

## Start command

```
$ rman
$ rman NOCATALOG
$ rman TARGET SYS/pwd@target
$ rman TARGET SYS/pwd@target NOCATALOG
$ rman CATALOG rman/pwd@catdb
$ rman TARGET=SYS/pwd@target CATALOG=rman/pwd@cat
$ rman TARGET / CATALOG rman/rman@cat
$ rman | tee rman.log
$ rman help=yes
```

---

## CONNECT command

Establish a connection between RMAN and a target, auxiliary, or recovery catalog database.

```
RMAN> CONNECT TARGET;
RMAN> CONNECT TARGET /
RMAN> CONNECT TARGET sys@tgt;
RMAN> CONNECT TARGET sys/pwd@tgt;
RMAN> CONNECT CATALOG rman@catdb;
RMAN> CONNECT CATALOG rman/pwd@catdb;
RMAN> CONNECT AUXILIARY /
RMAN> CONNECT AUXILIARY rman@auxdb;
RMAN> CONNECT AUXILIARY rman/pwd@auxdb;
```

---

## CREATE CATALOG command

Create Oracle schema for the recovery catalog.

```
RMAN> CREATE CATALOG;
RMAN> CREATE CATALOG TABLESPACE rmants;
RMAN> CREATE VIRTUAL CATALOG;
RMAN> SQL "EXEC
catown.DBMS_RCVCAT.CREATE_VIRTUAL_CATALOG";
```

---

## DROP CATALOG command

Remove Oracle schema from the recovery catalog.

```
RMAN> DROP CATALOG;
```

---

## RESYNC CATALOG command

Perform a full resynchronization, which creates a snapshot control file and then copies any new or changed information from that snapshot control file to the recovery catalog.

```
RMAN> RESYNC CATALOG;
RMAN> RESYNC CATALOG FROM DB_UNIQUE_NAME prod_db;
RMAN> RESYNC CATALOG FROM DB_UNIQUE_NAME ALL;
```

---

## UPGRADE CATALOG command

Upgrade the recovery catalog schema from an older version to

the version required by the RMAN executable.  
RMAN> UPGRADE CATALOG;

---

## IMPORT CATALOG command

Import the metadata from one recovery catalog into another recovery catalog.

```
RMAN> IMPORT CATALOG rcat@srcdb;
RMAN> IMPORT CATALOG rcat@srcdb DBID=<dbid>;
RMAN> IMPORT CATALOG cat@srcdb DBID=<dbid1>, <dbid2>;
RMAN> IMPORT CATALOG cat@srcdb DB_NAME=<dbname>;
RMAN> IMPORT CATALOG cat@srcdb DB_NAME=<dbname1>,
<dbname2>;
RMAN> IMPORT CATALOG cat@srcdb DB_NAME=<dbname> NO
UNREGISTER;
RMAN> IMPORT CATALOG rman/oracle@catdb1 NO UNREGISTER;
```

---

## REGISTER command

Register the target database in the recovery catalog.

```
RMAN> REGISTER DATABASE;
RMAN> REGISTER CATALOG;
RMAN> REGISTER CATALOG TABLESPACE <tbs-name>;
```

---

## UNREGISTER command

Unregister a Oracle database from the recovery catalog.

```
RMAN> UNREGISTER DATABASE;
RMAN> UNREGISTER DATABASE NOPROMPT;
RMAN> UNREGISTER DATABASE prod1;
RMAN> UNREGISTER DATABASE prod2 NOPROMPT;
RMAN> UNREGISTER DB_UNIQUE_NAME prod2;
RMAN> UNREGISTER DB_UNIQUE_NAME prod1 NOPROMPT;
RMAN> UNREGISTER DB_UNIQUE_NAME prod2 INCLUDING
BACKUPS;
RMAN> UNREGISTER DB_UNIQUE_NAME prod3 INCLUDING
BACKUPS NOPROMPT;
```

---

## GRANT command

Grant privileges to a recovery catalog user.

```
RMAN> GRANT CATALOG FOR DATABASE prod1 TO vpc1;
RMAN> GRANT REGISTER DATABASE TO bckop2;
RMAN> GRANT RECOVERY_CATALOG_OWNER TO rmanop1,
rmanop3;
```

---

## REVOKE command

Revoke privileges from a recovery catalog user.

```
RMAN> REVOKE CATALOG FOR DATABASE prod1 FROM vpc1;
RMAN> REVOKE REGISTER DATABASE FROM bckop2;
RMAN> REVOKE RECOVERY_CATALOG_OWNER FROM bckop;
```

---

## RESET DATABASE command

```
RMAN> RESET DATABASE TO INCARNATION 3;
```

---

**STARTUP command**

Startup the target database.

```

RMAN> STARTUP;
RMAN> STARTUP PFILE='/tmp/pfile/init<sid>.ora'
RMAN> STARTUP NOMOUNT;
RMAN> STARTUP MOUNT;
RMAN> STARTUP FORCE;
RMAN> STARTUP FORCE DBA;
RMAN> STARTUP FORCE DBA PFILE='/tmp/pfile/init<sid>.ora'
RMAN> STARTUP FORCE NOMOUNT;
RMAN> STARTUP FORCE MOUNT DBA
PFILE='/tmp/pfile/init<sid>.ora'
RMAN> STARTUP AUXILIARY nomount;
    
```

---

**SHUTDOWN command**

Shutdown the target database.

```

RMAN> SHUTDOWN;
RMAN> SHUTDOWN NORMAL;
RMAN> SHUTDOWN TRANSACTIONAL;
RMAN> SHUTDOWN IMMEDIATE;
RMAN> SHUTDOWN ABORT;
    
```

---

**ALTER DATABASE command**

Mount or open a database.

```

RMAN> ALTER DATABASE MOUNT;
RMAN> ALTER DATABASE OPEN;
RMAN> ALTER DATABASE OPEN RESETLOGS;
    
```

---

**SHOW command**

Display the current CONFIGURE settings.

```

RMAN> SHOW ALL;
RMAN> SHOW RETENTION POLICY;
RMAN> SHOW RETENTION POLICY FOR DB_UNIQUE_NAME ALL;
RMAN> SHOW DEVICE TYPE;
RMAN> SHOW DEVICE TYPE FOR DB_UNIQUE_NAME
<db_unique_name>
RMAN> SHOW DEFAULT DEVICE TYPE;
RMAN> SHOW CHANNEL;
RMAN> SHOW MAXSETSIZE;
RMAN> SHOW BACKUP OPTIMIZATION;
RMAN> SHOW SNAPSHOT CONTROLFILE NAME;
RMAN> SHOW CONTROLFILE AUTOBACKUP;
RMAN> SHOW COMPRESSION ALGORITHM;
RMAN> SHOW ENCRYPTION ALGORITHM;
RMAN> SHOW ALL FOR DB_UNIQUE_NAME ALL;
RMAN> SHOW ALL FOR DB_UNIQUE_NAME 'STANDBY';
    
```

