Below are some commands/syntaxes which would be useful in your daily life

**Disconnecting a Session**  
begin

rdsadmin.rdsadmin\_util.disconnect(

sid => sid,

serial => serial\_number);

end;

/

**Killing a Session**  
begin

rdsadmin.rdsadmin\_util.kill(

sid => sid,

serial => serial\_number);

end;

/

**Cancelling a SQL Statement in a Session**  
begin

rdsadmin.rdsadmin\_util.cancel(

sid => sid,

serial => serial\_number,

sql\_id => sql\_id);

end;

/

**Enabling and Disabling Restricted Sessions**

*/\* Verify that the database is currently unrestricted. \*/*

select LOGINS from V$INSTANCE;

LOGINS  
——-

ALLOWED

*/\* Enable restricted sessions \*/*

exec rdsadmin.rdsadmin\_util.restricted\_session(p\_enable => true);

*/\* Verify that the database is now restricted. \*/*

select LOGINS from V$INSTANCE;

LOGINS  
———-

RESTRICTED

*/\* Disable restricted sessions \*/*

exec rdsadmin.rdsadmin\_util.restricted\_session(p\_enable => false);

*/\* Verify that the database is now unrestricted again. \*/*

select LOGINS from V$INSTANCE;

LOGINS  
——-

ALLOWED

**Flushing the Shared Pool**

**In Non RDS :**

Alter system flush shared pool;

**In RDS :**

*exec rdsadmin.rdsadmin\_util.flush\_shared\_pool;*

**Flushing the Buffer Cache**

**In Non RDS :**

Alter system flush buffer cache;

**In RDS :**

*exec rdsadmin.rdsadmin\_util.flush\_buffer\_cache;*

**Privileges**

The following privileges are not available for the DBA role on an Amazon RDS DB instance using the Oracle engine:

* ALTER DATABASE
* ALTER SYSTEM
* CREATE ANY DIRECTORY
* DROP ANY DIRECTORY
* GRANT ANY PRIVILEGE
* GRANT ANY ROLE

When you create a DB instance, the master user account that you use to create the instance gets DBA privileges (with some limitations). Use the master user account for any administrative tasks such as creating additional user accounts in the database. You can’t use the SYS user, SYSTEM user, and other Oracle-supplied administrative accounts.

**Granting SELECT or EXECUTE Privileges to SYS Objects**

The following example grants select privileges on an object named V\_$SESSION to a user named USER1:

begin

rdsadmin.rdsadmin\_util.grant\_sys\_object(

p\_obj\_name => ‘V\_$SESSION‘,

p\_grantee => ‘USER1‘,

p\_privilege => ‘SELECT‘);

end;

/

The following example grants select privileges on an object named V\_$SESSION to a user named USER1 with the grant option :

begin

rdsadmin.rdsadmin\_util.grant\_sys\_object(

p\_obj\_name => ‘V\_$SESSION‘,

p\_grantee => ‘USER1‘,

p\_privilege => ‘SELECT‘,

p\_grant\_option => true);

end;

/

The following example grants the SELECT\_CATALOG\_ROLE and EXECUTE\_CATALOG\_ROLE to USER1. Since the with admin option is used, USER1 can now grant access to SYS objects that have been granted to SELECT\_CATALOG\_ROLE.

*grant SELECT\_CATALOG\_ROLE to USER1 with admin option;*

*grant EXECUTE\_CATALOG\_ROLE to USER1 with admin option;*

**Revoking SELECT or EXECUTE Privileges on SYS Objects**

begin

rdsadmin.rdsadmin\_util.revoke\_sys\_object(

p\_obj\_name => ‘V\_$SESSION‘,

p\_revokee => ‘USER1‘,

p\_privilege => ‘SELECT‘);

end;

/

**Granting Privileges to Non-Master Users**  
*grant SELECT\_CATALOG\_ROLE to user1;*

*grant EXECUTE\_CATALOG\_ROLE to user1;*

Creating of non master users is the same process as we follow in non RDS oracle database.

