10 Creating Other Schema Objects

Lesson Agenda

- Overview of views:
  - Creating, modifying, and retrieving data from a view
  - Data manipulation language(UML) operations on a view
  - Dropping a view
- Overview of sequences:
  - Creating, using, and modifying a sequence
  - Cache sequence values
  - NEXTVAL and CURRVAL pseudo column

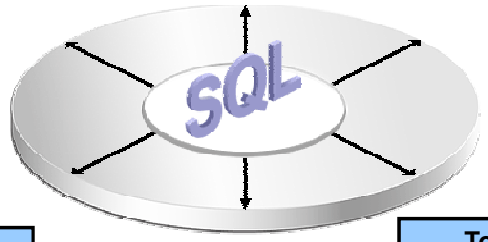
What is a View ?

EMPLOYEES table

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY
1	100	Steven	King	SKING	515.123.4567	17-JUN-87	AD_PRES	24000
2	101	Neena	Kochhar	NKOCHH...	515.123.4568	21-SEP-89	AD_VP	17000
3	102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-93	AD_VP	17000
4	103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-90	IT_PROG	9000
5								6000
6								4200
7								5800
								3500
								3100
								2600
								2500
								10500
							EP	11000
							SA_REP	8600
						-99	SA_REP	7000
						17-SEP-87	AD_ASST	4400
						17-FEB-96	MK_MAN	13000
					66	17-AUG-97	MK_REP	6000
19	205	Shelley	Higgins	SHIGGINS	515.123.8080	07-JUN-94	AC_MGR	12000
20	206	William	Gietz	WGIETZ	515.123.8181	07-JUN-94	AC_ACC...	8300

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	SALARY
100	Steven	King	24000
101	Neena	Kochhar	17000
102	Lex	De Haan	17000
103	Alexander	Hunold	9000
104	Bruce	Ernst	6000

Advantages of Views



- To restrict data access
- To make complex queries easy
- To provide data independence
- To present different views of the same data

Creating a View

- You embed a subquery in the CREATE VIEW statement:
- The subquery can contain complex SELECT syntax.

Syntax:

```
CREATE VIEW view-name
  [(alias[, alias]...)]
 AS subquery;
```


Creating a View

Create the EMP80 view, which contains details of the employees in department 80:

SQL Statement:

```
CREATE VIEW emp80
 AS SELECT   employee_id, last_name,salary
    FROM     employees
    WHERE    department_id = 80;
```

```
CREATE VIEW succeeded.
```


Retrieving Structure from a View

Describe the structure of the view by using the iSQL*Plus DESCRIBE command:

SQL Statement:

```
DESCRIBE emp80;
```

Output:

```
Name                           Null     Type
------------------------------ -------- --------------
EMPLOYEE_ID                    NOT NULL NUMBER(6)
LAST_NAME                      NOT NULL VARCHAR2(25)
SALARY                                  NUMBER(8,2)
```


Retrieving Data from a View

Select data of the `EMP80` view by using the SQL SELECT command:

SQL Statement:

```
SELECT   * FROM emp80;
```

Output:

	EMPLOYEE_ID	LAST_NAME	SALARY
1	145	Russell	14000
2	146	Partners	13500
3	147	Errazuriz	12000
4	148	Cambrault	11000
5	149	Zlotkey	10500
6	150	Tucker	10000
7	151	Bernstein	9500
8	152	Hall	9000

Practice

Create the EMPIT view which contains employee number,name,and job title where job number is IT_PROG

	EMPLOYEE_ID	LAST_NAME	JOB_TITLE
1	103	Hunold	Programmer
2	104	Ernst	Programmer
3	105	Austin	Programmer
4	106	Pataballa	Programmer
5	107	Lorentz	Programmer

Practice (Answer)

Creating a View with Column Aliases

- Create the SAL50 view by using column aliases of the employees in department 50:
- Select the columns from this view by the given alias names.

