**My Lab Journal**

Contents

[Lab #0 – Journal Example 3](#_Toc883108256)

[Lab #1 – Basic Tasks 4](#_Toc1158958598)

[Lab #2 – Database Creation (Command Line) 13](#_Toc849693182)

[Demonstrate 17](#_Toc55156121)

[Lab #3 – Control Files 19](#_Toc599975743)

[Lab #4 – Redo Log Files and Archived Log Files 26](#_Toc209071092)

[Lab #5 – Data Files (Storage) 34](#_Toc1755718050)

[Lab #6 – User Management 36](#_Toc1146977874)

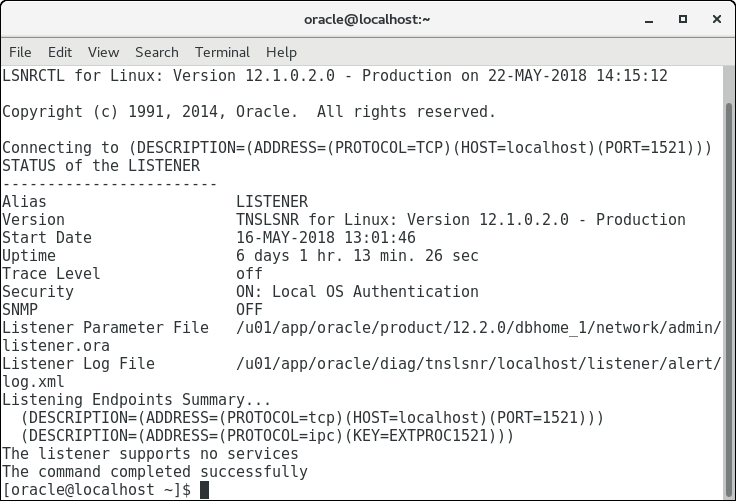
[Lab #7 – Backup and Recovery (Basic Tasks) 37](#_Toc2050158913)

[Lab #8 – Backup and Recovery (Fix) 38](#_Toc1150579537)

# Lab #0 – Journal Example

Start the database in *open* mode:

1. Check to see if databases are currently started on the server
   1. Open a terminal windows
   2. Type: lsnrctl status
   3. No databases currently started



1. Open a terminal window
2. Type: export ORACLE\_SID=FIRST
   1. This identifies the name of the database that will be used in this terminal window for Oracle utilities
3. Go into the SQL\*Plus utility to start the database
   1. Type: sqlplus sys as sysdba
   2. Enter the password when prompted
4. Type: startup open
5. Database is now started and in an open state

# Lab #1 – Basic Tasks

1. Graphical user interface, application

   Description automatically generatedGraphical user interface, application

   Description automatically generatedSETUP a shared folder from the VM to your real machine.

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

2. CONFIGURE the client side connectivity using NET MANAGER so clients can connect  
to the FIRST database.  
• Test the client configuration by connecting to SQL\*Plus using the client connection  
syntax:  
o sqlplus system@FIRST  
o Remember that your database should be fully open when performing a client  
connection.

Graphical user interface, application

Description automatically generated

Graphical user interface, text

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

# Text Description automatically generated

# Graphical user interface, text, application, email Description automatically generated

3. MODIFY the /etc/oratab file to prevent automatic database startup for the FIRST database  
when the Operating System boots up.  
• Remember that we want the flag set to S so the database does not automatically start,  
but it will be shut down in a controlled manner if it is running when the O/S shuts  
down.  
• Use vi to edit this file.

Text, letter

Description automatically generated

Demonstrate  
Demonstrate the following to your instructor for lab sign off:  
1. SHOW your instructor the folder on the VM that connects to your real machine.  
2. SHOW your instructor that a client would be able to connect to the database.  
• This would be done by connecting to SQL\*Plus using the client connection syntax:

sqlplus system@FIRST

3. SHOW your instructor the changes made to the oratab file by opening this file in vi and  
then cleanly exiting the vi utility while your instructor observes.

vi /etc/oratab

4. DISPLAY the status of the listener through the command line.

Lsnrctl status

5. STOP the listener through the command line.

Lsnrctl stop

6. START the listener through the command line.

Lsnrctl start

7. SHUTDOWN the FIRST database through the command line (shut the database down  
using IMMEDIATE). Make sure to export ORACLE\_SID=FIRST

export ORACLE\_SID=FIRST

sqlplus

sys as sysdba

password

shutdown immediate

8. DELETE the FIRST database using the database configuration assistant (DBCA) while  
your instructor observes

# Lab #2 – Database Creation (Command Line)

1. **Shutdown all currently running database instances**
   1. Open Terminal
   2. Check status of instances *“lsnrctl status*”
   3. No currently running instances

Text

Description automatically generated

1. **Locate the zipfile of the MANUAL database. Check the labs folder**  
   Graphical user interface, application, Word

   Description automatically generated
2. Extract the MANUAL folder with the creation scripts
   1. Make sure to use the extract tool
   2. Don’t extract them any deeper than */u01/admin*Chart, waterfall chart

      Description automatically generated
3. Explore the files extracted
   1. The .sql scripts contain all the initial setup commands and the initial data to insert into those. The MANUAL.sh file is responsible for executing all those other scripts in the correct order and making directories for the database.  
        
