последовательностью. Если оно не указано, Oracle применяет значение по умолчанию NOMAXVALUE
* nomaxvalue — указывает, что максимальное значение равно 1027, если последовательность возрастает, или -1, если последовательность убывает. По умолчанию принимается NOMAXVALUE
* cycle — позволяет последовательности повторно использовать созданные значения при достижении MAXVALUE или MINVALUE. Т.е. последовательность будет продолжать генерировать значения после достижения своего максимума или минимума. Возрастающая последовательность после достижения своего максимума генерирует свой минимум. Убывающая последовательность после достижения своего минимума генерирует свой максимум.
* nocycle — указывает, что последовательность не может продолжать генерировать значения после достижения своего максимума или минимума
* cache n — указывает, сколько значений последовательности ORACLE распределяет заранее и поддерживает в памяти для быстрого доступа. Минимальное значение этого параметра равно 2. Для циклических последовательностей это значение должно быть меньше, чем количество значений в цикле. Если кэширование нежелательно или не установлено явным образом, Oracle применяет значение по умолчанию – 20 значений.
* order — гарантирует, что номера последовательности генерируются в порядке запросов.
* noorder — не гарантирует, что номера последовательности генерируются в порядке запросов

1. **Какие привилегии необходимы для создания и удаления последовательности?**

CREATE SEQUENCE

1. **Что такое кластер?**

Кластер – объект БД, который хранит значения общих столбцов нескольких таблиц. Если две таблицы имеют идентичный столбец и вам, к примеру, часто приходится соединять таблицы по нему, то становится выгодно хранить значения общих столбцов в одном и том же блоке данных. Строки из отдельных таблиц сохраняются в одних и тех же блоках, поэтому объединяющие запросы выполняются быстрее.

1. **Что означает параметр hash?**

Хэш-кластеры используют функции хэширования кластерного ключа строки для определения физической локализации места, где строку следует хранить.

1. **Какие привилегии необходимы для создания и удаления кластера?**

CREATE CLUSTER

1. **Что такое синоним?**

Синонимы— это псевдонимы объектов базы данных, которые служат в основном для облегчения пользователям доступа к объектам, принадлежащим другим пользователям, а также в целях безопасности.

1. **Чем отличается публичный синоним от частного синонима?**

Общедоступные (public) синонимы относятся к специальной схеме базы данных Oracle, именуемой PUBLIC. Общедоступные синонимы видны всем пользователям базы данных. Приватные синонимы, в отличие от общедоступных, видны только в схеме, владеющей таблицей или объектом. Приватные синонимы можно создать, когда нужно обращаться к одной и той же таблице в разных контекстах под разными именами.

1. **Что такое материализованное представление?**

Представление (view), или виртуальная таблица в базе данных Oracle Database — это специфический образ таблицы или набора таблиц, определенный оператором SELECT. Представление не существует физически как обычная таблица, являющаяся частью табличного пространства.

Привилегия – CREATE VIEW OR REPLACE – если представление уже существует оно будет удалено перед созданием нового

FORCE или NOFORCE – Использование FORCE приведёт к созданию представления даже если базовые таблицы не существуют. NOFORCE значение по умолчанию и если таблицы не существуют команды выполняется с ошибкой

WITH CHECK OPTION – эта директива влияет на DML команды. Если подзапрос включает условие WHERE, тогда эта директива предотвратит возможность вставки строк которые не видно в представлении, или совершать обновления данных которое приведёт к пропаже данных из представления.

WITH READ ONLY – отключения возможности использование DML команд к представлению

CONSTRAINT constraintname – позволяет назначить имя ограничениям WITH CHECK OPTION и WITH READ ONLY и сообщения об ошибке станут более понятными

Материализованное представление - это объект базы данных, который хранит результат запроса (SELECT-запроса) в физическом виде в базе данных. Это позволяет улучшить производительность при выполнении запросов к данным, особенно при часто повторяющихся и сложных запросах. Привилегия – CREATE MATERIALIZED VIEW

• BUILD IMMEDIATE заполняет материализованное представление во время его создания (значение по умолчанию).

•REFRESH указывает, как оракл обновляет данные материализованного представления. COMPLETE — полное обновление данных из базовых таблиц. FAST – используются журналы фиксации изменений базовых таблиц. FORCE – попытка быстрого обновления; если быстрое обновление невозможно, то выполняется полное обновление

•REFRESH также указывается, когда оракл обновляет данные материализованного представления. В этом примере, материализованное представление будет обновляться по требованию (ON DEMAND) – только, когда вы явно обновляете его. (ON COMMIT) – каждый раз когда выполняется фиксация транзакции для мастер таблицы (таблиц) представления.