---

**LIST command**

Produce a detailed listing of backup sets or copies.

```

RMAN> LIST INCARNATION;
RMAN> LIST INCARNATION OF DATABASE;
RMAN> LIST INCARNATION OF DATABASE <DB_NAME>;
    
```

```

RMAN> LIST DB_UNIQUE_NAME ALL;
RMAN> LIST DB_UNIQUE_NAME OF DATABASE;
RMAN> LIST BACKUP;
RMAN> LIST BACKUP SUMMARY;
RMAN> LIST BACKUP BY FILE;
RMAN> LIST BACKUP OF DATABASE;
RMAN> LIST BACKUP OF DATABASE BY BACKUP;
RMAN> LIST BACKUP OF TABLESPACE <tablespace_name>
SUMMARY;
RMAN> LIST BACKUP OF DATAFILE <file #>;
RMAN> LIST BACKUP OF DATAFILE <file #> SUMMARY;
RMAN> LIST BACKUP OF CONTROLFILE;
RMAN> LIST BACKUP OF ARCHIVELOG FROM SEQUENCE <seq_no>;
RMAN> LIST BACKUP OF ARCHIVELOG FROM TIME 'sysdate-1';
RMAN> LIST BACKUP OF ARCHIVELOG ALL COMPLETED BEFORE
'sysdate-2';
RMAN> LIST BACKUP RECOVERABLE;
RMAN> LIST EXPIRED BACKUP;
RMAN> LIST EXPIRED BACKUP OF ARCHIVELOG ALL SUMMARY;
RMAN> LIST COPY;
RMAN> LIST COPY OF DATABASE ARCHIVELOG ALL;
RMAN> LIST COPY OF TABLESPACE <tablespace name>;
RMAN> LIST COPY OF DATAFILE <file #>;
RMAN> LIST COPY OF CONTROLFILE;
RMAN> LIST EXPIRED COPY;
RMAN> LIST BACKUPSET SUMMARY;
RMAN> LIST BACKUPSET 109;
RMAN> LIST BACKUPSET OF DATAFILE 1;
RMAN> LIST ARCHIVELOG;
RMAN> LIST ARCHIVELOG ALL LIKE '%5515%';
RMAN> LIST CONTROLFILECOPY "/tmp/cntrlfile.copy";
RMAN> LIST SCRIPT NAMES;
RMAN> LIST ALL SCRIPT NAMES;
RMAN> LIST GLOBAL SCRIPT NAMES;

RMAN> LIST FAILURE;
RMAN> LIST FAILURE 420 DETAIL;
RMAN> LIST FAILURE ALL;
RMAN> LIST RESTORE POINT ALL;
    
```

---

**SQL command**

Execute a SQL statement from within RMAN

```

RMAN> SQL 'ALTER TABLESPACE users ONLINE';
RMAN> SQL 'ALTER DATABASE DATAFILE 64 OFFLINE';
RMAN> SQL "ALTER SYSTEM ARCHIVE LOG CURRENT";
RMAN> SQL "ALTER SYSTEM SWITCH LOGFILE";
RMAN> SQL "ALTER DATABASE BACKUP CONTROLFILE TO TRACE";
    
```

**With 12c**

```

RMAN> SQL 'ALTER TABLESPACE users ONLINE';
RMAN> SQL 'ALTER DATABASE DATAFILE 64 OFFLINE';
RMAN> ALTER SYSTEM ARCHIVE LOG CURRENT;
RMAN> ALTER SYSTEM SWITCH LOGFILE;
    
```

---

**CONFIGURE command**

```

RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP OFF;
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR
DEVICE TYPE DISK TO 'cf%F';
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR
DEVICE TYPE DISK TO '+BACKUP';
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR
DEVICE TYPE DISK CLEAR;
RMAN> CONFIGURE RETENTION POLICY TO REDUNDANCY 3;
RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF
7 DAYS;
RMAN> CONFIGURE RETENTION POLICY CLEAR;
RMAN> CONFIGURE DATAFILE BACKUP COPIES FOR DEVICE TYPE
DISK TO 2;

```

```

RMAN> CONFIGURE ARCHIVELOG BACKUP COPIES FOR DEVICE TYPE
DISK TO 2;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY CLEAR;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO NONE;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO SHIPPED TO
STANDBY;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO SHIPPED TO
ALL STANDBY;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON
STANDBY;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON
ALL STANDBY;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO BACKED UP
2 TIMES TO sbt;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO BACKED UP
3 TIMES TO disk;

```

```

RMAN> CONFIGURE DEFAULT DEVICE TYPE TO sbt_tape;
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO DISK;
RMAN> CONFIGURE DEVICE TYPE sbt PARALLELISM 3;
RMAN> CONFIGURE DEVICE TYPE DISK PARALLELISM 4;
RMAN> CONFIGURE DEVICE TYPE DISK PARALLELISM 3 BACKUP TYPE
TO BACKUPSET;
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt;
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt
PARMS='ENV=mml_env_settings';
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt PARMS
'ENV=(NSR_SERVER=bksrv1)';
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt PARMS
'BLKSIZE=1048576';
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt FORMAT 'bkup_%U';
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt CLEAR;
RMAN> CONFIGURE CHANNEL 2 DEVICE TYPE sbt CONNECT
'SYS/pwd@node2' PARMS 'ENV=(NSR_SERVER=bksrv2)';
RMAN> CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT
'/tmp/%U';
RMAN> CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT
'C:\backup\df%t_s%s_s%p';
RMAN> CONFIGURE CHANNEL 2 DEVICE TYPE DISK FORMAT
'/backup/db_%s%d_%p';
RMAN> CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT CLEAR;

```

```

RMAN> CONFIGURE CHANNEL DEVICE TYPE DISK DEBUG 5;

```

```

RMAN> CONFIGURE BACKUP OPTIMIZATION ON;
RMAN> CONFIGURE BACKUP OPTIMIZATION OFF;
RMAN> CONFIGURE SNAPSHOT CONTROLFILE NAME TO
'/backup/snapcf_%d.f';
RMAN> CONFIGURE SNAPSHOT CONTROLFILE NAME TO
'+FRA/snap/snapcf_%d.f';
RMAN> CONFIGURE SNAPSHOT CONTROLFILE NAME TO
'/ocfs/oradata/snapcf';
RMAN> CONFIGURE SNAPSHOT CONTROLFILE NAME TO '/dev/sda';
RMAN> CONFIGURE MAXSETSIZE TO 100M;
RMAN> CONFIGURE MAXSETSIZE TO UNLIMITED;
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt MAXPIECESIZE 1G;
RMAN> CONFIGURE EXCLUDE FOR TABLESPACE example;
RMAN> CONFIGURE EXCLUDE CLEAR;
RMAN> CONFIGURE AUXNAME FOR DATAFILE 4 TO
'/oracle/auxfiles/aux_4.f';
RMAN> CONFIGURE AUXNAME FOR DATAFILE 2 CLEAR;

```

```

RMAN> CONFIGURE COMPRESSION ALGORITHM 'BZIP2';
RMAN> CONFIGURE COMPRESSION ALGORITHM 'ZLIB';

```

**With 11gR2**

```

RMAN> CONFIGURE COMPRESSION ALGORITHM 'LOW';
RMAN> CONFIGURE COMPRESSION ALGORITHM 'MEDIUM';
RMAN> CONFIGURE COMPRESSION ALGORITHM 'HIGH';
RMAN> CONFIGURE COMPRESSION ALGORITHM 'BASIC';

```

```

RMAN> CONFIGURE DB_UNIQUE_NAME 'standby' CONNECT
IDENTIFIER 'standby_cs';
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO DISK FOR
DB_UNIQUE_NAME 'standby';
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO DISK FOR
DB_UNIQUE_NAME ALL;
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO SBT FOR
DB_UNIQUE_NAME po;

