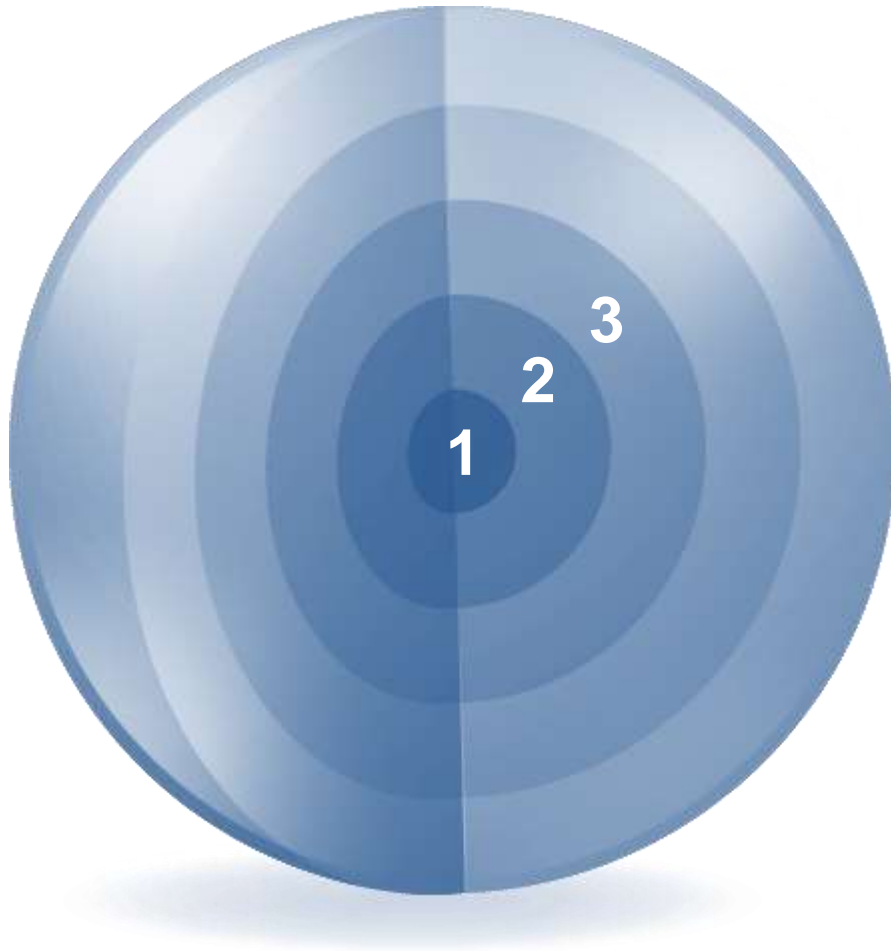


## Creating Other Schema Objects

# What you will learn at the end of this Session?



**1. Create simple and complex views**

**2. Retrieve data from views**

**3. Create, maintain, and use sequences**

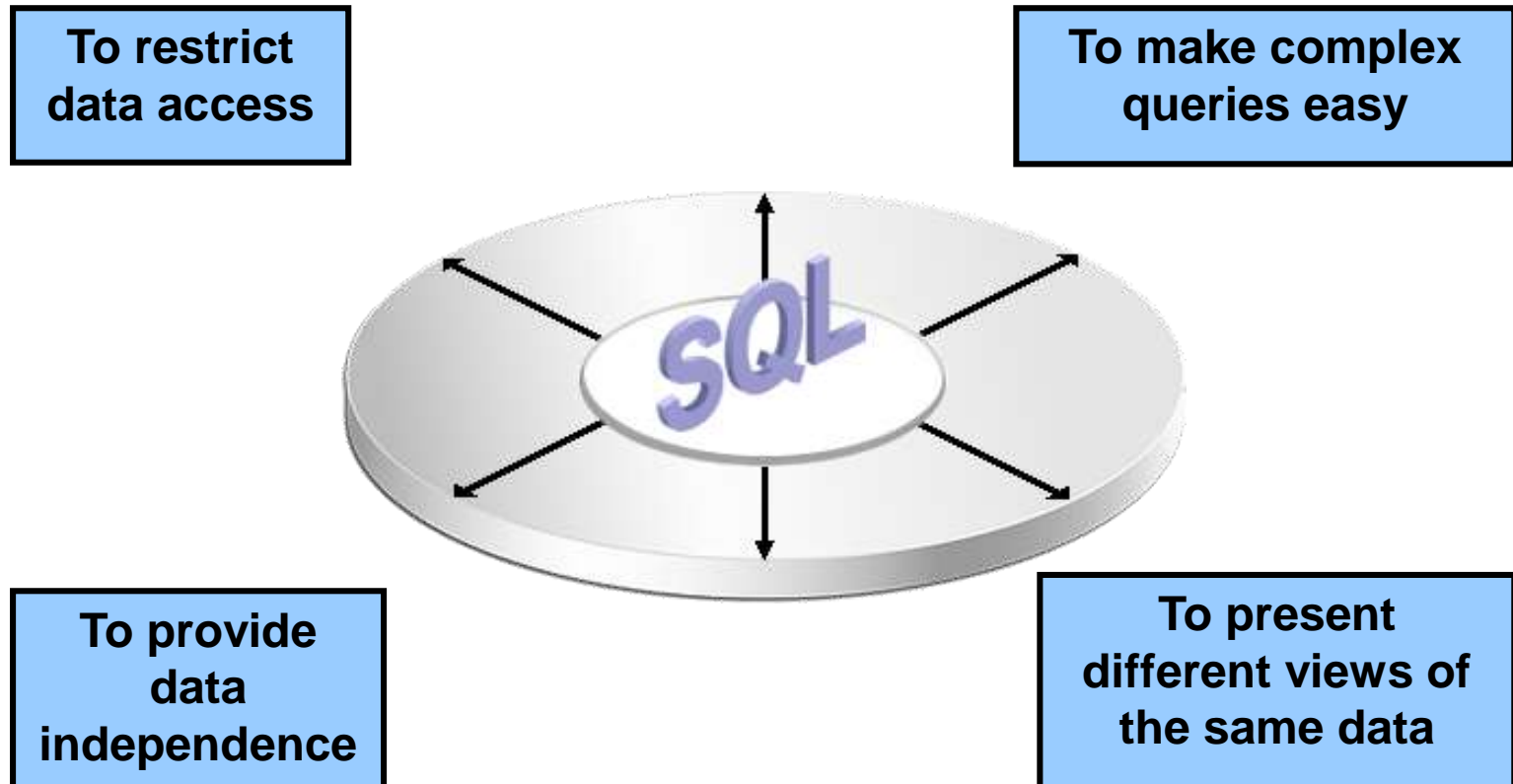
Object	Description
Table	Basic unit of storage; composed of rows
View	Logically represents subsets of data from one or more tables
Sequence	Generates numeric values
Index	Improves the performance of data retrieval queries
Synonym	Gives alternative names to objects

