DUMPSBOSS.

Oracle Database Administration I

Oracle 1z0-082

Version Demo

Total Demo Questions: 17

Total Premium Questions: 172

Buy Premium PDF

https://dumpsboss.co

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

Topic Break Down

Topic	No. of Questions
Topic 1, Oracle Database Architecture	15
Topic 2, Creating an Oracle Database	2
Topic 3, Managing the Oracle Instance	8
Topic 4, Configuring the Oracle Network Environment	8
Topic 5, Managing Database Storage Structures	25
Topic 6, Administering User Security	12
Topic 7, Managing Schema Objects	64
Topic 8, Managing Data Concurrency	6
Topic 9, Managing Undo Data	8
Topic 10, Performing Database Backups	1
Topic 11, Moving Data	7
Topic 12, Mix Questions	16
Total	172

QUESTION NO: 1

Examine this SQL statement:

SELECT cust_id, cust_last_name “Last Name”

FROM customers

WHERE country_id = 10

UNION

SELECT cust_id CUST_NO, cust_last_name

FROM customers

WHERE country_id = 30

Identify three ORDER BY clauses, any one of which can complete the query successfully. (Choose three.)

A. ORDER BY “Last Name”

B. ORDER BY 2, 1

C. ORDER BY 2, cust_id

D. ORDER BY CUST_NO

E. ORDER BY “CUST_NO”

ANSWER: A B C

Explanation:

For a compound query using `UNION`, the `ORDER BY` clause applies to the final combined result set. Oracle permits ordering that result set by the ordinal position of a select-list item or by a column alias defined in the first component query. `ORDER BY "Last Name"` is valid because `"Last Name"` is the quoted alias assigned to `cust_last_name` in the first `SELECT`. The quotation marks are required because the alias contains a space and is therefore case-sensitive.

`ORDER BY 2, 1` is also valid: position `2` denotes the result set’s last-name column and position `1` denotes its customer-ID column. Positional ordering is supported for compound-query results. `ORDER BY 2, cust_id` succeeds for the same reason: it first uses the second select-list position, then uses `cust_id`, which is the name of the first result column as determined by the first `SELECT`. Oracle documents that a compound query’s output column names derive from the first component query and that ordering may use select-list positions or valid aliases. See the Oracle SQL Language Reference on compound queries and the SELECT statement reference.

QUESTION NO: 2

What is true about non-equijoin statement performance? (Choose two.)

A. Table aliases can improve performance

B. The BETWEEN condition always performs better than using the >= and <= conditions

C. The join syntax used makes no difference to performance

D. The BETWEEN condition always performs less well than using the >= and <= conditions

E. The Oracle join syntax performs better than the SQL:1999 compliant ANSI join syntax

F. Non-equijoin performance depends on the execution plan, statistics, indexes, predicate selectivity, and data volume.

ANSWER: C F

Explanation:

The join syntax used makes no difference to performance when equivalent inner-join predicates are expressed. Oracle's optimizer evaluates the query semantics and can transform logically equivalent formulations into an efficient execution plan; using ANSI SQL join notation rather than Oracle's legacy join notation does not, by itself, make a non-equijoin faster or slower. A non-equijoin commonly compares a value to a range, such as when a salary falls between the low and high values in a grade table. Its actual performance depends on the optimizer's chosen access paths and join method, which are influenced by factors such as object statistics, available indexes, predicate selectivity, data volume, and the join order. Therefore, tuning should focus on the execution plan and the relevant statistics and indexing rather than on aliases or a presumed inherent advantage of one equivalent join syntax. Oracle documents non-equijoins and their range-based predicates in its SQL language reference, and explains the optimizer's role in selecting execution plans in the database performance documentation. See Oracle Database SQL Language Reference: Joins and Oracle Database SQL Tuning Guide: Query Optimizer Concepts.

QUESTION NO: 3

Which two statements are true about the DUAL table? (Choose two.)

A. It can be accessed only by the SYS user

B. It consists of a single row and single column of VARCHAR2 data type

C. It can display multiple rows but only a single column

D. It can be used to display only constants or pseudo columns

E. It can be accessed by any user who has the SELECT privilege in any schema

F. It can display multiple rows and columns

ANSWER: B F

Explanation:

The statement "It consists of a single row and single column of VARCHAR2 data type" is correct for the underlying DUAL table. DUAL is a special one-row table owned by SYS. Its sole column is named DUMMY and is defined as VARCHAR2(1); the single stored value is traditionally X. Oracle provides DUAL primarily as a convenient source table when a SQL statement needs a FROM clause but the values being selected are expressions rather than values retrieved from an application table.