```

**EXECUTE SCRIPT command**

```

Run an RMAN stored script.
RMAN> RUN {EXECUTE SCRIPT backup_whole;}
RMAN> RUN {EXECUTE SCRIPT backup_ts_any USING 'example';}
RMAN> RUN {EXECUTE SCRIPT backup_df USING 3 test_backup df3;}
RMAN> RUN {EXECUTE GLOBAL SCRIPT global_backup_db;}

```

**DELETE SCRIPT command**

```

Delete a stored script from the recovery catalog.
RMAN> DELETE SCRIPT backup_db;
RMAN> DELETE GLOBAL SCRIPT global_backup_db;

```

**SET command**

Set the value of various attributes that affect RMAN behavior for the duration of a RUN block or a session.

```

RMAN> SET ECHO ON;
RMAN> SET ECHO OFF;
RMAN> SET DATABASE <DB_NAME>;
RMAN> SET DBID=<DB_ID>;
RMAN> SET DBID <DB_ID>;
RMAN> SET COMMAND ID TO 'rman';
RMAN> SET MAXCORRUPT FOR DATABASE TO 2;
RMAN> SET MAXCORRUPT FOR DATAFILE 15 TO 200;
RMAN> SET BACKUP COPIES = 2;

RMAN> SET NEWNAME FOR DATABASE TO '/oradata1/%b';
RMAN> SET NEWNAME FOR TABLESPACE users TO '/oradata2/%U';
RMAN> SET NEWNAME FOR DATAFILE 1 to '/oradata/system01.dbf';
RMAN> SET NEWNAME FOR DATAFILE '/disk7/tbs11.f' TO
'/disk9/tbs11.f';
RMAN> SET NEWNAME FOR TEMPFILE 1 TO '/newdisk/dbs/temp1.f';

RMAN> SET CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE
sbt TO 'cf_%F';
RMAN> SET CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE
DISK TO 'cf_%F.bak';
RMAN> SET UNTIL TIME '04-23-2010:23:50:04';
RMAN> SET ARCHIVELOG DESTINATION TO '/oracle/temp_restore';
RMAN> SET COMPRESSION ALGORITHM 'LOW';
RMAN> SET COMPRESSION ALGORITHM 'LOW' OPTIMIZE FOR LOAD
FALSE;
RMAN> SET COMPRESSION ALGORITHM 'MEDIUM';
RMAN> SET COMPRESSION ALGORITHM 'HIGH';

```

**BACKUP command**

Backs up Oracle database files, copies of database files, archived logs, or backup sets.

```

RMAN> BACKUP DATABASE;
RMAN> BACKUP DATABASE TAG='test backup';
RMAN> BACKUP DATABASE COMMENT='full backup';
RMAN> BACKUP TAG 'weekly_full_db_bkup' DATABASE MAXSETSIZE
10M;
RMAN> BACKUP MAXSETSIZE 500M DATABASE PLUS ARCHIVELOG;
RMAN> BACKUP DURATION 00:60 DATABASE;
RMAN> BACKUP DURATION 00:30 MINIMIZE TIME DATABASE;
RMAN> BACKUP DURATION 00:45 MINIMIZE LOAD DATABASE;

RMAN> BACKUP DATABASE PLUS ARCHIVELOG;
RMAN> BACKUP DATABASE KEEP FOREVER;
RMAN> BACKUP DATABASE KEEP UNTIL TIME='SYSDATE+30';
RMAN> BACKUP DATABASE UNTIL 'SYSDATE+365' NOLOGS;
RMAN> BACKUP DATABASE NOEXCLUDE;
RMAN> BACKUP DATABASE NOEXCLUDE KEEP FOREVER TAG='abc';
RMAN> BACKUP DATABASE SKIP READONLY;
RMAN> BACKUP DATABASE SKIP OFFLINE;

```

```

RMAN> BACKUP DATABASE SKIP INACCESSIBLE;
RMAN> BACKUP DATABASE SKIP READONLY SKIP OFFLINE SKIP
INACCESSIBLE;
RMAN> BACKUP DATABASE FORCE; -- backup read only database
also
RMAN> BACKUP DATABASE NOT BACKED UP;
RMAN> BACKUP DATABASE NOT BACKED UP SINCE TIME='SYSDATE-
3';
RMAN> BACKUP NOT BACKED UP SINCE TIME 'SYSDATE-10'
MAXSETSIZE 500M DATABASE PLUS ARCHIVELOG;

RMAN> BACKUP DATABASE COPIES=2;
RMAN> BACKUP DATABASE FORMAT '/disk1/backups/db_%U.bck'
TAG quarterly KEEP UNTIL TIME 'SYSDATE+365' RESTORE POINT
Q1FY12;
RMAN> BACKUP DEVICE TYPE DISK DATABASE;
RMAN> BACKUP DEVICE TYPE sbt DATABASE PLUS ARCHIVELOG;
RMAN> BACKUP DEVICE TYPE sbt DATAFILECOPY FROM TAG 'latest'
FORMAT 'df%f_%d';
RMAN> BACKUP DEVICE TYPE sbt ARCHIVELOG LIKE '/disk%arc%'
DELETE ALL INPUT;
RMAN> BACKUP DEVICE TYPE sbt BACKUPSET COMPLETED BEFORE
'SYSDATE-14'DELETE INPUT;
RMAN> BACKUP CHECK LOGICAL DATABASE;
RMAN> BACKUP VALIDATE CHECK LOGICAL DATABASE;
RMAN> BACKUP VALIDATE DATABASE;
RMAN> BACKUP VALIDATE DATABASE ARCHIVELOG ALL;

RMAN> BACKUP TABLESPACE test;
RMAN> BACKUP TABLESPACE system, users, tools;
RMAN> BACKUP TABLESPACE 4;
RMAN> BACKUP TABLESPACE USERS PLUS ARCHIVELOG;
RMAN> BACKUP TABLESPACE USERS INCLUDE CURRENT
CONTROLFILE;
RMAN> BACKUP TABLESPACE USERS INCLUDE CURRENT
CONTROLFILE PLUS ARCHIVELOG;
RMAN> BACKUP TABLESPACE USERS SECTION SIZE 100M;
RMAN> BACKUP SECTION SIZE 250M TABLESPACE USERS;

RMAN> BACKUP DATAFILE 9;
RMAN> BACKUP DATAFILE '/u01/data/...';
RMAN> BACKUP DATAFILE 1 PLUS ARCHIVELOG;
RMAN> BACKUP KEEP FOREVER FORMAT '?/dbs/%U_longterm.cpy'
TAG longterm_bck DATAFILE 1 DATAFILE 2;
RMAN> BACKUP SECTION SIZE 500M DATAFILE 6;

RMAN> BACKUP ARCHIVELOG ALL;
RMAN> BACKUP ARCHIVELOG ALL DELETE INPUT;
RMAN> BACKUP ARCHIVELOG LIKE '/arch%' DELETE ALL INPUT;
RMAN> BACKUP ARCHIVELOG FROM TIME 'SYSDATE-3';
RMAN> BACKUP ARCHIVELOG FROM SEQUENCE 100;
RMAN> BACKUP ARCHIVELOG FROM SEQUENCE 999 DELETE INPUT;
RMAN> BACKUP ARCHIVELOG FROM SEQUENCE 123 DELETE ALL
INPUT;
RMAN> BACKUP ARCHIVELOG FROM SEQUENCE 21531 UNTIL
SEQUENCE 21590 FORMAT '/tmp/archive_backup.bkp';
RMAN> BACKUP ARCHIVELOG ALL FROM SEQUENCE 1200 DELETE
ALL INPUT;