•QUERY REWRITE необходимо включить, чтобы оптимизатор оракл переписывал запросы таким образом, чтобы использовались материализованные представления.

•Предложение AS описывает столбцы и строки материализованного представления, используя определяющий запрос.

•START WITH – показывает, когда выполнится в первый раз (если не был построен сразу)

•NEXT– показывает, когда выполнится в следующий раз

1. **Что такое DBlink?**

Database Link (dblink) - объект базы данных, предназначенный для доступа к объектам базы данных, управляемой другим сервером.

1. **Какие виды DBlink вы знаете?**

Приватный и общедоступный.

1. **Поясните, чем отличается public DBlink от обычного.**

Приватная связь базы данных принадлежит пользователю, который ее создал. Общедоступная связь базы данных позволяет любому пользователю обращаться к объектам удаленной базы данных.

1. **Какие привилегии необходимы для создания и удаления DBlink?**

GRANT CREATE DATABASE LINK TO USERNAME

1. Временные таблицы

Для создания временных таблиц используется оператор CREATE GLOBAL TEMPORARY TABLE с ключевыми словами ON COMMIT PRESERVE ROWS (хранение данных на время сеанса, возможны все DML-операторы, TCL-операторы) или ON COMMIT DELETE ROWS (хранение данных на время транзакции, возможны все DML-операторы, после выполнения COMMIT или ROLLBACK таблица становится пустой).

Место для временных таблиц выделяется во временных сегментах только после первого применения команды INSERT с этими таблицами (по умолчанию во временном табличном пространстве TEMP). Каждый пользователь видит только свои данные (свой сегмент данных).

**Лаба 10**

1. **Поясните тезис «PL/SQL – процедурное расширение SQL»**

Язык PL/SQL является собственным расширением языка SQL от Oracle и предлагает функциональность серьезного языка программирования. Одно из главных его преимуществ состоит в том, что он позволяет использовать в базе данных такие программные единицы, как процедуры и пакеты

1. **Чувствителен ли к регистру язык PL/SQL?**

Нет

1. **Какие виды блоков бывают в PL/SQL?**

В PL/SQL код не выполняется в однострочном формате, а всегда выполняется путем группировки кода в один элемент, называемый блоками.

1. Заголовок. Применяется лишь в именованных блоках, служит для определения способа вызова программы либо именованного блока.

2. Раздел объявлений. Включает описания переменных, вложенных блоков и курсоров.

3. Раздел исключений. Служит для обработки исключений (предупреждений и ошибок).

4. Исполняемый раздел. Речь идёт о командах, которые выполняются ядром PL/SQL в процессе работы приложения. Обратите внимание, что это ОБЯЗАТЕЛЬНЫЙ раздел.

1. **Что такое анонимный блок?**

* Не имеет секции заголовка
* Не может быть вызван из другого блока
* Начинается с DECLARE или BEGIN

1. **Как выглядит простейший анонимный блок PL/SQL?**

Begin

Null;

End;

1. **Что необходимо предпринять, для того что бы вывод процедуры dbms\_output.put\_line отображался в SQL+, SQLDevеloper.**

Для того, чтобы вывод DBMS\_OUPUT отображался в SQLDeveloper необходимо это установить с помощью соответствующей команды в меню View и подключить появившееся окно к одному из подключений (на окне зеленый +). Для того, чтобы вывод DBMS\_OUPUT отображался в SQL+ необходимо выполнить команду set serveroutput on

1. **Что такое исключение PL/SQL?**

Исключение возникает, когда механизм PL / SQL встречает инструкцию, которую он не может выполнить из-за ошибки, возникающей во время выполнения. Исключения будут препятствовать дальнейшему выполнению программы, поэтому, чтобы избежать такого условия, они должны быть записаны и обработаны отдельно. Этот процесс называется обработкой исключений, при которой программист обрабатывает исключение, которое может возникнуть во время выполнения.

1. **Что такое вложенные блоки PL/SQL?**

Блок может быть вложен в другой блок. Каждый внутренний блок снова является блоком PL / SQL, поэтому все свойства и характеристики внутреннего блока будут такими же, как внешний блок.

