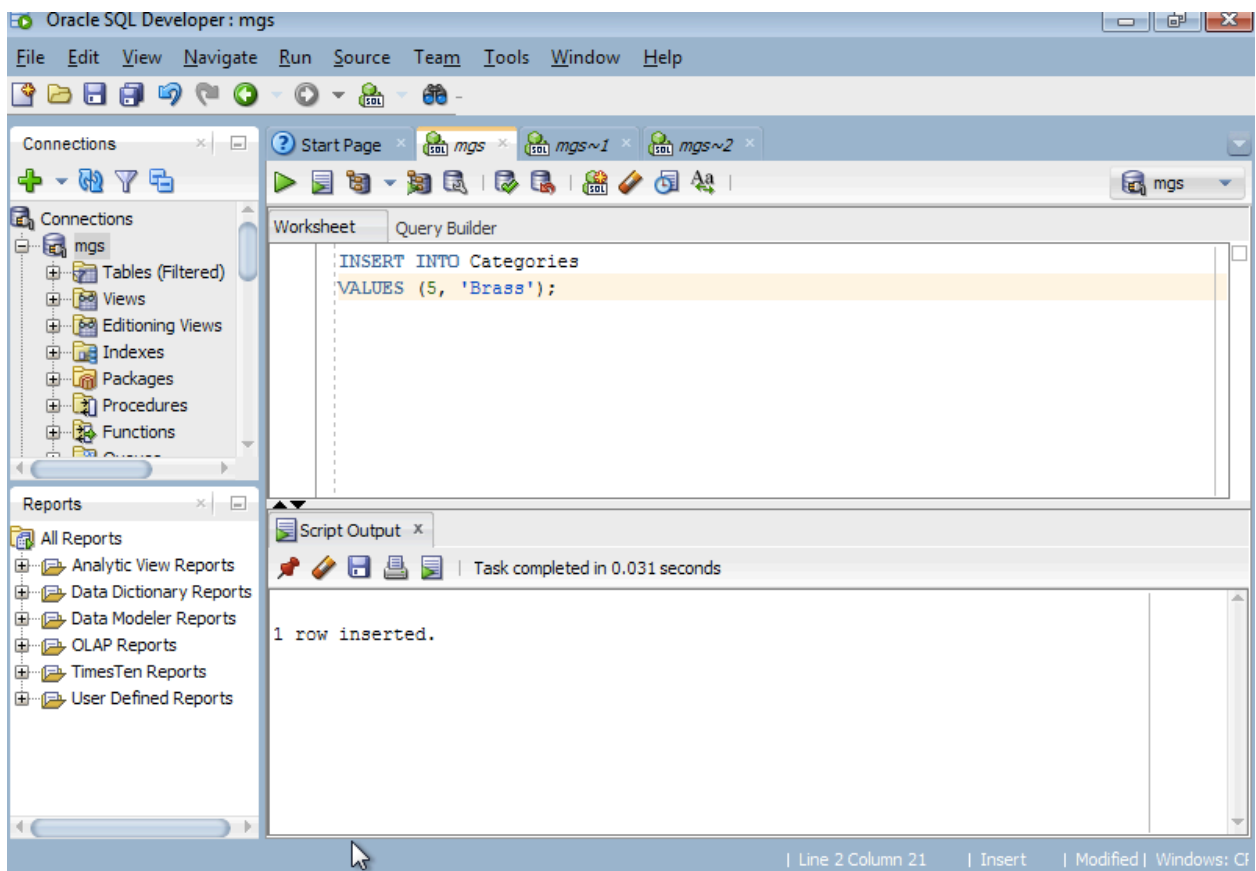


## Domas Budrys – Assignment 4

### CHAPTER 7

1.

