

Oracle® Tutor™



Section5 注意事项、考点和答案

0. 注意事项、考点：

exp/imp
expdp/impdp

```
[oracle@rac1 ~]$ impdp hr/oracle@prod directory=dir dumpfile=exp_mag.dmp  
tables=MAGAZINE_ARTICLES
```

Import: Release 10.2.0.1.0 - Production on Friday, 09 March, 2012 15:17:23

Copyright (c) 2003, 2005, Oracle. All rights reserved.

Connected to: Oracle Database 10g Enterprise Edition Release 10.2.0.1.0 - Production

With the Partitioning, OLAP and Data Mining options

Master table "HR"."SYS_IMPORT_TABLE_01" successfully loaded/unloaded

Starting "HR"."SYS_IMPORT_TABLE_01": hr/*****@prod directory=dir dumpfile=exp_mag.dmp
tables=MAGAZINE_ARTICLES

Processing object type TABLE_EXPORT/TABLE/TABLE

ORA-39151: Table "HR"."MAGAZINE_ARTICLES" exists. All dependent metadata and data will be skipped
due to table_exists action of skip

Processing object type TABLE_EXPORT/TABLE/TABLE_DATA

Job "HR"."SYS_IMPORT_TABLE_01" completed with 1 error(s) at 15:17:29

此时发现表已存在，无法导入，于是在导入时加上一个参数 CONTENT=data_only
在导入的时候将 HR 用户退出 Sqlplus，否则导入时会卡住不动。

传输表空间
创建分区表
如何处理违反约束的行
非标准块的表
创建分区索引 global hash 和 local
truncate partition
创建带 clob 字段的表
并行
使用 FGA 对一张表做审计 (可以参考联机帮助)
flashback table|query|drop
使用 FLASHBACK 特性恢复指定内容的已删除的表

1. Transportable Tablespace 表空间迁移：

1. 提供 sst.dmp 文件使用 imp，导入所有对象到 PROD 数据库，oltp_user 用户下。（SST 用户导出的 dmp 文件。）

答案：

步骤：

1. 创建 oltp_user 用户

```
create user oltp_user identified by oracle account unlock;
```

User created.

2. 给 OLTP_USER 赋予基本的权限

```
grant connect,resource to oltp_user;
```

Grant succeeded.

3. 使用 imp 导入 sst.dmp 到 oltp_user schema

```
imp system/oracle file=/home/oracle/sst.dmp buffer=100000 fromuser=SST  
touser=OLTP_USER
```

步骤:

1. 创建 oltp_user 用户

```
SQL> create user oltp_user identified by oracle account unlock;
```

User created.

2. 给 OLTP_USER 赋予基本的权限

```
SQL> grant connect,resource to oltp_user;
```

Grant succeeded.

3. 使用 imp 导入 sst.dmp 到 oltp_user schema

```
imp oltp_user/oracle file=sst.dmp fromuser=sst
```

Import: Release 10.2.0.2.0 - Production on Sat Jan 10 03:05:01 2015

Copyright (c) 1982, 2005, Oracle. All rights reserved.

Connected to: Oracle Database 10g Enterprise Edition Release 10.2.0.2.0 - Production
With the Partitioning, OLAP and Data Mining options

Export file created by EXPORT:V10.02.01 via conventional path
IMP-00013: only a DBA can import a file exported by another DBA
IMP-00000: Import terminated unsuccessfully

```
[oracle@rac1 ~]$ imp system/oracle@prod file=sst.dmp buffer=100000 fromuser=sst touser=oltp_user
```

Import: Release 10.2.0.1.0 - Production on Fri Mar 9 10:34:51 2012

Copyright (c) 1982, 2005, Oracle. All rights reserved.

Connected to: Oracle Database 10g Enterprise Edition Release 10.2.0.1.0 - Production
With the Partitioning, OLAP and Data Mining options

Export file created by EXPORT:V10.02.01 via conventional path

Warning: the objects were exported by SST, not by you

import done in ZHS16GBK character set and AL16UTF16 NCHAR character set
import server uses AL32UTF8 character set (possible charset conversion)
. importing SST's objects into OLTP_USER
.. importing table "T" 9612 rows imported
Import terminated successfully without warnings.