**The create\_verify\_function Procedure**

The create\_verify\_function procedure is supported for Oracle version 11.2.0.4.v9 and later, Oracle version 12.1.0.2.v5 and later, all 12.2.0.1 versions, all 18.0.0.0 versions, and all 19.0.0 versions.

**Custom Password Function**

You can create a custom function to verify passwords by using the Amazon RDS procedure rdsadmin.rdsadmin\_password\_verify.create\_verify\_function.

begin

rdsadmin.rdsadmin\_password\_verify.create\_verify\_function(

p\_verify\_function\_name => ‘CUSTOM\_PASSWORD\_FUNCTION‘,

p\_min\_length => 10,

p\_min\_uppercase => 1,

p\_min\_digits => 1,

p\_min\_special => 1,

p\_disallow\_at\_sign => true);

end;

/

**Changing the Global Name of a Database**  
**In NON RDS :**

ALTER DATABASE RENAME GLOBAL\_NAME TO database.domain;

**In RDS :**

*exec rdsadmin.rdsadmin\_util.rename\_global\_name(p\_new\_global\_name => ‘new\_global\_name‘);*

**Creating and Sizing Tablespaces**

Amazon RDS only supports Oracle Managed Files (OMF) for data files, log files, and control files. When you create data files and log files, you can’t specify the physical file names.

By default, the tablespace created is a bigfile tablespace.

To create a smallfile tablespace, you need to mention the “smallfile” keyword after create in your syntax.

*create smallfile tablespace users2 datafile size 1G autoextend on maxsize 10G;*

*create temporary tablespace temp01;*

Don’t use smallfile tablespaces because you can’t resize smallfile tablespaces with Amazon RDS for Oracle. However, you can add a datafile to a smallfile tablespace.

*alter tablespace users2 add datafile size 100000M autoextend on next 250m maxsize UNLIMITED;*

**Setting the Default Tablespace**

**In Non RDS :**  
*alter user username default tablespace tablespace\_name;*

**In RDS :**

*exec rdsadmin.rdsadmin\_util.alter\_default\_tablespace(tablespace\_name => ‘users2’);*

**Checkpointing a Database**

**In Non RDS :**

*ALTER SYSTEM CHECKPOINT*

**In RDS :**

*exec rdsadmin.rdsadmin\_util.checkpoint;*

**Creating New Directories in the Main Data Storage Space**

*exec rdsadmin.rdsadmin\_util.create\_directory(p\_directory\_name => ‘directory\_name‘);*

**Listing Files in a DB Instance Directory**

*select \* from table (rdsadmin.rds\_file\_util.listdir(p\_directory => ‘directory\_name‘));*