      Graphical user interface, application, Word

      Description automatically generated  
      MANUAL.sh  
      Text

      Description automatically generated
4. Run the MANUAL.sh file
   1. Open a terminal if one isn’t already open
   2. Run the MANUAL.sh file using the command   
      “. /u01/app/oracle/admin/MANUAL/scripts/MANUAL.sh”
   3. Make sure the period is included and that there is a space between it and the “/uo1/….”
   4. You will be to enter a bunch of passwords before the script runs. Enter “password” for all of these prompts.  
      Table

      Description automatically generated with medium confidence
   5. Confirm the database instance was created and running  
      Text

      Description automatically generated



1. CONFIGURE the new database so users can connect
   1. Open Net Manager from the applications menu
   2. Follow the same setup as before with service name “manual” (See Lab 1 Journal for setup steps). **Make sure to save the configuration in Net Manager!**

Graphical user interface

Description automatically generated

1. Edit the /etc/oratab file to prevent the database from start on boot using command   
   “vi /etc/oratab”. Add the line “MANUAL:/u01/app/oracle/product/12.2.0/dbhome\_1:S”. Hit escape if you’re still stuck in edit mode to exit edit mode. Then enter :wq to save and quit vi.  
   Graphical user interface, text, application

   Description automatically generated

### Demonstrate

1. SHOW the instructor that a client can connect to the database
   1. If not already done, make sure to export the instance with   
      “export ORACLE\_SID=MANUAL”
   2. Make sure the database is open before trying to use a client connection (Check with “lsnrctl status”)
      1. If the database isn’t open then enter the following two command
      2. “sqlplus sys as sysdba” – password = password
      3. “startup open”  
         Text

         Description automatically generated
   3. Connect using a client connection with the command “sqlplus system@MANUAL”  
      Text, letter

      Description automatically generated
2. Demonstrate the shutting down of the MANUAL database through SQL\*Plus
   1. Make sure to exit from the client sql connection as it doesn’t have sufficient permission to shutdown the database.
   2. Connect to the database again using the command “sqlplus sys as sysdba”
   3. Enter “Shutdown immediate”  
      Graphical user interface, text, application, email

      Description automatically generated
3. Demonstrate the starting up of the MANUAL database through SQL\*Plus
   1. Enter “startup open”  
      Text

      Description automatically generated
4. Show your instructor the changes made to the oratab file in vi and then cleanly exiting the vi utility
   1. In a terminal window enter “vi /etc/oratab”
   2. To exit cleanly enter “:” followed by “wq” and enter. The oratab file should look like this  
      Graphical user interface, text, application

      Description automatically generated

# Lab #3 – Control Files

**Step 1**: Shutdown the LABS database

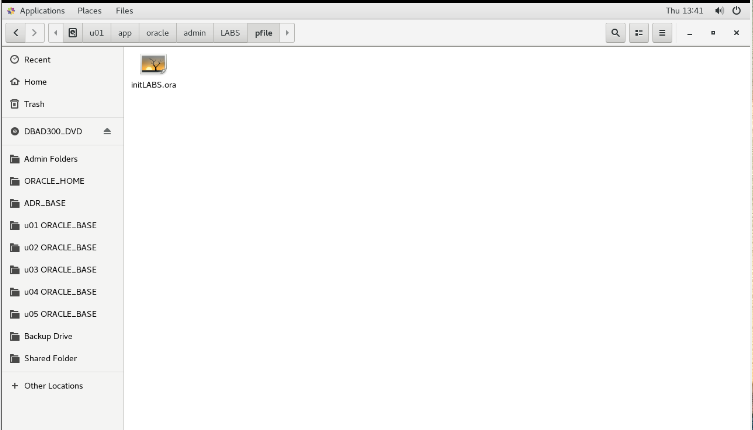
Check lsnrctl status > if none appear open then they are shutdown.

\*\* to double check you can log on using a Server connection (export ORACLE\_SID=LABS; sqlplus sys as sysdba; password; “”check for idle instance””)

**Step 2**: MOVE control file #2 to /u03 (following OFA standards) and make any other changes

that are necessary to have the database look at this new location.

**2a**. Find where the control files are located in the PFILE (Admin Folders > LABS > pfile > initLABS.ora)



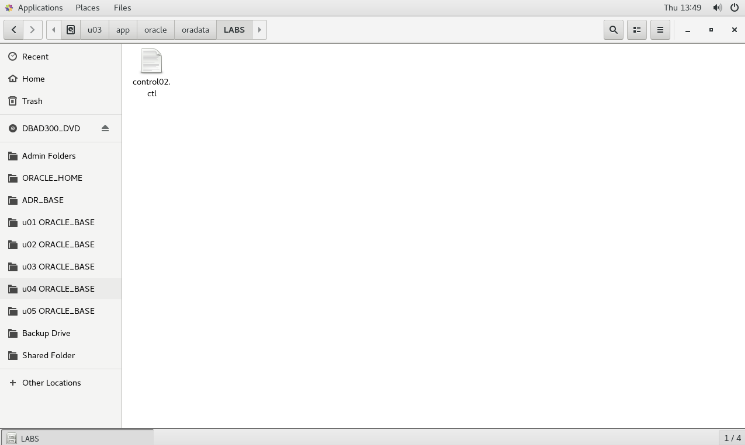
inside the .ora file



**2b**. Go to the folder with the control files; in this case its u02 > oradata > LABS > control02.ctl

**2c**. Cut the file and place in the u03 folder; in this case its u03 > oradata > LABS > control02.ctl

note: you need to create the directory structure in u03 to move the control file into



**2d**. Update the PFILE to show the changes (same pfile from step 2a.)

you will change the 2nd control file to reflect u03 instead of u02 because that’s where the file is now located.