SQL Statement:

```
CREATE VIEW sal50
 AS SELECT   employee_id ID_NUMBER,
            last_name ,
            salary*12 ANN_SALARY
    FROM    employees
    WHERE   department_id = 50;
```

```
CREATE VIEW succeeded.
```


Retrieving Data from a View

Select data of the SAL80 view by using the SQL SELECT command:

SQL Statement:

```
SELECT * FROM sal50;
```

Output:

	ID_NUMBER	NAME	ANN_SALARY
1	120	Weiss	96000
2	121	Fripp	98400
3	122	Kaufling	94800
4	123	Vollman	78000
5	124	Mourgos	69600
6	125	Nayer	38400
7	126	Mikkilineni	32400

Column Aliases

45 Rows...

Practice

Create the DEPT_VIEW view by using column aliases Dep No,Dep Name,Emp Name,which column department number,department name,first name where department number is 100

	Dep No	Dep Name	Eme Name
1	100	Finance	Nancy
2	100	Finance	Daniel
3	100	Finance	John
4	100	Finance	Ismael
5	100	Finance	Jose Manuel

Practice (Answer)

Modifying a View

Modify the EMP80 view by using a CREATE OR REPLACE VIEW clause. Add an alias for each column name:

SQL Statement:

```
CREATE OR REPLACE VIEW emp80
  (id_number, name, sal, department_id)
AS SELECT  employee_id, first_name || ' '
           || last_name, salary, department_id
   FROM    employees
   WHERE   department_id = 80;
```

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


Retrieving Structure from a View

Describe the structure of the view by using the iSQL*Plus DESCRIBE command:

SQL Statements:

```
DESCRIBE  emp80;
```

Output:

```
Name                          Null     Type
----------------------------- -------- ----------------
ID_NUMBER                     NOT NULL NUMBER(6)
NAME                                   VARCHAR2(46)
SAL                                    NUMBER(8,2)
DEPARTMENT_ID                          NUMBER(4)
```

Retrieving Data from a View

Select data of the EMP80 view by using the SQL SELECT command:

SQL Statement:

```
SELECT   * FROM emp80;
```

Output:

	EMPLOYEE_ID	L
1	145	Russ
2	146	Part
3	147	Erra
4	148	Camb
5	149	Zlot
6	150	Tuck
7	151	Berr
8	152	Hall

	ID_NUMBER	NAME	SAL	DEPARTMENT_ID
1	145	John Russell	14000	80
2	146	Karen Partners	13500	80
3	147	Alberto Errazuriz	12000	80
4	148	Gerald Cambrault	11000	80
5	149	Eleni Zlotkey	10500	80
6	150	Peter Tucker	10000	80
7	151	David Bernstein	9500	80
8	152	Peter Hall	9000	80
9	153	Christopher Olsen	8000	80
10	154	Nanette Cambrault	7500	80

Practice

Modify the EMPIT view with Column Aliases CODE, Name, SOCIAL, Job Name which employee number,first name,last name,3% of salary,job title where job number are “Purchasing Clerk ” or “Stock Clerk”

	CODE	Name	SOCIAL	Job Name
1	116	ShelliBaida	87 BAHT	Purchasing Clerk
2	117	SigalTobias	84 BAHT	Purchasing Clerk
3	118	GuyHimuro	78 BAHT	Purchasing Clerk
4	119	KarenColmenares	75 BAHT	Purchasing Clerk
5	125	JuliaNayer	96 BAHT	Stock Clerk
6	126	IreneMikkilineni	81 BAHT	Stock Clerk
7	127	JamesLandry	72 BAHT	Stock Clerk
8	128	StevenMarkle	66 BAHT	Stock Clerk
9	129	LauraBissot	99 BAHT	Stock Clerk

Practice (Answer)

Creating a Complex View

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

SQL Statements:

```
CREATE OR REPLACE VIEW dept_sum_vu
  (name, minsal, maxsal, avgsal)
AS SELECT   department_name, MIN(salary),
           MAX(salary),AVG(salary)
   FROM     employees e,departments d
   WHERE    e.department_id = d.department_id
   GROUP BY department_name;
```

