|  |
| --- |
|  |

1. Получите список всех существующих PDB в рамках экземпляра ORA12W. Определите их текущее состояние.

|  |
| --- |
| select name, open\_mode, con\_id from v$pdbs;  select pdb\_name, pdb\_id, status from SYS.dba\_pdbs; |

1. Создайте собственный экземпляр PDB (необходимо подключиться к серверу с серверного компьютера и используйте Database Configuration Assistant) с именем XXX\_PDB, где XXX – инициалы студента.

|  |
| --- |
|  |

1. Получите список всех существующих PDB в рамках экземпляра ORA12W. Убедитесь, что созданная PDB-база данных существует.

|  |
| --- |
| 1. select name,open\_mode,con\_id from v$pdbs; |

1. Подключитесь к XXX\_PDB c помощью SQL Developer создайте инфраструктурные объекты (табличные пространства, роль, профиль безопасности, пользователя с именем U1\_XXX\_PDB).

|  |
| --- |
| ALTER SESSION SET "\_ORACLE\_SCRIPT"=true; ALTER SESSION SET "\_ORACLE\_SCRIPT"=false;  CREATE TABLESPACE TS\_DDD  DATAFILE 'TS\_DDD\_pdb\_2.dbf'  SIZE 7 m  AUTOEXTEND ON NEXT 5M  MAXSIZE 20M  EXTENT MANAGEMENT LOCAL;  CREATE TEMPORARY TABLESPACE TS\_DDD\_TEMP  TEMPFILE 'TS\_DDD\_temp\_pdb\_2.dbf'  SIZE 5 m  AUTOEXTEND ON NEXT 3M  MAXSIZE 30M  EXTENT MANAGEMENT LOCAL;  CREATE ROLE RL\_DDDCORE;  drop role RL\_DDDCORE;  grant create session,  create table,  create view,  create procedure,  drop any table,  drop any view,  drop any procedure to RL\_DDDCORE; commit;  grant create session to RL\_DDDCORE; select \* from dba\_sys\_privs where grantee = 'RL\_DDDCORE';  CREATE PROFILE PF\_DDDCORE LIMIT  PASSWORD\_LIFE\_TIME 180  SESSIONS\_PER\_USER 3  FAILED\_LOGIN\_ATTEMPTS 7  PASSWORD\_LOCK\_TIME 1  PASSWORD\_REUSE\_TIME 10  PASSWORD\_GRACE\_TIME DEFAULT  CONNECT\_TIME 180  IDLE\_TIME 30;  select \*from v$pdbs;   create user U1\_DDD\_PDB identified by 12345 default tablespace TS\_DDD quota unlimited on TS\_DDD temporary tablespace TS\_DDD\_TEMP account unlock;   grant create session to U1\_DDD\_PDB; grant create table to U1\_DDD\_PDB; |

1. Подключитесь к пользователю U1\_XXX\_PDB, с помощью SQL Developer, создайте таблицу XXX\_table, добавьте в нее строки, выполните SELECT-запрос к таблице.

|  |
| --- |
| 1. create table DDD\_table ( x int , y varchar(5));  insert into DDD\_table values (1, 'frst'); insert into DDD\_table values (3, 'thrd');  select \*from DDD\_table; |

1. С помощью представлений словаря базы данных определите, все табличные пространства, все файлы (перманентные и временные), все роли (и выданные им привилегии), профили безопасности, всех пользователей базы данных XXX\_PDB и назначенные им роли.

|  |
| --- |
| 1. select \* from DBA\_TABLESPACES; select \* from DBA\_ROLES; select \* from DBA\_PROFILES; select \* from ALL\_USERS;  select \* from DBA\_DATA\_FILES; select \* from DBA\_TEMP\_FILES; select GRANTEE, PRIVILEGE from DBA\_SYS\_PRIVS; |

1. Подключитесь к CDB-базе данных, создайте общего пользователя с именем C##XXX, назначьте ему привилегию, позволяющую подключится к базе данных XXX\_PDB.
2. Выдайте пользователю C##XXX привилегию на создание таблицы

|  |
| --- |
| 1. GRANT CREATE TABLE TO C##DDD; |

1. Создайте 2 подключения пользователя C##XXX в SQL Developer к CDB-базе данных и XXX\_PDB – базе данных. Проверьте их работоспособность.
2. Подключитесь пользователем C##XXX, создайте таблицу и добавьте в нее данные.

|  |
| --- |
| 1. create table DDD1\_table ( x int , y varchar(5));  insert into DDD1\_table values (1, 'frst'); insert into DDD1\_table values (3, 'thrd');  select \* from DDD1\_table; |

1. Просмотрите все объекты, доступные пользователям C##XXX и U1\_XXX\_PD.

|  |  |
| --- | --- |
| |  | | --- | | 1. select \* from \_TABLESPACES; select \* from \_ROLES; select \* from \_PROFILES; select \* from \_USERS;  select \* from \_DATA\_FILES; select \* from \_TEMP\_FILES; select GRANTEE, PRIVILEGE from \_SYS\_PRIVS; | |

1. Назначьте привилегию, разрешающему подключение к XXX\_PDB общему пользователю C##YYY, созданному другим студентом. Убедитесь в работоспособности этого пользователя в базе данных XXX\_PDB.

