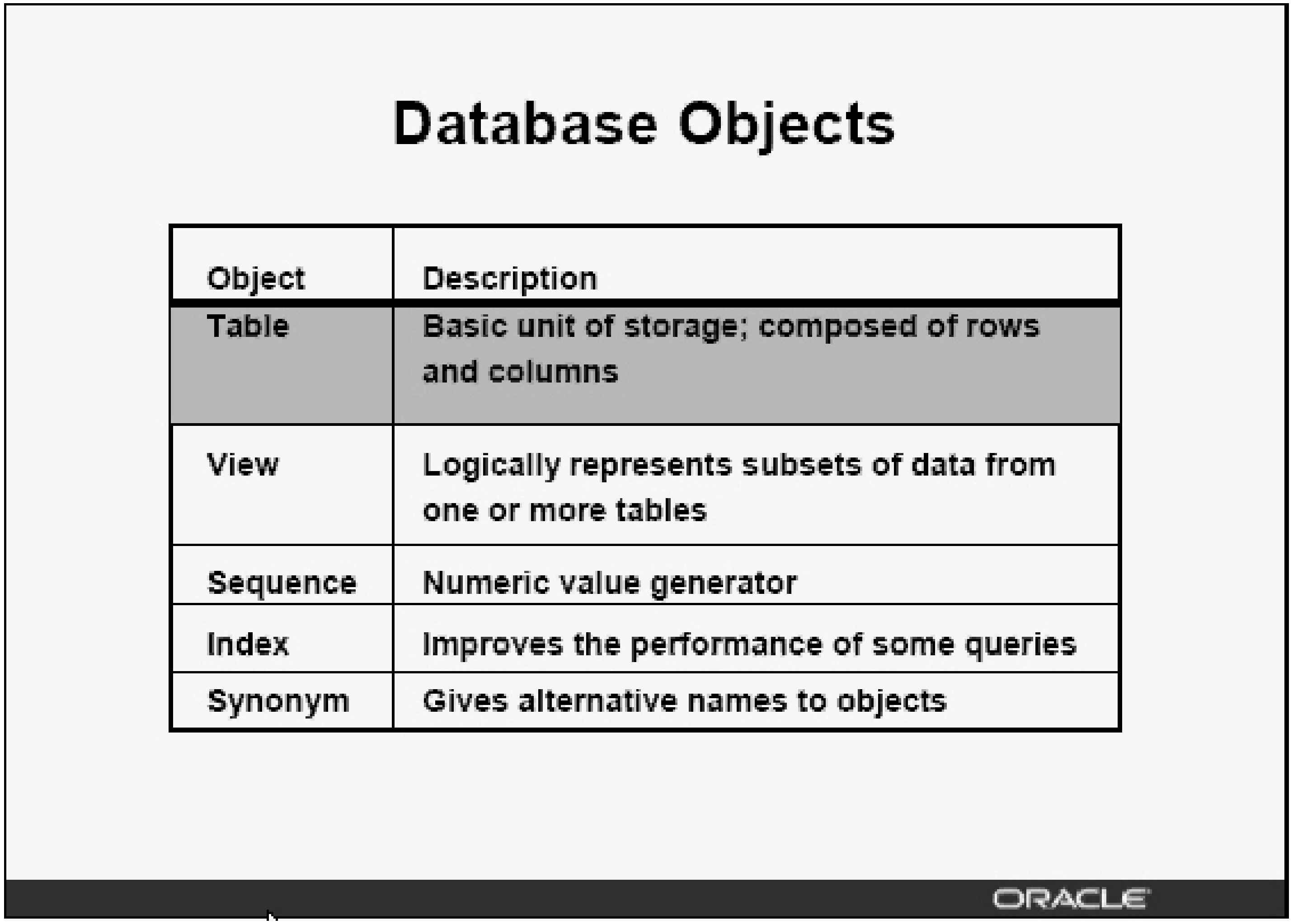
ORACLE®

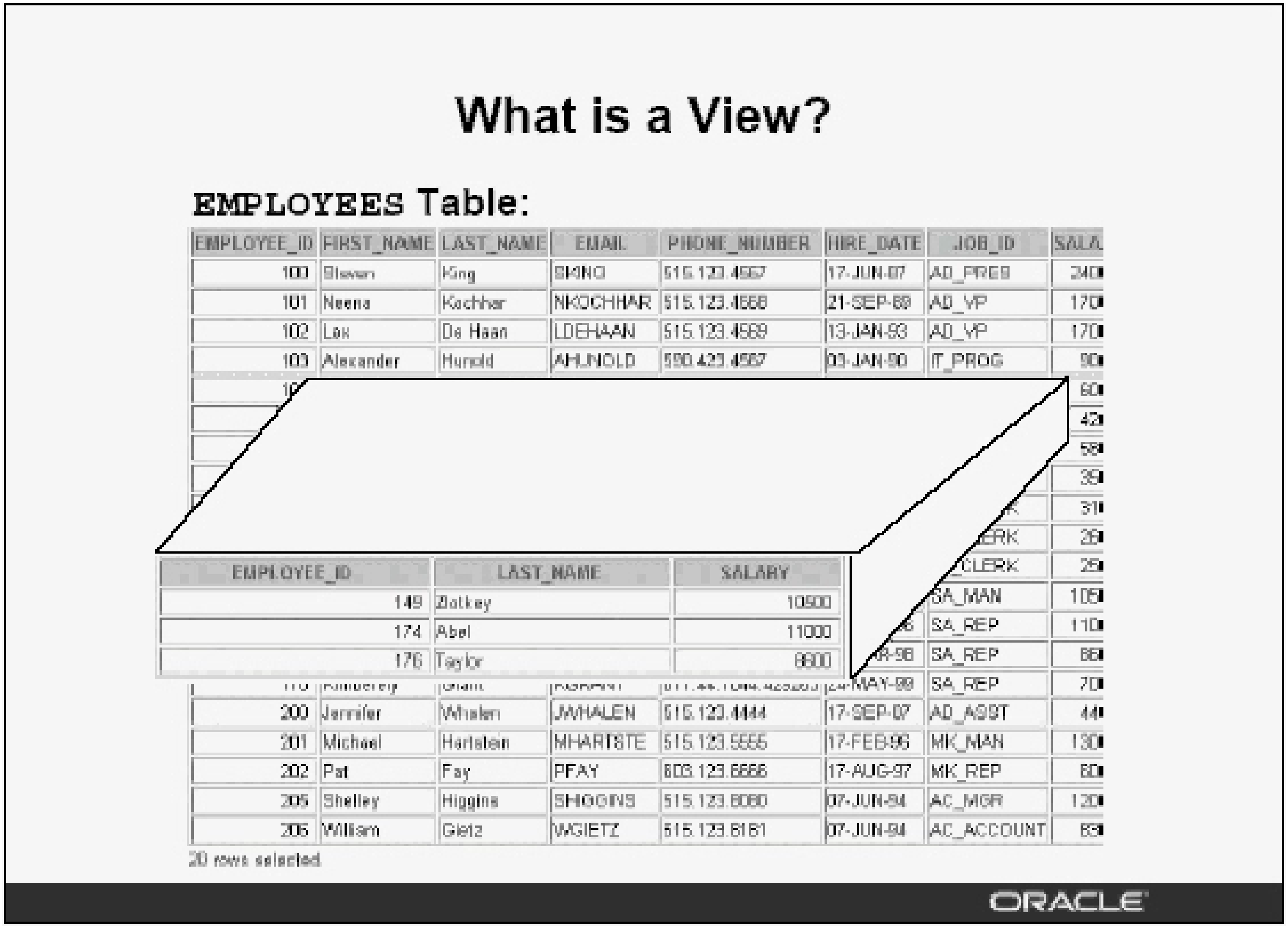
Database Objects



Database Objects

Object	Description
Table	Basic unit of storage; composed of rows and columns
View	Logically represents subsets of data from one or more tables
Sequence	Numeric value generator
Index	Improves the performance of some queries
Synonym	Gives alternative names to objects

Views



What is a View?