INSERT INTO Categories  
VALUES (5, 'Brass');



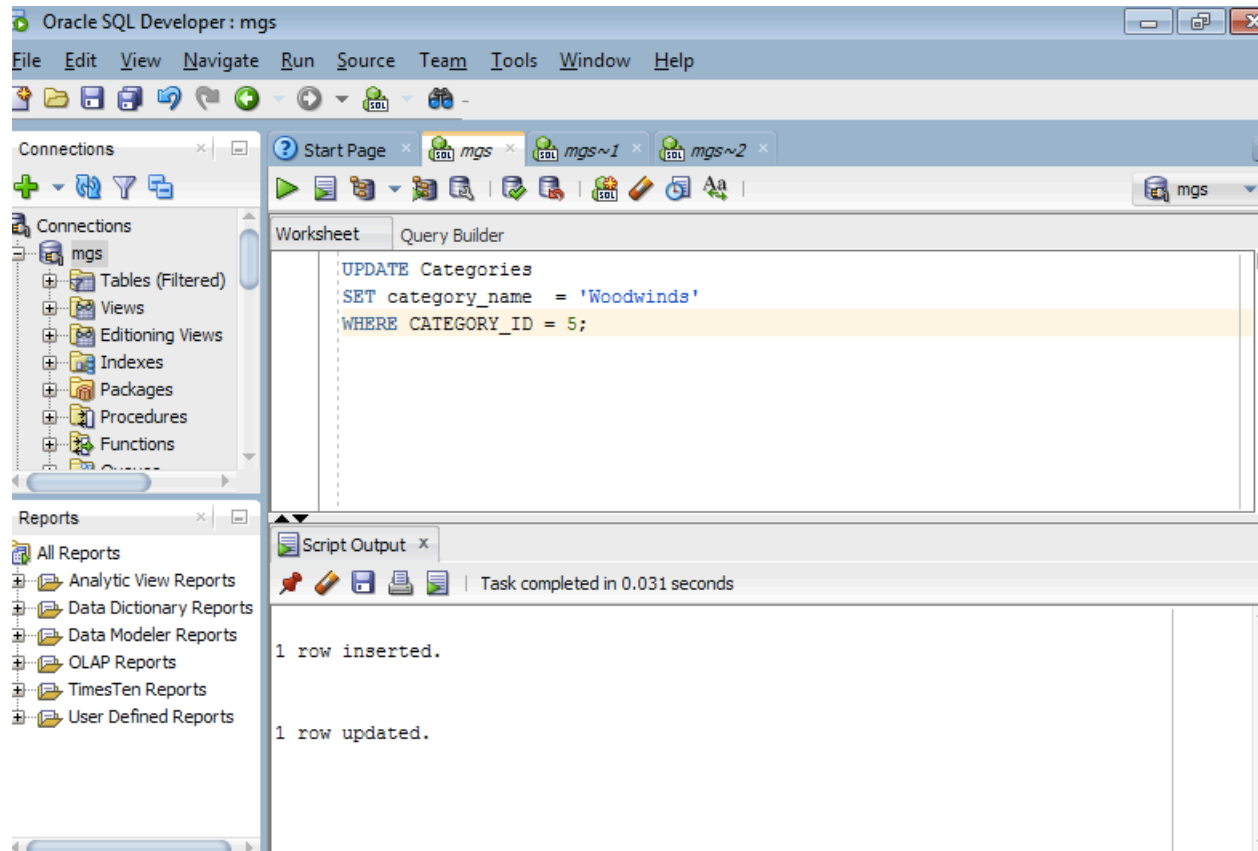
Category\_ID = 5 needed to be insert instead of 4, because category\_id is a primary key and 4 was already in the database.

2.

