Cognizant | Academy

Business Focused. Learner Centric.

Oracle
Version 12c

Schema Objects

Enabling Objectives

After completing this chapter, in the next 90 minutes you will be able to :

- Create, Alter, Rename, and Drop at least one views in Oracle.
- Create and Drop at least one index on table in Oracle.
- Create one synonym in Oracle.
- Create at least one sequence on a table in Oracle.

Key Topics

- Using Views.
- Create index.
- Create synonym.
- Create sequence

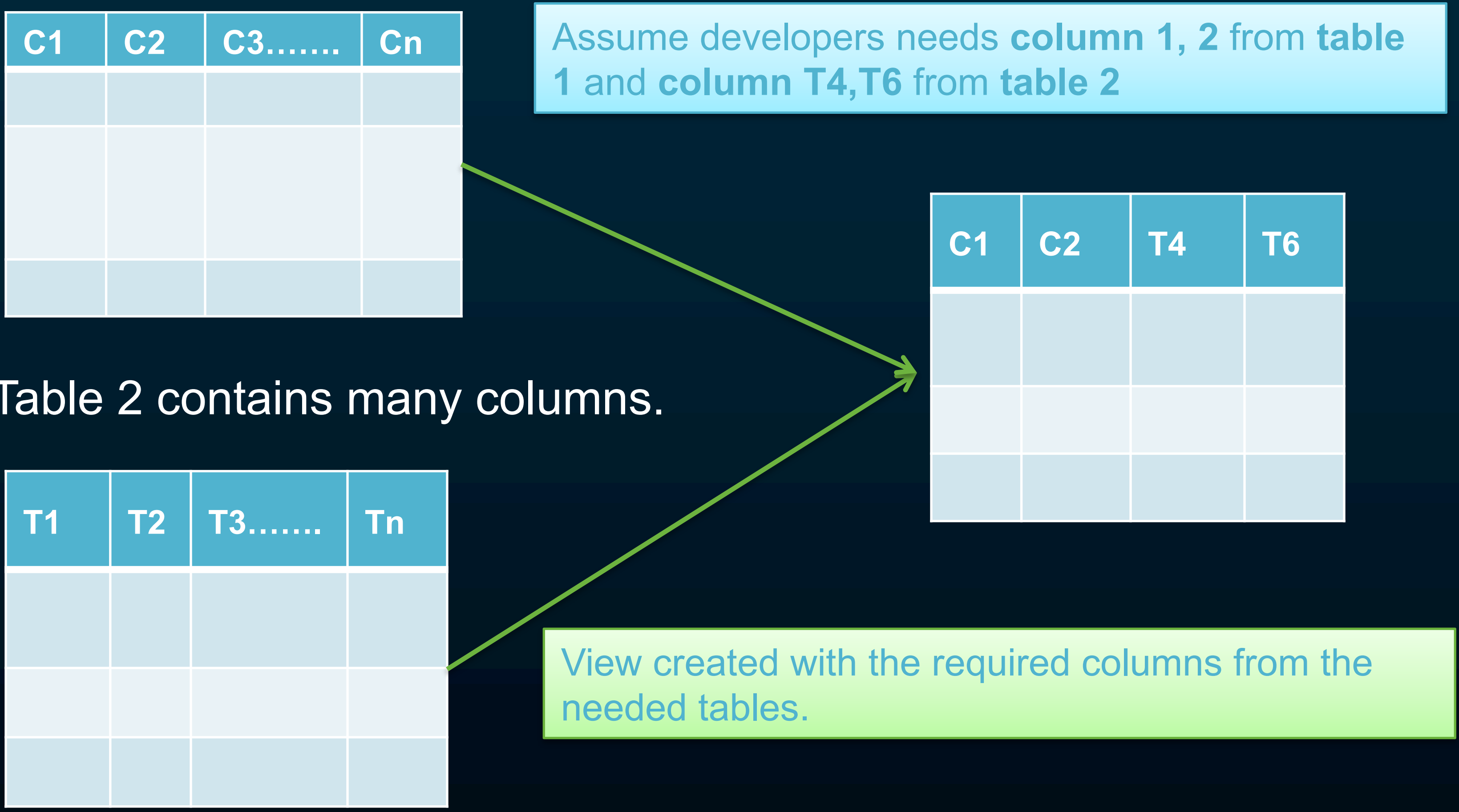
Views

View

- A view is a logical table based on one or more tables or another view.
- A view holds data's of one or more columns from selected tables (or) views.
- A view contains no data of its own but is like a window
  - through which data from tables can be viewed or changed.
- The tables on which a view is based are called base tables.
- The view is stored as a SELECT statement in the data dictionary.

Views illustration

- Table 1 contains many columns.



C1	C2	C3…….	Cn

Assume developers needs **column 1, 2** from **table 1** and **column T4,T6** from **table 2**

- Table 2 contains many columns.

T1	T2	T3…….	Tn

C1	C2	T4	T6

View created with the required columns from the needed tables.