EMPLOYEES Table:

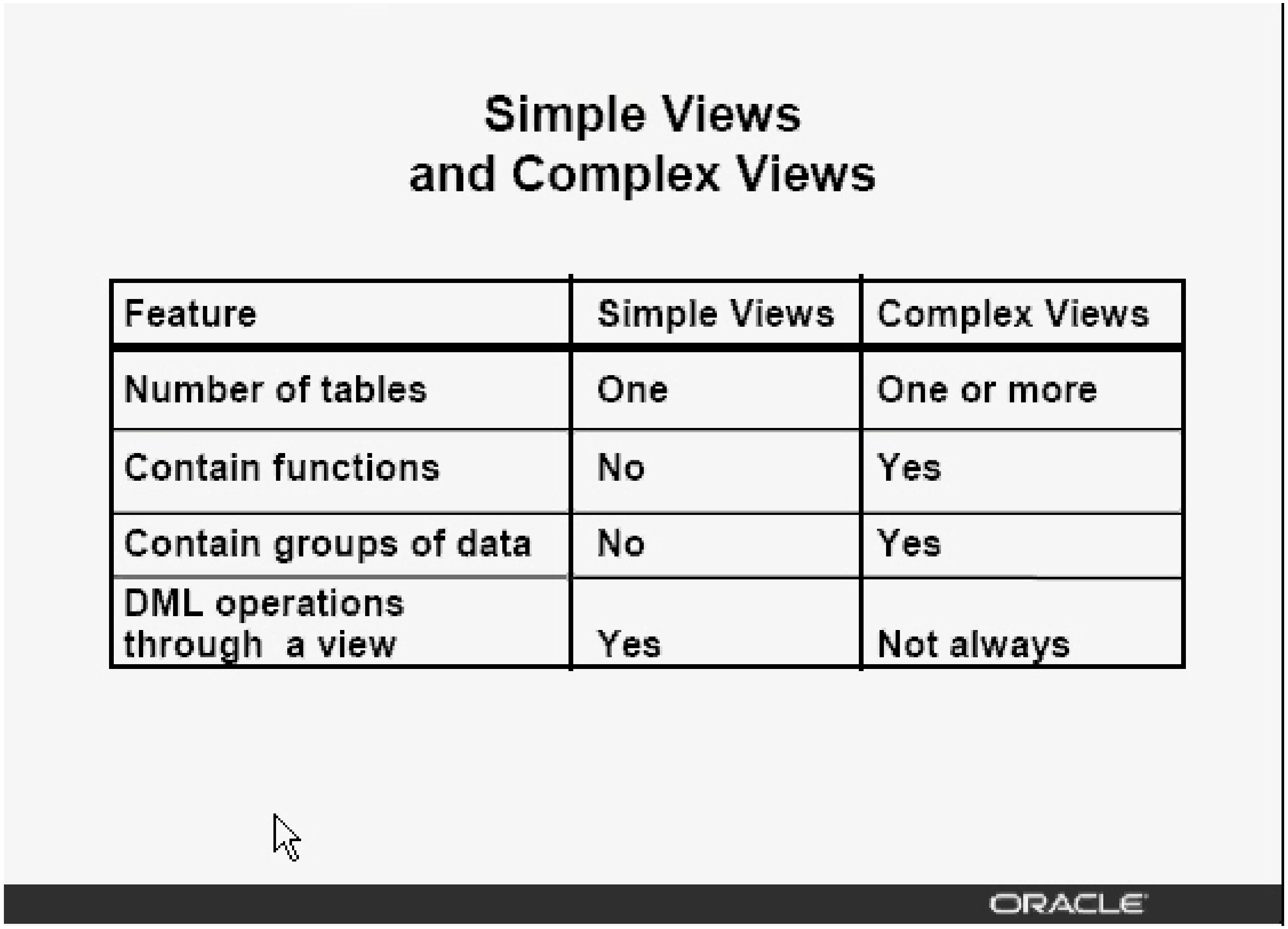
EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALA
100	Steven	King	SKING	515.123.4567	17-JUN-87	AD_PRES	240
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-89	AD_VP	170
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-93	AD_VP	170
103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-90	IT_PROG	90
10	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	60
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	42
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	58
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	35
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	K	31
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	ERK	26
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	_CLERK	25
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	SA_MAN	105
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	6	SA_REP	110
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	R-98	SA_REP	86
178	Kimberely	Grant	KGRANT	011.44.1644.429263	24-MAY-99	SA_REP	70
200	Jennifer	Whalen	JWHALEN	515.123.4444	17-SEP-87	AD_ASST	44
201	Michael	Hartstein	MHARTSTE	515.123.5555	17-FEB-96	MK_MAN	130
202	Pat	Fay	PFAY	603.123.6666	17-AUG-97	MK_REP	60
205	Shelley	Higgins	SHIGGINS	515.123.8080	07-JUN-94	AC_MGR	120
206	William	Gietz	WGIETZ	515.123.8181	07-JUN-94	AC_ACCOUNT	83

20 rows selected.