UPDATE Categories

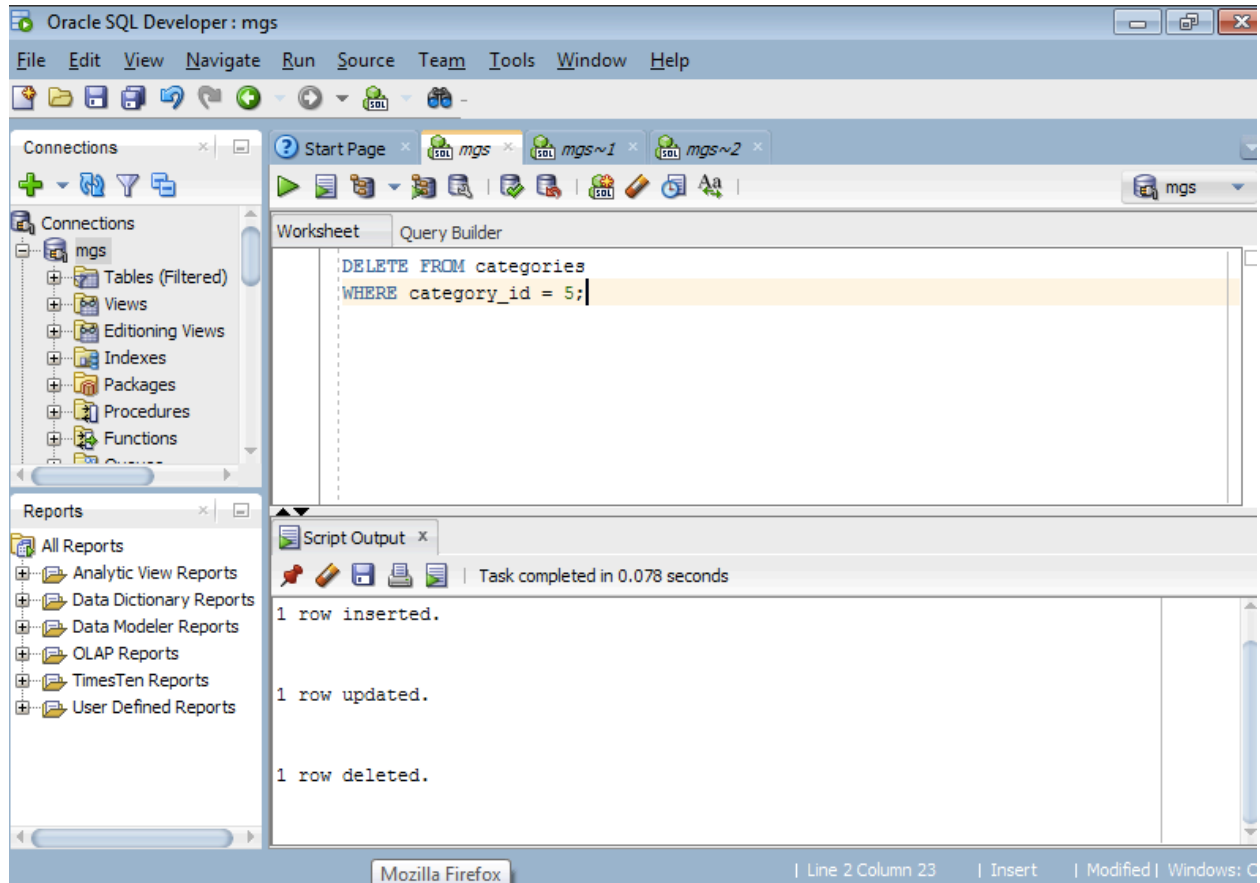
SET category\_name = 'Woodwinds'

WHERE CATEGORY\_ID = 5;



3.