# What Is a View?

## EMPLOYEES table

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY
100	Steven	King	SKING	515.123.4567	17-JUN-87	AD_PRES	24000
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-89	AD_VP	17000
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-93	AD_VP	17000
103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-90	IT_PROG	9000
104	Bruce	Ernst	BERNST	590.423.4566	17-FEB-96	IT_PROG	6000
105	David	Turner	DTURNER	590.423.4565	17-SEP-98	IT_PROG	4200
106	Walter	Clayton	WCLAYTON	590.423.4564	17-SEP-98	IT_PROG	4200
107	Jenna	Ford	JFORD	590.423.4563	17-SEP-98	IT_PROG	4200
108	Peter	Dutton	PDUTTON	590.423.4562	17-SEP-98	IT_PROG	4200
109	John	Ford	JFORD	590.423.4561	17-SEP-98	IT_PROG	4200
110	Christina	Mavris	CMAVRIS	590.423.4560	17-SEP-98	IT_PROG	4200
111	Ismael	Sciarra	ISCIARRA	590.423.4559	17-SEP-98	IT_PROG	4200
112	Oliver	Tobias	OTOBIAS	590.423.4558	17-SEP-98	IT_PROG	4200
113	Peter	Gravenor	PGRVENOR	590.423.4557	17-SEP-98	IT_PROG	4200
114	Renske	Adams	RADAMS	590.423.4556	17-SEP-98	IT_PROG	4200
115	Shelley	Stevens	SSTEVENS	590.423.4555	17-SEP-98	IT_PROG	4200
116	Walter	Clayton	WCLAYTON	590.423.4554	17-SEP-98	IT_PROG	4200
117	Walter	Clayton	WCLAYTON	590.423.4553	17-SEP-98	IT_PROG	4200
118	Walter	Clayton	WCLAYTON	590.423.4552	17-SEP-98	IT_PROG	4200
119	Walter	Clayton	WCLAYTON	590.423.4551	17-SEP-98	IT_PROG	4200
120	Walter	Clayton	WCLAYTON	590.423.4550	17-SEP-98	IT_PROG	4200
121	Walter	Clayton	WCLAYTON	590.423.4549	17-SEP-98	IT_PROG	4200
122	Walter	Clayton	WCLAYTON	590.423.4548	17-SEP-98	IT_PROG	4200
123	Walter	Clayton	WCLAYTON	590.423.4547	17-SEP-98	IT_PROG	4200
124	Walter	Clayton	WCLAYTON	590.423.4546	17-SEP-98	IT_PROG	4200
125	Walter	Clayton	WCLAYTON	590.423.4545	17-SEP-98	IT_PROG	4200
126	Walter	Clayton	WCLAYTON	590.423.4544	17-SEP-98	IT_PROG	4200
127	Walter	Clayton	WCLAYTON	590.423.4543	17-SEP-98	IT_PROG	4200
128	Walter	Clayton	WCLAYTON	590.423.4542	17-SEP-98	IT_PROG	4200
129	Walter	Clayton	WCLAYTON	590.423.4541	17-SEP-98	IT_PROG	4200
130	Walter	Clayton	WCLAYTON	590.423.4540	17-SEP-98	IT_PROG	4200
131	Walter	Clayton	WCLAYTON	590.423.4539	17-SEP-98	IT_PROG	4200
132	Walter	Clayton	WCLAYTON	590.423.4538	17-SEP-98	IT_PROG	4200
133	Walter	Clayton	WCLAYTON	590.423.4537	17-SEP-98	IT_PROG	4200
134	Walter	Clayton	WCLAYTON	590.423.4536	17-SEP-98	IT_PROG	4200
135	Walter	Clayton	WCLAYTON	590.423.4535	17-SEP-98	IT_PROG	4200
136	Walter	Clayton	WCLAYTON	590.423.4534	17-SEP-98	IT_PROG	4200
137	Walter	Clayton	WCLAYTON	590.423.4533	17-SEP-98	IT_PROG	4200
138	Walter	Clayton	WCLAYTON	590.423.4532	17-SEP-98	IT_PROG	4200
139	Walter	Clayton	WCLAYTON	590.423.4531	17-SEP-98	IT_PROG	4200
140	Walter	Clayton	WCLAYTON	590.423.4530	17-SEP-98	IT_PROG	4200
141	Walter	Clayton	WCLAYTON	590.423.4529	17-SEP-98	IT_PROG	4200
142	Walter	Clayton	WCLAYTON	590.423.4528	17-SEP-98	IT_PROG	4200
143	Walter	Clayton	WCLAYTON	590.423.4527	17-SEP-98	IT_PROG	4200
144	Walter	Clayton	WCLAYTON	590.423.4526	17-SEP-98	IT_PROG	4200
145	Walter	Clayton	WCLAYTON	590.423.4525	17-SEP-98	IT_PROG	4200
146	Walter	Clayton	WCLAYTON	590.423.4524	17-SEP-98	IT_PROG	4200
147	Walter	Clayton	WCLAYTON	590.423.4523	17-SEP-98	IT_PROG	4200
148	Walter	Clayton	WCLAYTON	590.423.4522	17-SEP-98	IT_PROG	4200
149	Walter	Clayton	WCLAYTON	590.423.4521	17-SEP-98	IT_PROG	4200
150	Walter	Clayton	WCLAYTON	590.423.4520	17-SEP-98	IT_PROG	4200
151	Walter	Clayton	WCLAYTON	590.423.4519	17-SEP-98	IT_PROG	4200
152	Walter	Clayton	WCLAYTON	590.423.4518	17-SEP-98	IT_PROG	4200