EMPLOYEE_ID	LAST_NAME	SALARY
149	Zlotkey	10500
174	Abel	11000
176	Taylor	8600

ORACLE

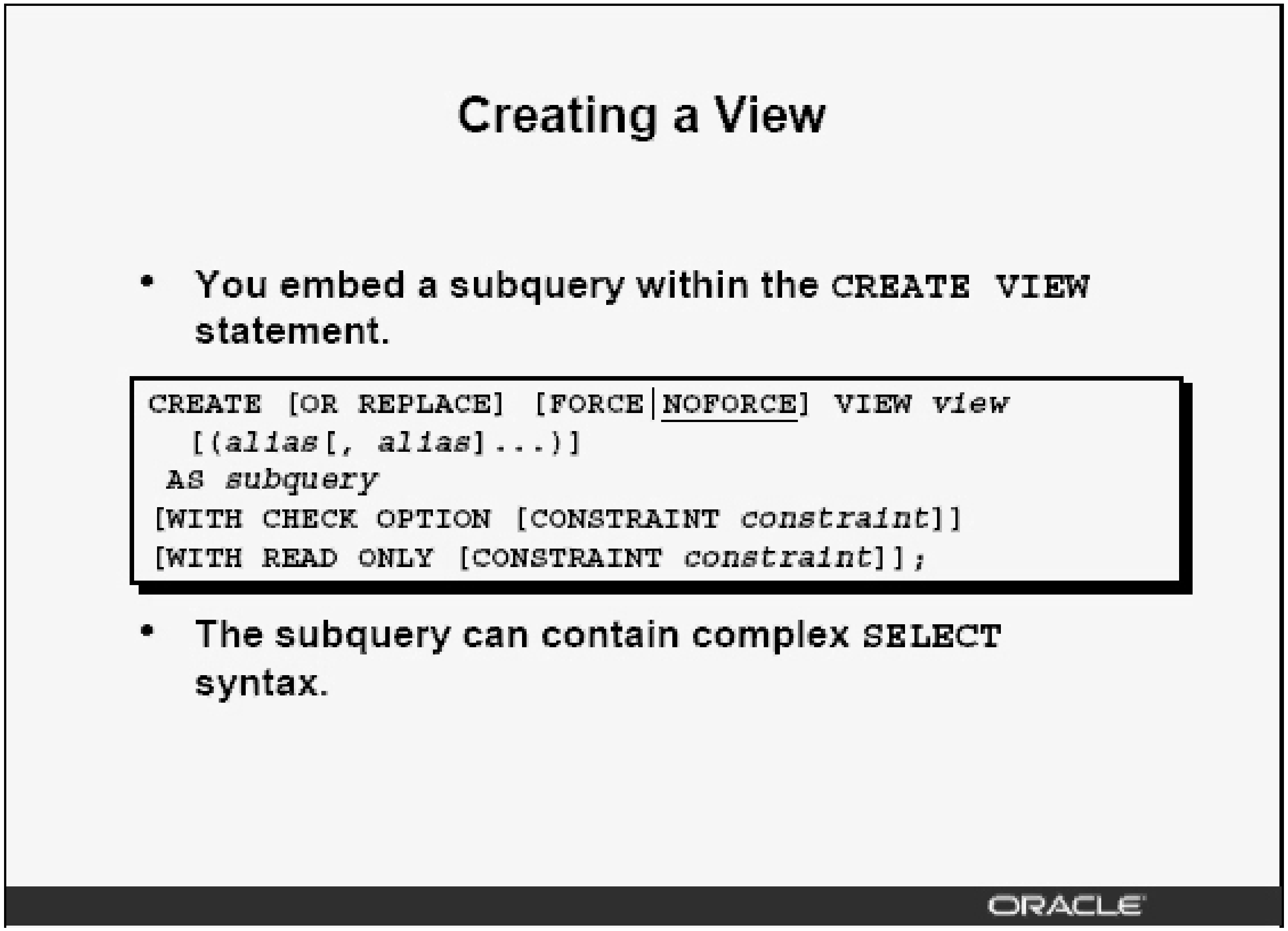
Views



Simple Views and Complex Views

Feature	Simple Views	Complex Views
Number of tables	One	One or more
Contain functions	No	Yes
Contain groups of data	No	Yes
DML operations through a view	Yes	Not always

ORACLE



Creating a View

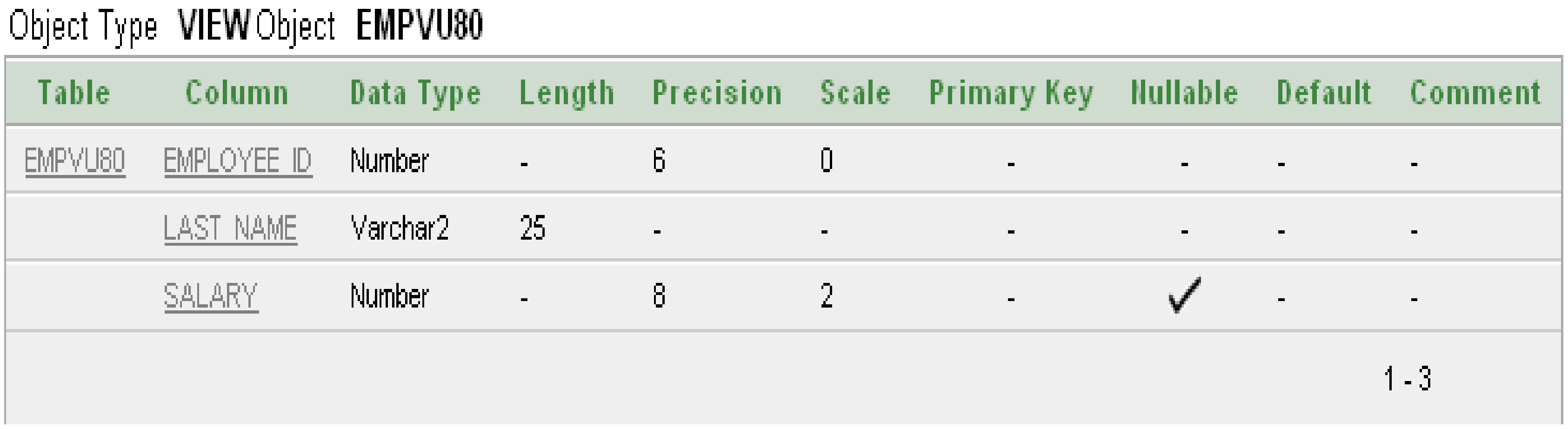
- You embed a subquery within the `CREATE VIEW` statement.

```
CREATE [OR REPLACE] [FORCE|NOFORCE] VIEW view
  [(alias[, alias]...)]
 AS subquery
[WITH CHECK OPTION [CONSTRAINT constraint]]
[WITH READ ONLY [CONSTRAINT constraint]];
```

- The subquery can contain complex `SELECT` syntax.

Views

```
CREATE VIEW empvu80
AS
SELECT employee_id,last_name,salary
FROM employees
WHERE department_id=80

DESC empvu80
```



Object Type **VIEW** Object **EMPVU80**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPVU80	EMPLOYEE_ID	Number	-	6	0	-	-	-	-
	LAST_NAME	Varchar2	25	-	-	-	-	-	-
	SALARY	Number	-	8	2	-	✓	-	-

1 - 3

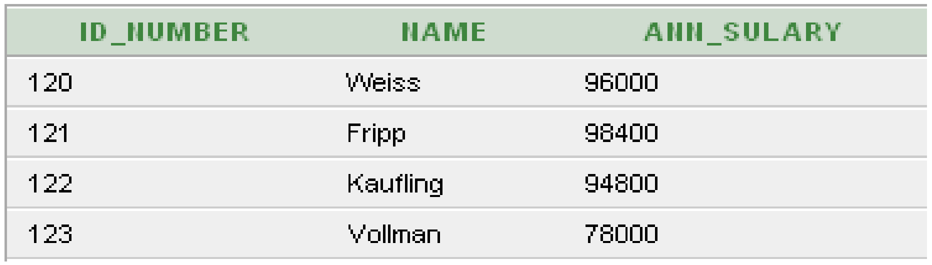
Views

```
CREATE VIEW salvu50
AS
SELECT employee_id id_number,last_name name,salary * 12
ann_sulary
FROM employees
WHERE department_id=50
```

```
CREATE VIEW salvu5 (id_number,name,ann_sulary)
AS
SELECT employee_id,last_name,salary*12
FROM employees
WHERE department_id=50
```

Views

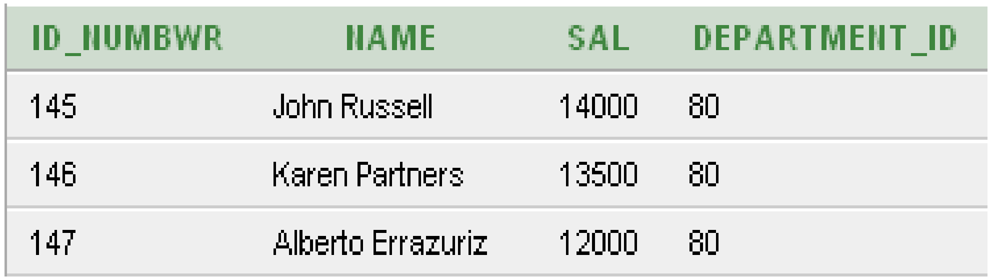
```
SELECT *
FROM  salvu5
```



ID_NUMBER	NAME	ANN_SULARY
120	Weiss	96000
121	Fripp	98400
122	Kaufling	94800
123	Vollman	78000

```
CREATE OR REPLACE VIEW empvu80
(id_number,name,sal,department_id)
AS
SELECT employee_id,first_name ||' '||last_name,salary,department_id
FROM employees
WHERE department_id=80
```

```
SELECT * FROM empvu80
```