4. 从 PROD 数据库，迁移 OLTP 表空间，到 EMREP 数据库。OLTP 表空间应提供读和写两个数据库的所有对象的拥有权限。由用户 oltp_user 在 PROD 数据库，对应应在 emrep 数据库的 OLTP 表空间迁移后运行。

答案：

检查 OLTP 表空间是否是自包含表空间（要被传输的表空间中的对象没有引用被传输的表空间之外的对象，这种表空间就是自包含表空间）

```
exec dbms_tts.transport_set_check('OLTP',true);
```

```
SELECT * FROM TRANSPORT_SET_VIOLATIONS;
```

检查源端和目标端的 BLOCK_SIZE 是否一致

源端：

```
select block_size from dba_tablespaces where tablespace_name='OLTP';
```

目标端：

```
show parameter db_block_size
```

如不一致可在目标端通过诸如 alter system set db_4k_cache_size=8M; 语句来修改表空间只读。

```
expdp system/oracle DUMPFILE=oltp_tsp.dmp DIRECTORY=***
```

```
TRANSPORT_TABLESPACES =OLTP
```

目标端：

```
CREATE USER oltp_user IDENTIFIED BY oltp_user;
```

```
grant dba to oltp_user;
```

将 OLTP 表空间对应的数据文件拷贝到 EMREP 数据库中

```
cp /u01/app/oracle/oradata/PROD/Disk1/oltp1.dbf /u01/app/oracle/oradata/EMREP/
```

导入传输集到 EMREP 数据库中，使 OLTP 表空间注册到 EMREP 数据库中

```
impdp system/oracle dumpfile=oltp_tr.dmp DIRECTORY=IMP_TR_DIR
```

```
TRANSPORT_DATAFILES=/u02/oradata/EMREP/oltp01.dbf
```

将 OLTP 表空间在 PROD 和 EMREP 数据库中均置为可读可写状态

参考文档：Administrator's Guide=> 8 Managing Tablespaces=> Transporting Tablespaces Between Databases

拷贝 oltp.dmp 和 datafile

```
select distinct owner from dba_tables where tablespace_name='OLTP';
```

```
impdp system/a DIRECTORY=mydir1 DUMPFILE=oltp.dmp
```

```
TRANSPORT_DATAFILES='/opt/oracle/oracle/oradata/proc/OLTP01.dbf'
```

2.Create Additional Buffer Cache :

题目说明：在 PROD 数据库的 SGA 中创建额外的块大小为 16k 的 buffer cache，保证它将在 SGA 中可用。

```
alter system set db_16k_cache_size=16M;
```

3.Working with LOB Data

1. 在 PROD 数据库创建一个新的表空间，名为 LOB_DATA,来满足以下的数据存储和指标要求：

1.1 在不同的位置，创建 2 个数据文件。

1.2 每个文件应该是 64M 大小。

1.3 块大小为 16K

1.4 决定哪种类型的管理将是最适合工作的数据和配置适当的程度。

答案：

方法一：在 Sqlplus 中用命令行创建：

```
create tablespace LOB_DATA datafile
'/u01/app/oracle/oradata/PROD/Disk1/lob_data01.dbf' size 64M
autoextend on next 2M,
'/u01/app/oracle/oradata/PROD/Disk2/lob_data02.dbf' size 64M
autoextend on next 2M
extent management local uniform size 2M
segment space management auto
blocksize 16k;
```

方法二：用 GC 图形界面创建：

Database Instance: PROD

Home

Performance

Administration

Maintenance

The Administration tab displays links that allow you to administer database objects and initiate database operations inside an Oracle database. The Maintenance tab displays links that provide functions that control the flow of data between or outside Oracle databases.

Database Administration

Storage

Control Files

Tablespaces

Temporary Tablespace Groups

Datafiles

Rollback Segments

Redo Log Groups

Archive Logs

Statistics Management

Automatic Workload Repository

Manage Optimizer Statistics

Database Configuration

Memory Parameters

Undo Management

All Initialization Parameters

Database Feature Usage

Change Database

Migrate to ASM

Convert to Cluster Database

Make Tablespace Locally Managed

Oracle Scheduler

Jobs

Chains

Schedules

Programs

Job Classes

Windows

Window Groups

Global Attributes

Resource Manager

Monitors

Consumer Groups

Consumer Group Mappings

Plans

Database Login

* Username sys

* Password *****

Database PROD

* Connect As SYSDBA

☐ Save as Preferred Credential

Cancel

Login

Database Instance: PROD >

Logged in As SYS

Tablespaces

Object Type Tablespace

Search

Select an object type and optionally enter an object name to filter the data that is displayed in your results set.

Object Name

Go

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode Single

Create

GeneralStorage

* NameLOB_DATA

Extent Management

☒ Locally Managed

☐ Dictionary Managed

Type

☒ Permanent

☐ Set as default permanent tablespace

☐ Temporary

☐ Set as default temporary tablespace

☐ Undo

Undo Retention Guarantee ☐ Yes ☒ No

Status

☒ Read Write

☐ Read Only

☐ Offline

Datfiles

☐ Use bigfile tablespace
Tablespace can have only one datafile with no practical size limit.