```

```

RMAN> BACKUP ARCHIVELOG NOT BACKED UP 2 TIMES;
RMAN> BACKUP ARCHIVELOG COMPLETION TIME BETWEEN
'SYSDATE-28' AND 'SYSDATE-7';
RMAN> BACKUP FORMAT='AL_%d/%t/%s/%p' ARCHIVELOG LIKE
'%arc_dest%';

```

```

RMAN> BACKUP CURRENT CONTROLFILE TO '/backup/cntrlfile.copy';
RMAN> BACKUP CONTROLFILE COPY '/u10/backup/control.bkp';
RMAN> BACKUP SPFILE;
RMAN> BACKUP DEVICE TYPE sbt SPFILE ARCHIVELOG ALL;
RMAN> BACKUP DEVICE TYPE sbt DATAFILECOPY ALL
NODUPLICATES;

```

```

RMAN> BACKUP RECOVERY FILES;

```

**With 12c**

```

RMAN>BACKUP PLUGGABLE DATABASE PDB1, PDB2;

```

**BACKUP set**

```

RMAN> BACKUP BACKUPSET ALL;
RMAN> BACKUP BACKUPSET ALL FORMAT =
'/u01/.../backup_%u.bak';
RMAN> BACKUP BACKUPSET COMPLETED BEFORE 'SYSDATE-3'
DELETE INPUT;
RMAN> BACKUP DEVICE TYPE sbt BACKUPSET COMPLETED BEFORE
'SYSDATE-14' DELETE INPUT;
RMAN> BACKUP COPIES 2 DEVICE TYPE sbt BACKUPSET ALL;
RMAN> BACKUP AS COMPRESSED BACKUPSET;
RMAN> BACKUP AS COMPRESSED BACKUPSET DEVICE TYPE DISK
COPIES 2 DATABASE FORMAT '/disk1/db_%U', '/disk2/db_%U';
RMAN> BACKUP AS COMPRESSED BACKUPSET INCREMENTAL FROM
SCN 4111140000000 DATABASE TAG 'RMAN_RECOVERY';
RMAN> BACKUP AS BACKUPSET DATAFILE
'$ORACLE_HOME/oradata/users01.dbf', '$ORACLE_HOME/oradata/t
ools01.dbf';
RMAN> BACKUP AS BACKUPSET DATAFILECOPY ALL;
RMAN> BACKUP AS BACKUPSET DATAFILECOPY ALL NODUPLICATES;

```

**IMAGE copy**

```

RMAN> BACKUP AS COPY DATABASE;
RMAN> BACKUP AS COPY COPY OF DATABASE FROM TAG 'test'
CHECK LOGICAL TAG 'dupstest';
RMAN> BACKUP AS COPY TABLESPACE 8;
RMAN> BACKUP AS COPY TABLESPACE test;
RMAN> BACKUP AS COPY TABLESPACE system, tools, users, undotbs;
RMAN> BACKUP AS COPY DATAFILE 1;
RMAN> BACKUP AS COPY DATAFILE 2 FORMAT '/disk2/df2.cpy' TAG
my_tag;
RMAN> BACKUP AS COPY CURRENT CONTROLFILE;
RMAN> BACKUP AS COPY CURRENT CONTROLFILE FORMAT '/...';
RMAN> BACKUP AS COPY ARCHIVELOG ALL;
RMAN> BACKUP AS COPY KEEP FOREVER NOLOGS CURRENT
CONTROLFILE FORMAT '?/oradata/cf_longterm.cpy', DATAFILE 1
FORMAT '?/oradata/df1_longterm.cpy', DATAFILE 2 FORMAT
'?/oradata/df2_longterm.cpy';
RMAN> BACKUP AS COPY DATAFILECOPY 'bar' FORMAT 'foobar';
RMAN> BACKUP AS COPY DATAFILECOPY '/disk2/df2.cpy' FORMAT
'/disk1/df2.cpy';

```

```

RMAN> BACKUP AS COPY REUSE TARGETFILE
'/u01/oracle/11.2.0.2/dbs/orapwcrd' AUXILIARY FORMAT
'/u01/oracle/11.2.0.2/dbs/orapwcrd';
RMAN> BACKUP AS COPY CURRENT CONTROLFILE FOR STANDBY
AUXILIARY format '+DATA/crd/data1/control01.ctl';

```

**Incremental backups**

```

RMAN> BACKUP INCREMENTAL LEVEL=0 DATABASE;
RMAN> BACKUP INCREMENTAL LEVEL=1 DATABASE;
RMAN> BACKUP INCREMENTAL LEVEL=0 DATABASE PLUS
ARCHIVELOG;
RMAN> BACKUP INCREMENTAL LEVEL 1 CUMULATIVE SKIP
INACCESSIBLE DATABASE;
RMAN> BACKUP INCREMENTAL LEVEL 1 FOR RECOVER OF COPY
WITH TAG 'incr' DATABASE;
RMAN> BACKUP DEVICE TYPE DISK INCREMENTAL LEVEL 1 FOR
RECOVER OF COPY WITH TAG 'oltp' DATABASE;
RMAN> BACKUP DEVICE TYPE DISK INCREMENTAL FROM SCN
351986 DATABASE FORMAT '/tmp/incr_standby_%U';
RMAN> BACKUP INCREMENTAL FROM SCN 629184 DATAFILE 5
FORMAT '/tmp/ForStandby_%U' TAG 'FORSTANDBY';

```

```

RMAN> BACKUP INCREMENTAL LEVEL = --- tablespace/datafile

```

```

RMAN> BACKUP BLOCKS ALL CHECK LOGICAL VALIDATE DATAFILE
1398;

```

**REPORT command**

Report backup status: database, files, and backups. Perform detailed analyses of the content of the recovery catalog.

```

RMAN> REPORT OBSOLETE;
RMAN> REPORT NEED BACKUP;
RMAN> REPORT NEED BACKUP DAYS=5;
RMAN> REPORT NEED BACKUP REDUNDANCY=3;
RMAN> REPORT NEED BACKUP RECOVERY WINDOW OF 7 DAYS;
RMAN> REPORT NEED BACKUP DATABASE;
RMAN> REPORT NEED BACKUP INCREMENTAL 1;
RMAN> REPORT UNRECOVERABLE;
RMAN> REPORT SCHEMA;
RMAN> REPORT SCHEMA AT TIME 'sysdate-20/1440';

```

**CHANGE command**

Update the status of a backup in the RMAN repository. Mark a backup piece, image copy, or archived redo log as having the status UNAVAILABLE or AVAILABLE; remove the repository record for a backup or copy; override the retention policy for a backup or copy; update the recovery catalog with the DB\_UNIQUE\_NAME for the target database.

```

RMAN> CHANGE BACKUPSET 666 KEEP FOREVER;
RMAN> CHANGE BACKUPSET 431 KEEP FOREVER NOLOGS;