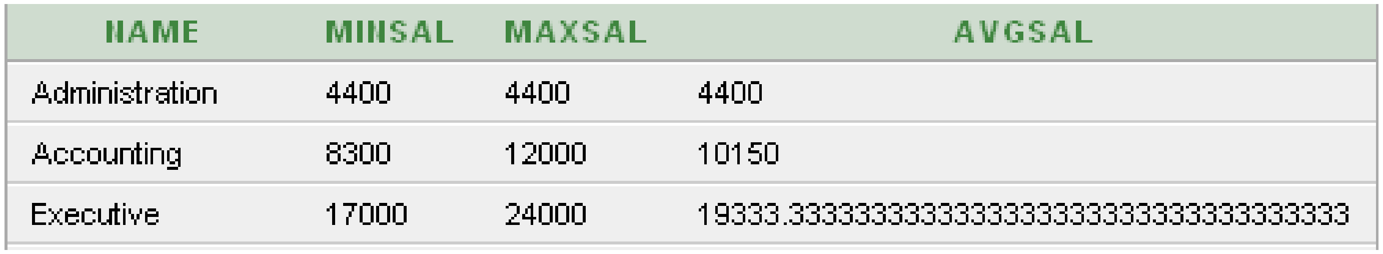


ID_NUMBWR	NAME	SAL	DEPARTMENT_ID
145	John Russell	14000	80
146	Karen Partners	13500	80
147	Alberto Errazuriz	12000	80

Views

```
CREATE VIEW dept_sum_vu
(name,minsal,maxsal,avgsal)
AS
SELECT
d.department_name,min(e.salary),max(e.salary),avg(e.salary)
FROM employees e, departments d
WHERE e.department_id=d.department_id
GROUP BY d.department_name
```

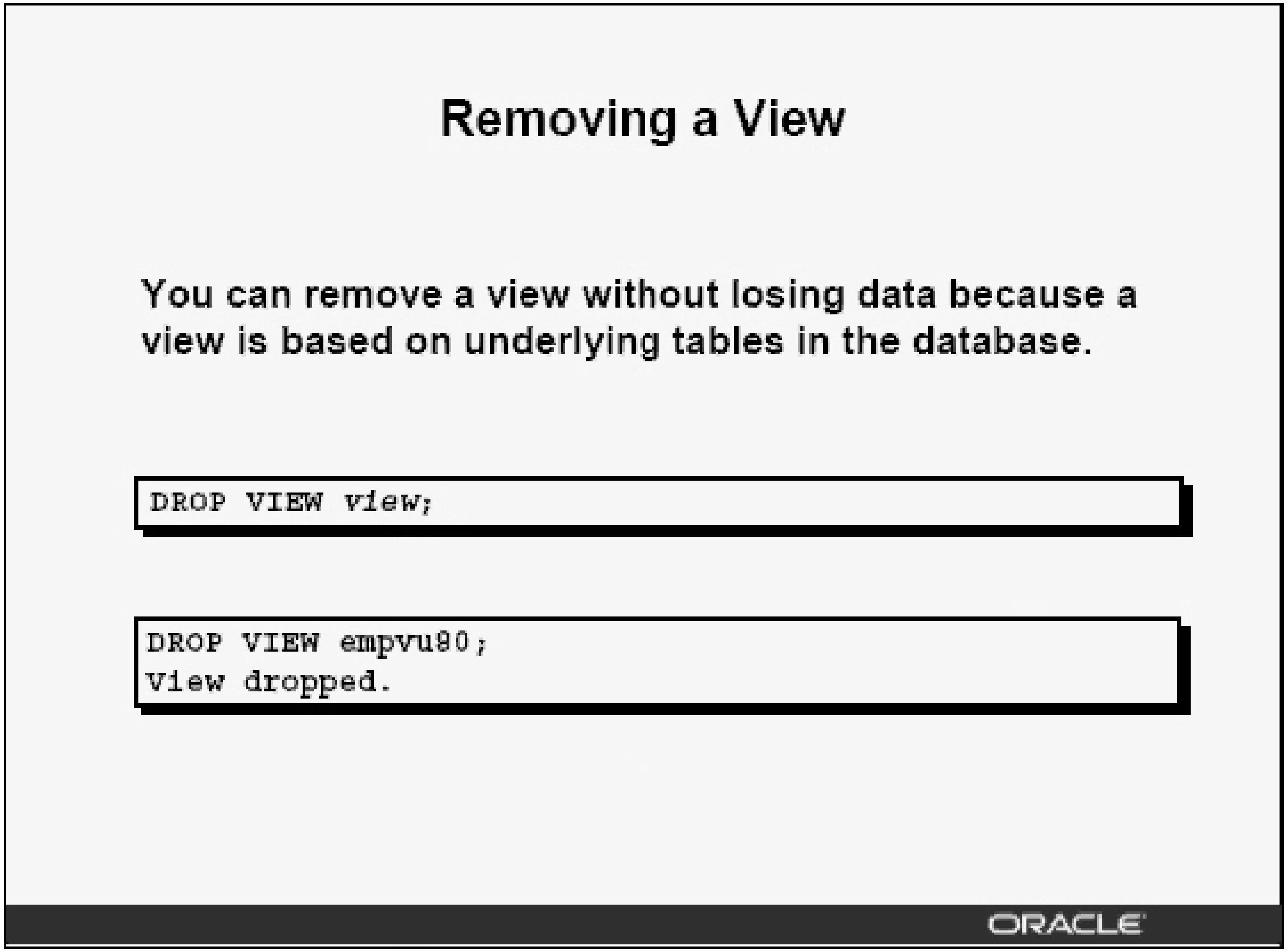
```
SELECT * FROM dept_sum_vu
```



NAME	MINSAL	MAXSAL	AVGSAL
Administration	4400	4400	4400
Accounting	8300	12000	10150
Executive	17000	24000	19333.3333333333333333333333333333333333

Views

```
CREATE OR REPLACE VIEW empvu20
AS SELECT * FROM employees
WHERE department_id=20
WITH CHECK OPTION CONSTRAINT empvu20_ck

UPDATE empvu20
SET employee_id=20
WHERE employee_id=201

CREATE OR REPLACE VIEW empvu10
(employee_number,employee_name,job_title)
AS SELECT employee_id,last_name,job_id FROM employees
WHERE department_id=10
WITH READ ONLY

DELETE FROM  empvu10
WHERE employee_number=200
```