View Syntax

Syntax of a view.

```
CREATE [OR REPLACE] [FORCE|NOFORCE] VIEW
view_name
[  (alias)  [,alias]…)   ]
AS select statement
[WITH CHECK OPTION [CONSTRAINT constraint_name]]
[WITH READ ONLY [CONSTRAINT constraint_name]];
```

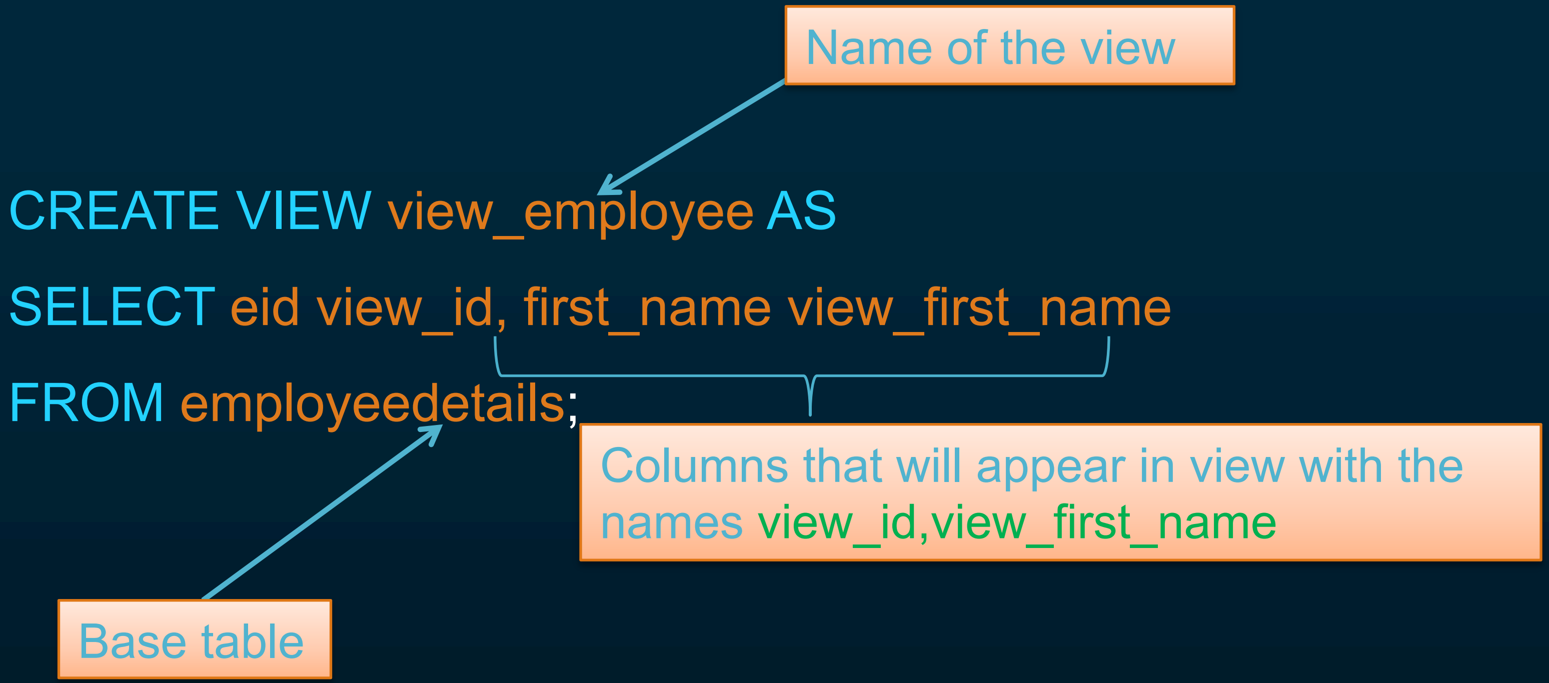
View Example

- Let us assume that we have a table employeedetails with the following data.

EID	FIRST_NAME	LAST_NAME	SALARY
01	Jason	Martin	12000
02	Alison	Mathews	15000
…….	……..	……..	…..
07	David	Larry	17000
08	James	Stevenson	21000

- Let us create view from this table which contains only employee id and first name of employees.

View Example



Name of the view

```
CREATE VIEW view_employee AS
SELECT eid view_id, first_name view_first_name
FROM employeedetails;
```

Columns that will appear in view with the names view_id,view_first_name

Base table

Retrieving from View

- To retrieve from a view

```
select * from view_name;
```

- Let us select from the view which we have created earlier.

```
select * from view_employee;
```

VIEW_ID	VIEW_FIRST_NAME
01	Jason
02	Alison
……	…….
07	David
08	James
VIEW_ID	VIEW_FIRST_NAME

With Check Option

```
CREATE VIEW view_employee AS
SELECT eid view_id, first_name view_first_name
FROM employeedetails WHERE eid>3 WITH CHECK OPTION
```

- Assume that a record is inserted into the view as shown below

```
INSERT INTO view_employee VALUES(2,'chan')
```

- While executing the statement oracle will throw an error because, the value for eid does not conform to the definition eid>3.