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

Retrieving Structure from a View

Describe the structure of the view by using the iSQL*Plus DESCRIBE command:

SQL Statement:

```
DESCRIBE  dept_sum_vu;
```

Output:

```
Name                          Null     Type
----------------------------- -------- ----------------
NAME                          NOT NULL VARCHAR2(30)
MINSAL                                 NUMBER
MAXSAL                                 NUMBER
AVGSAL                                 NUMBER
```


Retrieving Data from a View

Select data of the EMP80 view by using the SQL SELECT command:

SQL Statement:

```
SELECT   * FROM dept_sum_vu;
```

Output:

	NAME	MINSAL	MAXSAL	AVGSAL
1	Administration	4400	4400	4400
2	Accounting	8300	12000	10150
3	Executive	17000	24000	19333.333333333333333333333333333333333333
4	IT	4200	9000	5760
5	Purchasing	2500	11000	4150
6	Human Resources	6500	6500	6500
7	Public Relations	10000	10000	10000
8	Shipping	2100	8200	3475.555555555555555555555555555555555556
9	Finance	6900	12000	8600
10	Sales	6100	14000	8955.882352941176470588235294117647058824
11	Marketing	6000	13000	9500

Practice

Create the JOB_VIEW view with Column Aliases JOB NAME,AVG SAL,TOTAL SAL which job title, average salary,total salary group by job title where total salary more than 10000 and job title start letter with P or S order by job title

	JOB NAME	AVG SAL	TOTAL SAL
1	President	24,000.00	24,000.00
2	Programmer	5,760.00	28,800.00
3	Purchasing Clerk	2,780.00	13,900.00
4	Purchasing Manager	11,000.00	11,000.00
5	Sales Manager	12,200.00	61,000.00
6	Sales Representative	8,350.00	250,500.00
7	Shipping Clerk	3,215.00	64,300.00
8	Stock Clerk	2,785.00	55,700.00
9	Stock Manager	7,280.00	36,400.00

Practice (Answer)

Removing a View

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

SQL Statements:

```
DROP VIEW view-name;
```

SQL Statements:

```
DROP VIEW emp80;
```

```
DROP VIEW empvu80 succeeded.
```

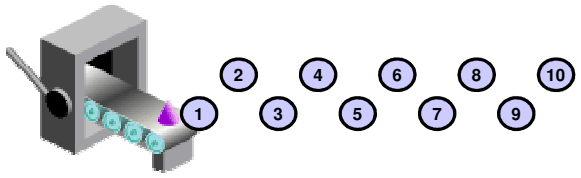

Sequences

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 some queries
Synonym	Gives alternative names to objects

Sequences

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

Define a sequence to generate sequential numbers automatically:

Syntax:

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

CREATE SEQUENCE Statement

CREATE SEQUENCE	คือคีย์เวิร์ด ที่ต้องพิมพ์ตามนี้
sequence-name	คือ ชื่อของ sequence ที่ต้องการสร้าง
INCREMENT BY n	คือ การเพิ่มค่าถัดไปของ sequence โดยปกติถ้าไม่ระบุ การเพิ่มค่าจะเพิ่มทีละ 1
START WITH n	คือ การกำหนดค่าเริ่มต้นของ sequence ที่สร้างขึ้น โดยปกติถ้าไม่ระบุ ค่าเริ่มต้นของ sequence ที่สร้างขึ้นจะมีค่าเท่ากับ 1
MAXVALUE n	คือ ค่าสูงสุดของ sequence ที่สร้างขึ้น(ไม่จำเป็นต้องระบุ)
NOMAXVALUE	คือ การสร้าง sequence โดยไม่ระบุค่าสูงสุด(เป็นค่า default)
MINVALUE n	คือ ค่าต่ำสุดของ sequence ที่สร้างขึ้น(ไม่จำเป็นต้องระบุ)
NOMINVALUE	คือ การสร้าง sequence โดยไม่ระบุค่าต่ำสุด(เป็นค่า default)