* Элементы, объявленные во внешнем блоке, и значение, определенное до определения внутреннего блока, видны внутри внутреннего блока.
* Элементы, объявленные во внутреннем блоке, не видны во внешнем блоке. Они видны только внутри внутреннего блока.
* Внешний блок и внутренний блок могут иметь переменную с одинаковым именем.
* В случае переменных с одинаковым именем внутренний блок по умолчанию будет ссылаться только на переменную, объявленную во внутреннем блоке.
* Если внутренний блок хочет ссылаться на переменную внешнего блока, имеющую то же имя, что и у внутреннего блока, тогда внешний блок должен быть LABELLED, а переменная внешнего блока может называться ' . '

1. **Для чего они применяются?**

Вложенные блоки очень распространены, когда мы хотим выполнить определенный процесс, и в то же время код для этого процесса должен храниться в отдельном блоке.

1. **Как работают секции исключения во вложенных блоках?**

В PL/SQL секции исключений могут быть использованы во вложенных блоках для обработки исключений на разных уровнях. Когда исключение возникает во вложенном блоке, PL/SQL ищет соответствующий обработчик исключений в самом вложенном блоке. Если обработчик не найден, поиск продолжается во внешних блоках, начиная с ближайшего и постепенно двигаясь к наружным блокам.

Если обработчик исключений найден во вложенном блоке, он выполняется, и управление переходит к следующему оператору после блока исключений. Если обработчик не найден в текущем блоке, PL/SQL ищет обработчик в его родительском блоке и так далее до главного блока программы.

1. **Совпадают ли типы данных базы данных Oracle и PL/SQL?**

Хотя большинство типов данных в PL/SQL совпадает с типами данных в Oracle, PL/SQL также имеет свои собственные особенности, такие как BOOLEAN, которого нет в базе данных Oracle, а также композитные типы данных RECORD и TABLE. Также стоит отметить, что PL/SQL может работать с типами данных, определенными в базе данных Orac

* Скалярные (Scalar)
* Ссылочные (Reference)
* Составные (Composite)
* Большие объекты (LOB)

1. **Какая семантика символов используется в Oracle?**

* Байтовая семантика рассматривает строки как последовательность байтов
* Символьная семантика рассматривает строки как последовательность символов
* Задается параметром NLS\_LENGTH\_SEMANTICS
* По умолчанию - BYTE

1. **Перечислите все типы данных для хранения символьной информации в базе данных Oracle и программе на PL/SQL.**

* CHAR(n):
* NCHAR(n):
* VARCHAR2(n):
* NVARCHAR2(n):
* LONG:
* RAW(n)
* LONG RAW
* CLOB:
* NLOB:
* BLLOB
* BFILE

1. **Перечислите все типы данных для хранения числовых данных в базе данных Oracle и программе на PL/SQL.**

* NUMBER

1. **Перечислите все операции, которые использовались при выполнении работы.**

**IF**:

* Оператор условия, который выполняет блок кода, если определенное условие истинно.

**CASE**:

* Позволяет выполнить различные действия в зависимости от значения выражения.

**LOOP**:

* Оператор, который позволяет выполнить блок кода несколько раз.
* Может быть бесконечным (без условия завершения) или иметь условие выхода.

**WHILE**:

* Оператор цикла, который выполняет блок кода, пока условие истинно.
* Проверка условия выполняется перед выполнением каждой итерации цикла.

**FOR**:

* Оператор цикла, который выполняет блок кода для каждого значения в указанном диапазоне или коллекции.
* Обычно используется для итерации по числовым диапазонам или коллекциям.

1. **Перечислите представления словаря базы данных, хранящие информацию связанную PL/SQL.**

v$parameter, v$reserved\_words,v$timezone\_names

1. **Как получить все параметры Oracle связанные с PL/SQL?**

select name,value from v$parameter where name like 'plsql%'

SELECT \* FROM ALL\_PLSQL\_OBJECT\_SETTINGS;

**Лаба 11**

**Курсор** — объект БД, который позволяет приложениям работать с записями построчно

1. **Что такое точная и неточная выборки?**

Точная выборка – выборка, возвращающая одну строку.

Неточная выборка – выборка, возвращающая несколько строк

1. **Объясните действие конструкций WHEN OTHERS, WHEN TO\_MANY\_ROWS, WHEN NO\_DATA\_FOUND в секции исключения.**

**WHEN OTHERS**: Эта конструкция обрабатывает все исключения, которые не совпадают с явно указанными в блоке обработки исключений. Она действует как "поймать все", обрабатывая любые исключения, которые не были обработаны другими конструкциями WHEN.

