Session 5-2 Data Manipulation

Kyunghee Lee, PhD

Data Manipulation Language (DML)

- Create or modify data values
- Insert/update records into tables
- Copy records from other tables
- Delete records

* DDL is for manipulating *tables* whereas DML for manipulating *records* in the tables

WAYNE STATE
Mike Ilitch School of Business

INSERT Command to Add Rows

- Identify the table in the INSERT INTO clause

- Specify data in the VALUES clause

- Can only add one row at a time to a table

- Enclose nonnumeric data in single quotes

- If a column list is not provided, a value must be assigned to each column in the table

```
INSERT INTO tablename [(columnname, …)]
VALUES (datavalue, …);
```

INSERT Command Examples

Enter SQL Statement:

```
INSERT INTO acctmanager
 VALUES ('T500', 'NICK', 'TAYLOR', '05-SEP-09', 42000, 3500, 'NE');
```

No Column List in INSERT clause

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

1 rows inserted

Enter SQL Statement:

```
INSERT INTO acctmanager (amid, amfirst, amlast, amsal, amcomm, region)
 VALUES ('J500', 'Sammie', 'Jones', 39500, 2000, 'NW');
```

Column List in INSERT clause

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

1 rows inserted

INSERT Command Error – Not Enough Values

Enter SQL Statement:

```
INSERT INTO acctmanager
 VALUES ('J500', 'Sammie', 'Jones', 39500, 2000, 'NW');
```

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

acctmanager table
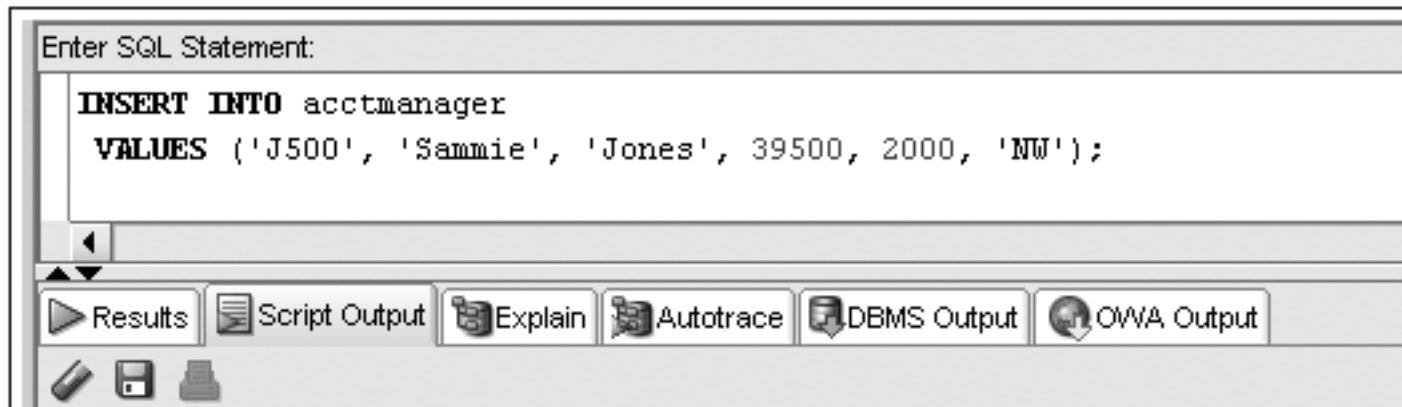
	COLUMN_NAME	DATA_TYPE	NULLABLE	
1	AMID	CHAR(4 BYTE)	No	
2	AMFIRST	VARCHAR2(12 BYTE)	No	
3	AMLAST	VARCHAR2(12 BYTE)	No	
4	AMEDATE	DATE	Yes	S
5	AMSAL	NUMBER(8,2)	Yes	
6	AMCOMM	NUMBER(7,2)	Yes	(
7	REGION	CHAR(2 BYTE)	Yes	

WAYNE STATE
Mike Ilitch School of Business

Inserting NULL Value

- Omit column name from INSERT INTO clause column list

- Substitute two single quotation marks(")

- Use NULL keyword

NULL value input

Enter SQL Statement:

```
INSERT INTO acctmanager
VALUES ('L500', 'MANDY', 'LOPEZ', '01-OCT-09', 47000, 1500, NULL);
```