CREATE SEQUENCE Statement

CYCLE|NOCYCLE	ถ้ากำหนดเป็น CYCLE เมื่อค่าของ sequence ถูกกำหนดมาจนถึงค่าสูงสุดจะมีการวนกลับไปใช้ค่าเริ่มต้น ถ้าไม่ระบุคำว่า CYCLE ในตอนสร้าง sequence ค่า default คือ NOCYCLE นั่นคือจะไม่มีการวนกลับไปใช้ค่าเริ่มต้น (กรณีที่สร้าง sequence เพื่อนำไปกำหนดค่าให้แก่ primary key ต้องไม่ระบุให้เป็น CYCLE)
CACHE n |NOCACHE	ถ้าระบุ NO CHACHE หมายความว่าจะไม่มีการกำหนดค่าของ sequence ที่สร้างขึ้นไว้ล่วงหน้า ปกติถ้าไม่มีการระบุค่า default ในการสร้าง sequence โดยทั่วไป คือ CACHE 20 นั่นคือค่าที่จะกำหนดให้ sequence จะถูกกำหนดไว้ล่วงหน้า 20 ค่า

Creating a Sequence

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

SQL Statement:

```
CREATE SEQUENCE dept_deptid_seq
               INCREMENT BY 10
               START WITH 400
               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.

Using a Sequence

Insert a new department named “Support” in location ID 2500:

SQL Statement:

```
INSERT INTO departments(department_id,
            department_name, location_id)
VALUES      (dept_deptid_seq.NEXTVAL,
            'Support', 2500);
```

```
1 rows inserted
```


Using a Sequence

SQL Statement:

```
SELECT * FROM department;
```

Output:

	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
17	170	Manufacturing	(null)	1700
18	180	Construction	(null)	1700
19	190	Contracting	(null)	1700
20	200	Operations	(null)	1700
21	210	IT Support	(null)	1700
22	220	NOC	(null)	1700
23	230	IT Helpdesk	(null)	1700
24	240	Government Sales	(null)	1700
25	250	Retail Sales	(null)	1700
26	260	Recruiting	(null)	1700
27	270	Payroll	(null)	1700
28	300	Support	(null)	2500

Practice

Insert a new department named “HR” in location ID 1400:

20	200	Operations	(null)	1700
21	210	IT Support	(null)	1700
22	220	NOC	(null)	1700
23	230	IT Helpdesk	(null)	1700
24	240	Government Sales	(null)	1700
25	250	Retail Sales	(null)	1700
26	260	Recruiting	(null)	1700
27	270	Payroll	(null)	1700
28	300	Support	(null)	2500
29	310	HR	(null)	1400

Practice (Answer)

Using a Sequence

View the current value for the DEPT_DEPTID_SEQ sequence:

SQL Statement:

```
SELECT  dept_deptid_seq.CURRVAL FROM dual;
```

Output:

	CURRVAL
1	310

Modifying a Sequence

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

SQL Statement:

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

```
ALTER SEQUENCE dept_deptid_seq succeeded.
```


Practice

Insert a new department named “Engineer” in location ID 1800:

20	200	Operations	(null)	1700
21	210	IT Support	(null)	1700
22	220	NOC	(null)	1700
23	230	IT Helpdesk	(null)	1700
24	240	Government Sales	(null)	1700
25	250	Retail Sales	(null)	1700
26	260	Recruiting	(null)	1700
27	270	Payroll	(null)	1700
28	300	Support	(null)	2500
29	310	HR	(null)	1400
30	330	Engineer	(null)	1800

Practice (Answer)

Guidelines for Modifying a Sequence

- You must be the owner or have the ALTER privilege for the sequence.
- Only future sequence numbers are affected.
- The sequence must be dropped and re-created to restart the sequence at a different number.
- To remove a sequence, use the DROP statement:

SQL Statement:

```
DROP  SEQUENCE  dept_deptid_seq;
```

```
DROP SEQUENCE dept_deptid_seq succeeded.
```


Summary

This lesson,you should be able to do the following:

- Create simple and complex views
- Retrieve data from views
- Create,maintain,and use sequences