Add

EditRemove

Select Name	Directory	Size (MB)
<input checked="" type="radio"/> lob_data01.dbf	/u01/app/oracle/oradata/PROD/Disk1/	64.00
<input type="radio"/> lob_data02.dbf	/u01/app/oracle/oradata/PROD/Disk2/	64.00

GeneralStorage

Database Instance: PROD > Tablespaces > Logged in As SYS

Create Tablespace: Edit Datafile

CancelContinue

* File Namelob_data01.dbf

* File Directory/u01/app/oracle/oradata/PROD/Disk1/

TablespaceLOB_DATA

File Size64MB

☐ Reuse Existing File

Storage

☒ Automatically extend datafile when full (AUTOEXTEND)

Increment2MB

Maximum File Size ☒ Unlimited

☐ ValueMB

☒ TIP Changes made on this page will NOT take effect until you click "Apply" button on the Tablespace page.

CancelContinue

Database Instance: PROD > Tablespaces > Logged in As SYS

Create Tablespace: Edit Datafile

CancelContinue

* File Namelob_data02.dbf

* File Directory/u01/app/oracle/oradata/PROD/Disk2/

TablespaceLOB_DATA

File Size64MB

☐ Reuse Existing File

Storage

☒ Automatically extend datafile when full (AUTOEXTEND)

Increment2MB

Maximum File Size ☒ Unlimited

☐ ValueMB

☒ TIP Changes made on this page will NOT take effect until you click "Apply" button on the Tablespace page.

CancelContinue

General

Storage

Extent Allocation

☐ Automatic

☒ Uniform

Size MB

Segment Space Management

☒ Automatic

Objects in the tablespace automatically manage their free space. It offers high performance for free space management.

☐ Manual

Objects in the tablespace will manage their free space using free lists. It is provided for backward compatibility.

Enable logging

☒ Yes

Generate redo logs for creation of tables, indexes and partitions, and for subsequent inserts. Recoverable

☐ No

Redo log entries are smaller, the above operations are not logged and not recoverable.

Block information

Block Size (B)

General

Storage

Database Instance: PROD > [Tablespaces](#) > [Create Tablespace](#) >

Logged in As SYS

Show SQL

Execute On Multiple Databases

Return

```
CREATE SMALLFILE TABLESPACE "LOB_DATA" DATAFILE '/u01/app/oracle/oradata/PROD/Disk1/lob_data01.dbf' SIZE 64M AUTOEXTEND ON NEXT 2M MAXSIZE UNLIMITED , '/u01/app/oracle/oradata/PROD/Disk2/lob_data02.dbf' SIZE 64M AUTOEXTEND ON NEXT 2M MAXSIZE UNLIMITED LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 2M SEGMENT SPACE MANAGEMENT AUTO BLOCKSIZE 16384
```

Execute On Multiple Databases

Return

点击 OK

Update Message

The object has been created successfully

Tablespaces

Object Type

Search

Select an object type and optionally enter an object name to filter the data that is displayed in your results set.

Object Name

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode

<div><div>Edit</div><div>View</div><div>Delete</div><div>Actions</div><div>Add Datafile</div><div>Go</div></div>											
Select	Name	Allocated Size(MB)	Space Used(MB)	Allocated Space Used(%)	Allocated Free Space(MB)	Status	Datafiles	Type	Extent Management	Segment Management	
<input type="radio"/>	EXAMPLE	400.0	290.0	<div><div></div></div> <div>72.5</div>	110.0	✓	1	PERMANENT	LOCAL	AUTO	
<input type="radio"/>	INDX	40.0	0.1	<div><div></div></div> <div>0.2</div>	39.9	✓	1	PERMANENT	LOCAL	AUTO	
<input type="radio"/>	LOB_DATA	128.0	4.0	<div><div></div></div> <div>3.1</div>	124.0	✓	2	PERMANENT	LOCAL	AUTO	

4. Manage Schema Data

答案：

方法一：用 Sqlplus 命令行，以 hr 身份登录 PROD，执行以下 SQL 语句：

```
CREATE TABLE MAGAZINE_ARTICLES  
( AUTHOR VARCHAR2(30),  
  ARTICLE_NAME VARCHAR2(50),  
  ARTICLE_DATE DATE,  
  ARTICLE_DATA CLOB  
)
```

```
TABLESPACE USERS LOB(ARTICLE_DATA) STORE AS (TABLESPACE LOB_DATA  
STORAGE (INITIAL 2m NEXT 2m) CHUNK 16384 NOCACHE DISABLE STORAGE IN ROW);
```

方法二：用 GC 图形界面：

Schema

Database Objects

Tables

[Indexes](#)

[Views](#)

[Synonyms](#)

[Sequences](#)

[Database Links](#)

[Directory Objects](#)

[Reorganize Objects](#)

Programs

[Packages](#)

[Package Bodies](#)

[Procedures](#)

[Functions](#)

[Triggers](#)

Database Login

* Username

* Password

Database **PROD**

* Connect As

☐ Save as Preferred Credential

Tables

Object Type

Search

Select an object type and optionally enter a schema name and an object name to filter the data that is displayed in your results set.