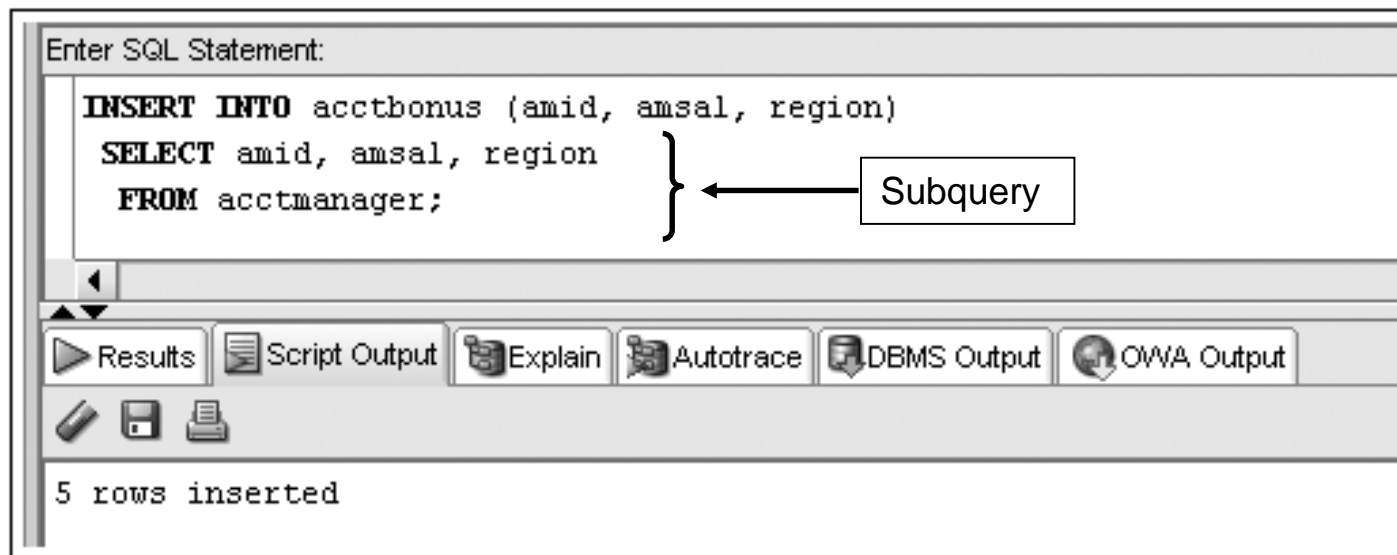Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

1 rows inserted

Inserting Data from an Existing Table

• Substitute subquery for VALUES clause


```
Enter SQL Statement:
    INSERT INTO acctbonus (amid, amsal, region)
    SELECT amid, amsal, region
    FROM acctmanager;                     }  ←—— [Subquery]

    ◄
▲▼
  ► Results  |  ≣ Script Output  |  Explain  |  Autotrace  |  DBMS Output  |  OWA Output
  ✎  💾  🖨
5 rows inserted
```

JustLeeDB.txt Script

Create Customers table

```
CREATE TABLE Customers
(Customer#  NUMBER(4),
LastName VARCHAR2(10) NOT NULL,
FirstName VARCHAR2(10) NOT NULL,
Address VARCHAR2(20),
City VARCHAR2(12),
State VARCHAR2(2),
Zip VARCHAR2(5),
Referred NUMBER(4),
Region CHAR(2),
  CONSTRAINT customers_customer#_pk PRIMARY KEY(customer#),
  CONSTRAINT customers_region_ck
             CHECK (region IN ('N', 'NW', 'NE', 'S', 'SE', 'SW', 'W', 'E')) );
```

Insert new rows into Customers table

```
INSERT INTO CUSTOMERS
VALUES (1001, 'MORALES', 'BONITA', 'P.O. BOX 651', 'EASTPOINT', 'FL', '32328', NULL, 'SE');
INSERT INTO CUSTOMERS
VALUES (1002, 'THOMPSON', 'RYAN', 'P.O. BOX 9835', 'SANTA MONICA', 'CA', '90404', NULL, 'W');
INSERT INTO CUSTOMERS
VALUES (1003, 'SMITH', 'LEILA', 'P.O. BOX 66', 'TALLAHASSEE', 'FL', '32306', NULL, 'SE');
INSERT INTO CUSTOMERS
VALUES (1004, 'PIERSON', 'THOMAS', '69821 SOUTH AVENUE', 'BOISE', 'ID', '83707', NULL, 'NW');
INSERT INTO CUSTOMERS
VALUES (1005, 'GIRARD', 'CINDY', 'P.O. BOX 851', 'SEATTLE', 'WA', '98115', NULL, 'NW');
INSERT INTO CUSTOMERS
VALUES (1006, 'CRUZ', 'MESHIA', '82 DIRT ROAD', 'ALBANY', 'NY', '12211', NULL, 'NE');
INSERT INTO CUSTOMERS
VALUES (1007, 'GIANA', 'TAMMY', '9153 MAIN STREET', 'AUSTIN', 'TX', '78710', 1003, 'SW');
INSERT INTO CUSTOMERS
VALUES (1008, 'JONES', 'KENNETH', 'P.O. BOX 137', 'CHEYENNE', 'WY', '82003', NULL, 'N');
INSERT INTO CUSTOMERS
VALUES (1009, 'PEREZ', 'JORGE', 'P.O. BOX 8564', 'BURBANK', 'CA', '91510', 1003, 'W');
INSERT INTO CUSTOMERS
VALUES (1010, 'LUCAS', 'JAKE', '114 EAST SAVANNAH', 'ATLANTA', 'GA', '30314', NULL, 'SE');
INSERT INTO CUSTOMERS
VALUES (1011, 'MCGOVERN', 'REESE', 'P.O. BOX 18', 'CHICAGO', 'IL', '60606', NULL, 'N');
INSERT INTO CUSTOMERS
VALUES (1012, 'MCKENZIE', 'WILLIAM', 'P.O. BOX 971', 'BOSTON', 'MA', '02110', NULL, 'NE');
INSERT INTO CUSTOMERS
```