**Reading Files in a DB Instance Directory**

*select \* from table*  
*(rdsadmin.rds\_file\_util.read\_text\_file(*

*p\_directory => ‘directory\_name‘,*

*p\_filename => ‘file\_name‘*

*)*

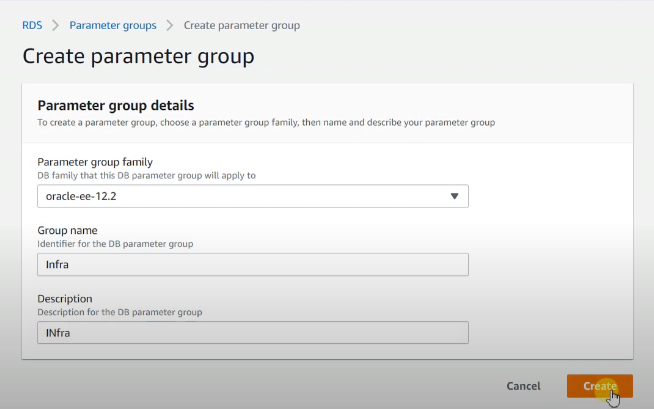
*);*

**Enabling Auditing for the SYS.AUD$ Table**

If your auditing is not set for the database, perform the below actions

1. Click on the Parameter groups from your RDS Dashboard

2. Create a Parameter Group

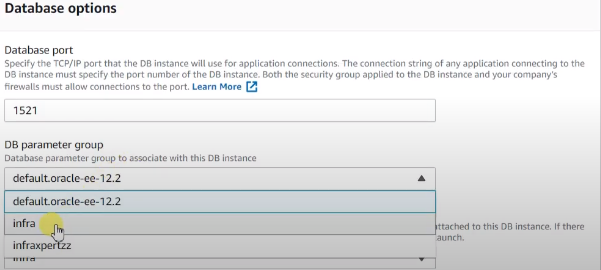


3. Edit the concerned Parameter group and click on **Save Changes** after making the changes as given below :

https://infraxpertzz.com/wp-content/uploads/2020/12/dd-27.png

4. Click on your RDS and click on MODIFY

5. From the Database Options under MODIFY, choose your recently added Parameter Group



6. Apply the changes and BOUNCE the RDS instance, since the parameter is a Static parameter.

7. Run the below commands to enable the changes after RDS restart

*exec rdsadmin.rdsadmin\_master\_util.audit\_all\_sys\_aud\_table;*

*exec rdsadmin.rdsadmin\_master\_util.audit\_all\_sys\_aud\_table(p\_by\_access => true);*

**Disabling Auditing for the SYS.AUD$ Table**

*exec rdsadmin.rdsadmin\_master\_util.noaudit\_all\_sys\_aud\_table;*

**Purging the Recycle Bin**

*exec rdsadmin.rdsadmin\_util.purge\_dba\_recyclebin;*

**Setting Force Logging**

In force logging mode, Oracle logs all changes to the database except changes in temporary tablespaces and temporary segments (NOLOGGING clauses are ignored).

*exec rdsadmin.rdsadmin\_util.force\_logging(p\_enable => true);*

**Setting Supplemental Logging**

Supplemental logging ensures that LogMiner and products that use LogMiner technology have sufficient information to support chained rows and storage arrangements such as cluster tables.

Oracle Database doesn’t enable supplemental logging by default. To enable and disable supplemental logging, use the Amazon RDS procedure rdsadmin.rdsadmin\_util.alter\_supplemental\_logging.

*begin*

*rdsadmin.rdsadmin\_util.alter\_supplemental\_logging(*

*p\_action => ‘ADD‘);*

*end;*

*/*

**Adding Online Redo Logs**  
An Amazon RDS DB instance running Oracle starts with four online redo logs, 128 MB each.

To add additional redo logs, use the Amazon RDS procedure rdsadmin.rdsadmin\_util.add\_logfile.

*exec rdsadmin.rdsadmin\_util.add\_logfile(p\_size => ‘Size in M‘);*

*exec rdsadmin.rdsadmin\_util.drop\_logfile(grp => Number of groups);*

**Dropping Online Redo Logs**

Drop each inactive log using the group number

*exec rdsadmin.rdsadmin\_util.drop\_logfile(grp => 1);*

**Switch Log Files**

*exec rdsadmin.rdsadmin\_util.switch\_logfile;*

**Retaining Archived Redo Logs**

You can retain archived redo logs locally on your DB instance for use with products like Oracle LogMiner (DBMS\_LOGMNR). After you have retained the redo logs, you can use LogMiner to analyze the logs.

*begin*

*rdsadmin.rdsadmin\_util.set\_configuration(*

*name => ‘archivelog retention hours’,*

*value => ‘24‘);*

*end;*

*/*

*commit;*

The following example shows the log retention time.

*set serveroutput on*

*exec rdsadmin.rdsadmin\_util.show\_configuration;*

**View Files present in  BDUMP Directory**

*SELECT \* FROM table(rdsadmin.rds\_file\_util.listdir(‘BDUMP’)) order by mtime;*

**View the contents of a file in the BDUMP directory**

*SELECT text FROM table(rdsadmin.rds\_file\_util.read\_text\_file(‘BDUMP’,’rds-rman-validate-nnn.txt’));*

**Oracle file size limits in AWS RDS**

The maximum file size on Amazon RDS Oracle DB instances is 16 TiB (tebibytes).