Schema

Object Name

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Select	Schema	Table Name	Tablespace	Partitioned	Rows	Last Analyzed
<input type="checkbox"/>	No search conducted					

Database Instance: PROD > Tables >

Logged in As HR

Create Table: Table Organization

Specifying the table organization tells Oracle how to store this table in memory. The first step in creating a table is deciding which organization should be used.

- ☒ Standard, Heap Organized
- ☐ Temporary
- ☐ Index Organized Table (IOT)

Create Table

General Constraints Storage Options Partitions

* Name

Schema

Tablespace

Organization **Standard, Heap Organized**

Define Using

Columns

Insert Column:

Select	Name	Data Type	Size	Scale	Not NULL	Default Value
<input checked="" type="checkbox"/>	<input type="text" value="AUTHOR"/>	<input type="text" value="VARCHAR2"/>	<input type="text" value="30"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
<input type="checkbox"/>	<input type="text" value="ARTICLE_NAME"/>	<input type="text" value="VARCHAR2"/>	<input type="text" value="50"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
<input type="checkbox"/>	<input type="text" value="ARTICLE_DATE"/>	<input type="text" value="DATE"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
<input type="checkbox"/>	<input type="text" value="ARTICLE_DATA"/>	<input type="text" value="CLOB"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
<input type="checkbox"/>	<input type="text"/>	<input type="text" value="VARCHAR2"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

☐ Indicates a Primary Key column

☒ Indicates a Unique Key column

Set Default LOB Attributes

Cancel OK

These properties will be used as the default for all LOB storage.

LOB Attributes

Segment Name <System Assigned>

Size of Data Accessed at One Time (CHUNK) 16 KB

Space Used for Maintaining Old Versions of the LOB (PCTVERSION)(%)

Cache NOCACHE

☐ Store LOB Data in the Row (ENABLE STORAGE IN ROW)

LOB Storage Attributes

☒ **TIP** Default values for storage parameters are acceptable in most cases. In some cases these options may be changed to improve object performance.

Tablespace

Name LOB_DATA Change

Extent Management Local

Segment Management Automatic

Allocation Type UNIFORM

Logging Yes

Extents

Initial Size 2 MB

Buffer Pool

Buffer Pool DEFAULT

Cancel OK

Update Message

Table HR.MAGAZINE_ARTICLES has been created successfully

Tables

Object Type Table

Search

Select an object type and optionally enter a schema name and an object name to filter the data that is displayed in your results set.

Schema HR

Object Name

Go

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode Single

Create

Select	Schema	Table Name	Tablespace	Partitioned	Rows	Last Analyzed
<input type="radio"/>	HR	COUNTRIES	USERS	NO	25	Mar 7, 2012 3:22:56 PM CST
<input type="radio"/>	HR	DEPARTMENTS	USERS	NO	27	Mar 7, 2012 3:22:56 PM CST
<input type="radio"/>	HR	EMPLOYEES	USERS	NO	107	Mar 7, 2012 3:22:56 PM CST
<input type="radio"/>	HR	JOBS	USERS	NO	19	Mar 7, 2012 3:22:57 PM CST
<input type="radio"/>	HR	JOB_HISTORY	USERS	NO	10	Mar 7, 2012 3:22:57 PM CST
<input type="radio"/>	HR	LOCATIONS	USERS	NO	23	Mar 7, 2012 3:22:57 PM CST
<input type="radio"/>	HR	MAGAZINE_ARTICLES	USERS	NO		

```
imp hr/hr file=exp_mag.dmp tables=MAGAZINE_ARTICLES ignore=y
```

For any row in the HR.MAGAZINE_ARTICLES table that contains three or more references to Oracle9i insert the corresponding rowid and a timestamp for the time that it was inserted into the ORACLE9I_REFERENCES table.

```
CREATE TABLE "HR"."ORACLE9I_REFERENCES"
(
  "ORACLE9I_ARTICLE" ROWID,
  "INSERT_TIME"    TIMESTAMP WITH LOCAL TIME ZONE
)
TABLESPACE "USERS";
```

```
insert into HR.ORACLE9I_REFERENCES
select ROWID,
SCN_TO_TIMESTAMP(ORA_ROWSCN)
from HR.MAGAZINE_ARTICLES where instr(AUTHOR||ARTICLE_NAME||ARTICLE_DATE||
ARTICLE_DATA, 'Oracle9i',1,3)>0;
```

INSTR(源字符串, 要查找的字符串, 从第几个字符开始, 要找到第几个匹配的序号)