|  |
| --- |
| 1. create user c##ddd1 identified by 12345;  grant create session to c##ddd1;  select \*from v$session; |

1. Подключитесь к пользователю U1\_XXX\_PDB со своего компьютера, а к пользователям C##XXX и C##YYY с другого (к XXX\_PDB-базе данных). На своем компьютере получите список всех текущих подключений к XXX\_PDB (найдите в списке созданные вами подключения). На своем компьютере получите список всех текущих подключений к СDB (найдите в списке созданные вами подключений).

|  |
| --- |
|  |

1. Удалите созданную базу данных XXX\_PDB. Убедитесь, что все файлы PDB-базы данных удалены. Удалите пользователя C##XXX. Удалите в SQL Developer все подключения к XXX\_PDB.

|  |
| --- |
|  |

**Ответьте на следующие вопросы**

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

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

Создается при генерации базы данных.

Обновляется и обслуживается сервером Oracle в фоновом режиме после выполнения операторов DDL.

Позволяет запрашивать данные в виде представлений.

Содержит следующую информацию:

- Имена пользователей сервера Oracle

- Уровни привилегий пользователей

- Имена объектов базы данных

- Табличные ограничения

- Учетные данные

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

Прочие представления (DICTIONARY и пр.)

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

«Клиент — сервер» — вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между поставщиками услуг, называемыми серверами, и заказчиками услуг, называемыми клиентами.

До появления технологии клиент/сервер большинство приложений Oracle функционировало на одной ЭВМ. Одна система отвечала не только за всю обработку базы данных, но и за выполнение логики приложения. Кроме того, та же система обрабатывала весь обмен с каждым терминалом; все нажатия клавиш и элементы отображения обслуживались тем же процессором, который обрабатывал запросы к базе данных и логику приложения.

Системы клиент/сервер значительно изменили эту архитектуру, переместив все интерфейсные функции и часть обработки приложения с основного процессора системы на процессор клиента.

1. Что такое экземпляр (инстанс) сервера СУБД?

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.
2. **STARTUP NOMOUNT**: Запуск экземпляра Oracle без монтирования базы данных. Доступ пользователей запрещен. Используется для создания БД и пересоздания управляющих файлов;
3. **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;
4. **STARTUP OPEN:** Запустить экземпляр. Монтировать и открыть БД. Могут присоединяться пользователи. Из состояния STARTUP MOUNT переводится ALTER DATABASE OPEN.

**Остановка**

1. **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;
2. **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.
3. **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.
4. **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.
5. Где указывается, что при старте операционной системы будет запущен инстанс Oracle 12c?

**В службах**

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.

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

**CDB** - container DB – контейнер базы данных

**PDB** - pluggable DB – подключаемые базы данных

**Начальная PDB** — это системный шаблон, который CDB может использовать для создания новых PDB. Начальный PDB называется PDB $ SEED. Вы не можете добавлять или изменять объекты в PDB $ SEED.

1. Что такое общий пользователь Oracle 12c Multitenant?

Вам нужна общая учетная запись (общий пользователь) для выполнения административных операций с CDB (база данных контейнеров), таких как изменение состояния подключаемой базы данных, подключение и отключение подключаемых баз данных (PDBs) и т. Д., что не может сделать локальный пользователь.

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

**Табличное пространство** – логическая структура хранения данных, контейнер сегментов.

С одним табличным пространством связаны один или несколько файлов операционной системы, с каждым файлом связано только одно табличное пространство.

**PERMANENT** - хранение постоянных объектов БД.

**TEMPORARY** - хранение временных данных.

**UNDO** - хранение сегментов отката, используется всегда один.

Oracle сначала автоматически создает табличное пространство System, а за ним — Sysaux

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

SYS и SYSTEM.

1. Какие табличные пространства дублируются в каждой PDB?

По умолчанию в каждой PDB (Pluggable Database) в Oracle дублируются табличные пространства SYSTEM и SYSAUX.

Табличное пространство SYSTEM содержит системные объекты базы данных, такие как таблицы, индексы и процедуры, а также объекты, необходимые для функционирования самой базы данных. Это пространство не должно использоваться для хранения пользовательских данных.

Табличное пространство SYSAUX содержит служебные объекты базы данных, такие как дополнительные компоненты базы данных, такие как Oracle Text, Oracle XML DB и Oracle Spatial, а также объекты, необходимые для функционирования этих компонентов.

Помимо этих двух табличных пространств, каждая PDB может содержать собственные пользовательские табличные пространства. Каждое из этих пространств можно дублировать в других PDB или размещать в единой CDB (Common Database), в зависимости от требований приложения и конфигурации базы данных.

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

**Консолидация данных** — это способ получения итоговой информации, при котором данные, расположенные в нескольких различных областях, объединяются в соответствии с выбранной функцией обработки. Для объединения связанной между собой информации, расположенной в разных таблицах.

1. Какие преимущества дает Oracle 12c Multitenant?

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

Словарь разбивается на две части: общую часть и локальную.

Управление многими БД как одной 
Резервирование всех БД как одной; восстановление на уровне PDB 
Copyright © 2013, Oracle ...

Мультиарендность для Провизионирования 
Быстрое клонирование PDB баз данных 
Copyright © 2013, Oracle and/or its affiliate...