**WHEN TOO\_MANY\_ROWS**: Эта конструкция используется, когда операция SQL запроса возвращает более одной строки, а ожидается, что будет возвращена только одна строка.

**WHEN NO\_DATA\_FOUND**: Эта конструкция используется, когда операция SQL запроса не возвращает ни одной строки, а ожидается, что будет найдена хотя бы одна строка.

1. **Объясните назначение функций SQLERRM и SQLCODE.**

**SQLERRM**: Эта функция возвращает сообщение об ошибке, связанное с последней операцией SQL, которая была выполнена в блоке PL/SQL. Это сообщение обычно является текстовым описанием ошибки, которое может быть полезным для понимания причины возникновения ошибки.

**SQLCODE**: Эта функция возвращает код ошибки, связанный с последней операцией SQL, выполненной в блоке PL/SQL. Код ошибки является числовым значением и обычно используется для определения типа ошибки и принятия соответствующих действий на основе этого типа ошибки.

1. **Что такое атрибут курсора? Перечислите все атрибуты курсора и объясните их назначение**

**Атрибуты курсора** - это свойства, которые позволяют получить информацию о текущем состоянии курсора в PL/SQL.

**%FOUND:** Этот атрибут возвращает значение TRUE, если последняя операция FETCH из курсора успешно извлекла строку. Если ни одна строка не была извлечена или операция FETCH еще не была выполнена, атрибут возвращает FALSE.

**%NOTFOUND:** Атрибут %NOTFOUND возвращает значение TRUE, если последняя операция FETCH из курсора не извлекла строки (то есть не было найдено строк, удовлетворяющих критериям запроса). В противном случае он возвращает FALSE.

**%ROWCOUNT**: Этот атрибут возвращает количество строк, обработанных последней операцией SQL (например, INSERT, UPDATE или DELETE), выполненной с использованием курсора.

**%ISOPEN**: Атрибут %ISOPEN возвращает TRUE, если курсор открыт (OPEN), и FALSE в противном случае. Он позволяет проверить состояние открытия курсора перед выполнением операций с ним.

1. **Объясните назначение конструкции RETURNING в операторах INSERT, DELETE, UPDATE.**

**RETURNING** в операторах INSERT, DELETE и UPDATE в SQL используется для получения значений столбцов, которые были изменены, вставлены или удалены в результате выполнения соответствующей операции. Это позволяет получить новые значения столбцов после выполнения операции без необходимости выполнения дополнительного запроса.

1. **В чем отличие явного и неявного курсоров.**

явный — объявляется разработчиком;

неявный — не требует объявления

1. **Объясните схему работы с явным курсором в PL/SQL-блоке.**

* Сначала необходимо объявить курсор в PL/SQL блоке с помощью ключевого слова CURSOR, указав запрос, который будет выполняться при открытии курсора. Курсор можно объявить как с параметрами, так и без них.

|  |
| --- |
| CURSOR cursor\_name IS  SELECT column1, column2  FROM table\_name  WHERE condition; |

* После объявления курсора его нужно открыть с помощью оператора OPEN. Это действие запускает выполнение запроса, и курсор становится активным.

OPEN cursor\_name;

* После открытия курсора можно использовать циклы или операторы FETCH для извлечения данных из курсора по одной строке за раз. Данные могут быть присвоены переменным или использованы для выполнения каких-либо действий.

|  |
| --- |
| LOOP  FETCH cursor\_name INTO variable1, variable2;  EXIT WHEN cursor\_name%NOTFOUND;  -- Выполнение действий с данными  END LOOP; |

* По завершении работы с курсором его следует закрыть с помощью оператора CLOSE. Это освободит ресурсы, занятые курсором, и завершит его работу.

1. **Что происходит по команде OPEN курсора?**

* При вызове OPEN происходит инициализация запроса, связанного с курсором.
* После инициализации запроса он отправляется к базе данных для выполнения. База данных обрабатывает запрос и возвращает первую порцию данных, удовлетворяющих условиям запроса.
* После успешного выполнения запроса курсор становится активным. Это означает, что он готов к использованию для извлечения данных из результирующего набора.
* После открытия курсора текущая позиция курсора устанавливается на первую строку результирующего набора данных. Таким образом, курсор готов начать извлечение данных с первой строки.

1. **Объясните особенность применения FOR-цикла при работе с явным курсором.**

Сначала необходимо объявить курсор в PL/SQL блоке, как это делается обычно.