例如:

```
SQL> select instr('Oracle9iOracle9iOracle9iOracle9i','Oracle9i',1,3) from dual;
```

```
INSTR('ORACLE9IORACLE9IORACLE9IORACLE9I','ORACLE9I',1,3)
```

17

返回的是第三个 Oracle9i 开始的位置

或者

```
select rowid,a_date from MAGAZINE_ARTICLES
```

```
where dbms_lob.substr(ARTICLE_DATA) like '%Oracle9i%Oracle9i%Oracle9i%';
```

5.Partitioning

创建分区表，参考联机文档 Administrator's Guide=>17 Managing Partitioned Tables and Indexes=>Creating Partitioned Tables

```
create tablespace data01 datafile '/u01/app/oracle/oradata/PROD/Disk1/DATA01.dbf'
size 250M
autoextend on next 10M
extent management local uniform size 4M
segment space management auto
blocksize 16k;
```

```
create tablespace data02 datafile '/u01/app/oracle/oradata/PROD/Disk2/DATA02.dbf'
size 250M
```

```
autoextend on next 10M
extent management local uniform size 4M
segment space management auto
blocksize 16k;
```

```
create tablespace data03 datafile '/u01/app/oracle/oradata/PROD/Disk3/DATA03.dbf'
size 250M
autoextend on next 10M
extent management local uniform size 4M
segment space management auto
blocksize 16k;
```

```
create tablespace data04 datafile '/u01/app/oracle/oradata/PROD/Disk4/DATA04.dbf'
size 250M
autoextend on next 10M
extent management local uniform size 4M
segment space management auto
blocksize 16k;
```

```
create tablespace data05 datafile '/u01/app/oracle/oradata/PROD/Disk5/DATA05.dbf'
size 250M
autoextend on next 10M
extent management local uniform size 4M
segment space management auto
blocksize 16k;
```

Create a partitioned table named SALES_HISTORY in the SH schema in the PROD database. Use the following specifications:

5.2.1 The column names and definitions will be the same as the OLTP_USER.SALES table

5.2.2 Partition the table into 5 different partitions on the SDATE column using the following specifications:

5.2.2.1 Partition P1 will contain data for 1998 and should be placed in the DATA01 tablespace.

5.2.2.2 Partition P2 will contain data for 1999 and should be placed in the DATA02 tablespace.

5.2.2.3 Partition P3 will contain data for 2000 and should be placed in the DATA03 tablespace.

5.2.2.4 Partition P4 will contain data for 2001 and should be placed in the DATA04 tablespace.

5.2.2.5 Partition P5 will contain data for 2002 and should be placed in the DATA05 tablespace.

答案：

```

CREATE TABLE sh.SALES_HISTORY
  PARTITION BY RANGE ( SDATE )
  ( PARTITION p1 VALUES LESS THAN (to_date('1999-01-01 00:00:00','yyyy-mm-dd
hh24:mi:ss'))
    TABLESPACE DATA01,
    PARTITION p2 VALUES LESS THAN (to_date('2000-01-01 00:00:00','yyyy-mm-
ddhh24:mi:ss' ))
    TABLESPACE DATA02,
    PARTITION p3 VALUES LESS THAN (to_date('2001-01-01 00:00:00','yyyy-mm-
ddhh24:mi:ss' ))
    TABLESPACE DATA03,
    PARTITION p4 VALUES LESS THAN (to_date('2002-01-01 00:00:00','yyyy-mm-
ddhh24:mi:ss' ))
    TABLESPACE DATA04,
    PARTITION p5 VALUES LESS THAN (to_date('2003-01-01 00:00:00','yyyy-mm-dd
hh24:mi:ss' ))
    TABLESPACE DATA05
  )as select * from OLTP_USER.SALES where 1=2;

```

```

@/home/oracle/scripts/populate_sales_hist.sql
select * from SALES_HISTORY partition (p1);

```

```

select * from SALES_HISTORY partition (p2);

```

```

select * from SALES_HISTORY partition (p3);

```

```

select * from SALES_HISTORY partition (p4);

```

```

select * from SALES_HISTORY partition (p5)

```

4. Create a unique index named SALES_HISTORY_PK in the SH schema of the PROD database on the SALES_HISTORY table. Partition the index into 4 partitions with each partition containing approximately the same amount of entries. Use the following specifications:

- 4.1 Create the index to include the SALES_ID column.
- 4.2 Create the index in the INDX tablespace.
- 4.3 Create the index with parallelism degree 4.

global index 能减少索引的争用

Create Index

[Execute On Multiple Databases](#) [Show SQL](#) [Cancel](#) [OK](#)

General

[Storage](#) [Options](#) [Partitions](#) [Statistics](#)* Name Schema Tablespace [Estimate Index Size](#)Index Type ☒ Standard - B-tree ☐ Bitmap