The statement "It can display multiple rows and columns" is also correct because the result of a query that uses DUAL is not restricted to the physical shape of DUAL itself. A SELECT list can contain multiple expressions, producing multiple result columns. In addition, row-generating SQL constructs can produce more than one result row, for example by using a hierarchical query with CONNECT BY LEVEL. Thus, DUAL remains a one-row, one-column base table, while a query that references it can return a result set with multiple rows and columns. Oracle also optimizes many queries that select expressions from DUAL through its FAST DUAL optimization. See Oracle's Selecting from the DUAL Table documentation and its SELECT statement reference.

QUESTION NO: 4

Examine this command and some partial output:

```
LSNRCTL> start LISTENER_1

Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=PROTOCOL=tcp) (HOST=host01.abc.com) (PORT=1562)))
Services Summary...
Service "DB01.abc.com " has 1 instance(s).
  Instance "DB01", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

Why does the DB01.abc.com service show unknown status?

A. The service DB01.abc.com is dynamically registered

B. The LOCAL_LISTENER database parameter is not set to a service name that refers to LISTENER_1

C. The service DB01.abc.com is statically registered

D. The listener is not listening on the default port 1521

E. The SID_LIST_LISTENER section is not contained in the LISTENER.ORA file

ANSWER: C

Explanation:

The service DB01.abc.com is statically registered. An `UNKNOWN` status in `lsnrctl services` output is the normal status that the listener reports for a service defined statically in its `listener.ora` configuration, typically through a `SID_LIST_listener_name` entry. With static registration, the listener has configuration information that tells it how to direct a connection to the database, but it does not receive the database instance health and availability information that is supplied by dynamic service registration. Consequently, it cannot establish whether the instance is currently ready to accept connections and displays the handler or service status as `UNKNOWN`.

Static registration is useful in situations such as enabling connections when dynamic registration is unavailable, and for certain administrative connection scenarios. In contrast, services registered dynamically by the database's listener-registration process provide runtime status information to the listener. Oracle documents the listener service-status values and explains that `UNKNOWN` applies to statically configured services: Oracle Net Services Administrator's Guide: Monitoring the Listener. The static listener configuration mechanism is described in Oracle Net Services Administrator's Guide: Configuring the Listener.

QUESTION NO: 5

Which three statements are true about advanced connection options supported by Oracle Net for connection to Oracle Database instances? (Choose three.)

A. Connect Time Failover requires the use of Transparent Application Failover (TAF)

B. Source Routing requires the use of a name server

C. Source Routing enables the use of Connection Manager (CMAN) which enables network traffic to be routed through a firewall

D. Load Balancing can balance the number of connections to dispatchers when using a Shared Server configuration

E. Load Balancing requires the use of a name server

F. Connect Time Failover requires the connect string to have two or more listener addresses configured

ANSWER: C D F

Explanation:

Source Routing enables the use of Connection Manager (CMAN) which enables network traffic to be routed through a firewall because Oracle Connection Manager can act as an intermediary between clients and database servers. A source-routed connect descriptor explicitly identifies the route that a connection must follow, including the Connection Manager address. This allows clients to reach database services without requiring direct client-to-database network access, a common design for firewall-separated networks.

Load Balancing can balance the number of connections to dispatchers when using a Shared Server configuration. In a shared server environment, the listener selects an appropriate dispatcher for an incoming client session, distributing connections among available dispatchers to support efficient use of the shared server architecture.

Connect Time Failover requires the connect string to have two or more listener addresses configured. The client must have alternative addresses in its connect descriptor or resolved connect data so that, if one listener address cannot be reached during initial connection establishment, Oracle Net can attempt another configured listener address. This feature applies at connection time and is distinct from failover handling after a session has already been established. See Oracle's Oracle Net Services advanced configuration documentation and Oracle Net naming and connection configuration documentation.

QUESTION NO: 6

Examine this description of the books table containing 100 rows:

```
Name                 Null?    Type
-------------------- -------- ----------------------------
TRANSACTION_ID       NOT NULL VARCHAR2(6)
TRANSACTION_DATE              DATE
AMOUNT                        NUMBER(10,2)
CUSTOMER_ID                   VARCHAR2(6)
```

Now examine this sequence of statements Issued In a new session;

```
INSERT INTO books VALUES ('ADV112', 'Adventures of Tom Sawyer', NULL, NULL);

SAVEPOINT a;

DELETE FROM books;

ROLLBACK TO SAVEPOINT a;

ROLLBACK;
```

Examine this description of the books table containing 100 rows:

Which three statements are true?

A. The second rollback command restores the row that was inserted.

B. The first rollback command leaves the table ' s 100 original rows locked.

C. The second rollback command rolls back the rollback to savepoint a command.

D. The first rollback command leaves the inserted row locked.

E. The first RollBack command restores the row that was inserted.

F. The second rollback command restores the 100 rows that were in the table originally.

G. The first rollback command restores the 100 rows that were in the table originally.

ANSWER: D E F

Explanation:

The correct statements follow from the transaction boundaries established by the savepoint. The insert occurs before savepoint `a`, while the delete occurs after it. Therefore, `ROLLBACK TO SAVEPOINT a` reverses the delete and returns the inserted row to the table, because that insert is part of the transaction state at the savepoint. Oracle specifies that a rollback to a savepoint undoes statements executed after that savepoint but retains the transaction and the work performed before it. See the Oracle ROLLBACK SQL Reference.

Because the insert was performed before the savepoint and has not been committed or fully rolled back, its transaction lock remains in place after the rollback to the savepoint. Thus, the inserted row remains locked by the current session until the transaction ends. Oracle's discussion of transaction concurrency and locking is available in the Oracle Database Concepts documentation.

The subsequent plain `ROLLBACK` ends the transaction and undoes all remaining uncommitted work, including the insert that occurred before the savepoint. The table consequently returns to its original state of 100 rows.

QUESTION NO: 7

In one of your databases, user KING is:

1. Not a DBA user

2. An operating system (OS) user

Examine this command and its output:

```
SHOW PARAMETER OS_AUTHENT_PREFIX
NAME                 TYPE                 VALUE
-------------------  -------------------  -----------------
os_authen_prefix     string
```

What must you do so that KING is authenticated by the OS when connecting to the database instance?

A. Set OS_AUTHENT_PREFIX to OPS$

B. Have the OS administrator add KING to the OSDBA group

C. Grant DBA to KING

D. Unset REMOTE_LOGIN_PASSWORDFILE

E. Alter user KING to be IDENTIFIED EXTERNALLY

ANSWER: E

Explanation:

Alter user KING to be IDENTIFIED EXTERNALLY is required because Oracle must explicitly define the database account as an externally authenticated user before it accepts operating-system authentication for that account. The `IDENTIFIED EXTERNALLY` clause changes the authentication method from database-managed credentials to external authentication, allowing Oracle Database to rely on the authenticated OS identity when KING connects locally without supplying a database password.

Oracle maps the OS username to the database username according to the `OS_AUTHENT_PREFIX` initialization parameter. If the prefix is null, the mapped database account is simply `KING`; if it has a value, the externally identified database username must use that configured prefix. The shown account name and parameter output determine the applicable mapping, but the necessary account-level action is to alter KING to use external identification. Membership in privileged OS groups is not required for ordinary external authentication; it is relevant to administrative authentication and privileges. Oracle documents

external authentication and the use of `IDENTIFIED EXTERNALLY` in the SQL language reference and the security guide: Oracle ALTER USER reference and Oracle Database Security Guide: Configuring Authentication.

QUESTION NO: 8

The EMPLOYEES table contains DEPARTMENT_ID and SALARY columns.

You want to display each department identifier and its average salary only for departments whose average salary is greater than 5000.

Which query can be used?

A.
```
SELECT department_id, AVG(salary)
FROM employees
WHERE AVG(salary) > 5000
GROUP BY department_id;
```

B.
```
SELECT department_id, AVG(salary)
FROM employees
GROUP BY department_id
HAVING AVG(salary) > 5000;
```

C.
```
SELECT department_id, AVG(salary)
FROM employees
HAVING AVG(salary) > 5000;
```

D.
```
SELECT department_id, salary
FROM employees
GROUP BY department_id
HAVING salary > 5000;
```

ANSWER: B

Explanation:

The query must group rows by `department_id`, calculate `AVG(salary)` for each group, and use a `HAVING` condition to restrict groups based on the aggregate result. A `WHERE` clause cannot contain this aggregate condition because WHERE filters individual rows before grouping occurs.

See the Oracle Database SQL Language Reference: SELECT.

QUESTION NO: 9

A sequence named ORDER_SEQ is used to generate order numbers.

A session obtains a value by selecting `ORDER_SEQ.NEXTVAL` and subsequently rolls back its transaction.

Which statement is true?

A. The generated sequence number is not reused because of the rollback.

B. The generated sequence number is returned to the sequence after the rollback.

C. The sequence is reset to its START WITH value after the rollback.

D. The session cannot roll back after selecting NEXTVAL.

ANSWER: A

Explanation:

A sequence value returned by `NEXTVAL` is incremented independently of transaction commit or rollback processing. Therefore, rolling back the transaction does not return the generated number to the sequence, and that number is not reused merely because the transaction was rolled back. Gaps in sequence values are normal and can also result from caching and concurrent sessions.

See the Oracle Database SQL Language Reference: CREATE SEQUENCE.

QUESTION NO: 10

A script abc.sql must be executed to perform a job.

A database user HR, who is defined in this database, executes this command:

$ sqlplus hr/hr@orcl @abc.sql

What will happen upon execution?

A. The command succeeds and HR will be connected to the orcl and abc.sql databases

B. The command succeeds and HR will be connected to the orcl database instance, and the abc.sql script will be executed

C. The command fails because the script must refer to the full path name

D. The command fails and reports an error because @ is used twice

ANSWER: B

Explanation:

The command succeeds by starting SQL*Plus, connecting the HR user to the database service identified by `orcl`, and then executing the SQL*Plus script named `abc.sql`. In the connection string `hr/hr@orcl`, the at sign separates the password from the connect identifier, which SQL*Plus uses to locate the target database or service through its configured naming method. After a successful connection, the separate command-line argument `@abc.sql` is interpreted as a SQL*Plus script invocation. SQL*Plus processes the commands contained in that script in the connected HR session. The script name does not identify a database; it identifies a command file available to SQL*Plus, ordinarily in the current working directory or in a location SQL*Plus can find through its script-search behavior. This syntax permits a connection and an initial script execution to be supplied in one operating-system command. Oracle documents SQL*Plus startup and connection syntax at Starting SQL*Plus and documents use of the `@` script command at Using Scripts in SQL*Plus.

QUESTION NO: 11

Which three functions are performed by dispatchers in a shared server configuration? (Choose three.)

A. writing inbound request to the common request queue from all shared server connections

B. checking for outbound shared server responses on the common outbound response queue

C. receiving inbound requests from processes using shared server connections