Removing a View

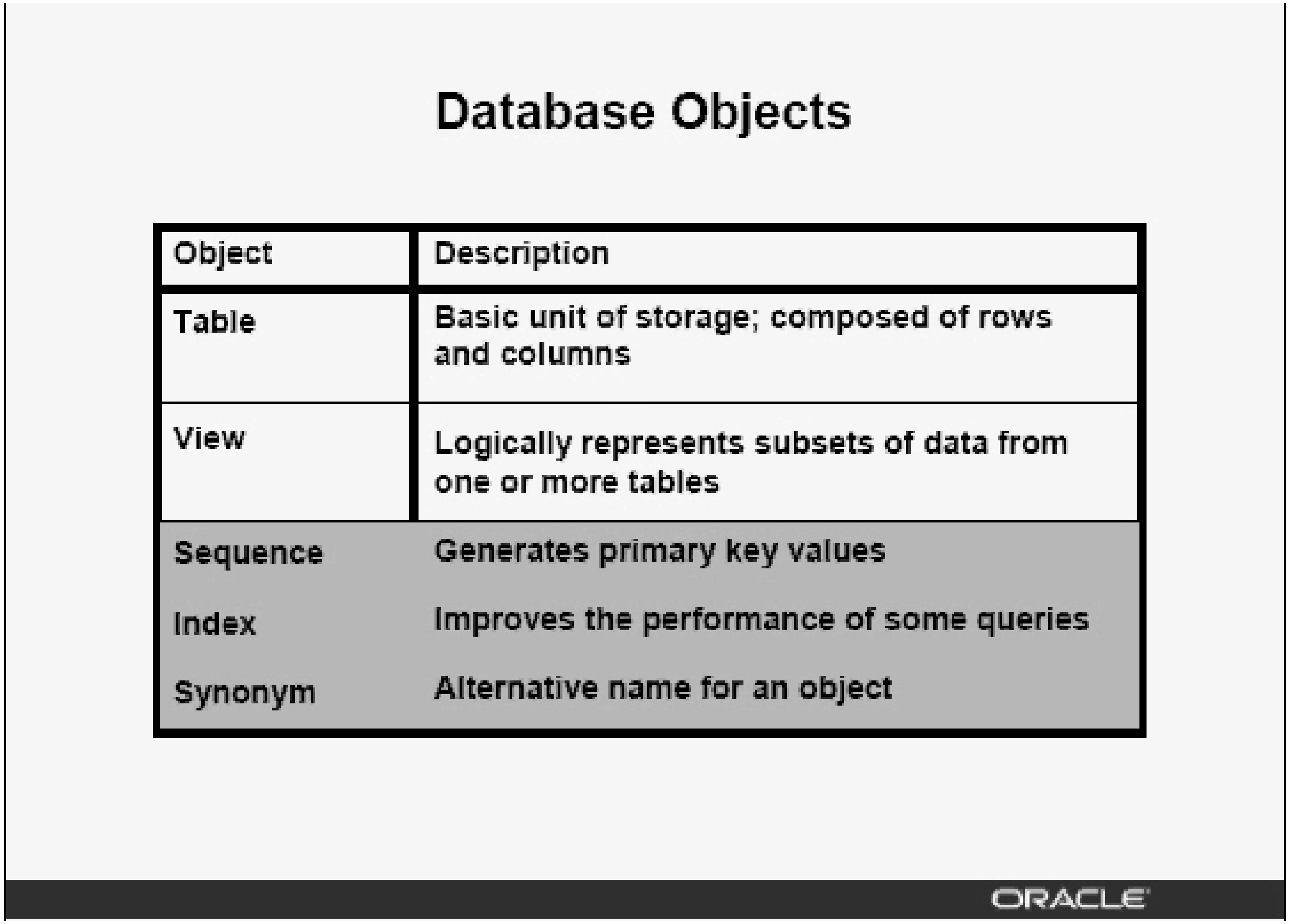
You can remove a view without losing data because a view is based on underlying tables in the database.

```
DROP VIEW view;
```

```
DROP VIEW empvu80;
View dropped.
```

ORACLE

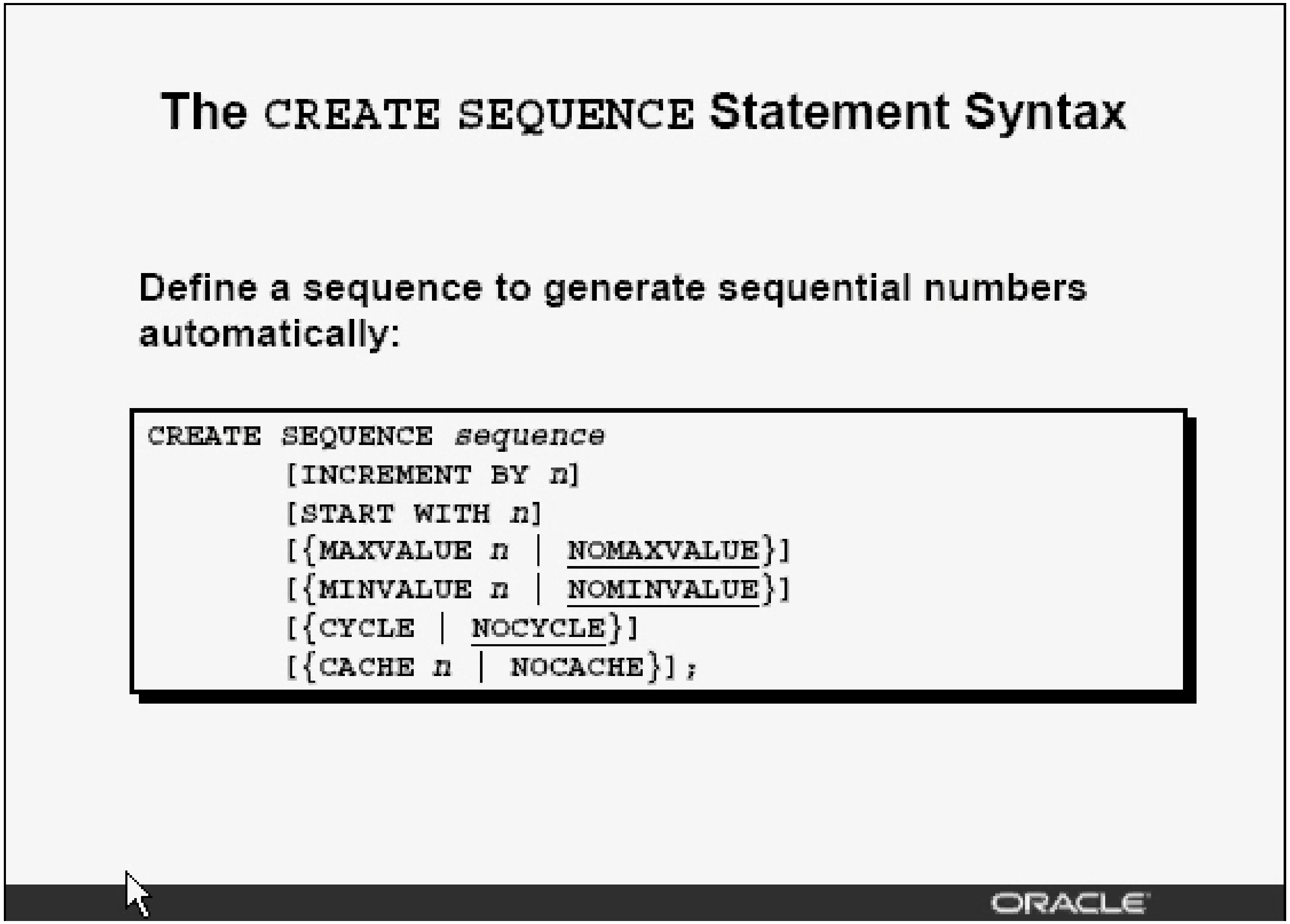
SEQUENCE-INDEX-SYNONYM



Database Objects

Object	Description
Table	Basic unit of storage; composed of rows and columns
View	Logically represents subsets of data from one or more tables
Sequence	Generates primary key values
Index	Improves the performance of some queries
Synonym	Alternative name for an object

SEQUENCE



The CREATE SEQUENCE Statement Syntax

Define a sequence to generate sequential numbers automatically:

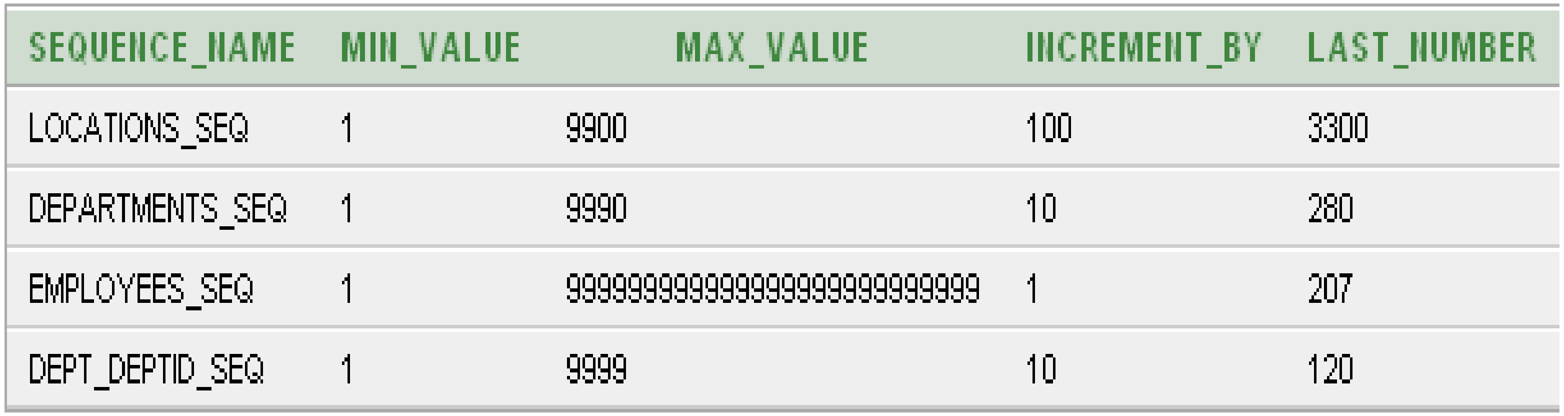
```
CREATE SEQUENCE sequence
        [INCREMENT BY n]
        [START WITH n]
        [{MAXVALUE n | NOMAXVALUE}]
        [{MINVALUE n | NOMINVALUE}]
        [{CYCLE | NOCYCLE}]
        [{CACHE n | NOCACHE}];
```

ORACLE

SEQUENCE

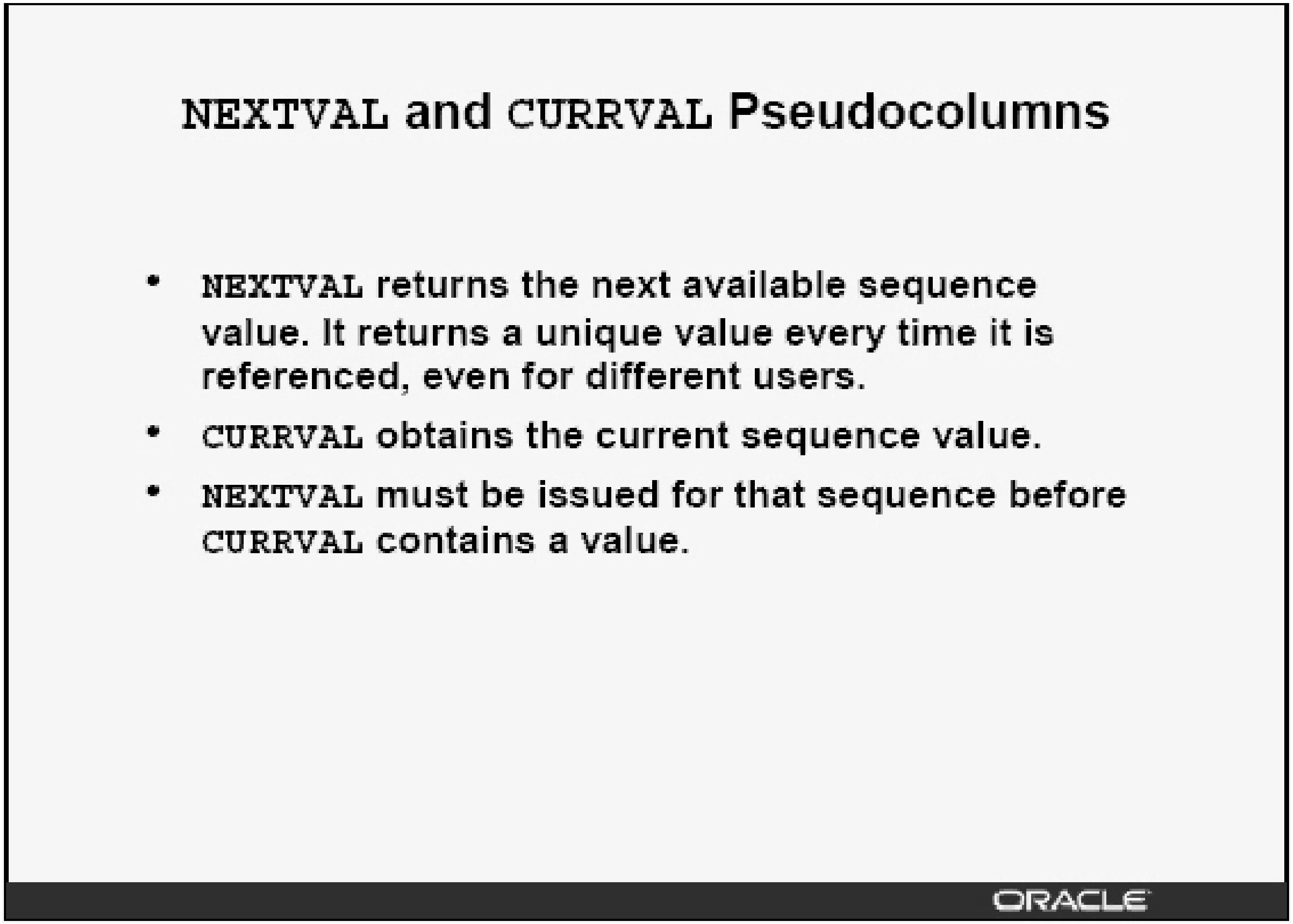
```
CREATE SEQUENCE a_num
INCREMENT BY 1
START WITH 1
MAXVALUE 9999
NOCACHE
NOCYCLE
```

```
SELECT
sequence_name,min_value,max_value,increment_by,last_number
FROM user_sequences
```



SEQUENCE_NAME	MIN_VALUE	MAX_VALUE	INCREMENT_BY	LAST_NUMBER
LOCATIONS_SEQ	1	9900	100	3300
DEPARTMENTS_SEQ	1	9990	10	280
EMPLOYEES_SEQ	1	999999999999999999999999999	1	207
DEPT_DEPTID_SEQ	1	9999	10	120

SEQUENCE

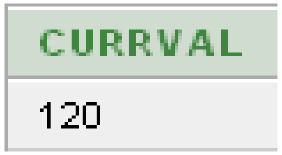


NEXTVAL and CURRVAL Pseudocolumns

- NEXTVAL returns the next available sequence value. It returns a unique value every time it is referenced, even for different users.
- CURRVAL obtains the current sequence value.
- NEXTVAL must be issued for that sequence before CURRVAL contains a value.