Indexed Table Object


Index On ☒ Table ☐ Cluster* Table Name [Populate Columns](#) **TIP** The indexed columns and their orders are indicated by the Order field

Table Columns

Column Name	Data Type	Sorting Order	Order
INVC_ID	NUMBER	ASC	
ORDER_ID	NUMBER	ASC	1
PROD_ID	NUMBER	ASC	
CUST_ID	NUMBER	ASC	
TIME_ID	DATE	ASC	
SDATE	DATE	ASC	
CHANNEL_ID	NUMBER	ASC	
PROMO_ID	NUMBER	ASC	
QUANTITY_SOLD	NUMBER	ASC	
AMOUNT_SOLD	NUMBER	ASC	

[Add Column Expression](#)

General

[Storage](#) [Options](#) [Partitions](#) [Statistics](#)

Create Index

[Execute On Multiple Databases](#) [Show SQL](#) [Cancel](#) [OK](#)

General

[Storage](#) [Options](#) [Partitions](#) [Statistics](#)

Index Options

☒ **Unique** - Specify that the value of the columns upon which the index is based must be unique.☐ **Reverse** - Store the bytes of the index block in reverse order, excluding the row id.☒ **Parallel** - Use this option to change the default degree of parallelism for queries and DML on the index.Parallel Degree ☐ Default ☒ Value ☐ **Compress** - Eliminate repeated occurrence of key column values, may substantially reduce storage.Prefix Length

Execution Options

☐ **Online** - DML operations on the table will be allowed during creation/rebuilding of the index.☐ **Compute Statistics** - Collect statistics at relatively little cost during the creation/rebuilding of the index.☐ **No Sort** - Rows will not be sorted when creating index since they are already in ascending order.

General

[Storage](#) [Options](#) [Partitions](#) [Statistics](#)[Execute On Multiple Databases](#) [Show SQL](#) [Cancel](#) [OK](#)

http://rac2.localdomain:4889/em/console/database/schema/index?target=PROD&type=oracle_database&canc

Oracle Enterprise Manager (SYST... Oracle Enterprise Manager

Database Instance: PROD > Tables > Logged in As SH

Create Index

Execute On Multiple Databases Show SQL Cancel OK

General Storage Options **Partitions** Statistics

☒ **Enable Partitioning**
Partitions can be used to improve the availability and manageability of an index.

Partitioning Method: Global-Hash

Partitioning Method: Global-Hash

When using hash-partitioned global indexes, each index partition contains values determined by Oracle's hash function. Hash-partitioned global indexes provide improved performance and manageability over non-partitioned indexes. They also offer improved performance over range-partitioned global indexes for OLTP applications when most index insertions occur at the right edge of the index.

Partitioning Columns

The values of these columns will be hashed to determine the index partitioning key.

Column Name	Data Type	Order
ORDER_ID	NUMBER	1

Remove

☒ **TIP** This column list must be a prefixed subset of the index column list, so columns can only be added and removed in order.

Partition Definitions

Number of Partitions: 4 Set Tablespaces (round-robin distribution): INDX
(example: tablespace1, tablespace2, tablespace3) Add

Advanced Options Insert Before Delete Delete All Partitions

Select	Partition Name	Tablespace
<input checked="" type="radio"/>	SALES_HISTORY_PK_P1	INDX
<input type="radio"/>	SALES_HISTORY_PK_P2	INDX
<input type="radio"/>	SALES_HISTORY_PK_P3	INDX
<input type="radio"/>	SALES_HISTORY_PK_P4	INDX

Add Another Partition

☒ **TIP** When using hash partitioning, the number of partitions should be a power of 2 so that the data is evenly spread across the partitions.

General Storage Options **Partitions** Statistics

最后，Show SQL 检查下

Database Instance: PROD > Tables > Logged in As SH

Show SQL

Execute On Multiple Databases Return

```
CREATE UNIQUE INDEX "SH"."SALES_HISTORY_PK" ON "SH"."SALES_HISTORY" ("ORDER_ID") PARALLEL 4 TABLESPACE "INDX" GLOBAL PARTITION
BY HASH ("ORDER_ID") (PARTITION "SALES_HISTORY_PK_P1" TABLESPACE "INDX", PARTITION "SALES_HISTORY_PK_P2" TABLESPACE "INDX",
PARTITION "SALES_HISTORY_PK_P3" TABLESPACE "INDX", PARTITION "SALES_HISTORY_PK_P4" TABLESPACE "INDX")
```

Execute On Multiple Databases Return

方法二：用 Sqlplus 命令行做

CREATE unique INDEX sales_index_pk ON sales_history (SALES_ID)

GLOBAL PARTITION BY HASH (SALES_ID)

(PARTITION p1 tablespace indx,

PARTITION p2 tablespace indx,

PARTITION p3 tablespace indx,

PARTITION p4 tablespace indx)