153	Walter	Clayton	WCLAYTON	590.423.4517	17-SEP-98	IT_PROG	4200
154	Walter	Clayton	WCLAYTON	590.423.4516	17-SEP-98	IT_PROG	4200
155	Walter	Clayton	WCLAYTON	590.423.4515	17-SEP-98	IT_PROG	4200
156	Walter	Clayton	WCLAYTON	590.423.4514	17-SEP-98	IT_PROG	4200
157	Walter	Clayton	WCLAYTON	590.423.4513	17-SEP-98	IT_PROG	4200
158	Walter	Clayton	WCLAYTON	590.423.4512	17-SEP-98	IT_PROG	4200
159	Walter	Clayton	WCLAYTON	590.423.4511	17-SEP-98	IT_PROG	4200
160	Walter	Clayton	WCLAYTON	590.423.4510	17-SEP-98	IT_PROG	4200
161	Walter	Clayton	WCLAYTON	590.423.4509	17-SEP-98	IT_PROG	4200
162	Walter	Clayton	WCLAYTON	590.423.4508	17-SEP-98	IT_PROG	4200
163	Walter	Clayton	WCLAYTON	590.423.4507	17-SEP-98	IT_PROG	4200
164	Walter	Clayton	WCLAYTON	590.423.4506	17-SEP-98	IT_PROG	4200
165	Walter	Clayton	WCLAYTON	590.423.4505	17-SEP-98	IT_PROG	4200
166	Walter	Clayton	WCLAYTON	590.423.4504	17-SEP-98	IT_PROG	4200
167	Walter	Clayton	WCLAYTON	590.423.4503	17-SEP-98	IT_PROG	4200
168	Walter	Clayton	WCLAYTON	590.423.4502	17-SEP-98	IT_PROG	4200
169	Walter	Clayton	WCLAYTON	590.423.4501	17-SEP-98	IT_PROG	4200
170	Walter	Clayton	WCLAYTON	590.423.4500	17-SEP-98	IT_PROG	4200
171	Walter	Clayton	WCLAYTON	590.423.4499	17-SEP-98	IT_PROG	4200
172	Walter	Clayton	WCLAYTON	590.423.4498	17-SEP-98	IT_PROG	4200
173	Walter	Clayton	WCLAYTON	590.423.4497	17-SEP-98	IT_PROG	4200
174	Walter	Clayton	WCLAYTON	590.423.4496	17-SEP-98	IT_PROG	4200
175	Walter	Clayton	WCLAYTON	590.423.4495	17-SEP-98	IT_PROG	4200
176	Walter	Clayton	WCLAYTON	590.423.4494	17-SEP-98	IT_PROG	4200
177	Walter	Clayton	WCLAYTON	590.423.4493	17-SEP-98	IT_PROG	4200
178	Walter	Clayton	WCLAYTON	590.423.4492	17-SEP-98	IT_PROG	4200
179	Walter	Clayton	WCLAYTON	590.423.4491	17-SEP-98	IT_PROG	4200
180	Walter	Clayton	WCLAYTON	590.423.4490	17-SEP-98	IT_PROG	4200
181	Walter	Clayton	WCLAYTON	590.423.4489	17-SEP-98	IT_PROG	4200
182	Walter	Clayton	WCLAYTON	590.423.4488	17-SEP-98	IT_PROG	4200
183	Walter	Clayton	WCLAYTON	590.423.4487	17-SEP-98	IT_PROG	4200
184	Walter	Clayton	WCLAYTON	590.423.4486	17-SEP-98	IT_PROG	4200
185	Walter	Clayton	WCLAYTON	590.423.4485	17-SEP-98	IT_PROG	4200
186	Walter	Clayton	WCLAYTON	590.423.4484	17-SEP-98	IT_PROG	4200
187	Walter	Clayton	WCLAYTON	590.423.4483	17-SEP-98	IT_PROG	4200
188	Walter	Clayton	WCLAYTON	590.423.4482	17-SEP-98	IT_PROG	4200
189	Walter	Clayton	WCLAYTON	590.423.4481	17-SEP-98	IT_PROG	4200
190	Walter	Clayton	WCLAYTON	590.423.4480	17-SEP-98	IT_PROG	4200
191	Walter	Clayton	WCLAYTON	590.423.4479	17-SEP-98	IT_PROG	4200
192	Walter	Clayton	WCLAYTON	590.423.4478	17-SEP-98	IT_PROG	4200
193	Walter	Clayton	WCLAYTON	590.423.4477	17-SEP-98	IT_PROG	4200
194	Walter	Clayton	WCLAYTON	590.423.4476	17-SEP-98	IT_PROG	4200
195	Walter	Clayton	WCLAYTON	590.423.4475	17-SEP-98	IT_PROG	4200
196	Walter	Clayton	WCLAYTON	590.423.4474	17-SEP-98	IT_PROG	4200
197	Walter	Clayton	WCLAYTON	590.423.4473	17-SEP-98	IT_PROG	4200
198	Walter	Clayton	WCLAYTON	590.423.4472	17-SEP-98	IT_PROG	4200
199	Walter	Clayton	WCLAYTON	590.423.4471	17-SEP-98	IT_PROG	4200
200	Walter	Clayton	WCLAYTON	590.423.4470	17-SEP-98	IT_PROG	4200
201	Walter	Clayton	WCLAYTON	590.423.4469	17-SEP-98	IT_PROG	4200
202	Walter	Clayton	WCLAYTON	590.423.4468	17-SEP-98	IT_PROG	4200
203	Walter	Clayton	WCLAYTON	590.423.4467	17-SEP-98	IT_PROG	4200
204	Walter	Clayton	WCLAYTON	590.423.4466	17-SEP-98	IT_PROG	4200
205	Shelley	Higgins	SHIGGINS	515.123.8080	07-JUN-94	AC_MGR	12000
206	William	Gietz	WGIEZT	515.123.8181	07-JUN-94	AC_ACCOUNT	8300