CURSOR cursor\_name IS

SELECT column1, column2

FROM table\_name

WHERE condition;

Затем курсор может быть использован внутри конструкции FOR-цикла для автоматической итерации по результирующему набору данных. В этом случае, после слова FOR, указывается итератор, переменные, в которые будут записываться значения из результирующего набора, и ключевое слово IN, за которым следует курсор.

FOR record IN cursor\_name LOOP

-- Обработка данных, например:

DBMS\_OUTPUT.PUT\_LINE('Column1: ' || record.column1 || ', Column2: ' || record.column2);

END LOOP;

При использовании FOR-цикла PL/SQL автоматически открывает курсор перед началом цикла и закрывает его после завершения работы с ним. Это упрощает код и обеспечивает автоматическое управление ресурсами. При проходе явного курсора при помощи for-цикла создаётся переменная при начале описания цикла, которая будет содержать строку,

берущуюся из курсора на каждой итерации цикла for

1. **Для чего применяются параметры курсора? Где они указываются?**

**Параметры** **курсора** используются для передачи значений в запрос, который используется в курсоре, что позволяет делать запросы более динамическими и универсальными. Они позволяют использовать один и тот же запрос с разными наборами параметров без необходимости изменения самого запроса. Параметры курсора указываются в объявлении курсора.

DECLARE

CURSOR cursor\_name (param1 VARCHAR2, param2 NUMBER) IS

SELECT column1, column2

FROM table\_name

WHERE condition\_column = param1

AND another\_condition\_column = param2;

1. **Что такое курсорная переменная? Как ее можно объявить?**

**Курсорная переменная** в PL/SQL представляет собой переменную, которая содержит ссылку на курсор. Это позволяет создавать динамические курсоры и управлять ими во время выполнения программы. Курсорная переменная объявляется с использованием специального типа данных REF CURSOR в PL/SQL.

1. **Что значит курсорный подзапрос?\**

**Курсорный подзапрос** представляет собой подзапрос, который внешнему запросу возвращает курсор, а не просто набор значений. Этот курсор может затем использоваться внешним запросом для обработки данных.

Основная идея курсорного подзапроса состоит в том, что каждая строка, возвращаемая внешним запросом, вызывает выполнение подзапроса, который затем возвращает курсор для этой строки. Таким образом, каждая строка внешнего запроса имеет свой собственный набор данных, возвращаемый подзапросом.

1. **Объясните назначение конструкции CURRENT OF.**

CURRENT OF в SQL используется для обновления или удаления текущей строки, на которой находится курсор. Это позволяет обращаться к строке в результате курсора без явного указания условий в WHERE-предложении.

**Для чего применяется псевдостолбец ROWID? Поясните его структуру.**

ROWID представляет собой уникальный идентификатор строки в таблице Oracle.

Он содержит информацию о файле данных, блоке данных и местоположении строки внутри блока. ROWID может быть использован для быстрого доступа к конкретной строке в таблице, так как он является фактическим адресом строки внутри базы данных.

Первые шесть символов – это номер объекта данных, следующие три – относительный номер файла, следующие шесть – номер блока, последние три – номер строки.

1. **Для чего применяется псевдостолбец ROWNUM?**

ROWNUM - логический номер записи в запросе

Псевдостолбец ROWNUM в SQL используется для ограничения количества возвращаемых строк в результирующем наборе данных. Он присваивает каждой возвращенной строке уникальный номер в порядке их выборки из таблицы или представления.

**Лаба 12**

1. **Объясните разницу между функциями floor(), ceil(), round() и trunc().**

* Функция FLOOR() возвращает наибольшее целое число, которое меньше или равно заданному числу.
* Функция CEIL() возвращает наименьшее целое число, которое больше или равно заданному числу.
* Функция ROUND() округляет заданное число до ближайшего целого числа или до заданного количества десятичных знаков.
* Функция TRUNC() обрезает (усекает) число до указанного количества десятичных знаков без округления.

1. **Перечислите встроенные функции для работы с датами, исследованные в лабораторной работе.**

* SYSDATE Возвращает текущую дату и время системы базы данных.
* EXTRACT() Извлекает часть даты (год, месяц, день).
* TO\_CHAR() Преобразует дату в строку по заданному формату.
* ADD\_MONTHS() Добавляет указанное количество месяцев к дате.