```

```

RMAN> CHANGE BACKUPSET 100 UNAVAILABLE;
RMAN> CHANGE BACKUPSET 123 NOKEEP;
RMAN> CHANGE BACKUPSET 121,122,127,203,300 UNCATALOG;
RMAN> CHANGE BACKUP OF DATABASE TAG='abc' UNAVAILABLE;
RMAN> CHANGE BACKUP OF DATABASE DEVICE TYPE DISK
UNAVAILABLE;
RMAN> CHANGE COPY OF DATABASE CONTROLFILE NOKEEP;
RMAN> CHANGE BACKUP OF SPFILE COMPLETED BEFORE 'SYSDATE-
3' UNAVAILABLE;
RMAN> CHANGE BACKUP TAG 'consistent_db_bkup' KEEP FOREVER;
RMAN> CHANGE BACKUP TAG 'consistent_db_bkup' DATABASE
KEEP FOREVER;
RMAN> CHANGE BACKUP TAG 'consistent_db_bkup' KEEP FOREVER
NOLOGS;
RMAN> CHANGE BACKUP TAG 'consistent_db_bkup' NOKEEP;

```

```

RMAN> CHANGE ARCHIVELOG ALL UNCATALOG;
RMAN> CHANGE CONTROLFILECOPY '/tmp/cf.cpy' UNCATALOG;
RMAN> CHANGE FAILURE 5 PRIORITY LOW;
RMAN> CHANGE BACKUP FOR DB_UNIQUE_NAME standby1 RESET
DB_UNIQUE_NAME;
RMAN> CHANGE BACKUP FOR DB_UNIQUE_NAME standby3 RESET
DB_UNIQUE_NAME TO standby2;
RMAN> CHANGE DB_UNIQUE_NAME FROM rdbms4 TO rdbms_dev;

```

#### CROSSCHECK command

Check whether files managed by RMAN, such as archived logs, datafile copies, and backup pieces, still exist on disk or tape.

```

RMAN> CROSSCHECK BACKUP;
RMAN> CROSSCHECK BACKUP TAG='full db';
RMAN> CROSSCHECK BACKUP COMPLETED BETWEEN 'SYSDATE-7'
AND 'SYSDATE-1';
RMAN> CROSSCHECK BACKUP COMPLETED BETWEEN '01-JAN-10'
AND '14-FEB-10';
RMAN> CROSSCHECK BACKUP DEVICE TYPE sbt COMPLETED
BETWEEN '01-AUG-09' AND '31-DEC-09';
RMAN> CROSSCHECK BACKUP DEVICE TYPE DISK COMPLETED
BETWEEN '01-JAN-10' AND '23-MAR-10';

```

```

RMAN> CROSSCHECK BACKUP OF DATABASE;
RMAN> CROSSCHECK BACKUP OF TABLESPACE warehouse;
RMAN> CROSSCHECK BACKUP OF TABLESPACE userd COMPLETED
BEFORE 'SYSDATE-14';
RMAN> CROSSCHECK BACKUP OF TABLESPACES gld, invd;
RMAN> CROSSCHECK BACKUP OF DATAFILE 9;
RMAN> CROSSCHECK BACKUP OF DATAFILE 4 COMPLETED AFTER
'SYSDATE-14';
RMAN> CROSSCHECK BACKUP OF DATAFILE
"?/oradata/dwh/system01.dbf" COMPLETED AFTER 'SYSDATE-14';
RMAN> CROSSCHECK BACKUP OF CONTROLFILE;
RMAN> CROSSCHECK BACKUP OF SPFILE;
RMAN> CROSSCHECK BACKUP OF ARCHIVELOG ALL;
RMAN> CROSSCHECK BACKUP OF ARCHIVELOG ALL SPFILE;

```

```

RMAN> CROSSCHECK COPY;

```

```

RMAN> CROSSCHECK COPY OF DATABASE;
RMAN> CROSSCHECK DATAFILECOPY 113, 114, 115;
RMAN> CROSSCHECK CONTROLFILECOPY '/tmp/control01.ctl';
RMAN> CROSSCHECK ARCHIVELOG ALL;
RMAN> CROSSCHECK BACKUPSET;
RMAN> CROSSCHECK BACKUPSET 1338, 1339, 1340;
RMAN> CROSSCHECK BACKUPPIECE TAG = 'nightly_backup';
RMAN> CROSSCHECK PROXY 789;

```

#### RESTORE command

Restore files from backup sets or from disk copies to the default or a new location.

```

RMAN> RESTORE DATABASE;
RMAN> RESTORE DATABASE VALIDATE;
RMAN> RESTORE DATABASE PREVIEW;
RMAN> RESTORE DATABASE PREVIEW SUMMARY;
RMAN> RESTORE DATABASE SKIP TABLESPACE temp, history;
RMAN> RESTORE DATABASE UNTIL SCN 154876;

```

```

RMAN> RESTORE TABLESPACE users;
RMAN> RESTORE TABLESPACE dwh1, dwh2;
RMAN> RESTORE TABLESPACE tbs1 PREVIEW;
RMAN> RESTORE TABLESPACE users VALIDATE;

```

```

RMAN> RESTORE DATAFILE 45;
RMAN> RESTORE DATAFILE 23 PREVIEW;
RMAN> RESTORE DATAFILE 12 VALIDATE;

```

```

RMAN> RESTORE CONTROLFILE;
RMAN> RESTORE CONTROLFILE FROM AUTOBACKUP;
RMAN> RESTORE CONTROLFILE FROM TAG 'monday_cf_backup';
RMAN> RESTORE CONTROLFILE FROM '/u01/control01.ctl';
RMAN> RESTORE CONTROLFILE VALIDATE;
RMAN> RESTORE CONTROLFILE TO '/tmp/autobkp.dbf' FROM
AUTOBACKUP MAXSEQ 20 MAXDAYS 150;

```

```

RMAN> RESTORE SPFILE;
RMAN> RESTORE SPFILE FROM AUTOBACKUP;
RMAN> RESTORE ARCHIVELOG ALL VALIDATE;
RMAN> RESTORE ARCHIVELOG ALL PREVIEW;
RMAN> RESTORE ARCHIVELOG ALL PREVIEW RECALL;
RMAN> RESTORE ARCHIVELOG ALL DEVICE TYPE sbt PREVIEW;
RMAN> RESTORE ARCHIVELOG LOW LOGSEQ 78311 HIGH LOGSEQ
78340 THREAD 1 ALL;
RMAN> RESTORE ARCHIVELOG FROM LOGSEQ=<seq_no> UNTIL
LOGSEQ=<seq_no>;
RMAN> RESTORE STANDBY CONTROLFILE FROM TAG 'forstandby';
RMAN> RESTORE CLONE CONTROLFILE TO
'+DATA/pcrd/data2/control02.ctl' FROM
'+DATA/pcrd/data1/control01.ctl';

```

Restore the control file, (to all locations specified in the parameter file) then restore the database, using that control file:

```

STARTUP NOMOUNT;
RUN
{