# Advantages of 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

- You embed a subquery in 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.

- Create the EMPVU80 view, which contains details of the employees in department 80:

```
CREATE VIEW ordvu  
AS SELECT order_id , order_date , order_status  
   FROM orders  
   WHERE order_status = 10 ;
```

```
CREATE VIEW succeeded.
```

- Describe the structure of the view by using the SQL\*Plus

```
DESCRIBE ordvu ;
```



- Create a view by using column aliases in the subquery:

```
CREATE VIEW ordvu
AS SELECT order_id , order_status , order_total / 12 Total_per_Month
FROM orders
WHERE order_status = 10 ;
```

```
CREATE VIEW succeeded.
```

- Select the columns from this view by the given alias names.

```
SELECT *
FROM ordvu ;
```

[illegible]

- Modify the EMPVU80 view by using a CREATE OR REPLACE VIEW clause. Add an alias for each column name:

```
CREATE OR REPLACE VIEW ordvu
  (order_id, order_date, order_status)
AS SELECT  order_id, to_char(order_date, 'fmDD-Mon-YYYY')
           , order_status)
FROM      orders
WHERE     order_status = 10;
```

```
CREATE OR REPLACE VIEW succeeded.
```

- Column aliases in the CREATE OR REPLACE VIEW clause are listed in the same order as the columns in the subquery.