**2e**. Update the SPFILE to show the change

In the terminal, login to the db using a server connection

export ORACLE\_SID=LABS

sqlplus sys as sysdba

password

Once logged in, type following command (in idle mode) ‘:

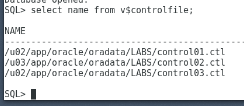
CREATE SPFILE FROM pfile=’/u01/app/oracle/admin/LABS/pfile/initLABS.ora’;

**Step 3**: check that the changes were done properly.

**3a**. Startup the db to check that it worked correctly.

In SQL: startup open

**3b**. Check data dictionary to ensure changes were made to the control file



**Step 4**: Shutdown the db using SHUTDOWN IMMEDIATE

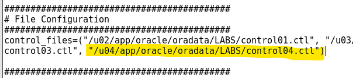
**Step 5**: CREATE a new control file (#4) and place it on /u04 (copy and paste from an existing

control file). Make any other changes that are necessary to have the database start up and

use this new control file.

**5a**. Copy one of the control files from u02 or u03 (see step 2b or c)

**5b**. Open the PFILE and add the path to the new control file



**5c**. Update the SPFILE to show the change

In the terminal, login to the db using a server connection

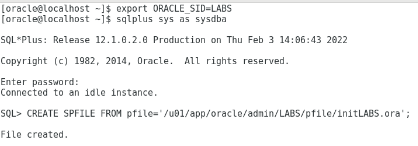
export ORACLE\_SID=LABS

sqlplus sys as sysdba

password

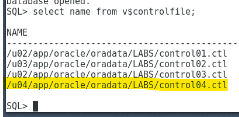
Once logged in, type following command:

CREATE SPFILE FROM pfile=’/u01/app/oracle/admin/LABS/pfile/initLABS.ora’;



NOTE: do NOT open the database, just enter the idle instance

**Step 6**: Startup the database in OPEN mode and check the controlfile was changed



**Step 7**: backup control file to trace

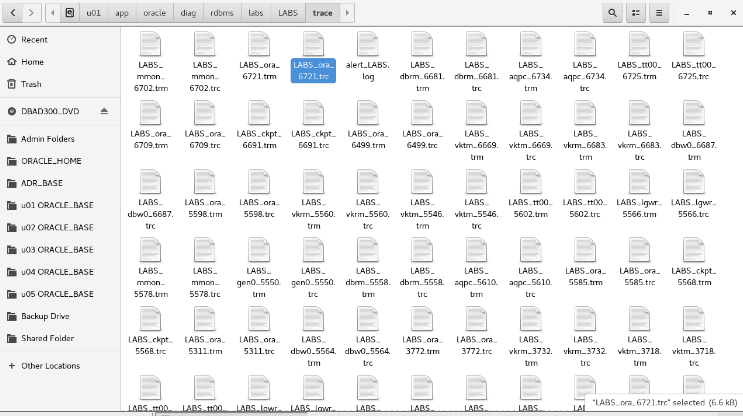
7a. in SQL > ALTER DATABASE BACKUP CONTROLFILE TO TRACE;



**Step 8**: find the trace file BEFORE shutting down the database

adr\_base > rdbms > labs > LABS > trace

note: it will be the last ora file that was modified, the name is also listed in the alert\_LABS.log file at the very end