D. sending each connection input request to the appropriate shared server input queue

E. broadcasting shared server session responses back to requesters on all connections

F. sending shared server session responses back to requesters on the appropriate connection

ANSWER: A C F

Explanation:

In Oracle shared server architecture, a dispatcher is the network-facing process for clients connected through shared server connections. It receives inbound requests from those client processes, which allows clients to communicate with the

database without each client requiring a dedicated server process. After receiving a request, the dispatcher writes it to the common request queue. Shared server processes retrieve work from that common queue and execute the SQL or other database operation on behalf of the client session.

Once processing is complete, the shared server places the result on the response queue associated with the dispatcher serving that client connection. The dispatcher then sends the shared server session response to the requester on the appropriate connection. This routing preserves the association between each client session and its network connection while allowing a pool of shared server processes to service many sessions efficiently. Oracle describes dispatchers as receiving client requests, placing them on the request queue, and returning responses to clients. See the Oracle Database Concepts: Process Architecture and the Oracle Database Administrator's Guide: Managing Processes.

QUESTION NO: 12

You want to use table compression suitable for OLTP that will:

1. Compress rows for all DML statements on that table 2. Minimize the overheads associated with compression Which compression option is best suited for this?

A. COLUMN STORE COMPRESS FOR QUERY LOW

B. ROW STORE COMPRESS BASIC

C. COLUMN STORE COMPRESS FOR ARCHIVE LOW

D. COLUMN STORE COMPRESS FOR ARCHIVE HIGH

E. ROW STORE COMPRESS ADVANCED

ANSWER: E

Explanation:

ROW STORE COMPRESS ADVANCED is the appropriate choice for an OLTP table because Oracle Advanced Row Compression is designed to compress data during conventional DML activity, including inserts and updates. Unlike basic row compression, which is principally intended for bulk-load operations, Advanced Row Compression maintains compression as rows are created or modified through normal OLTP processing. This directly satisfies the requirement to compress rows for all DML statements.

The feature is also designed to limit performance impact by using an efficient compression algorithm and by compressing data in database blocks. Oracle documents that Advanced Row Compression can reduce the storage consumed by tables and associated I/O, while supporting regular transactional workloads. Lower I/O can provide an additional operational benefit where queries or scans must read compressed table blocks. The SQL syntax for this capability is `ROW STORE COMPRESS ADVANCED` in a table definition or compression clause.

Oracle's CREATE TABLE SQL Reference documents the row-store compression clauses, and the Oracle Database Administrator's Guide describes Advanced Row Compression use for OLTP workloads and conventional DML.

QUESTION NO: 13

In one of your databases, the user HR has the password HRMGR.

You want to connect to a database instance whose listener listens on port 1531 by using this statement:

CONNECT HR/HRMGR@orcl

No name server is used.

Which statement is true about ORCL?

A. It must be the value of the SERVICE_NAMES parameter on the client side

B. It must resolve to a valid connect descriptor in the server's tnsnames.ora file

C. It must resolve to a valid connect descriptor in the client's tnsnames.ora file

D. It must be the name of the database to whose instance HR wishes to connect

E. It must be the name of the server running the database to whose instance HR wishes to connect

ANSWER: C

Explanation:

It must resolve to a valid connect descriptor in the client's tnsnames.ora file is correct. In the connection command, `orcl` is a net service name, not automatically the database name or server name. Oracle Net must translate that net service name into a connect descriptor that identifies the network location and target service. Because no directory naming service (name server) is used, local naming is the applicable naming method: the client uses its `tnsnames.ora` file to resolve `orcl`.

The connect descriptor associated with `orcl` must include the listener host, port `1531`, and the target database service name. For example, it can contain an `ADDRESS` section with `PORT=1531` and a `CONNECT_DATA` section with `SERVICE_NAME` set to the service registered with that listener. Oracle Net configuration and name resolution occur on the client initiating the connection, before the listener receives the request. Oracle documents local naming as resolving a net service name through entries in a client-side `tnsnames.ora` file. See Oracle Net Services naming methods documentation and Oracle local naming reference.

QUESTION NO: 14

In the ORCL database, UNDOTBS1 is the active undo tablespace with these properties:

1. A size of 100 MB

2. AUTOEXTEND is off

3. UNDO_RETENTION is set to 15 minutes

4. It has RETENTION GUARANTEE

UNDOTBS1 fills with uncommitted undo 10 minutes after the database opens.

What will happen when the next update is attempted by any transaction?

A. It succeeds and the generated undo is stored in SYSTEM.

B. It fails and returns the error message "ORA-30036: unable to extend segment by 8 in undo tablespace 'UNDOTBS1'".

C. It succeeds and the least recently written undo block of UNDOTBS1 is overwritten by the generated undo.

D. It succeeds and the generated undo is stored in SYSAUX.