# Creating a Complex View

Create a complex view that contains group functions to display values from two tables:

```
CREATE OR REPLACE 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 JOIN departments d
  ON        (e.department_id = d.department_id)
  GROUP BY  d.department_name;
```

```
CREATE OR REPLACE VIEW succeeded.
```

# Rules for Performing DML Operations on a View

**You can usually perform DML operations on simple views.**

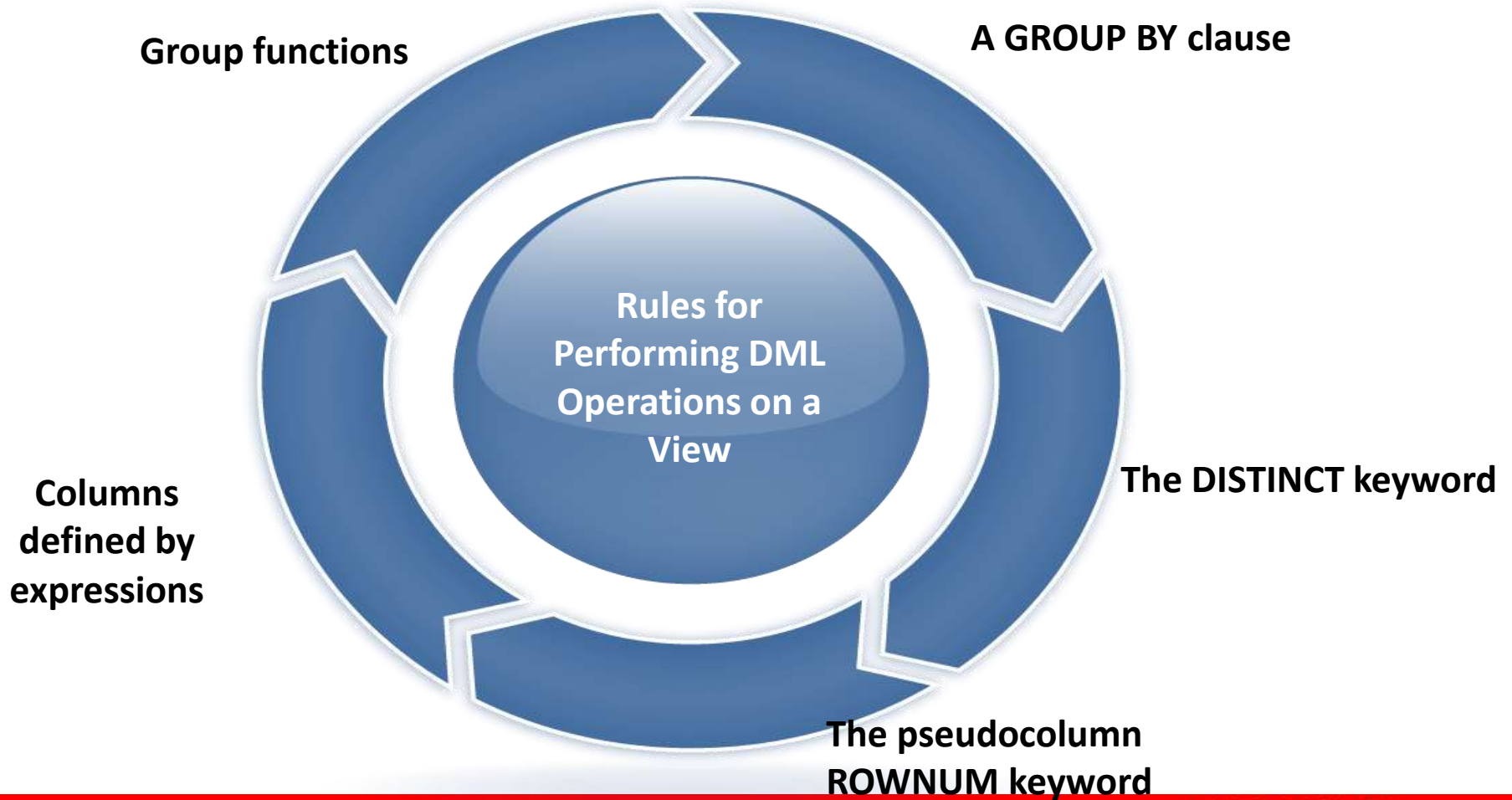
**You cannot**

**remove a row if the view contains the following:**

- **Group functions**
- **A GROUP BY clause**
- **The DISTINCT keyword**
- **The pseudo column ROWNUM keyword**