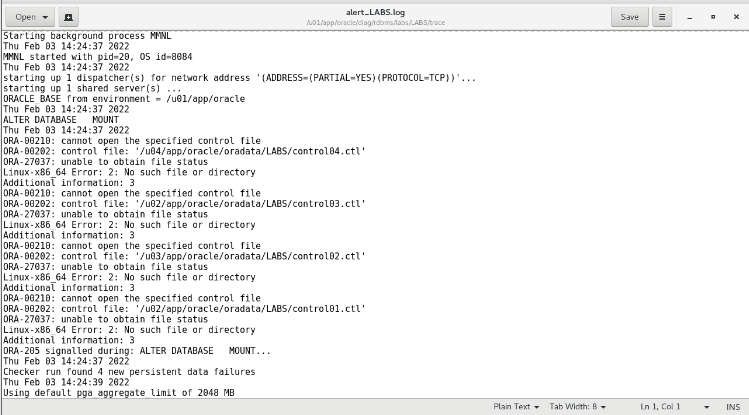
**Step 9**: Shutdown database

**Step 10**: Copy one of the control files to the desktop for ‘safe keeping’

**Step 11**: Delete ALL four of the control files for the data base (the ones in u02, u03 and u04)

**Step 12**: Try and login to the database using a server connection, then startup the database (startup open)

**12a**. Look in the alert\_LABS.log file to see the exact issue



**Step 13**: Shutdown database

**Step 14**: CREATE a script to generate a new control file. Use the trace file created when the

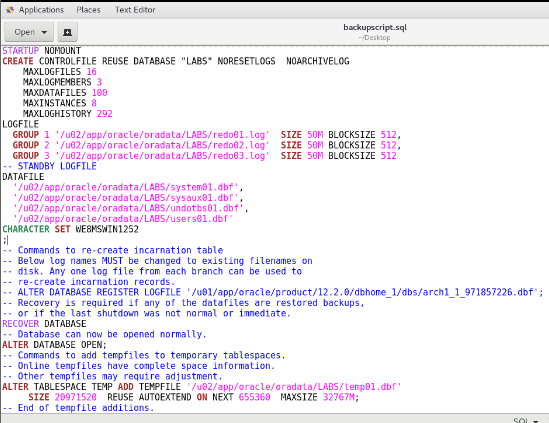
control file was backed up to trace. You will need to clean it up before it can be used as a

script. Use the version that does not reset the redo logs.

**14a**. Open a new gedit file.

**14b**. Go into the trace file from step 8 (LABS\_ora\_6721.trc) and copy the script from Set #1 NORESETLOGS case up until just before Set #2

**14c**. Paste the script into the new gedit file. Safe the file to the desktop as an SQL file.



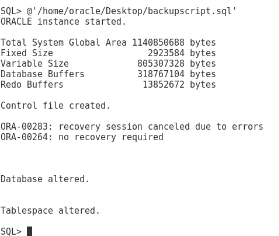
**Step 15**: EXECUTE script

**15a**. Sign back into the database

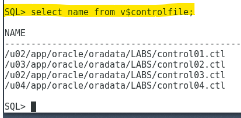
**15b**. Execute the script by typing the following:

SQL > @{file location}

note: you can drag the file into the terminal from desktop to copy the path

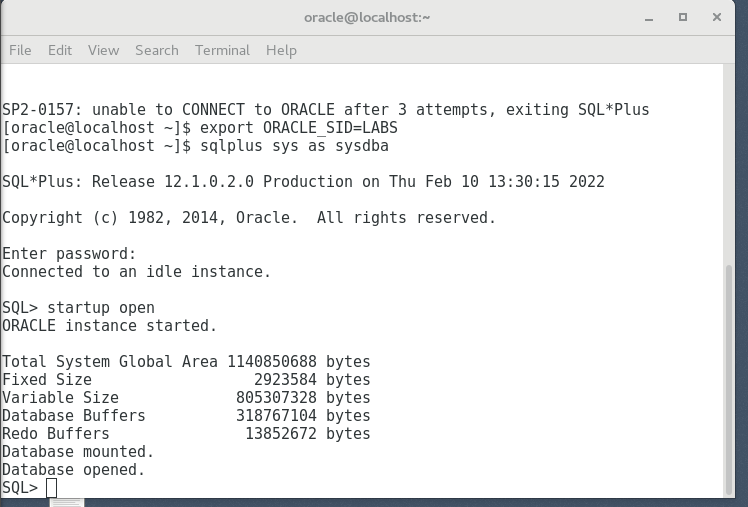


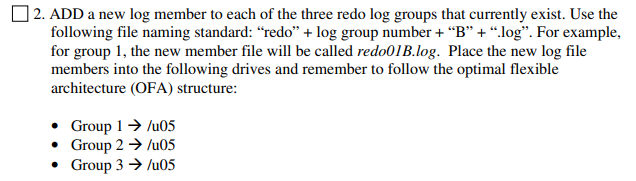
**15c**. check the data dictionary

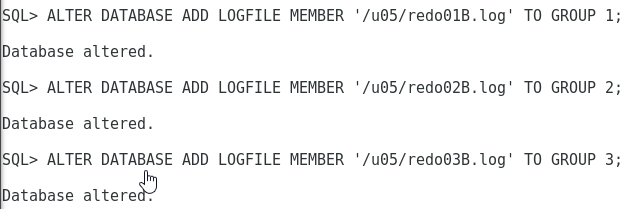


# Lab #4 – Redo Log Files and Archived Log Files





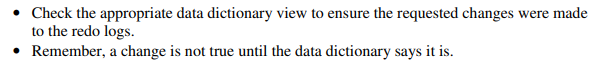


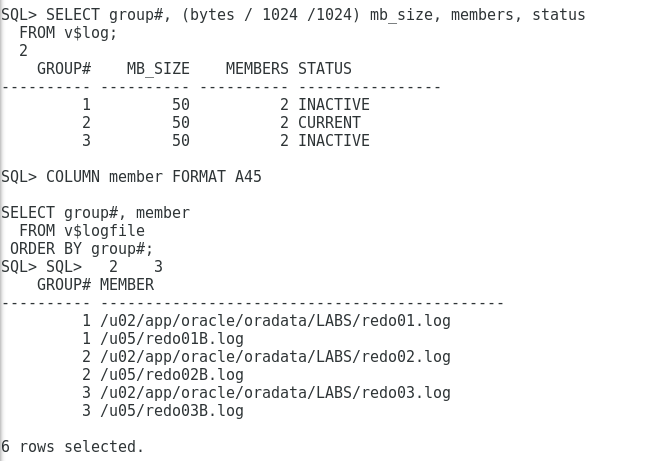


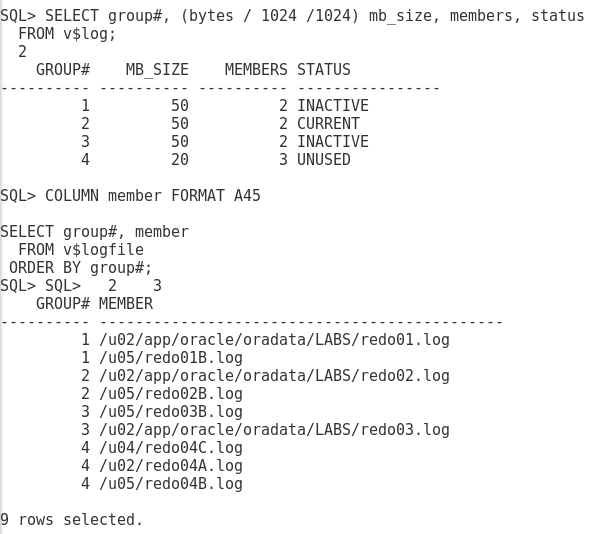
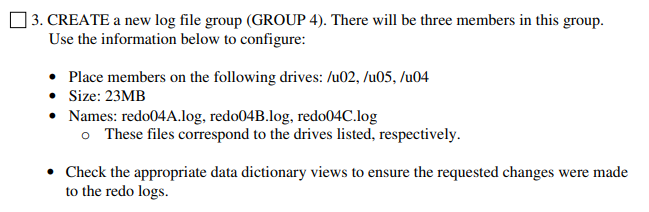
ALTER DATABASE ADD LOGFILE MEMBER '/u05/redo01B.log' TO GROUP 1;

ALTER DATABASE ADD LOGFILE MEMBER '/u05/redo02B.log' TO GROUP 2;

ALTER DATABASE ADD LOGFILE MEMBER '/u05/redo03B.log' TO GROUP 3;







To view redo log group information:

SELECT group#, (bytes / 1024 /1024) mb\_size, members, status

FROM v$log;

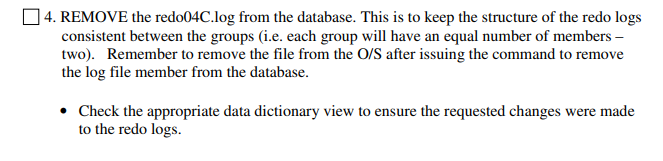
To view redo log group member information:

COLUMN member FORMAT A45

SELECT group#, member

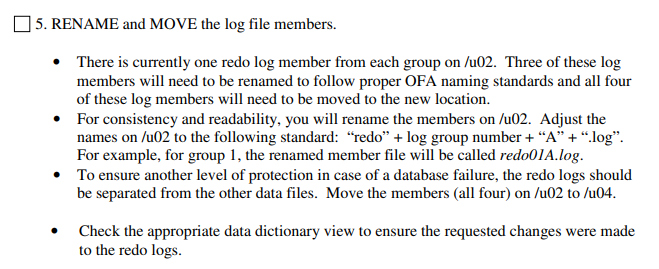
FROM v$logfile

ORDER BY group#;





**Make sure to delete the file on the OS after removing it as a member of the group.**



* Shutdown the database before trying to move/rename any of the files.
* Move the log files to intended directories
* Open the database in mount mode
* Run command to rename/move the file locations
* Startup the database as open to check

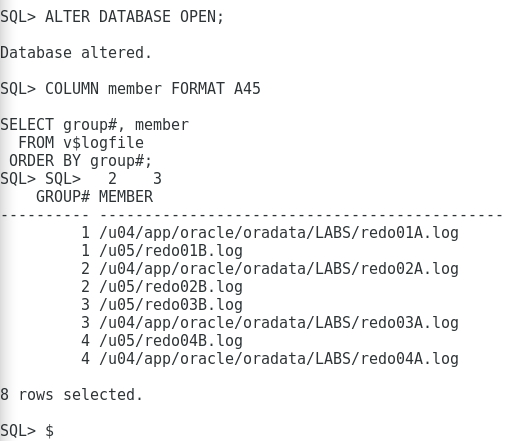


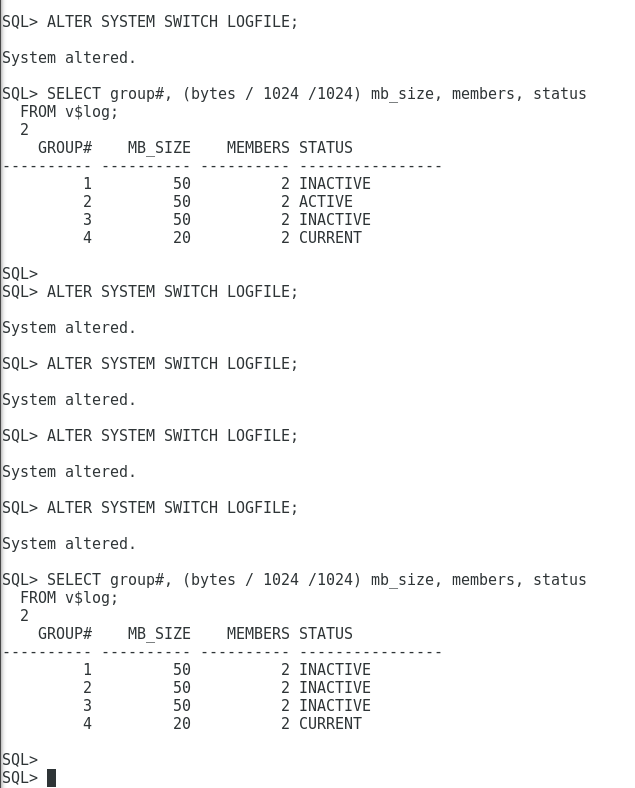
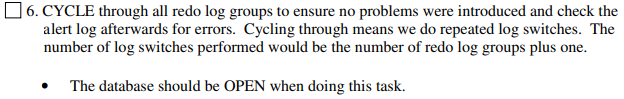
ALTER DATABASE