```

```

ALLOCATE CHANNEL c1 DEVICE TYPE sbt;
RESTORE CONTROLFILE;
ALTER DATABASE MOUNT;
RESTORE DATABASE;
}

```

**With 12c**

```

RMAN>RESTORE PLUGGABLE DATABASE PDB1, PDB2;

```

**RECOVER command**

Perform media recovery from RMAN backups and copies. Apply redo log files and incremental backups to datafiles or data blocks restored from backup or datafile copies, to update them to a specified time.

```

RMAN> RECOVER DATABASE;
RMAN> RECOVER DATABASE NOREDO;
RMAN> RECOVER DATABASE SKIP TABLESPACE temp;
RMAN> RECOVER DATABASE SKIP FOREVER TABLESPACE exam;
RMAN> RECOVER DATABASE UNTIL SCN 154876;

```

```

RMAN> RECOVER TABLESPACE users;
RMAN> RECOVER TABLESPACE dwh DELETE ARCHIVELOG MAXSIZE 2M;
RMAN> RECOVER DATAFILE 33;
RMAN> RECOVER DATAFILE 3 BLOCK 116 DATAFILE 4 BLOCK 10;
RMAN> RECOVER DATAFILE 2 BLOCK 204 DATAFILE 9 BLOCK 109 FROM TAG=sundaynight;
RMAN> RECOVER DATAFILECOPY '/disk1/img.df' UNTIL TIME 'SYSDATE-7';
RMAN> RECOVER COPY OF DATABASE WITH TAG 'incr';
RMAN> RECOVER COPY OF DATABASE WITH TAG 'upd' UNTIL TIME 'SYSDATE - 7';
RMAN> RECOVER CORRUPTION LIST;

```

Restore and recover the whole database

```

RMAN> STARTUP FORCE MOUNT;
RMAN> RESTORE DATABASE;
RMAN> RECOVER DATABASE;
RMAN> ALTER DATABASE OPEN;

```

Restore and recover a tablespace

```

RMAN> SQL 'ALTER TABLESPACE users OFFLINE';
RMAN> RESTORE TABLESPACE users;
RMAN> RECOVER TABLESPACE users;
RMAN> SQL 'ALTER TABLESPACE users ONLINE';

```

Restore and recover a datafile

```

RMAN> SQL 'ALTER DATABASE DATAFILE 12 OFFLINE';
RMAN> RESTORE DATAFILE 12;
RMAN> RECOVER DATAFILE 12;
RMAN> SQL 'ALTER DATABASE DATAFILE 12 ONLINE';

```

**DELETE command**

Delete backups and copies, remove references to them from the recovery catalog, and update their control file records to status DELETED.

```

RMAN> DELETE OBSOLETE;
RMAN> DELETE NOPROMPT OBSOLETE;
RMAN> DELETE NOPROMPT OBSOLETE RECOVERY WINDOW OF 7 DAYS;
RMAN> DELETE EXPIRED BACKUP;
RMAN> DELETE EXPIRED BACKUP DEVICE TYPE sbt;
RMAN> DELETE BACKUP OF DATABASE LIKE '/tmp%';
RMAN> DELETE NOPROMPT EXPIRED BACKUP OF TABLESPACE users COMPLETED BEFORE 'SYSDATE-14';
RMAN> DELETE BACKUP OF SPFILE TABLESPACE users DEVICE TYPE SBT;

```

```

RMAN> DELETE ARCHIVELOG ALL;
RMAN> DELETE ARCHIVELOG ALL COMPLETED BEFORE 'sysdate-2';
RMAN> DELETE ARCHIVELOG ALL BACKED UP 2 TIMES TO DEVICE TYPE SBT;
RMAN> DELETE ARCHIVELOG ALL LIKE '%755153075%';
RMAN> DELETE ARCHIVELOG UNTIL SEQUENCE=79228;
RMAN> DELETE FORCE ARCHIVELOG ALL COMPLETED BEFORE 'sysdate-1.5';
RMAN> DELETE FORCE ARCHIVELOG UNTIL SEQUENCE=16364;
RMAN> DELETE NOPROMPT ARCHIVELOG UNTIL SEQUENCE = 7300;
RMAN> DELETE EXPIRED ARCHIVELOG ALL;
RMAN> DELETE NOPROMPT EXPIRED ARCHIVELOG ALL;
RMAN> DELETE BACKUPSET 101, 102, 103;
RMAN> DELETE NOPROMPT BACKUPSET TAG weekly_bkup;
RMAN> DELETE FORCE NOPROMPT BACKUPSET TAG weekly_bkup;

```

```

RMAN> DELETE DATAFILECOPY "+DG_DATA/db/datafile/system.259.699468079";
RMAN> DELETE CONTROLFILECOPY '/tmp/cntrlfile.copy';
RMAN> DELETE BACKUP DEVICE TYPE SBT;
RMAN> DELETE BACKUP DEVICE TYPE DISK;
RMAN> DELETE COPY;
RMAN> DELETE EXPIRED COPY;
RMAN> DELETE COPY TAG 'lastest';

```

**DROP DATABASE command**

Delete the target database from disk and unregisters it.

```

RMAN> DROP DATABASE;
RMAN> DROP DATABASE NOPROMPT;
RMAN> DROP DATABASE INCLUDING BACKUPS;
RMAN> DROP DATABASE INCLUDING BACKUPS NOPROMPT;

```

**DUPLICATE command**

Use backups of the target database to create a duplicate database

that we can use for testing purposes or to create a standby database.

```

RMAN> DUPLICATE TARGET DATABASE;
RMAN> DUPLICATE TARGET DATABASE TO <DB_NAME>;
RMAN> DUPLICATE TARGET DATABASE TO test
PFILE=/u01/apps/db/inittest.ora;
RMAN> DUPLICATE TARGET DATABASE TO <DB_NAME>
NOFILENAMECHECK;
RMAN> DUPLICATE DATABASE 'prod' DBID 139525561 TO 'dupdb'
NOFILENAMECHECK;
RMAN> DUPLICATE DATABASE TO <DB_NAME> NOFILENAMECHECK
BACKUP LOCATION '/apps/oracle/backup';
RMAN> DUPLICATE TARGET DATABASE TO dup FROM ACTIVE
DATABASE NOFILENAMECHECK PASSWORD FILE SPFILE;

```

```

RMAN> DUPLICATE TARGET DATABASE TO dupdb
LOGFILE GROUP 1
('?'/dbs/dupdb_log_1_1.f','?'/dbs/dupdb_log_1_2.f') SIZE 200K,
GROUP 2 ('?'/dbs/dupdb_log_2_1.f','?'/dbs/dupdb_log_2_2.f') SIZE
200K REUSE;
RMAN> DUPLICATE TARGET DATABASE TO dup FOR STANDBY FROM
ACTIVE DATABASE PASSWORD FILE SPFILE
PARAMETER_VALUE_CONVERT '/disk1', '/disk2'
SET DB_FILE_NAME_CONVERT '/disk1', '/disk2'
SET LOG_FILE_NAME_CONVERT '/disk1', '/disk2'
SET SGA_MAX_SIZE 200M SET SGA_TARGET 125M;

```

```

RMAN> DUPLICATE TARGET DATABASE FOR STANDBY
NOFILENAMECHECK;
RMAN> DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE
DATABASE;
RMAN> DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE
DATABASE NOFILENAMECHECK;
RMAN> DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE
DATABASE
SPFILE PARAMETER_VALUE_CONVERT '/stg', '/muc/'
SET "DB_UNIQUE_NAME"="muc"
SET LOG_FILE_NAME_CONVERT '/stg', '/muc/'
SET DB_FILE_NAME_CONVERT '/stg', '/muc/'
DORECOVER;
RMAN> DUPLICATE DATABASE TO newdb
UNTIL RESTORE POINT firstquart12
DB_FILE_NAME_CONVERT='/u01/prod1/dbfiles', '/u01/newdb/dbfi
les'
PFILE = '/u01/newdb/admin/init.ora';