1. **Перечислите встроенные функции конвертации, исследованные в лабораторной работе.**

* Функция TO\_CHAR() преобразует дату или числовое значение в строку по заданному формату.
* Функция TO\_DATE() преобразует строку в дату по заданному формату.
* Функция TO\_NUMBER() преобразует строку в числовое значение.

1. **Объясните назначение встроенных функций sqlcode и sqlerrm.**

**SQLERRM**: Эта функция возвращает сообщение об ошибке, связанное с последней операцией SQL, которая была выполнена в блоке PL/SQL. Это сообщение обычно является текстовым описанием ошибки, которое может быть полезным для понимания причины возникновения ошибки.

**SQLCODE**: Эта функция возвращает код ошибки, связанный с последней операцией SQL, выполненной в блоке PL/SQL. Код ошибки является числовым значением и обычно используется для определения типа ошибки и принятия соответствующих действий на основе этого типа ошибки.

1. **Поясните принцип обработки исключений в PL/SQL.**

Каждое условие WHEN сопровождается именем исключения, которое, как ожидается, будет вызвано во время выполнения.

Когда какое-либо исключение возникает во время выполнения, механизм PL / SQL будет искать в этой части исключение в части обработки было исключений. Он будет начинаться с первого предложения WHEN и последовательно будет выполнять поиск. Если он обнаружил обработку исключения для возникшего исключения, то он выполнит эту конкретную часть кода обработки. Если ни одно из условий «WHEN» не присутствует в исключении, которое сгенерировано, то механизм PL / SQL выполнит часть «WHEN OTHERS» (если присутствует). Это общее для всех исключений. После выполнения исключения управление деталями выйдет из текущего блока.

**Лаба 13**

1. **Что такое локальные процедуры и функции PL/SQL?**

Локальные процедуры и функции в PL/SQL – это подпрограммы, которые объявляются и определяются внутри другого блока PL/SQL, например, внутри анонимного блока, процедуры или функции. Они доступны только в пределах этого блока и недоступны за его пределами.

* Объявление локальных процедур и функций должно размещаться в конце секции декларации после всех типов, записей, курсоров, переменных и исключений

1. **Чем отличаются процедуры от функций?**

**Функция** – именованный модуль, который выполняет ноль или более выражений через фразу Return.

* Всегда возвращают значение. Они должны включать ключевое слово RETURN, определяющее тип возвращаемого значения, и оператор RETURN внутри тела функции, чтобы фактически вернуть значение.
* Могут быть использованы в SQL выражениях, таких как SELECT, INSERT, UPDATE, и DELETE. Это возможно, потому что функции возвращают значение.
* Предназначены для вычислений и возвращения значений

**Процедура** – именованный модуль, который выполняет одно или

несколько выражений и может принимать или возвращать значения

через список параметров.

* Не возвращают значение непосредственно. Они могут использовать выходные параметры (OUT или IN OUT) для возврата значений, но нет обязательного оператора RETURN для завершения процедуры с результатом.
* Не могут быть вызваны непосредственно из SQL выражений. Они вызываются через команды PL/SQL
* Используются для выполнения операций, таких как обновление данных в таблицах, выполнение логики, отправка сообщений и т.д.

1. **Что такое хранимые процедуры и функции?**

Хранимые процедуры — это подпрограммы, которые выполняют одну или несколько задач и могут принимать входные параметры, но не обязательно возвращают значение. Они могут выполнять сложные операции, включая манипуляции с данными, выполнение бизнес-логики и вызов других подпрограмм

Хранимые функции — это подпрограммы, которые выполняют определённые вычисления и обязательно возвращают значение. Функции могут быть вызваны из SQL выражений, PL/SQL кода или других функций и процедур.

1. **Что такое параметрическая, позиционная и смешанная формы передачи параметров?**

* **Позиционная** передача параметров означает, что значения аргументов передаются подпрограмме в том порядке, в котором они определены в объявлении подпрограммы.
* **Параметрическая** (или именованная) передача параметров позволяет явно указывать имена параметров при вызове подпрограммы.
* **Смешанная** передача параметров сочетает в себе позиционную и параметрическую формы. В этом случае сначала передаются позиционные параметры, а затем параметрические. Однако при использовании смешанной передачи следует помнить, что все позиционные параметры должны быть указаны перед параметрическими.

1. **Что такое пакет?**

Пакет (package) в PL/SQL представляет собой контейнер, который группирует логически связанные типы данных, подпрограммы (процедуры и функции), переменные и другие элементы PL/SQL.