PARALLEL 4;

如果表当中有包含重复的值，则创建失败，需要把违反约束的行删除

1、@?/rdbms/admin/utlexpt1.sql

2、找出重复的行

```
SQL> alter table sales_history
```

```
add constraints c2_sales_uk unique (c2)
```

```
exceptions into exceptions;
```

3、删除重复的行

```
SQL> delete sales_history
```

```
where rowid in (select row_id from exceptions);
```

或者直接删除：

```
delete sales_history a
```

```
where rowid<(select max(rowid) from sales_history b where a.id=b.id)
```

Create an index named SALES_HISTORY_DATE_IDX in the SH schema on the SALES_HISTORY table. Use the following specifications:

Create the index on the SDATE column

Partition the index into 5 partitions that are based on the SDATE column

Query the data in the SALES_HISTORY table to validate the use of each of the indexes.

```
CREATE INDEX SALES_HISTORY_DATE_IDX ON sales (sdate) local;
```

```
set autot traceonly
```

```
select * from SH.SALES_HISTORY where ORDER_ID=1;
```

no rows selected

Execution Plan

Plan hash value: 480139289

Id	Operation	Name	Rows	Bytes
	Cost (%CPU) Time	Pstart Pstop		

```

| 0 | SELECT STATEMENT | | | 1 | 122 |
| 2 | (0)| 00:00:01 | | |
| 1 | PARTITION HASH SINGLE | | | 1 | 122 |
| 2 | (0)| 00:00:01 | 4 | 4 |
| 2 | TABLE ACCESS BY GLOBAL INDEX ROWID| SALES_HISTORY | 1 | 122 |
| 2 | (0)| 00:00:01 | ROWID | ROWID |
|* 3 | INDEX UNIQUE SCAN | SALES_HISTORY_PK | 1 |
| 1 | (0)| 00:00:01 | 4 | 4 |

```

Predicate Information (identified by operation id):

```

3 - access("ORDER_ID">=1)

```

Statistics

```

1 recursive calls
0 db block gets
1 consistent gets
0 physical reads
0 redo size
802 bytes sent via SQL*Net to client
374 bytes received via SQL*Net from client
1 SQL*Net roundtrips to/from client
0 sorts (memory)
0 sorts (disk)
0 rows processed

```

EXPLAIN PLAN for select * from hr.SALES_HISTORY where sdate=sysdate;
@ ?/rdbms/admin/utlxplp.sql

Truncate partition P1 of the SALES_HISTORY and ensure all indexes are available for use both during and after the truncate command completes.

```

alter table sh.sales_history truncate partition p1 update global indexes;

```

6.Fine-Grained Auditing

1. Set up FGA on the SALARY and COMMISSION_PCT columns of the EMPLOYEES table in the HR schema of the PROD database. An audit record should be created if either of these two columns are selected as part of the output of a query or are used in the where condition of a select statement and their values are not null.
2. Validate that the FGA is taking place by executing statements that should result in auditing records being created and by executing statements where no auditing records will be generated. Do not delete your audit records.

参考联机文档：Security Guide—> 12 Configuring and Administering Auditing —> Policies in Fine-Grained Auditing

PL/SQL Packages and Types Reference—>搜索 DBMS_FGA—>ADD_POLICY Procedure
begin

```

DBMS_FGA.ADD_POLICY(
OBJECT_SCHEMA=>'HR',
OBJECT_NAME=>'EMPLOYEES',

```

```

POLICY_NAME=>'SALARY',
AUDIT_CONDITION=>'SALARY IS NOT NULL AND COMMISSION_PCT IS NOT NULL',
AUDIT_COLUMN=>'SALARY,COMMISSION_PCT',
STATEMENT_TYPES=>'SELECT',
audit_trail => DBMS_FGA.DB_EXTENDED,
audit_column_opts=> DBMS_FGA.ALL_COLUMNS
);
end;
/

```

验证:

```

conn hr/oracle
Connected.

```

```

select * from employees where SALARY is not null and COMMISSION_PCT is not null
35 rows selected.

```

```

conn / as sysdba
Connected.

```

```

PROD..>select
AUDIT_TYPE,OBJECT_SCHEMA,OBJECT_NAME,POLICY_NAME,ACTION,SQL_TEXT from
dba_common_audit_trail;

```

AUDIT_TYPE	OBJEC	OBJECT_NAM	POLICY	ACTION	SQL_TEXT

Fine Grained Audit	HR	EMPLOYEES	SALARY	select *	from employees where
				SALARY is not null and COMMISS	
				ION_PCT is not null	

```

PROD..>select OBJECT_SCHEMA,OBJECT_NAME,POLICY_NAME,SQL_TEXT from
dba_fga_audit_trail;