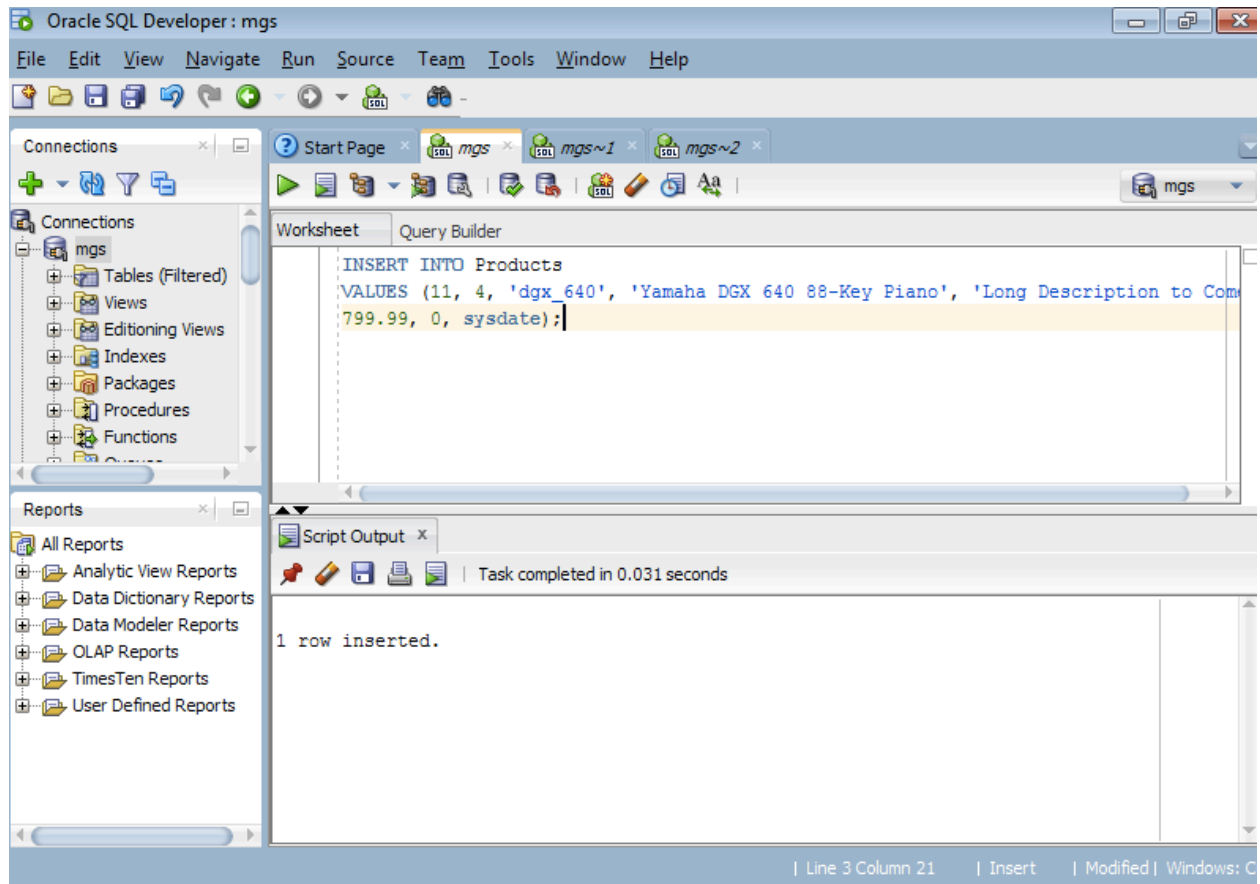
DELETE FROM categories  
WHERE category\_id = 5;



4.

INSERT INTO Products

VALUES (11, 4, 'dgx\_640', 'Yamaha DGX 640 88-Key Piano', 'Long Description to Come',  
799.99, 0, sysdate);