SEQUENCE

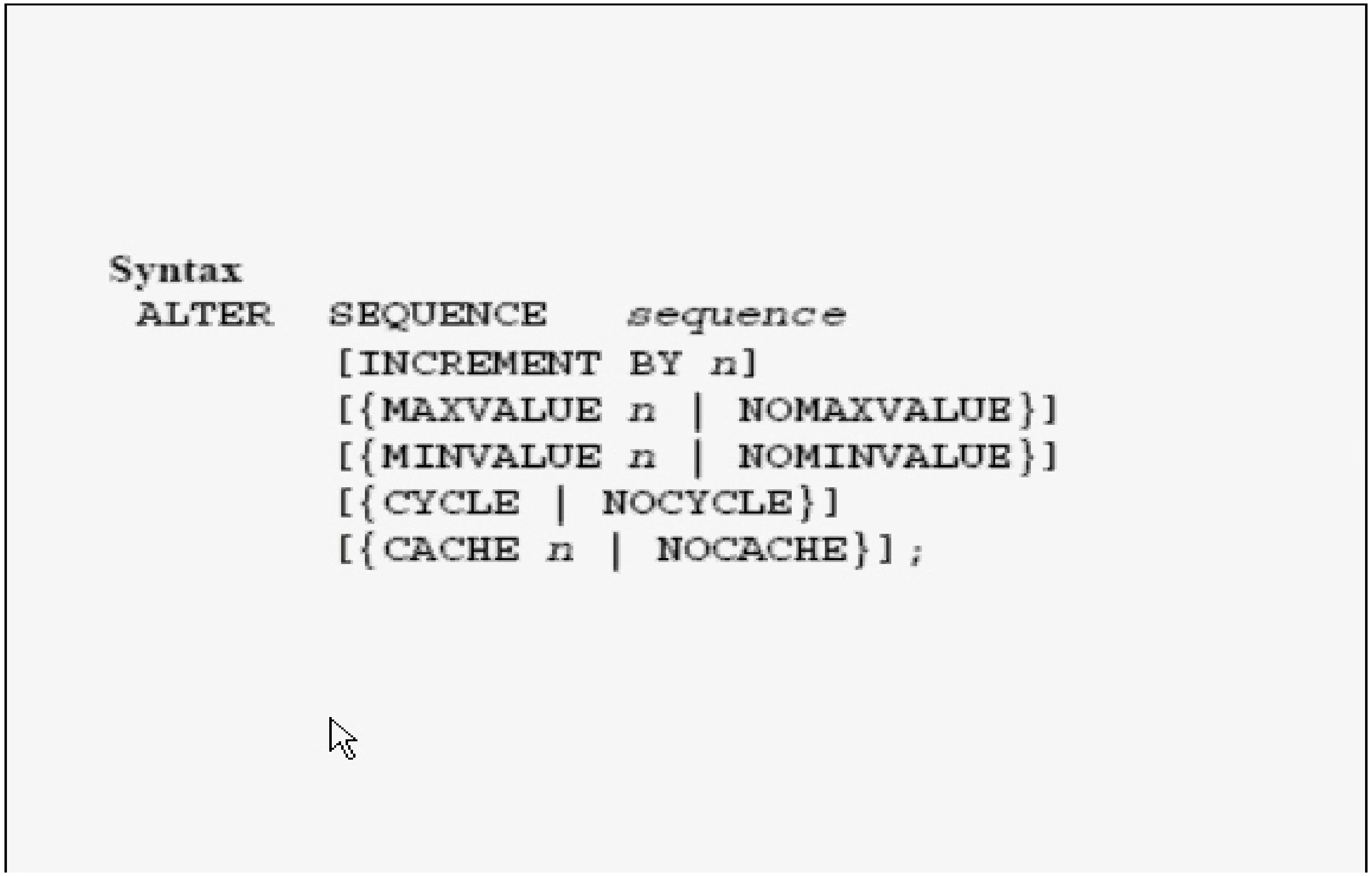
```
INSERT INTO departments
(department_id,department_name,location_id)
VALUES (dept_deptid_seq.nextval,'support',2500)

SELECT dept_deptid_seq.currval FROM dual
```



```
CURRVAL
120
```

```
INSERT INTO departments
(department_id,department_name,location_id)
VALUES (dept_deptid_seq.nextval,:dpt_name,:location)
```

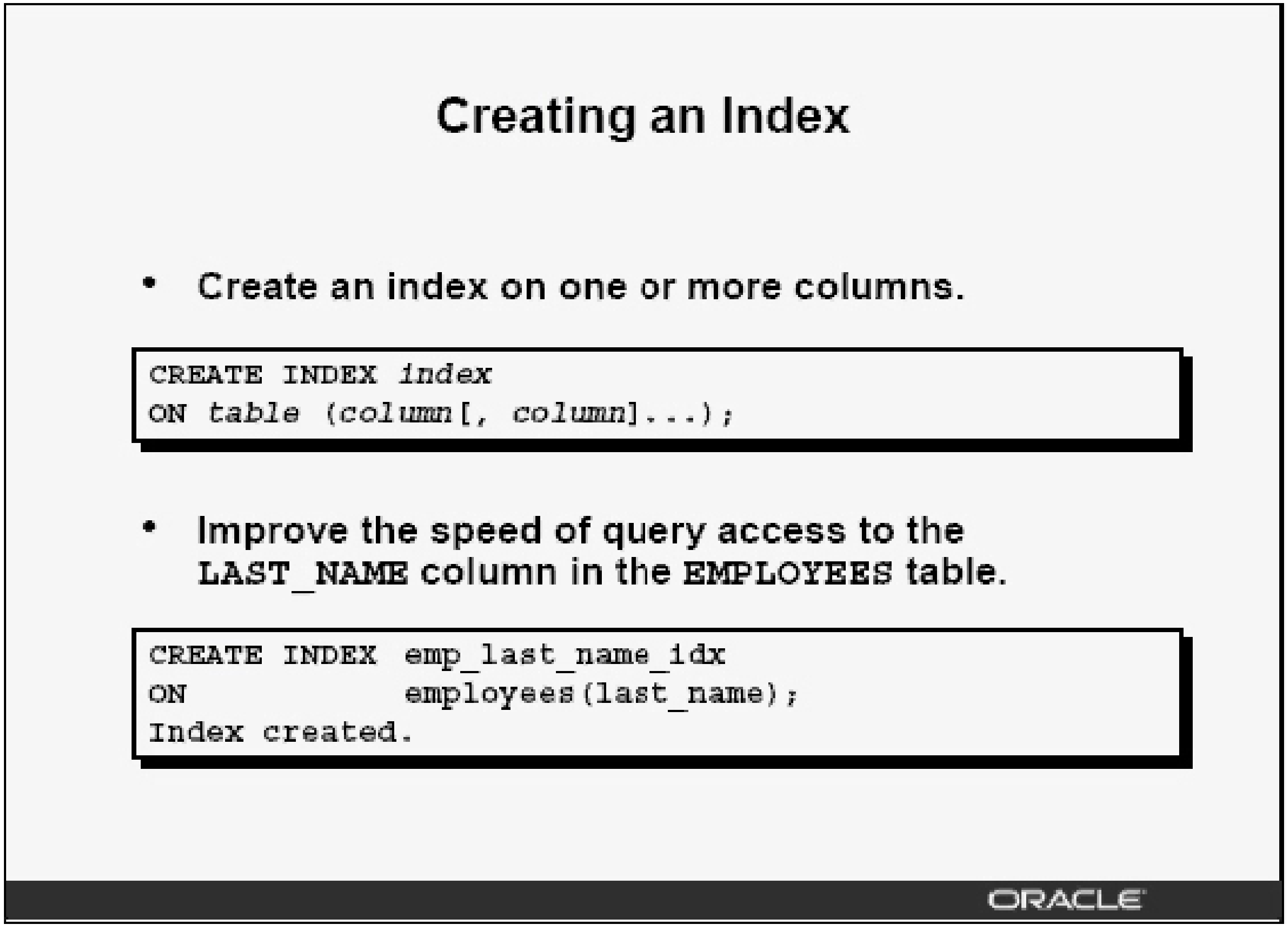
SEQUENCE


Syntax

```
ALTER  SEQUENCE   sequence
        [INCREMENT BY n]
        [{MAXVALUE n | NOMAXVALUE}]
        [{MINVALUE n | NOMINVALUE}]
        [{CYCLE | NOCYCLE}]
        [{CACHE n | NOCACHE}];
```