**Supported Features of AWS RDS**

Amazon RDS Oracle supports the following Oracle Database features:

* Advanced Compression
* Application Express (APEX)
* Automatic Memory Management
* Automatic Undo Management
* Automatic Workload Repository (AWR)
* Active Data Guard with Maximum Performance in the same AWS Region or across AWS Regions
* Continuous Query Notification (version 12.1.0.2.v7 and later)
* Data Redaction
* Database Change Notification (version 11.2.0.4.v11 and later 11g versions.
* Database In-Memory (version 12.1 and later)
* Distributed Queries and Transactions
* Edition-Based Redefinition
* Enterprise Manager Database Control (11g) and EM Express (12c)
* Fine-Grained Auditing
* Flashback Table, Flashback Query, Flashback Transaction Query
* Import/export (legacy and Data Pump) and SQL\*Loader
* Java Virtual Machine (JVM)
* Materialized Views
* Multimedia
* Network encryption
* Partitioning
* Spatial and Graph
* Streams and Advanced Queuing
* Summary Management – Materialized View Query Rewrite
* Text (File and URL data store types are not supported)
* Total Recall
* Transparent Data Encryption (TDE)
* XML DB (without the XML DB Protocol Server)
* Virtual Private Database

**Unsupported Features of AWS RDS**

Amazon RDS Oracle doesn’t support the following Oracle Database features:

* Automatic Storage Management (ASM)
* Database Vault
* Flashback Database
* Multitenant
* Oracle Enterprise Manager Cloud Control Management Repository
* Real Application Clusters (Oracle RAC)
* Real Application Testing
* Unified Auditing, Pure Mode
* Workspace Manager (WMSYS) schema

**RMAN Tasks in AWS RDS Oracle**

Since AWS RDS is a PaaS, we donot have login to the Database host, rather we can connect to the database and there are some built-in queries which can help make our life easier to perform the Backup Operations in AWS RDS

Below are some of the commands, which can be used to make our life simpler in AWS :

**Validate the DB instance using the default values for the parameters**

*exec rdsadmin.rdsadmin\_rman\_util.validate\_database;*

**Validate the DB instance using the specified values for the parameters.**  
*BEGIN*

*rdsadmin.rdsadmin\_rman\_util.validate\_database(*

*p\_validation\_type => ‘PHYSICAL+LOGICAL‘,*

*p\_parallel => 4,*

*p\_section\_size\_mb => 10,*

*p\_rman\_to\_dbms\_output => FALSE);*

*END;*

*/*

**Enabling and Disabling Block Change Tracking**

To determine whether block change tracking is enabled for your DB instance, run the following query.

*SELECT status, filename FROM V$BLOCK\_CHANGE\_TRACKING;*

Below are the commands to enable/disable block change tracking

**Enable :** *exec rdsadmin.rdsadmin\_rman\_util.enable\_block\_change\_tracking;*  
**Disable :** exec rdsadmin.rdsadmin\_rman\_util.disable\_block\_change\_tracking;

**Crosschecking Archived Redo Logs**

The following example  crosschecks and deletes the expired archived redo logs from the control file.

*BEGIN*

*rdsadmin.rdsadmin\_rman\_util.crosscheck\_archivelog(*

*p\_delete\_expired      => FALSE,*

*p\_rman\_to\_dbms\_output => FALSE);*

*END;*

*/*

**Setting up Archive log retention**

Here we are setting the archive log retention to 24 hours

*begin*

*rdsadmin.rdsadmin\_util.set\_configuration(*

*name  => ‘archivelog retention hours’,*

*value => ‘24‘);*

*end;*

*/*

*commit;*

**Take Archive logs backup**

Here we are backing up archive logs to MYDIRECTORY directory with owner as SYS and parallelism of 4