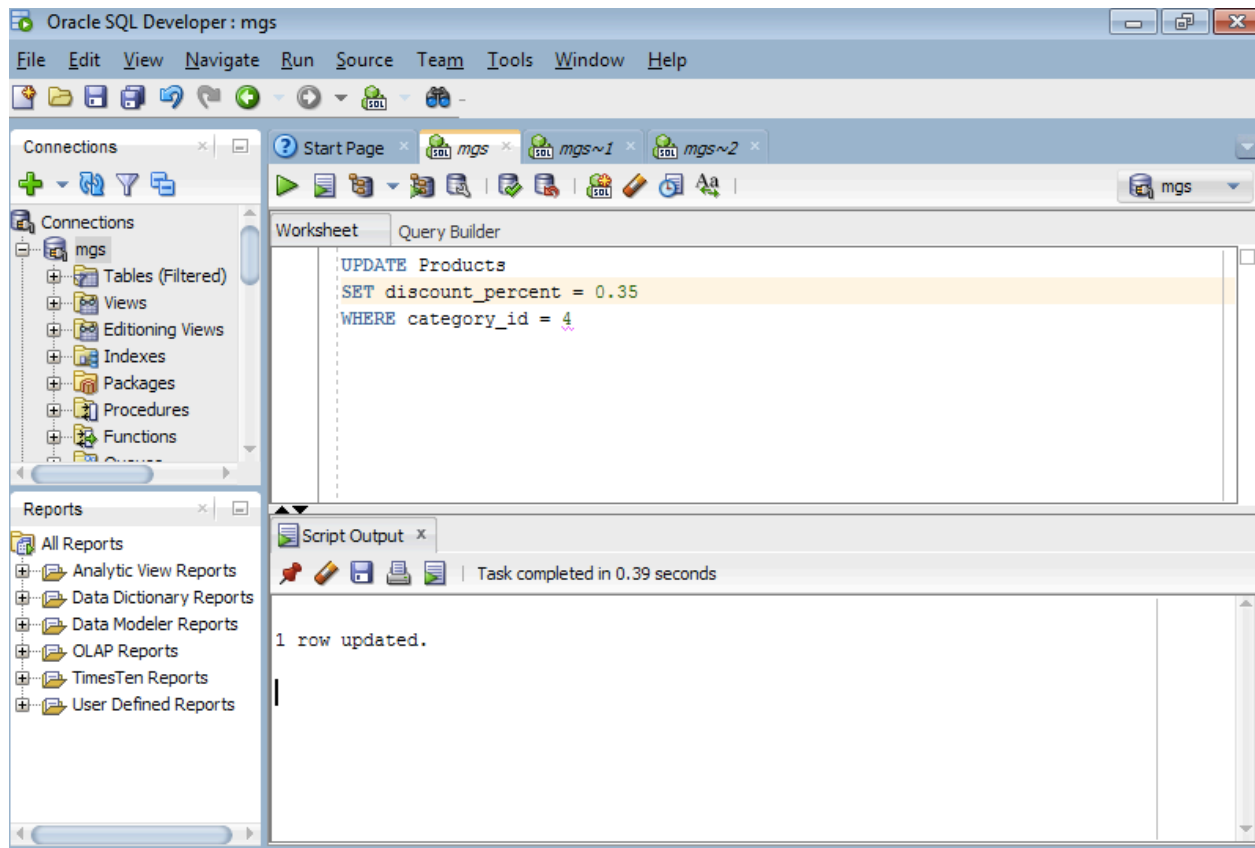


5.