DROP SEQUENCE dept_deptid_seq

INDEX



Creating an Index

- Create an index on one or more columns.

```
CREATE INDEX index
ON table (column[, column]...);
```

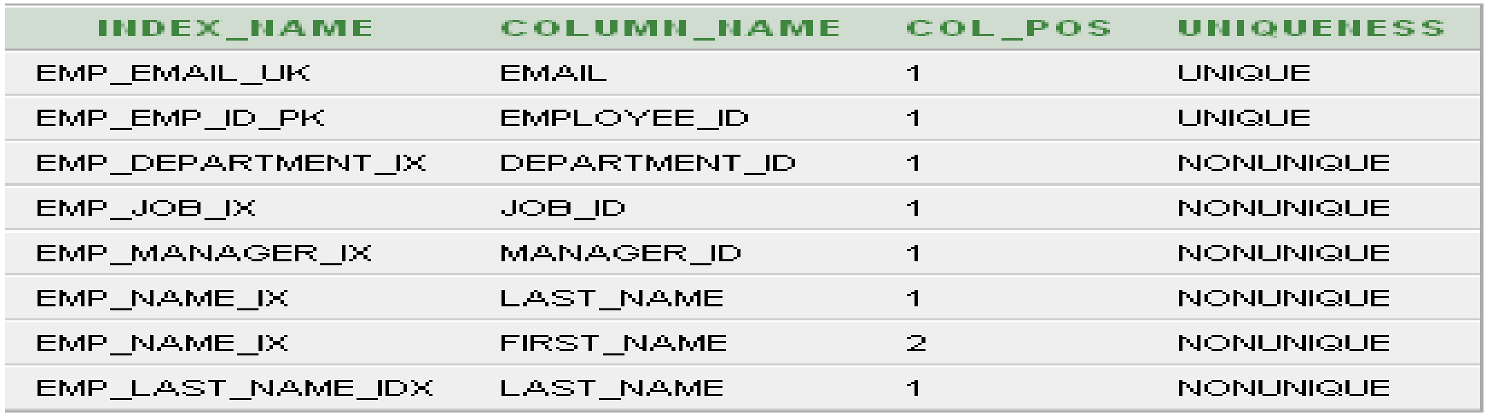
- Improve the speed of query access to the **LAST_NAME** column in the **EMPLOYEES** table.

```
CREATE INDEX  emp_last_name_idx
ON            employees(last_name);
Index created.
```

INDEX

```
CREATE INDEX emp_last_name_idx
ON employees (last_name)
```

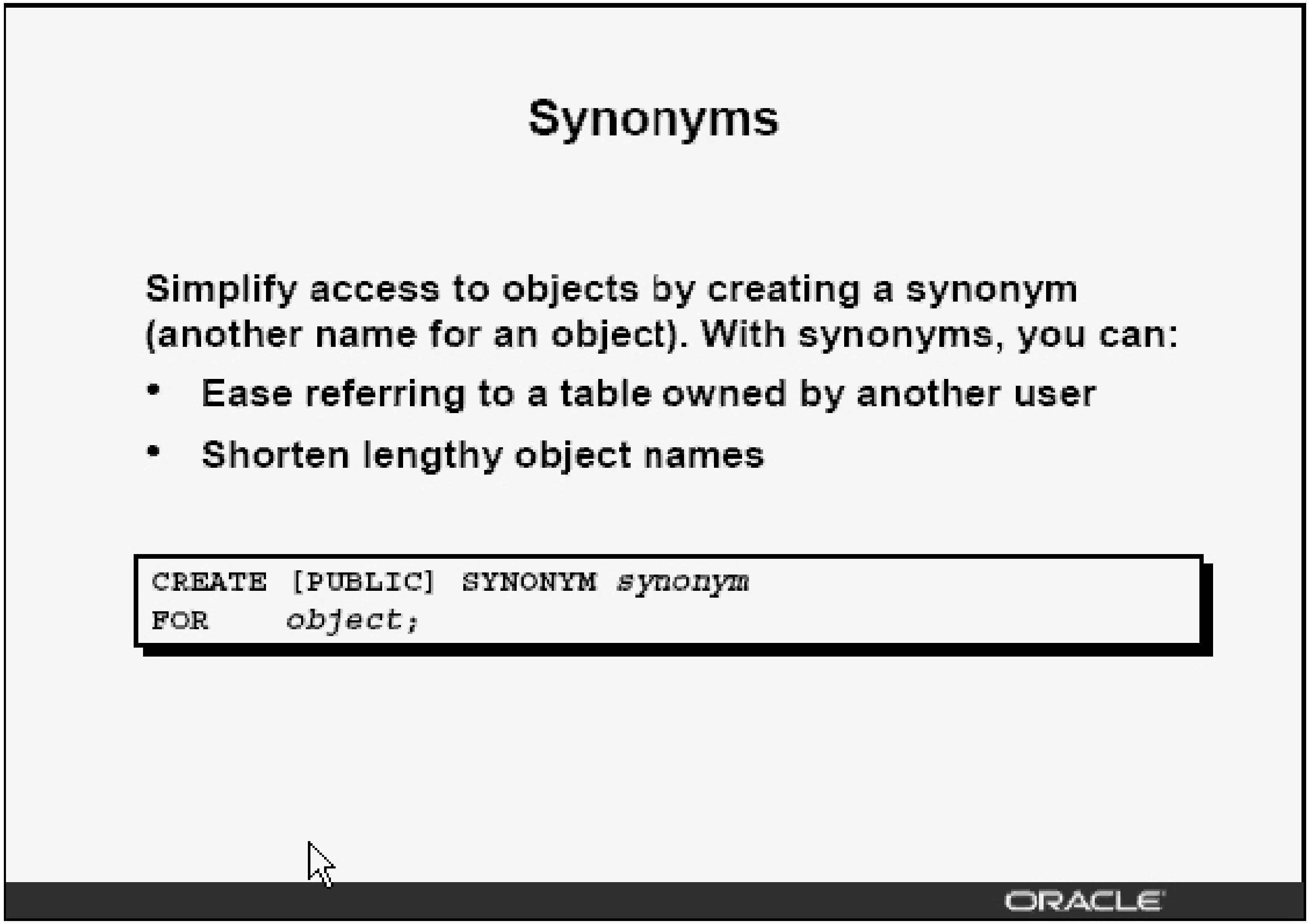
```
SELECT ic.index_name,ic.column_name,ic.column_position
col_pos,ix.uniqueness
FROM user_indexes ix,user_ind_columns ic
WHERE ic.index_name=ix.index_name
AND ic.table_name='EMPLOYEES'
```



INDEX_NAME	COLUMN_NAME	COL_POS	UNIQUENESS
EMP_EMAIL_UK	EMAIL	1	UNIQUE
EMP_EMP_ID_PK	EMPLOYEE_ID	1	UNIQUE
EMP_DEPARTMENT_IX	DEPARTMENT_ID	1	NONUNIQUE
EMP_JOB_IX	JOB_ID	1	NONUNIQUE
EMP_MANAGER_IX	MANAGER_ID	1	NONUNIQUE
EMP_NAME_IX	LAST_NAME	1	NONUNIQUE
EMP_NAME_IX	FIRST_NAME	2	NONUNIQUE
EMP_LAST_NAME_IDX	LAST_NAME	1	NONUNIQUE

```
DROP INDEX emp_last_name_idx
```

Synonym



Synonyms

Simplify access to objects by creating a synonym (another name for an object). With synonyms, you can:

- Ease referring to a table owned by another user
- Shorten lengthy object names

```
CREATE [PUBLIC] SYNONYM synonym
FOR     object;
```

SYNONYM

```
CREATE SYNONYM d_sum
FOR hr.employees

DROP SYNONYM d_sum
```