RENAME FILE '/u02/app/oracle/oradata/LABS/redo01.log', '/u02/app/oracle/oradata/LABS/redo02.log', '/u02/app/oracle/oradata/LABS/redo03.log',

'/u02/redo04A.log'

TO '/u04/app/oracle/oradata/LABS/redo01A.log', '/u04/app/oracle/oradata/LABS/redo02A.log', '/u04/app/oracle/oradata/LABS/redo03A.log', '/u04/app/oracle/oradata/LABS/redo04A.log';

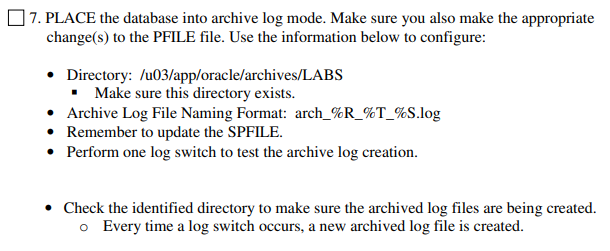


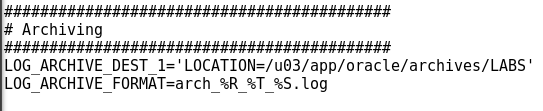


ALTER SYSTEM SWITCH LOGFILE;

SELECT group#, (bytes / 1024 /1024) mb\_size, members, status

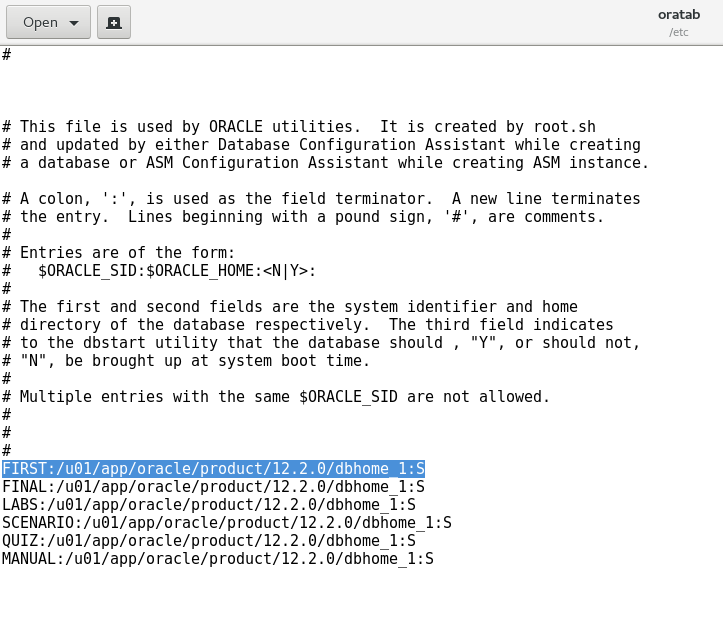
FROM v$log;



* **Edit the pfile and add a section for archiving**
* 

# Clean Database Install

For Clean installing the FIRST Database:

1. Under the disk drive, navigate to Database Clean Installs > FIRST >FIRST Clean.zip. Extract the zip to /u02
   1. NOTE: make sure you don’t have the app folder highlighted, highlight u02 from the root directory to ensure this
2. Make sure /u01/app/oracle/admin/FIRST exists.
3. Move the scripts folder from /u02/app/oracle/oradata/FIRST to /u01/app/oracle/admin/FIRST
4. Make sure /u02/app/oracle/archives/FIRST exists.
5. Execute the /u01/app/oracle/admin/FIRST/scripts/FIRST.sh script.
   1. in terminal, navigate to folder then use command: **sh FIRST.sh**
6. Copy and rename /u01/app/oracle/admin/FIRST/scripts/init.ora to /u01/app/oracle/admin/FIRST/pfile/initFIRST.ora.
7. Update /etc/oratab file to add the FIRST database so it wont autostart on bootup  
   
8. The database should be good to go, you can log in as a sys user to test that it properly opens.

# Lab #5 – Data Files (Storage)

o Look at the data dictionary views to see what tablespaces and/or datafiles already exist and figure out what is changing for those tablespaces that already exist.

Text

Description automatically generated

All of the following commands can be done in OPEN mode EXCEPT renaming/moving datafiles. Renaming/moving datafiles is done is MOUNT mode.