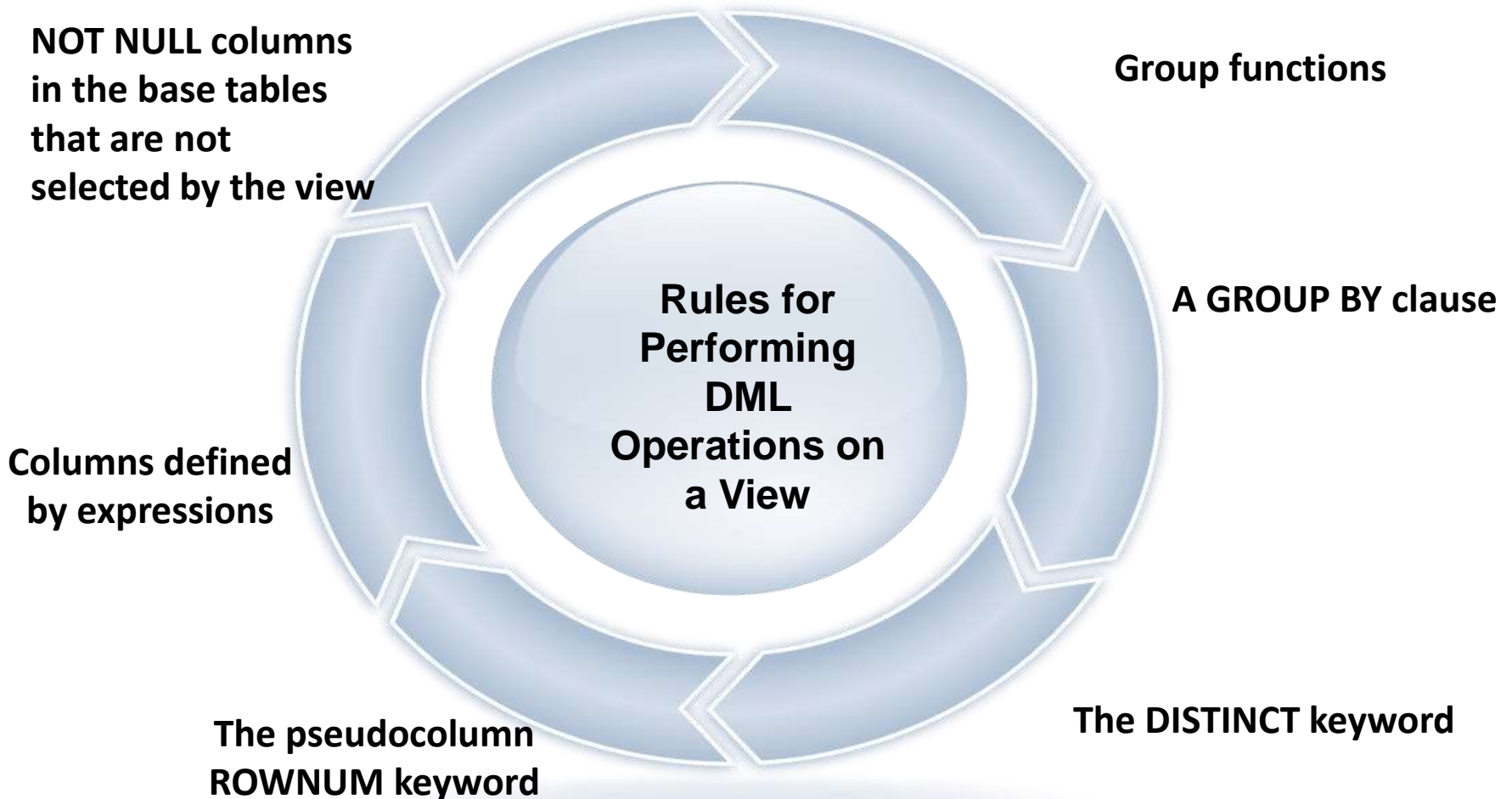
# Rules for Performing DML Operations on a View

- You cannot modify data in a view if it contains:



# Rules for Performing DML Operations on a View

•You cannot add data through a view if the view includes:



# Using the WITH CHECK OPTION Clause

- You can ensure that DML operations performed on the view stay in the domain of the view by using the WITH CHECK OPTION clause:

```
CREATE OR REPLACE VIEW ordvu
AS SELECT      *
   FROM      orders
   WHERE     order_status = 10
   WITH CHECK OPTION CONSTRAINT ordvu20_ck ;
```

```
CREATE OR REPLACE VIEW succeeded.
```

- Any attempt to INSERT a row with an order\_status other than 10, or to UPDATE the status number for any row in the view fails because it violates the WITH CHECK OPTION constraint.



- You can ensure that no DML operations occur by adding the **WITH READ ONLY** option to your view definition.
- Any attempt to perform a DML operation on any row in the view results in an Oracle server error.