UPDATE Command to Modify Existing Rows

- Modify rows using UPDATE command
- Use UPDATE command to:
  - Add values to an existing row (replace NULL values)
  - Change existing values

WAYNE STATE
Mike Ilitch School of Business

UPDATE Command Syntax

- UPDATE clause identifies table

- SET clause identifies column(s) being changed and new value(s)

- Optional WHERE clause specifies row(s) to be changed – if omitted, all rows will be updated!

```
UPDATE tablename
SET columnname = new_datavalue, …
[WHERE condition];
```

FIGURE 5-18 Syntax for the UPDATE command

UPDATE Command Example


```
Enter SQL Statement:

UPDATE acctmanager
  SET amedate = '10-OCT-09',
      region = 'S'
  WHERE amid = 'L500';
```

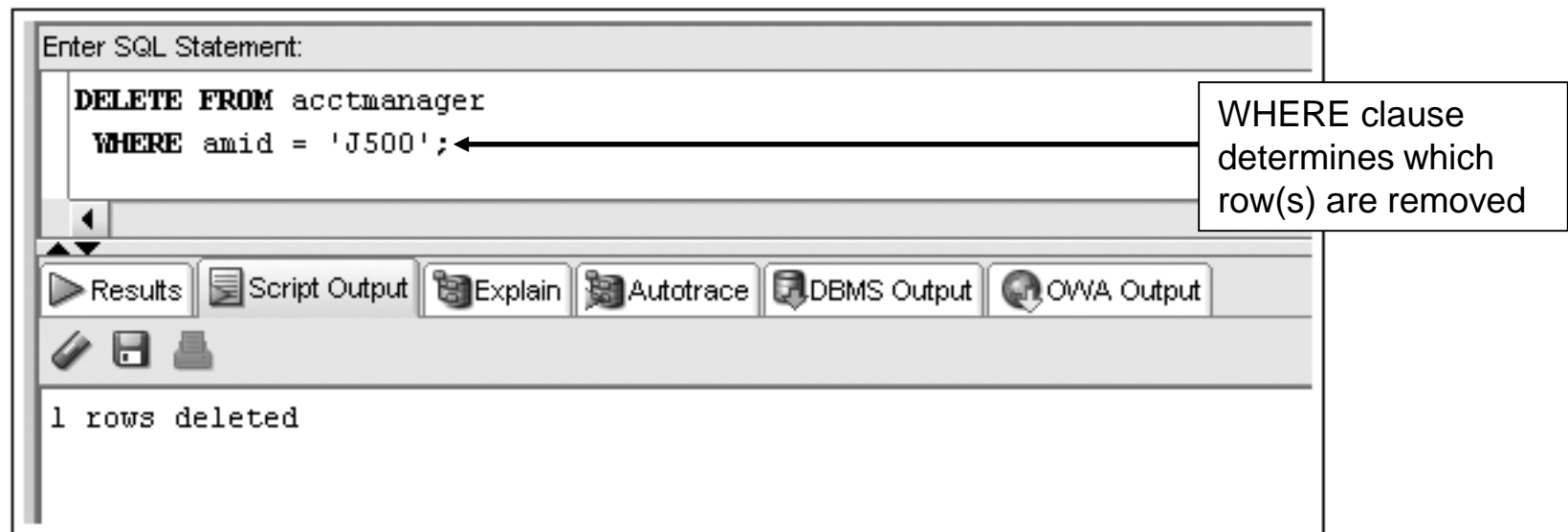Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

```
1 rows updated
```

DELETE Command to Delete Rows

- DELETE command removes a row from a table

Enter SQL Statement:

```
DELETE FROM acctmanager
  WHERE amid = 'J500';
```

WHERE clause determines which row(s) are removed

▷ Results   ▤ Script Output   ▦ Explain   ▦ Autotrace   ▦ DBMS Output   ▦ OWA Output

1 rows deleted

WAYNE STATE
Mike Ilitch School of Business

DELETE Command – Omitting WHERE Clause

- Omitting WHERE clause removes **all rows**
- Example below removes all rows from the acctmanager2 table

```
DELETE FROM acctmanager2;
```

FIGURE 5-27 DELETE command without the WHERE clause

Wrap-up

- INSERT to add rows
- UPDATE to modify existing rows
- DELETE to remove rows

WAYNE STATE
Mike Ilitch School of Business