Timeline

Description automatically generated

Text

Description automatically generated

Table

Description automatically generated with low confidence

Graphical user interface, text, email

Description automatically generated

Text, letter

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

# Lab #6 – User Management

Lab Tasks – Implement a user access plan

1. Create Developer user account

CREATE USER MONEY\_DEV

IDENTIFIED BY password

DEFAULT TABLESPACE MONEY\_DATA

TEMPORARY TABLESPACE TEMP

QUOTA UNLIMITED ON MONEY\_DATA

PASSWORD EXPIRE;

1. Create a table under the user account

CREATE TABLE MONEY\_DEV.MONEY\_APP(

ID NUMBER GENERATED BY DEFAULT AS IDENTITY,

PRIMARY KEY(ID)

);

1. Create all roles
   1. role with multiple object privileges

CREATE ROLE MONEY\_FULL\_ACCESS;

GRANT MONEY\_READ\_ONLY to MONEY\_FULL\_ACCESS;

GRANT INSERT, UPDATE, DELETE on MONEY\_DEV.MONEY\_APP to MONEY\_FULL\_ACCESS;

* 1. role with another roles privileges

GRANT HEREIAM\_READ\_ONLY TO HEREIAM\_FULL\_ACCESS;

* 1. role with system privileges

GRANT GRANT ANY ROLE, BACKUP ANY TABLE to MONEY\_ADMIN;

1. Create all profiles

CREATE PROFILE APP\_READ\_USERS LIMIT

SESSIONS\_PER\_USER 1

PASSWORD\_REUSE\_TIME 60

PASSWORD\_LIFE\_TIME 30;

1. Create all users

CREATE USER MONEY\_ADM\_U

IDENTIFIED BY password

DEFAULT TABLESPACE USERS

TEMPORARY TABLESPACE TEMP

QUOTA UNLIMITED ON USERS

PROFILE ADMIN\_USERS

PASSWORD EXPIRE;

Data Dictionary Views

View User Information

COLUMN username FORMAT a20

COLUMN dt FORMAT a15

COLUMN tt FORMAT a10

COLUMN profile FORMAT a20

SELECT username, default\_tablespace dt,

temporary\_tablespace tt, profile

FROM dba\_users

WHERE username = UPPER('&NAME');

View Profile Information

COLUMN profile FORMAT a20

COLUMN resource\_name FORMAT a30

COLUMN limit FORMAT a15

SELECT profile, resource\_name, limit

FROM dba\_profiles

WHERE profile = UPPER('&NAME')

ORDER BY profile, resource\_name;

View Role Information

SELECT role

FROM dba\_roles

WHERE role = UPPER('&NAME')

ORDER BY role;

View Roles granted to roles

COLUMN role FORMAT a25

COLUMN granted\_role FORMAT a25

SELECT role, granted\_role, admin\_option

FROM role\_role\_privs

WHERE role = UPPER('&NAME')

ORDER BY role, granted\_role;

View System privileges granted to roles

COLUMN role FORMAT a20

COLUMN privilege FORMAT a20

SELECT role, privilege, admin\_option

FROM role\_sys\_privs

WHERE role = UPPER('&NAME')

ORDER BY role, privilege;

View Object privileges granted to roles

COLUMN role FORMAT a20

COLUMN owner FORMAT a15

COLUMN table\_name FORMAT a15

COLUMN privilege FORMAT a15

SELECT role, owner, table\_name, privilege, grantable

FROM role\_tab\_privs

WHERE role = UPPER('&NAME')

ORDER BY role, table\_name, privilege;

View roles grants to users

COLUMN grantee FORMAT a15

COLUMN granted\_role FORMAT a20

SELECT grantee, granted\_role, admin\_option

FROM dba\_role\_privs

WHERE grantee = UPPER('&NAME')

ORDER BY grantee, granted\_role;

View table information

SELECT owner, table\_name

FROM dba\_tables

WHERE owner = UPPER('&NAME')

ORDER BY owner, table\_name;

View System privileges granted to users

COLUMN privilege FORMAT a25

SELECT grantee, privilege, admin\_option

FROM dba\_sys\_privs

WHERE grantee = UPPER('&NAME')

ORDER BY grantee, privilege;

View object privileges granted to users

COLUMN grantee FORMAT a10

COLUMN owner FORMAT a10

COLUMN table\_name FORMAT a30

COLUMN privilege FORMAT a20

SELECT grantee , owner, table\_name, privilege, grantable

FROM dba\_tab\_privs

WHERE grantee = UPPER('&NAME')

ORDER BY grantee, table\_name, privilege;

# Lab #7 – Backup and Recovery (Basic Tasks)

**The user creation script used in this lab**



## Creating a cold/offline backup

1. Shutdown the database
2. Copy datafile, control files, and redo log files in the operating system to a safe location (will be in the ORACLE\_BASE folder. (this lab was /u02/app/oracle/oradata)
3. If the database has not been started since the offline backup was performed, can simply copy the missing or damaged datafile directly from backup.

If the database has been started since the offline backup was performed, must restore all datafiles, control files, and redo log files.

