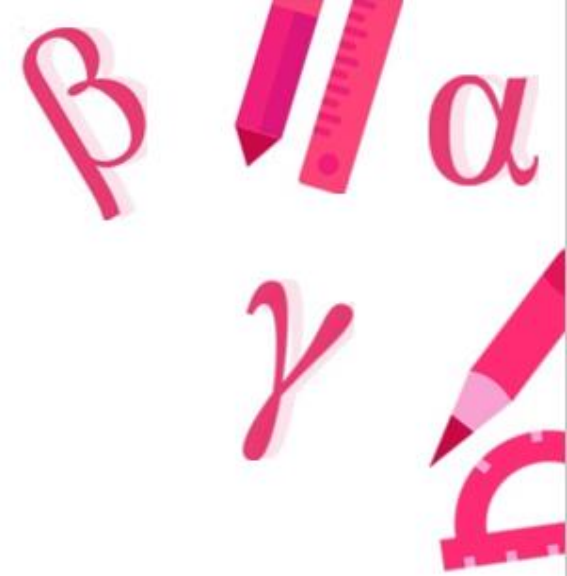


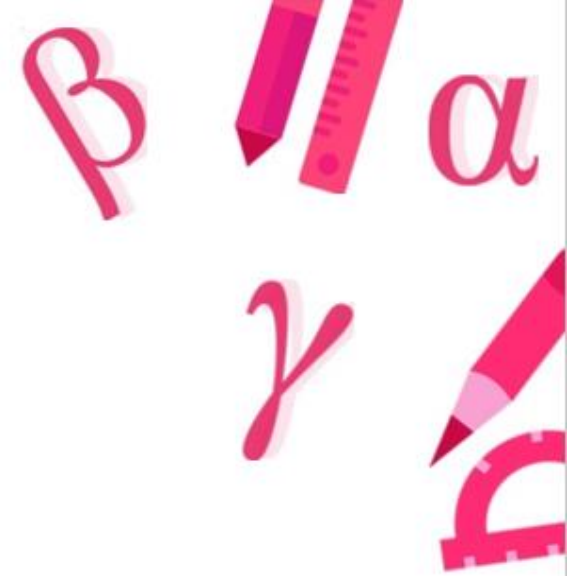
Database Design & Applications

Data Manipulation Language



Objective

- Describe each DML statement
- Insert rows into a table
- Update rows in a table
- Delete rows from a table



Data Manipulation Language

A DML statement is executed when you:

- Add new rows to a table
- Modify existing rows in a table
- Remove existing rows from a table



Inserting Row

New row

...Insert a new row into the Departments Table...

70	Public Relations	100	1700
----	------------------	-----	------

DEPARTMENT

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
50	Shipping	124	1500
60	IT	103	1400
80	Sales	149	2500
90	Executive	100	1700
110	Accounting	205	1700
190	Contracting		1700

INSERT Statement

- Add new rows to a table by using the Insert statement.

```
INSERT INTO  table [(column [, column...])]  
VALUES      (value [, value...])  
            (value [, value...]);
```

Inserting New Rows

- Insert a new row containing values for each column.
- List values in the default order of the columns in the table.
- Optionally, list the columns in the INSERT clause

```
INSERT INTO departments (department_id, department_name,  
                        manager_id, location_id)  
VALUES (70, 'Public Relations', 100, 1700);
```

- Enclose character and date values within single quotation marks.

Inserting More than One Rows

- INSERT statements that use VALUES syntax can insert multiple rows.
- Include multiple lists of comma-separated column values, with lists enclosed within parentheses and separated by commas.
- Each values list must contain exactly as many values as are to be inserted per row.

```
INSERT INTO departments (department_id, department_name,  
                        manager_id, location_id)  
VALUES      (70, 'Public Relations', 100, 1700),  
            (80, 'Communication', 142, 1500);
```

Inserting Rows with Null Values

- Implicit method: Omit the column from the column list.

```
INSERT INTO departments (department_id,  
                          department_name  )  
VALUES (30, 'Purchasing');
```

- Explicit method: Specify the NULL keyword in the VALUES clause.

```
INSERT INTO departments  
VALUES (100, 'Finance', , 
```


Changing Data in a Table

Employee

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	HIRE_DATE	JOB_ID	SALARY	DEPARTMENT_ID	COMMISSION_F
100	Steven	King	SKING	17-JUN-87	AD_PRES	24000	90	
101	Neena	Kochhar	NKOCHHAR	21-SEP-89	AD_VP	17000	90	
102	Lex	De Haan	LDEHAAN	13-JAN-93	AD_VP	17000	90	
103	Alexander	Hunold	AHUNOLD	03-JAN-90	IT_PROG	9000	60	
104	Bruce	Ernst	BERNST	21-MAY-91	IT_PROG	6000	60	
107	Diana	Lorentz	DLORENTZ	07-FEB-99	IT_PROG	4200	60	
124	Kevin	Mourgos	KMOURGOS	16-NOV-99	ST_MAN	5800	50	

Update rows in the EMPLOYEE table.

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	HIRE_DATE	JOB_ID	SALARY	DEPARTMENT_ID	COMMISSIO
100	Steven	King	SKING	17-JUN-87	AD_PRES	24000	90	
101	Neena	Kochhar	NKOCHHAR	21-SEP-89	AD_VP	17000	90	
102	Lex	De Haan	LDEHAAN	13-JAN-93	AD_VP	17000	90	
103	Alexander	Hunold	AHUNOLD	03-JAN-90	IT_PROG	9000	30	
104	Bruce	Ernst	BERNST	21-MAY-91	IT_PROG	6000	30	
107	Diana	Lorentz	DLORENTZ	07-FEB-99	IT_PROG	4200	30	
124	Kevin	Mourgos	KMOURGOS	16-NOV-99	ST_MAN	5800	50	

Update Statement

```
UPDATE    table
SET       column = value [, column = value, ...]
[WHERE    condition];
```

- Update more than one row at a time, if required.

Updating Rows in a Table

- Specific row or rows are modified if you specify the WHERE clause.

```
UPDATE employees
SET    department_id = 70
WHERE  employee_id = 113;
1 row updated.
```

- All rows in the table are modified if you omit the WHERE clause.

```
UPDATE    copy_emp
SET       department_id = 110;
22       rows updated.
```

Removing a Row from a Table

Departments

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing		
100	Finance		
50	Shipping	124	1500
60	IT	103	1400

Delete a row from the Department S table.

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing		
50	Shipping	124	1500
60	IT	103	1400

The DELETE Statement

- You can remove existing rows from a table by using the Delete statement.

```
DELETE [FROM] table  
[WHERE condition];
```


Deleting Rows from a Table

- Specific rows are deleted if you specify the WHERE clause.

```
DELETE FROM departments
WHERE      department_name = 'Finance';
1 row      deleted.
```

- All rows in the table are deleted if you omit the WHERE clause.

```
DELETE FROM copy_emp;
22      rows      deleted.
```


THANK YOU!