# Denying DML Operations

```
CREATE VIEW ordvu  
AS SELECT order_id , order_status , order_total / 12 Total_per_Month  
FROM orders  
WHERE order_status = 10 ;  
WITH READ ONLY ;
```

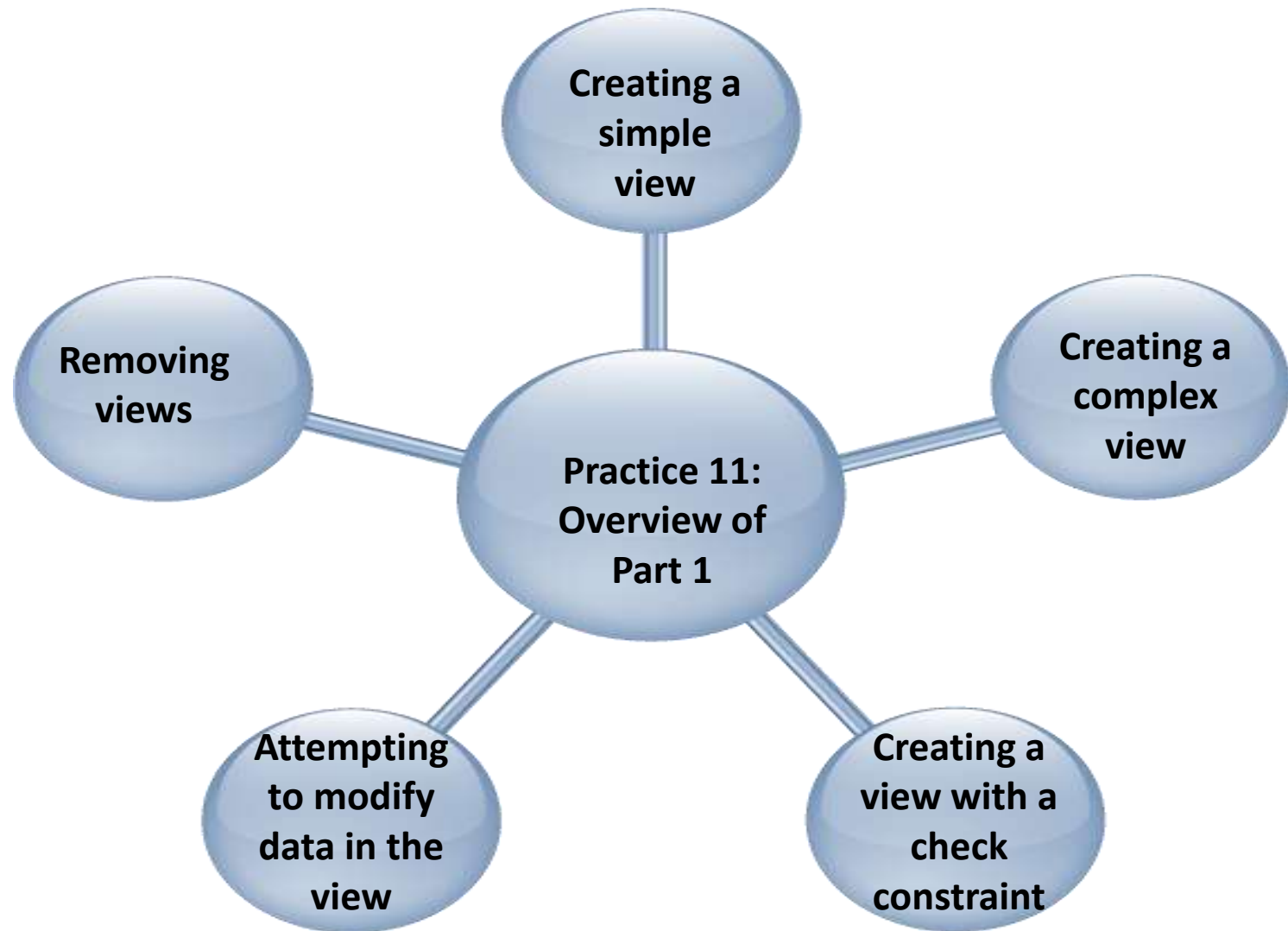
```
CREATE OR REPLACE VIEW succeeded.
```

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

```
DROP VIEW view;
```

```
DROP VIEW ordvu;
```

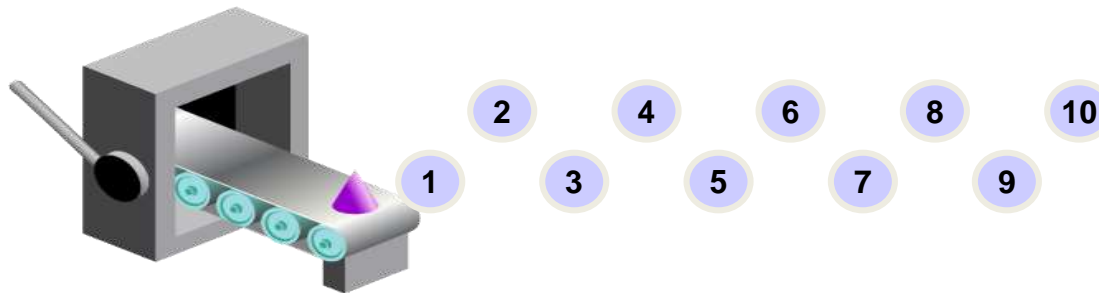
```
DROP VIEW empvu80 succeeded.
```



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

## A sequence:

- Can automatically generate unique numbers
- Is a shareable object
- Can be used to create a primary key value
- Replaces application code
- Speeds up the efficiency of accessing sequence values when cached in memory



# 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}];
```

- Create a sequence named DEPT\_DEPTID\_SEQ to be used for the primary key of the DEPARTMENTS table.
- Do not use the CYCLE option.