```

**SWITCH command**

Specify that a datafile copy is now the current datafile, i.e. the datafile pointed to by the control file. This command is equivalent to the SQL statement ALTER DATABASE RENAME FILE as it applies to datafiles.

```

RMAN> SWITCH DATABASE TO COPY;
RMAN> SWITCH TABLESPACE users TO COPY;

```

```

RMAN> SWITCH DATAFILE ALL;
RMAN> SWITCH DATAFILE '/disk1/tols.dbf' TO DATAFILECOPY
'/disk2/tols.copy';
RMAN> SWITCH DATAFILE
"+DG_OLD/db/datafile/sysaux.260.699468081" TO COPY;
RMAN> SWITCH TEMPFILE 1;
RMAN> SWITCH TEMPFILE 1 TO '/newdisk/dbs/temp1.f';
RMAN> SWITCH TEMPFILE ALL;
RMAN> SWITCH CLONE DATAFILE ALL;

```

**CATALOG command**

Add information about file copies and user-managed backups to the catalog repository.

```

RMAN> CATALOG DATAFILECOPY '<file location>';
RMAN> CATALOG DATAFILECOPY '<file location>' LEVEL 0;
RMAN> CATALOG CONTROLFILECOPY '/disk3/backup/cf_copy.bkp';
RMAN> CATALOG ARCHIVELOG '<arch location>';
RMAN> CATALOG BACKUPPIECE '<backup piece location>';
RMAN> CATALOG LIKE '/backup';
RMAN> CATALOG START WITH '/fs2/arch';
RMAN> CATALOG START WITH '/disk2/archlog' NOPROMPT;
RMAN> CATALOG START WITH '+FLASH';
RMAN> CATALOG RECOVERY AREA;

```

**ALLOCATE command**

Establish a channel, which is a connection between RMAN and a database instance.

```

RMAN> ALLOCATE CHANNEL c1 DEVICE TYPE sbt_tape;
RMAN> ALLOCATE CHANNEL c1 DEVICE TYPE DISK FORMAT
'C:\ORACLEBKP\DB_U%';
RMAN> ALLOCATE CHANNEL t1 DEVICE TYPE DISK CONNECT
'sys/pwd@bkp1';
RMAN> ALLOCATE CHANNEL c1 DEVICE TYPE sbt_tape PARMS
'ENV=(OB_MEDIA_FAMILY=wholedb_mf)';
RMAN> ALLOCATE CHANNEL t1 DEVICE TYPE sbt_tape PARMS
'ENV=(OB_DEVICE_1=tape1, OB_DEVICE_2=tape3)';
RMAN> ALLOCATE CHANNEL t1 TYPE 'sbt_tape'
PARMS='SBT_LIBRARY=/usr/opencv/netbackup/bin/libbk.so.1';
RMAN> ALLOCATE CHANNEL t1 TYPE 'sbt_tape' SEND
"NB_ORA_CLIENT=CLIENT_MACHINE_NAME";
RMAN> ALLOCATE CHANNEL 'dev1' TYPE 'sbt_tape' PARMS
'ENV=OB2BARTYPE=ORACLE8, OB2APPNAME=ORCL,
OB2BARLIST=MACHINENAME_ORCL_ARCHLOGS)';
RMAN> ALLOCATE CHANNEL y1 TYPE DISK RATE 70M;
RMAN> ALLOCATE AUXILIARY CHANNEL ac1 TYPE DISK;
RMAN> ALLOCATE AUXILIARY CHANNEL ac2 DEVICE TYPE sbt;

```

ALLOCATE CHANNEL FOR MAINTENANCE - allocate a channel in preparation for issuing maintenance commands such as DELETE.

```

RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE DISK;
RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE DISK
FORMAT '/disk2/%U';
RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE DISK

```

```
CONNECT '@test1';
RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE sbt;
RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE sbt
PARMS 'SBT_LIBRARY=/usr/local/oracle/backup/lib/libobk.so,
ENV=(OB_DEVICE_1=tape2)';
```

#### RELEASE CHANNEL command

Release a channel that was allocated with an ALLOCATE CHANNEL or ALLOCATE CHANNEL FOR MAINTENANCE command.

```
RMAN> RELEASE CHANNEL;
RMAN> RELEASE CHANNEL c1;
```

#### BLOCKRECOVER command

Will recover the corrupted blocks.

```
RMAN> BLOCKRECOVER CORRUPTION LIST;
RMAN> BLOCKRECOVER DATAFILE 8 BLOCK 22;
RMAN> BLOCKRECOVER DATAFILE 7 BLOCK 233,235 DATAFILE 4
BLOCK 101;
RMAN> BLOCKRECOVER DATAFILE 2 BLOCK 12,13 DATAFILE 3 BLOCK
5,98,99 DATAFILE 4 BLOCK 19;
RMAN> BLOCKRECOVER DATAFILE 3 BLOCK 2,4,5 TABLESPACE sales
DBA 4194405,4194412 FROM DATAFILECOPY;
RMAN> BLOCKRECOVER TABLESPACE dwh DBA 4194404,4194405
FROM TAG "weekly";
RMAN> BLOCKRECOVER TABLESPACE dwh DBA 94404 RESTORE
UNTIL TIME 'SYSDATE-2';
```

#### ADVISE FAILURE command

Display repair options.

```
RMAN> ADVISE FAILURE;
RMAN> ADVISE FAILURE 555, 242;
RMAN> ADVISE FAILURE ALL;
RMAN> ADVISE FAILURE CRITICAL;
RMAN> ADVISE FAILURE HIGH;
RMAN> ADVISE FAILURE LOW;
RMAN> ADVISE FAILURE HIGH EXCLUDE FAILURE 625;
```

#### REPAIR FAILURE command

Repair one or more failures recorded in the automated diagnostic repository.

```
RMAN> REPAIR FAILURE;
RMAN> REPAIR FAILURE PREVIEW;
RMAN> REPAIR FAILURE NOPROMPT;
RMAN> REPAIR FAILURE USING ADVISE OPTION integer;
```

#### VALIDATE command

Examine a backup set and report whether its data is intact. RMAN scans all of the backup pieces in the specified backup sets and looks at the checksums to verify that the contents can be successfully restored.

```
RMAN> VALIDATE BACKUPSET 218;
RMAN> VALIDATE BACKUPSET 3871, 3890;
RMAN> VALIDATE DATABASE;
RMAN> VALIDATE CHECK LOGICAL DATABASE;
RMAN> VALIDATE SKIP INACCESSIBLE DATABASE;
RMAN> VALIDATE COPY OF DATABASE;
RMAN> VALIDATE TABLESPACE dwh;
RMAN> VALIDATE COPY OF TABLESPACE dwh;
RMAN> VALIDATE DATAFILE 2;
RMAN> VALIDATE DATAFILE 4,8;
RMAN> VALIDATE DATAFILE 4 BLOCK 56;
RMAN> VALIDATE DATAFILE 8 SECTION SIZE = 200M;
RMAN> VALIDATE CURRENT CONTROLFILE;
RMAN> VALIDATE SPFILE;
RMAN> VALIDATE RECOVERY FILES;
RMAN> VALIDATE RECOVERY AREA;
RMAN> VALIDATE DB_RECOVERY_FILE_DEST;
```