Спецификация пакета (package specification): Это публичный интерфейс пакета, который определяет, какие элементы пакета доступны для внешнего использования. В спецификации объявляются типы данных, константы, переменные, исключения, курсоры и подпрограммы (процедуры и функции).

Тело пакета (package body): Это реализация пакета, где определяются тела процедур и функций, а также содержится любая другая логика, скрытая от внешнего использования. Тело пакета должно соответствовать спецификации.

1. **Что может входить в пакет?**

* Процедуры (Procedures)
* Функции (Functions
* Переменные (Variables)
* Константы (Constants)
* Пользовательские типы данных (User-defined types)
* Курсоры (Cursors)
* Пользовательские исключения (User-defined exceptions)
* Псевдонимы для типов данных (Synonyms for data types)

1. **Что значит «пакет сохраняет состояние переменных, объявленных в спецификации, на время сессии»?**

Это означает, что переменные, объявленные в спецификации пакета, являются сессионными и сохраняют свои значения на протяжении всей сессии пользователя в базе данных.

* Переменные, объявленные в спецификации пакета, существуют и сохраняют свои значения на протяжении всей сессии пользователя. Это означает, что их значение может изменяться и сохраняться между вызовами подпрограмм пакета в пределах одной сессии.
* Сессия начинается, когда пользователь подключается к базе данных, и заканчивается, когда пользователь отключается. Переменные пакета сохраняют свое состояние в течение всего этого времени.

**Лаба 14**

1. **Перечислите типы триггеров, поддерживаемых Oracle.**

По привязанному объекту:

На таблице

* На представлении - instead of trigger

По событиям запуска:

* Вставка записей - insert
* Обновление записей - update
* Удаление записей – delete

По области действия:

* Уровень оператора - statement level triggers
* Уровень записи - row level triggers
* Составные триггеры - compound triggers

По времени срабатывания:

* Перед выполнением операции – before
* После выполнения операции – after

1. **Поясните правило: триггер является частью транзакции.**

Все действия, выполняемые в триггере, происходят в контексте той же транзакции, что и вызывающее его событие (например, вставка, обновление или удаление).

Если транзакция отменяется (ROLLBACK), все изменения, сделанные триггером, также откатываются.

1. **Перечислите привилегии необходимые для работы с триггерами.**

* CREATE TRIGGER - создавать, удалять, изменять в своей подсхеме
* CREATE ANY TRIGGER - создать любой триггер в любой схеме, кроме SYS, не рекомендуется для словаря, не разрешает менять текст триггера
* ALTER ANY TRIGGER - разрешать, запрещать, изменять, компилировать, любые, кроме SYS-триггеров, триггеры
* DROP ANY TRIGGER - удалять любой триггер, кроме SYS-триггеров
* ADMINISTER DATABASE TRIGGER - создавать, изменять, удалять системные триггеры, должен иметь привилегию CREATE TRIGGER или CREATE ANY TRIGGER

1. **Перечислите события, на которые могут срабатывать DML-триггеры.**

DML-триггеры (триггеры на языке управления данными) в Oracle могут срабатывать на различные события, связанные с операциями вставки (INSERT), обновления (UPDATE) или удаления (DELETE) данных в таблицах базы данных.

* BEFORE INSERT: Триггер срабатывает перед вставкой новой строки в таблицу.
* AFTER INSERT : Триггер срабатывает после вставки новой строки в таблицу.
* BEFORE UPDATE : Триггер срабатывает перед обновлением существующей строки в таблице.
* AFTER UPDATE : Триггер срабатывает после обновления существующей строки в таблице.
* BEFORE DELETE : Триггер срабатывает перед удалением строки из таблицы.
* AFTER DELETE :Триггер срабатывает после удаления строки из таблицы.

1. **Поясните принцип применения INSTEADOF-триггеров в Oracle.**

INSTEAD OF-триггеры в Oracle представляют собой специальный тип триггеров, который используется в контексте операций DML на представлениях (views). Они предлагают альтернативный способ обработки операций INSERT, UPDATE и DELETE для представлений, что дает большую гибкость и контроль над манипуляциями данными через представления. При использовании INSTEAD OF-триггеров, Oracle не выполняет стандартные DML операции, вместо этого передает управление указанному триггеру.