UPDATE Products

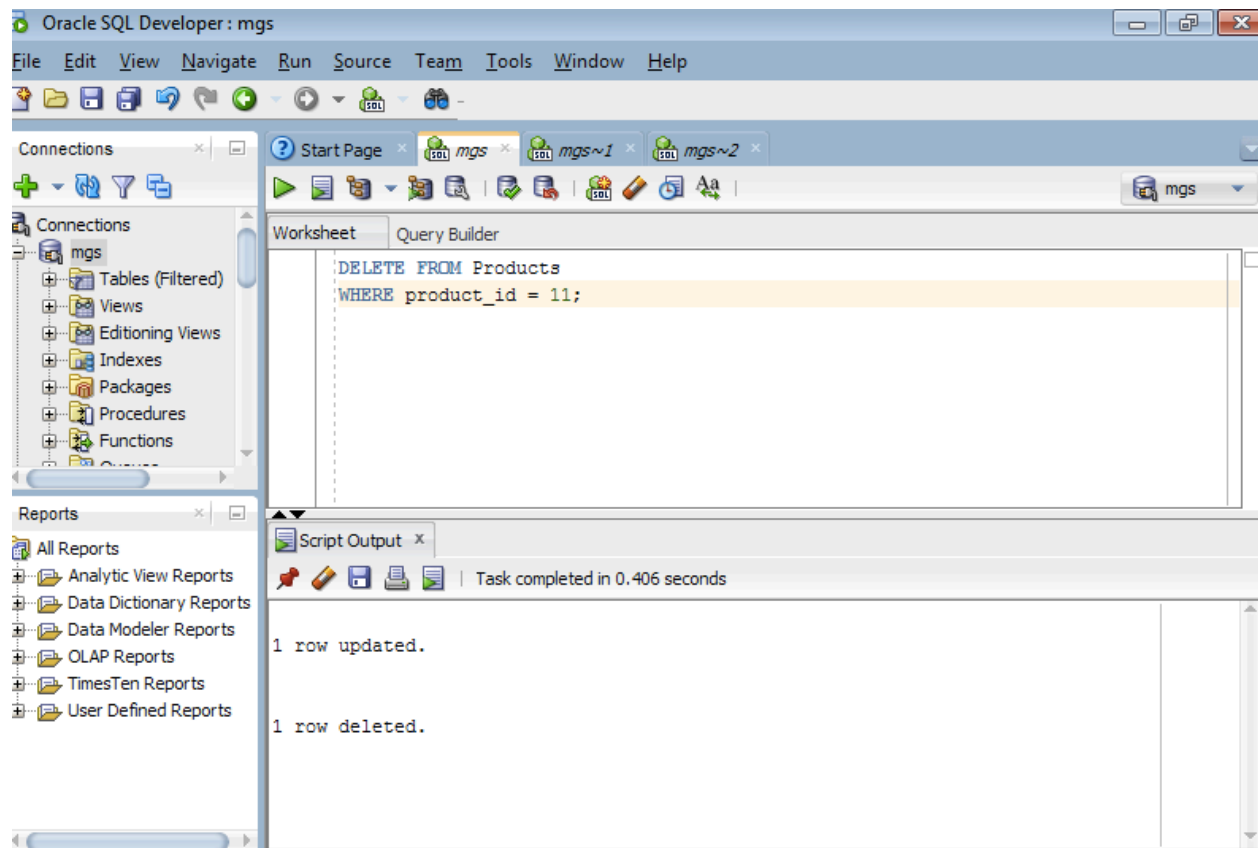
SET discount\_percent = 0.35

WHERE category\_id = 4



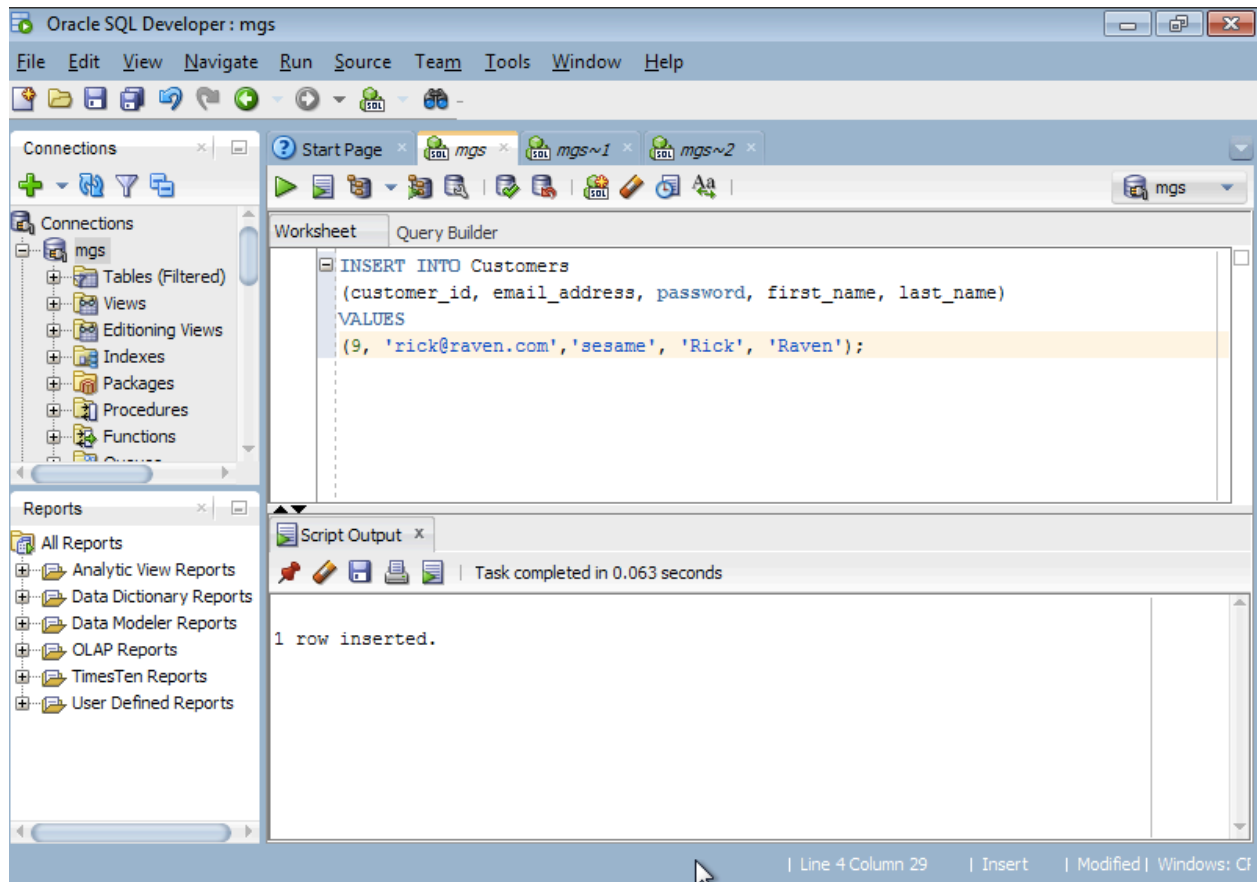
6.