*BEGIN*

*rdsadmin.rdsadmin\_rman\_util.backup\_archivelog\_all(*

*p\_owner               => ‘SYS‘,*

*p\_directory\_name      => ‘MYDIRECTORY‘,*

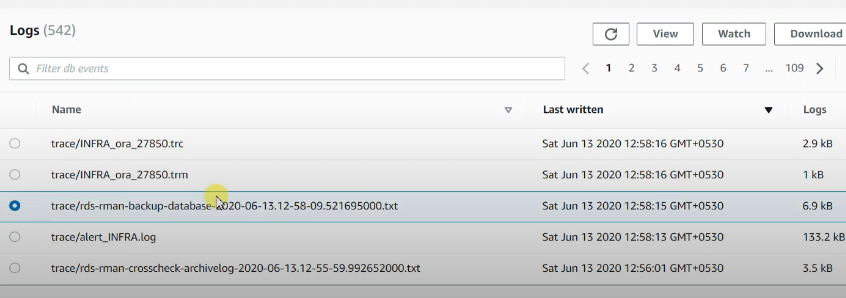
*p\_parallel            => 4,*

*p\_rman\_to\_dbms\_output => FALSE);*

*END;*

*/*

You can also view the logs of archive logs backup from your RDS console, traversing to Logs and then selecting the desired log and by clicking on view :



**Take archive logs backup for a particular date range**

*BEGIN*

*rdsadmin.rdsadmin\_rman\_util.backup\_archivelog\_date(*

*p\_owner               => ‘SYS‘,*

*p\_directory\_name      => ‘MYDIRECTORY‘,*

*p\_from\_date           => ‘17/11/2020 00:00:00‘,*

*p\_to\_date             => ‘17/12/2020 00:00:00‘,*

*p\_parallel            => 4,*

*p\_rman\_to\_dbms\_output => FALSE);*

*END;*

*/*

**Backup Archive logs for a particular SCN Range**

Here we are backing up archive logs from 15617-16617 SCN Rnage

*BEGIN*

*rdsadmin.rdsadmin\_rman\_util.backup\_archivelog\_scn(*

*p\_owner               => ‘SYS‘,*

*p\_directory\_name      => ‘MYDIRECTORY‘,*

*p\_from\_scn            => 15617,*

*p\_to\_scn              => 16617,*

*p\_parallel            => 4,*

*p\_rman\_to\_dbms\_output => FALSE);*

*END;*

*/*

**Backing Up an Archived Redo Log for a Sequence Number Range**

Here we are backing up archive logs for a Sequence ranging between 21216-21316

*BEGIN*

*rdsadmin.rdsadmin\_rman\_util.backup\_archivelog\_sequence(*

*p\_owner               => ‘SYS‘,*

*p\_directory\_name      => ‘MYDIRECTORY‘,*

*p\_from\_sequence       => 21216,*

*p\_to\_sequence         => 21316,*

*p\_parallel            => 4,*

*p\_rman\_to\_dbms\_output => FALSE);*

*END;*

*/*

**Performing a Full Database Backup**

*BEGIN*

*rdsadmin.rdsadmin\_rman\_util.backup\_database\_full(*

*p\_owner               => ‘SYS‘,*

*p\_directory\_name      => ‘MYDIRECTORY‘,*

*p\_parallel            => 4,*

*p\_section\_size\_mb     => 10,*

*p\_rman\_to\_dbms\_output => FALSE);*

*END;*

*/*

**Performing an Incremental Database Backup**

*BEGIN*

*rdsadmin.rdsadmin\_rman\_util.backup\_database\_incremental(*

*p\_owner               => ‘SYS‘,*

*p\_directory\_name      => ‘MYDIRECTORY‘,*

*p\_level               => 1,*

*p\_parallel            => 4,*

*p\_section\_size\_mb     => 10,*

*p\_rman\_to\_dbms\_output => FALSE);*

*END;*

*/*

**Performing a Tablespace Backup**

*BEGIN*

*rdsadmin.rdsadmin\_rman\_util.backup\_tablespace(*

*p\_owner               => ‘SYS‘,*

*p\_directory\_name      => ‘MYDIRECTORY‘,*

*p\_tablespace\_name     => Tablespace\_name,*

*p\_parallel            => 4,*

*p\_section\_size\_mb     => 10,*

*p\_rman\_to\_dbms\_output => FALSE);*

*END;*

*/*