E. It succeeds and the least recently read undo block of UNDOTBS1 is overwritten by the generated undo.

ANSWER: B

Explanation:

It fails and returns the error message "ORA-30036: unable to extend segment by 8 in undo tablespace 'UNDOTBS1'". The active undo tablespace has reached its fixed 100 MB capacity, and AUTOEXTEND is disabled, so Oracle cannot obtain additional space by growing an undo datafile. More importantly, the existing undo is uncommitted, meaning it is active undo required for transaction rollback and read consistency. Active undo cannot be overwritten to make room for a new update.

RETENTION GUARANTEE reinforces the protection of undo needed within the retention period: Oracle preserves eligible unexpired undo rather than reusing it for new DML. When an undo tablespace with retention guarantee cannot allocate the required undo extent, DML that needs to generate undo fails with ORA-30036. The fact that only 10 minutes have elapsed is also within the configured 15-minute retention target, but the decisive condition is that the tablespace is full of uncommitted undo, which must remain available. Oracle documents that retention guarantee can cause DML to fail when undo space is

exhausted, rather than allowing protected undo to be overwritten. See Oracle Database Administrator's Guide: Managing Undo and Oracle Error Help: ORA-30036.

QUESTION NO: 15

Which three statements are true concerning logical and physical database structures? (Choose three.)

A. All tablespaces may have one or more data files

B. The extents of a segment must always reside in the same datafile

C. A smallfile tablespace might be bigger than a bigfile tablespace

D. A segment can span multiple data files in some tablespaces

E. A segment's blocks can be of different sizes

F. A segment might have only one extent

G. Segments can span multiple tablespaces

ANSWER: C D F

Explanation:

A smallfile tablespace might be bigger than a bigfile tablespace because a smallfile tablespace can contain multiple data files, whereas a bigfile tablespace contains a single data file. Although each individual smallfile is subject to the database file-size limit, the combined capacity of its multiple files can exceed the capacity of a bigfile tablespace. This distinction is particularly relevant when planning tablespace growth and file-management practices.

A segment can span multiple data files in some tablespaces because a segment is composed of one or more extents, and Oracle can allocate successive extents in different data files belonging to the same tablespace. The segment remains a single logical storage object even when its extents are physically distributed among the tablespace's files. A segment might have only one extent when its initially allocated extent remains sufficient and no additional extent allocation has been required. Oracle documentation describes a segment as a set of extents, with each extent consisting of contiguous data blocks. See the Oracle Database logical storage structures documentation and the Oracle tablespace management documentation.

QUESTION NO: 16

The stores table has a column START_DATE of data type DATE, containing the date the row was inserted. You only want to display details of rows where START_DATE is within the last 25 months. Which where clause can be used?

A. WHERE MON'THS_BETWEEN (SYSDATE, start_date) <= 25

B. WHERE ADD_MONTHS

C. WHERE TO_XUMBER(start_date - SYSDATE) <= 25

D. WHERE MON'THS_BETWEEN (start_date, SYSDATE) <= 25

E. WHERE start_date >= ADD_MONTHS(SYSDATE, -25) AND start_date <= SYSDATE

ANSWER: E

Explanation:

`WHERE start_date >= ADD_MONTHS(SYSDATE, -25) AND start_date <= SYSDATE` correctly defines a rolling date range beginning 25 calendar months before the current database date and ending at the current database date. `SYSDATE` supplies the current date and time according to the database server's operating-system clock. Passing `-25` to `ADD_MONTHS` moves that date backward by 25 months, producing the inclusive lower boundary for the required period.

Comparing `start_date` with this boundary ensures that qualifying rows were inserted no earlier than that calculated date. The upper-bound predicate ensures the stored insertion date is not later than the current date, so the result is strictly within the elapsed 25-month period rather than merely after its lower boundary. This approach uses date arithmetic at the month level, which is appropriate because the requirement is expressed in months rather than as a fixed number of days. Oracle documents that `ADD_MONTHS` adds an integer number of months to a date and returns a `DATE` value, allowing a direct comparison to a `DATE` column. See Oracle Database SQL Language Reference: ADD_MONTHS and Oracle Database SQL Language Reference: SYSDATE.

QUESTION NO: 17

Which three statements are true about temporary tablespaces and tempfiles? (Choose three.)

A. A tempfile can belong only to a temporary tablespace.

B. A tempfile can be configured to extend automatically.

C. RMAN does not back up tempfiles.

D. A permanent table segment can be stored in a tempfile.

E. A tempfile can be added to a read-only tablespace.

F. A tempfile contains archived redo information.

ANSWER: A B C

Explanation:

Tempfiles belong to temporary tablespaces and are used for temporary segments required by operations such as sorts, hash joins, and temporary-table processing. A tempfile can be configured with `AUTOEXTEND ON` so that it grows when more temporary space is required. RMAN does not back up tempfiles because their contents are temporary and can be recreated when necessary.

Permanent table segments cannot be stored in a temporary tablespace. See the Oracle Database Administrator's Guide: Managing Tablespaces.