Read Only Clause

- This clause is used to create read only view where we cannot apply DML operations on it.

```
CREATE VIEW view_employee AS
SELECT eid view_id, first_name view_first_name
FROM employeedetails WHERE eid >3 WITH READ ONLY
```

- Assume that a record is inserted into the view as shown below

```
INSERT INTO view_employee VALUES(2,'chan')
```

- If you try to insert or delete or update a record in this view oracle will throw an error.

Alter, Rename, Drop Views

- Altering a view.

```
ALTER VIEW view_name COMPILE;
```

- Renaming a View:

```
RENAME view_name1 TO view_name2
```

- Dropping a View

```
DROP VIEW view_name.
```

Index

Index

- It is a schema object used by the oracle server to speed up the retrieval of rows.
- Indexes are optional structures associated with tables and clusters.
- You can create indexes on one or more columns of a table.
- Can reduce disk I/O by using a rapid path access method to locate data quickly.
- Is used and maintained automatically by the Oracle server.

Index

- Assume in the employee table we have 1.5 million records
- Developer develops a query to retrieve employee with Name “Dave” and assume Dave is the 1 Million’th record.
- The process will fully scan all the 1 million record before it hits the Jack record.
- The process of retrieving records is slow.

Index

Employee Name	Designation	Age	Salary	Department	PF Amount
Tim	President	32	53000	Sales	1000
Jack	Clerk	39	17000	Sales	200
....	....	......	......	......	....
Jim	SW Developer	24	16000	IT	
Dave	Manager	32	22000	IT	350

Index

- The retrieval process works with the employee name Indexed.
- The process will now directly retrieve the Jack record instead of fully scanning the table.

Employee Name	Designation	Age	Salary	Department	PF Amount
Tim	President	32	53000	Sales	1000
Jack	Clerk	39	17000	Sales	200
....	....	....	....	....	....
Jim	SW Developer	24	16000	IT	
Dave	Manager	32	22000	IT	350

Create Index

- Syntax.

```
CREATE INDEX index_name  ON table (column1[,column2]…);
```

- Data retrieval process will be fast only if the SQL's '*where*' condition is executed on the indexed column.

```
CREATE INDEX  lname_idx  ON employeedetails (last_name);
```

Drop Index

Syntax.

```
DROP INDEX index_name;
```

- To drop an index, you must be the owner of the index or have the DROP ANY INDEX privilege..

Example:

```
DROP INDEX lname_idx;
```

Synonym

Synonym

- A synonym is an alternative (nick) name given for objects such as tables, views, sequences, stored procedures, and other database objects.

Syntax:

```
CREATE [OR REPLACE] [PUBLIC] SYNONYM synonym_name
FOR object;
```

Example:

```
CREATE PUBLIC SYNONYM  employeeSynonym FOR
employeedetails;
```

Synonym

We can use synonyms in DML statements like

- SELECT
- INSERT
- UPDATE
- DELETE

- We can also refer synonyms in the following DDL statements like

- GRANT, REVOKE

Example

```
SELECT * FROM employeeSynonym;
```

Sequence

Sequence

Sequence is a schema object used for creating unique numbers automatically

- It is a can be shared by multiple users.
- A sequence number can be associated to a column of a table.
- It is typically used to generate value for primary key.

Sequence

Syntax

```
CREATE SEQUENCE sequence_name
[INCREMENT BY n]
[START WITH n]
[{MAXVALUE n | NOMAXVALUE}]
[{MINVALUE n | NOMINVALUE}]
[{CYCLE | NOCYCLE }]
[{CACHE n | NOCACHE];
```

Sequence

Example

- Consider the student id generation example. The sequence starts with the value 100 and gets incremented by 2.

The sequence generation,

```
CREATE SEQUENCE student_id_seq  INCREMENT BY 2
START WITH 100 MAXVALUE 10000 NO CACHE NO CYCLE;
```

Alter Sequence

- ALTER SEQUENCE command alters a sequence definition

Example

```
alter sequence student_id_seq
 increment by 10
 maxvalue 20000;
```

Result: The sequence is altered, next value generated will have a differential increment of 10 with maxvalue extended to 20000.

Alter Sequence

- DROP SEQUENCE command is used to delete the sequence definition from the database.

Example:

```
DROP SEQUENCE student_id_seq;
```

Practice Check 1

- Let us create a view which displays course details like course code, course name and course duration only whose course_code is greater than 2.

- Problem #1: Create view with columns course_code, course_name,course_duration from the table course_info

Practice Check 2

- Let us assume that we have table called book_order_id _<employee id>with columns book_id primary key, book name varchar2 , address varchar 2.

- Problem #1: Create a sequence named *book_id_seq* with the start value 1001 which gets incremented by 1 and can generate up to a maximum of ten thousand numbers.

- Problem #2: Insert a record into the book table using this sequence.

- Problem #3: Print the current sequence value of book id.

RECAP

In this chapter we have learnt how to:

- Create, Alter, Rename, and Drop at least one views in Oracle.
- Create and Drop at least one index on table in Oracle.
- Create one synonym in Oracle.
- Create at least one sequence on a table in Oracle.

You have successfully completed -

Schema Objects