## Creating a hot/online backup

1. Database must be in archive log mode.
2. Backup the datafiles for each tablespace issuing the appropriate commands to place the tablespaces into backup mode.
3. For each tablespace being backed up:
   1. ALTER TABLESPACE <name> BEGIN BACKUP;
   2. Copy the datafile for <name> tablespace to a safe location.  
      After the copy is complemented enter:
   3. ALTER TABLESPACE <name> END BACKUP;
   4. Repeat steps A-D for each tablespace

## Using online backups

1. Mount the database
2. Restore the specific datafile from online backups
3. If needed recover the datafile using:  
   RECOVER DATAFILE ‘<path>/<name>’;

## Perform an export

Note: Database must be open

Note: These commands must be launch from command line (not sqlplus).

Note: These commands need a server connection so check that you have used  
export ORACLE\_SID=<database>

Note: The Default directory for the dump files is “{ORACLE\_BASE}/admin/{DB\_NAME}/dpdump”



## Perform an import

Note: These commands must be launch from command line (not sqlplus).

Note: These commands need a server connection so check that you have used  
export ORACLE\_SID=<database>

To import a single table (structure and data):

impdp system DUMPFILE=<file name.dmp> LOGFILE=<file name.log> TABLES=<list separated by commas>

The tables names should be prefixed with the schema (user) that owns them.

To import a single table (data only):

impdp system DUMPFILE=<file name.dmp> LOGFILE=<file name.log> TABLES=<list separated by commas> INCLUDE=table\_data

The tables names should be prefixed with the schema (user) that owns them.

Import appends import data to the end of the existing rows of data.

# Lab #8 – Backup and Recovery (Fix)

## Scenario 1:

Error Message: ORA-00205: error in identifying control file, check alert log for more info

Problem: The error message was stating an issue in locating a control file. Immediately the first things we thought of causing the error was a missing control file or a control file defined in the wrong location in the pfile. Checking the directory “/u02/app/oracle/oradata/SCENARIO/” for control files we noticed that control01.ctl was missing, but was defined in the initSCENARIO.ora Pfile. More details were provided in the alert log “alert\_SCENARIO.log” that pointed to “control01.ctl” not being located.

Graphical user interface, text, email

Description automatically generated

Solution:

1. Shutdown database
2. Copy and paste new control file from an existing one
3. Rename copied control file to the missing control file’s name
4. Make sure pfile is correct
5. then, CREATE SPFILE FROM PFILE=‘pfile path’; (in database closed state)
6. Open database

Text

Description automatically generated with medium confidence

## Scenario 2:

Error Messages:   
ORA-01157: cannot identify/lock data file 1 – see DBWR trace file  
ORA-01110: data file 1: ‘/u02/app/oracle/oradata/SCENARIO/system01.dbf’

Problem: The problem is that a data file called system01.dbf is missing from the path it should be found in as defined in the database.

Table

Description automatically generated with medium confidence

Solution:

1. Startup database in mount mode
2. Copy and paste the backup datafile to its original location defined in the pfile
3. Recover the datafile if needed using RECOVER DATAFILE ‘path’;

Text, letter

Description automatically generated

## Scenario 3:

Error Message: ORA-00205: error in identifying control file, check alert log for more info

Problem: The first problem we discovered is that there are no control files. This was determined by looking in the directory structure where the control file control01.ctr should be located. When we tried to run the trace file script to get a new control file created, we discovered that users01.dbf was also missing.

Table

Description automatically generated

Solution:

1. Made a script from trace file to recover control files
2. Tried to run script but users datafile was missing
3. Performed recovery of users datafile using hot backup
4. Ran the script for recovering control files again
5. Used RECOVER DATAFILE ‘path’;
6. ALTER DATABASE OPEN;

Text

Description automatically generated

## Scenario 4:

Error Messages:  
ORA-10873: file 4 needs to be either taken out of backup mode or media recovered  
ORA-01110: data file 4: ‘path’

Problem: When the database attempted to go from mount to open mode, it ran into the problem that users01.dbf was currently in backup mode and first needed to be taken off that mode.

Table

Description automatically generated with low confidence

Solution:

1. ALTER TABLESPACE USERS END BACKUP.

Text

Description automatically generated

## Scenario 5:

Error Message: ORA-03113: end-of-file on communication channel

Problem: The first problem was identified in the alert log file saying that our three redo log members couldn’t be found on the path provided. Checking the file structure, we can see that the redo log files are all missing (redo01.log, redo02.log, redo03.log).

Graphical user interface, text, application

Description automatically generated

Solution:

1. Recreate 3 redo log files with empty files
2. Put backup archive log files into archive folder
3. STARTUP MOUNT;
4. RECOVER DATABASE UNTIL CANCEL;
5. ALTER DATABASE OPEN RESETLOGS;

## Scenario 6:

Error Message:   
ORA-01157: cannot identify/lock data file 1 – see DBWR trace file  
ORA-01110: data file 1: ‘/u02/app/oracle/oradata/SCENARIO/users01.dbf’

Problem:

Table

Description automatically generated with low confidence

Solution:

1. Copied **all** datafiles from hot backup to database
2. ALTER DATABASE OPEN;
3. RECOVER DATAFILE ‘path’; -> AUTO (for the missing/bad datafile)
4. RECOVER DATABASE UNTIL CANCEL;
5. ALTER DATABASE OPEN RESETLOGS;

Graphical user interface, text, application, email

Description automatically generated