```
CREATE SEQUENCE ord_ordid_seq  
            INCREMENT BY 10  
            START WITH 120  
            MAXVALUE 9999  
            NOCACHE  
            NOCYCLE ;
```

CREATE SEQUENCE succeeded.



# 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.

- Insert a new order with mode “Direct” and status 5:

```
INSERT INTO orders ( order_id ,  
                    order_date , order_mode , order_status )  
VALUES ( ord_ordid_seq.NEXTVAL,  
        to_char (SYSDATE , 'fmDD-Mon-YYYY' ) ,  
        'direct', 1) ;
```

```
1 rows inserted
```

- View the current value for the DEPT\_DEPTID\_SEQ sequence:

```
SELECT  ord_ordid_seq.CURRVAL  
FROM    dual ;
```

**Caching sequence values in memory gives faster access to those values.**

**Gaps in sequence values can occur when:**

- **A rollback occurs**
- **The system crashes**
- **A sequence is used in another table**

Change the increment value, maximum value, minimum value, cycle option, or cache option:

```
ALTER SEQUENCE ord_ordid_seq  
    INCREMENT BY 20  
    MAXVALUE 999999  
    NOCACHE  
    NOCYCLE ;
```

```
ALTER SEQUENCE dept_deptid_seq succeeded.
```

# Guidelines for Modifying a Sequence

## ALTER Privelege

You must be the owner or have the ALTER privilege for the sequence.

## Future sequence

Only future sequence numbers are affected.

## Re-create to restart the sequence

The sequence must be dropped and re-created to restart the sequence at a different number.

## Validation

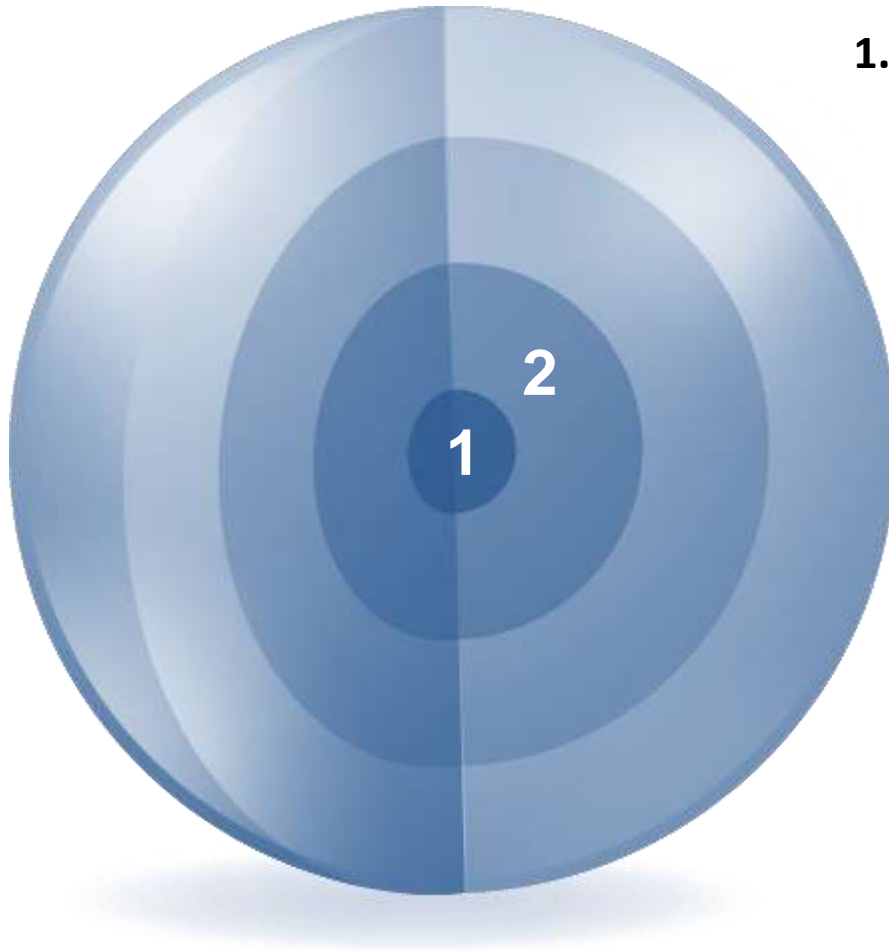
Some validation is performed.

## DROP statement

To remove a sequence, use the DROP statement

```
DROP SEQUENCE ord_ordid_seq ;
```

```
DROP SEQUENCE dept_deptid_seq succeeded.
```



**1. Create, use, and remove views**

**2. Automatically generate sequence numbers by using a sequence generator**