#### SPOOL command

Write RMAN output to a log file.

```
RMAN> SPOOL LOG TO '/tmp/spool.log';
RMAN> SPOOL LOG TO '/tmp/backup.log' APPEND;
RMAN> SPOOL LOG OFF;
```

#### run command

Execute a sequence of one or more RMAN commands, which are one or more statements executed within the braces of RUN.

```
RMAN> run {
ALLOCATE CHANNEL c1 TYPE DISK FORMAT '/orabak/%U';
BACKUP TABLESPACE users;
}
RMAN> run {
ALLOCATE CHANNEL c1 TYPE DISK FORMAT '&1/%U';
BACKUP TABLESPACE &2;
}
RMAN> run
{
ALLOCATE CHANNEL dev1 DEVICE TYPE DISK FORMAT '/fs1/%U';
ALLOCATE CHANNEL dev2 DEVICE TYPE DISK FORMAT '/fs2/%U';
BACKUP(TABLESPACE system,fin,mark FILESPERSET 20) (DATAFILE
2,3,6);
}
```

### CREATE SCRIPT command

Create a stored script and store it in the recovery catalog.

```

RMAN> CREATE SCRIPT backup_whole
COMMENT "backup whole database and archived redo log files"
{BACKUP INCREMENTAL LEVEL 0 TAG backup_whole FORMAT
"/disk2/backup/%U" DATABASE PLUS ARCHIVELOG;}
    
```

```

RMAN> CREATE SCRIPT backup_ts_users
COMMENT 'tablespace users backup'
{ALLOCATE CHANNEL c1 TYPE DISK FORMAT 'c:\temp\%U';
BACKUP TABLESPACE users;}
    
```

```

RMAN> CREATE SCRIPT df {BACKUP DATAFILE &1 TAG &2.1 FORMAT
'/disk1/&3_%U';}
RMAN> CREATE SCRIPT backup_ts_users FROM FILE
'backup_ts_users.rman';
RMAN> CREATE GLOBAL SCRIPT gl_backup_db {BACKUP DATABASE
PLUS ARCHIVELOG;}
RMAN> CREATE GLOBAL SCRIPT backup_db
COMMENT "back up any database from the recovery catalog, with
logs"
{BACKUP DATABASE;}
    
```

### PRINT SCRIPT command

Display a stored script.

```

RMAN> PRINT SCRIPT backup_db;
RMAN> PRINT GLOBAL SCRIPT backup_db;
RMAN> PRINT GLOBAL SCRIPT gl_backup_db TO FILE
"/tmp/gl_backupdb.rman";
    
```

### REPLACE SCRIPT command

Replace an existing script stored in the recovery catalog. If the script does not exist, then REPLACE SCRIPT creates it.

```

RMAN> REPLACE SCRIPT backup_db {BACKUP DATABASE PLUS
ARCHIVELOG;}
RMAN> REPLACE SCRIPT df {BACKUP DATAFILE &1 TAG &2.1
FORMAT '&3_%U';}
RMAN> REPLACE GLOBAL SCRIPT backup_db {BACKUP DATABASE
PLUS ARCHIVELOG;}
RMAN> REPLACE GLOBAL SCRIPT gl_full_bkp FROM FILE
'/tmp/script_file.txt';
    
```

### FLASHBACK DATABASE command

Return the database to its state at a previous time or SCN.

```

RMAN> FLASHBACK DATABASE TO SCN 411010;
RMAN> FLASHBACK DATABASE TO RESTORE POINT 'before_update';
    
```

### TRANSPORT TABLESPACE command

Create transportable tablespace sets from backup for one or more tablespaces.

```

RMAN> TRANSPORT TABLESPACE example, tools
TABLESPACE DESTINATION '/disk1/trans' AUXILIARY DESTINATION
'/disk1/aux' UNTIL TIME 'SYSDATE-15/1440';
RMAN> TRANSPORT TABLESPACE exam
TABLESPACE DESTINATION '/disk1/trans' AUXILIARY DESTINATION
'/disk1/aux' DATAPUMP DIRECTORY dpdir DUMP FILE 'dmpfile.dmp'
IMPORT SCRIPT 'impscript.sql' EXPORT LOG 'explog.log';
    
```

### CONVERT command

Convert datafile formats for transporting tablespaces and databases across platforms.

```

RMAN> CONVERT DATABASE NEW DATABASE 'prodwin' TRANSPORT
SCRIPT '/tmp/convertdb/transportscript' TO PLATFORM 'Microsoft
Windows IA (32-bit)' DB_FILE_NAME_CONVERT
'/disk1/oracle/dbs','/tmp/convertdb';
RMAN> CONVERT DATABASE ON DESTINATION PLATFORM
CONVERT SCRIPT '/tmp/convertdb/convertscript.rman' TRANSPORT
SCRIPT '/tmp/convertdb/transportscript.sql' NEW DATABASE
'prodwin' FORMAT '/tmp/convertdb/%U';
RMAN> CONVERT DATABASE ON DESTINATION PLATFORM
CONVERT SCRIPT '/tmp/convert_newdb.rman' TRANSPORT SCRIPT
'/tmp/transport_newdb.sql' NEW DATABASE 'prodaix'
DB_FILE_NAME_CONVERT '/u01/oradata/datafile','+DATA';
    
```

```

RMAN> CONVERT TABLESPACE tbs_2 FORMAT '/tmp/tbs_2_%U.df';
RMAN> CONVERT TABLESPACE fin, hr TO PLATFORM 'Solaris[tm] OE
(32-bit)';
RMAN> CONVERT TABLESPACE fin, hr TO PLATFORM 'Solaris[tm] OE
(32-bit)' FORMAT '/tmp/transport_to_solaris/%U';
    
```

```

RMAN> CONVERT DATAFILE '/disk1/oracle/dbs/tbs_f1.df',
'/disk1/oracle/dbs/ax1.f' FORMAT '+DATAFILE';
RMAN> CONVERT DATAFILE '/u01/oradata/datafile/system.dbf'
FROM PLATFORM 'Linux x86 64-bit' FORMAT '+DATA/system.dbf';
RMAN> CONVERT DATAFILE
'/tmp/from_solaris/fin/fin01.dbf', '/tmp/from_solaris/fin/fin02.dbf',
'/tmp/from_solaris/hr/hr01.dbf', '/tmp/from_solaris/hr/hr02.dbf'
DB_FILE_NAME_CONVERT
'/tmp/from_solaris/fin','/disk2/orahome/dbs/fin',
'/tmp/from_solaris/hr','/disk2/orahome/dbs/hr'
FROM PLATFORM 'Solaris[tm] OE (64-bit)';
RMAN> CONVERT DATAFILE '/tmp/PSMN.dbf' TO
PLATFORM='Solaris Operating System (x86-64)' FROM
PLATFORM='Solaris[tm] OE (64-bit)'
    
```

```

FORMAT '/tmp/test/%N.dbf'
DB_FILE_NAME_CONVERT='/ui/prod/oracle/oradata/SEARCHP/data
/', '/tmp/test';
    
```