**Лаба 15**

1. **Что такое секционирование таблиц?**

Секционирование таблиц - это процесс разбиения одной таблицы на более мелкие подмножества данных, называемые разделами или разделами таблицы. Эти разделы могут быть физически распределены по разным хранилищам данных или файлам, что помогает управлять и улучшать производительность работы с данными, особенно в случае больших объемов данных.

1. **В каких случаях целесообразно применять секционирование?**

* При управлении большими объемами данных.
* Для параллельной обработки данных и улучшения производительности запросов.
* Для оптимизации операций загрузки и выгрузки данных.
* Для разделения доступа к данным и улучшения безопасности.
* Для улучшения управляемости и обработки часто обновляемых данных.

Это достигается за счет того, что поиск и модификация строк в таблице идут не по всей таблице, а только в ее части (в одной или нескольких секциях). Кроме того, разбиение таблицы на секции позволяет увеличит скорость обработки таблицы за счет использования параллелизма.\

1. **Объясните принцип секционирования для всех типов секционирования, которые использовались в заданиях лабораторной работы.**

* **Диапазонное секционирование:** При диапазонном секционировании таблица разбивается на секции на основе диапазона значений ключевого столбца. Каждая секция отвечает за определенный диапазон значений этого ключевого столбца.
* **Интервальное секционирование**: При интервальном секционировании система автоматически создает новые секции по мере необходимости в зависимости от значений ключевого столбца. Это позволяет таблице динамически расширяться и автоматически создавать новые секции для данных, которые не подходят ни в одну из существующих секций.
* **Хэш-секционирование**: При хэш-секционировании строки таблицы распределяются по секциям на основе хэш-функции, применяемой к значению ключевого столбца. Хэш-секционирование равномерно распределяет данные по секциям, что может быть полезно для балансировки нагрузки и распределения данных.
* **Списочное секционирование**: При списочном секционировании каждая секция соответствует определенному списку значений ключевого столбца. Это позволяет явно указать, какие данные должны быть помещены в каждую секцию.

1. **Перечислите названия типов секционирования, которые не использовались в заданиях лабораторной работы.**

* Секционирование по виртуальному столбцу: Этот тип секционирования использует виртуальные столбцы, которые вычисляются на основе значений других столбцов. Секционирование по виртуальному столбцу позволяет определять секции с использованием значений, вычисленных на лету из других столбцов.
* Ссылочное секционирование: Этот тип секционирования используется в связи с использованием внешних ключей для секционирования родительской таблицы и связанных дочерних таблиц.
* Композитное секционирование : Этот тип секционирования используется для комбинации двух или более методов секционирования, таких как диапазонное и хэш-секционирование.

1. **Объясните действие оператора ALTER TABLE MERGE.**

Оператор ALTER TABLE MERGE используется для объединения смежных секций в одну большую секцию в секционированной таблице.

Оператор ALTER TABLE MERGE выбирает две смежные секции в таблице и объединяет их в одну. Все данные из двух объединяемых секций переносятся в одну новую секцию, а затем оригинальные секции удаляются. После объединения таблица будет иметь на одну секцию меньше, что может улучшить производительность выполнения запросов, так как база данных будет обрабатывать меньше секций при выполнении операций.

1. **Объясните действие оператора ALTER TABLE SPLIT.**

Оператор ALTER TABLE SPLIT используется для разделения одной секции таблицы на две или более новых секции.

Оператор ALTER TABLE SPLIT выбирает существующую секцию в таблице для разделения на две или более новых секции. Для каждого нового раздела определяется граница значений ключевого столбца, на основе которой произойдет разделение данных из исходной секции. Данные из исходной секции перераспределяются между новыми секциями в соответствии с их границами значений ключевого столбца. После выполнения оператора таблица будет содержать дополнительные секции, и данные будут распределены между ними в соответствии с заданными границами значений ключевого столбца.

1. **Объясните действие оператора ALTER TABLE EXCHANGE.**

Оператор ALTER TABLE EXCHANGE используется для обмена содержимым секции с содержимым некоторой внешней таблицы. Это позволяет перемещать данные между таблицами, сохраняя структуру и связи секций в секционированных таблицах.

Оператор ALTER TABLE EXCHANGE выбирает одну секцию из секционированной таблицы и одну внешнюю таблицу для обмена данными. Данные из выбранной секции перемещаются во внешнюю таблицу, а данные из этой внешней таблицы перемещаются в выбранную секцию. Важно, чтобы структура обоих таблиц (типы столбцов, индексы, ограничения) была совместима перед выполнением оператора ALTER TABLE EXCHANGE.