```

OBJEC	OBJECT_NAM	POLICY	SQL_TEXT

HR	EMPLOYEES	SALARY	select * from employees where
			SALARY is not null and COMMISS
			ION_PCT is not null

```

PROD..>select LSQLTEXT,OBJ$SCHEMA,OBJ$NAME ,POLICYNAME from SYS.FGA_LOG$;

```

LSQLTEXT	OBJ\$S	OBJ\$NAME	POLICY

```

select * from employees where SALARY is not null and COMMISSION_PCT is not null

```

另外：

```

audit_trail=>DBMS_FGA.XML + DBMS_FGA.EXTENDED
audit_trail      =>  DBMS_FGA.DB + DBMS_FGA.EXTENDED

```

Setting audit_trail to DBMS_FGA.DB + DBMS_FGA.EXTENDED sends the audit trail to the SYS.FGA_LOG\$ table in the database and includes SQL Text and SQL Bind.

Setting audit_trail to DBMS_FGA.XML writes the audit trail in XML files sent to the operating system and omits SQL Text and SQL Bind.

Setting audit_trail to DBMS_FGA.XML + DBMS_FGA.EXTENDED writes the audit trail in XML files sent to the operating system and includes SQL Text and SQL Bind.

甚至可以设成如下：

audit_trail=>DBMS_FGA.XML + DBMS_FGA.EXTENDED + DBMS_FGA.DB,

7.Flashback

1. Create a table named ORIGINAL_SALARY in the HR schema in the PROD database that includes the employee_id, commission_pct and salaries of all records in the HR.EMPLOYEES table.Note the date and time and then commit the changes.
2. Delete all employees in the HR.ORIGINAL_SALARY table whose HIRE_DATE is before 1994.Commit the chages.Create a view HR.SALARY_VIEW that will show all the original rows before the deletion.

```
SQL> create table hr.ORIGINAL_SALARY as select employee_id,commission_pct,salary from
hr.EMPLOYEES;
Table created.
```

```
SQL> col sys_scn new_value scn
SQL> select current_scn sys_scn from v$database;
```

```
SYS_SCN
-----
      348289
```

```
SQL> commit;
Commit complete.
```

```
SQL> delete from hr.ORIGINAL_SALARY where EMPLOYEE_ID in (select EMPLOYEE_ID from
hr.EMPLOYEES where HIRE_DATE<to_date(1994,'yyyy'));
6 rows deleted.
```

```
SQL> commit;
Commit complete.
```

```
SQL> create view HR.SALARY_VIEW as select * from hr.ORIGINAL_SALARY as of scn
&sys_scn;
```

Enter value for sys_scn: 348289

```
old 1: create view HR.SALARY_VIEW as select * from hr.ORIGINAL_SALARY as of scn
&sys_scn
```

```
new 1: create view HR.SALARY_VIEW as select * from hr.ORIGINAL_SALARY as of scn
348289
```

View created.

验证：

```
SQL> select count(*) from hr.ORIGINAL_SALARY;
```

```
COUNT(*)
```

```
-----
101
```

```
SQL> select count(*) from hr.SALARY_VIEW;
```

```
COUNT(*)
```

```
-----
107
```

闪回已删除的表：

```
SQL> drop table hr.ORIGINAL_SALARY;
```

Table dropped.

```
SQL> select OWNER,OBJECT_NAME,ORIGINAL_NAME from dba_recyclebin where
ORIGINAL_NAME='ORIGINAL_SALARY';
```

```
OWNER
```

```
-----
OBJECT_NAME
```

```
-----
ORIGINAL_NAME
```

```
-----
HR
```

```
BIN$uwQZgZEec9jgQKjADAodPA==$0
```

```
ORIGINAL_SALARY
```

```
SQL> flashback table HR."BIN$uwQZgZEec9jgQKjADAodPA==$0" to before drop rename to
ORIGINAL_SALARY2;
```

Flashback complete.

验证：

```
SQL> SELECT COUNT(*) FROM ORIGINAL_SALARY2;
```

```
COUNT(*)
```

```
-----
```

```
101
```

```
select versions_starttime, versions_endtime, versions_xid,versions_operation  
from ORIGINAL_SALARY versions between timestamp minvalue and maxvalue  
order by versions_starttime;
```

```
select count(*) from ORIGINAL_SALARY versions between timestamp  
to_date('2011-09-24 00:04:00','YYYY-MM-DD HH24:MI:SS')  
and to_date('2011-09-24 00:04:50','YYYY-MM-DD HH24:MI:SS')
```

```
alter system set recyclebin=on;
```

先查询被删除的表中的内容：

```
select * from "BIN$sZ3YK5wF8njgQKjAXxIcbA==$0";
```

```
flashback table d1 to before drop rename to xxx;
```

```
flashback table "BIN$f8ddM/q7ydvqQMCAAQAhJg==$0" to before drop;
```