DELETE FROM Products  
WHERE product\_id = 11;



7.

```
INSERT INTO Customers  
(customer_id, email_address, password, first_name, last_name)  
VALUES  
(9, 'rick@raven.com', 'sesame', 'Rick', 'Raven');
```

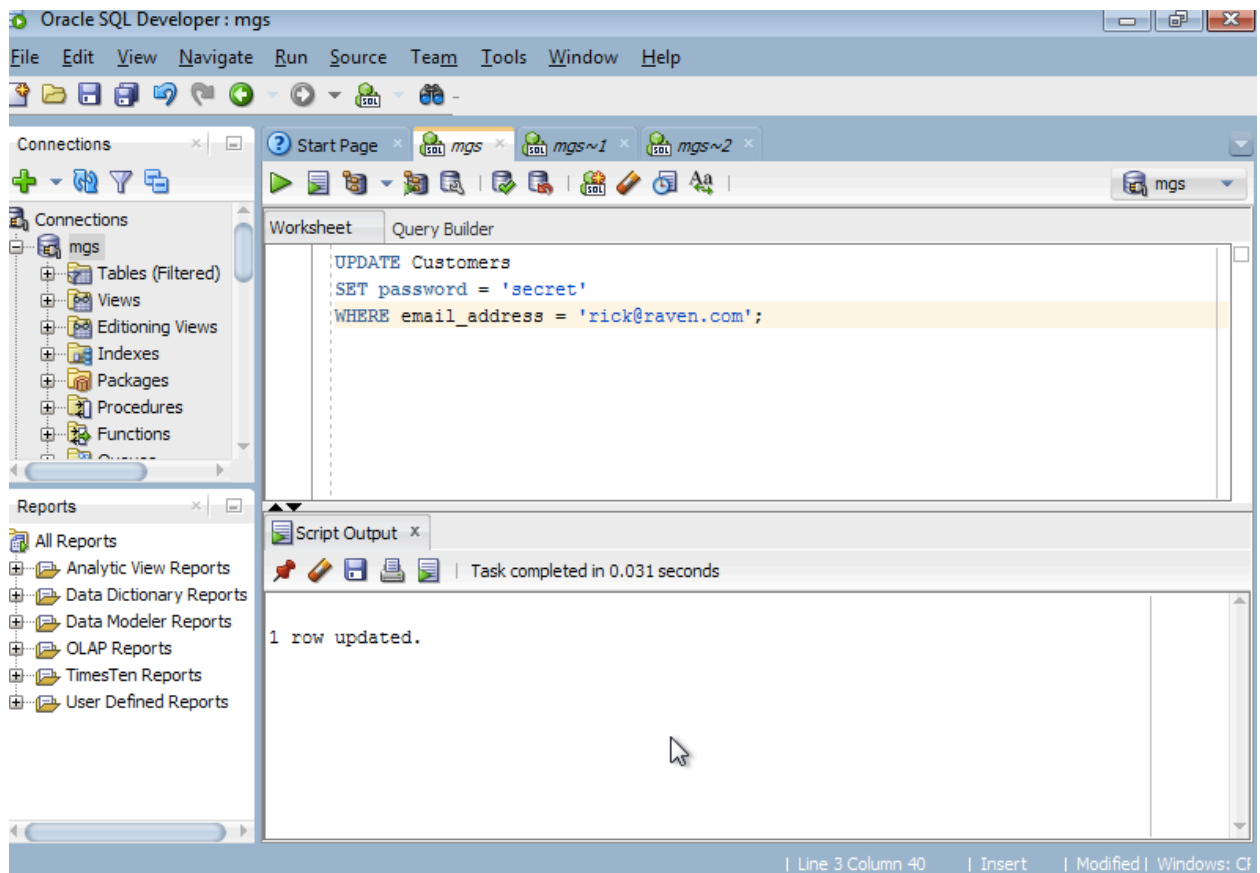


8.

UPDATE Customers

SET password = 'secret'

WHERE email\_address = 'rick@raven.com';

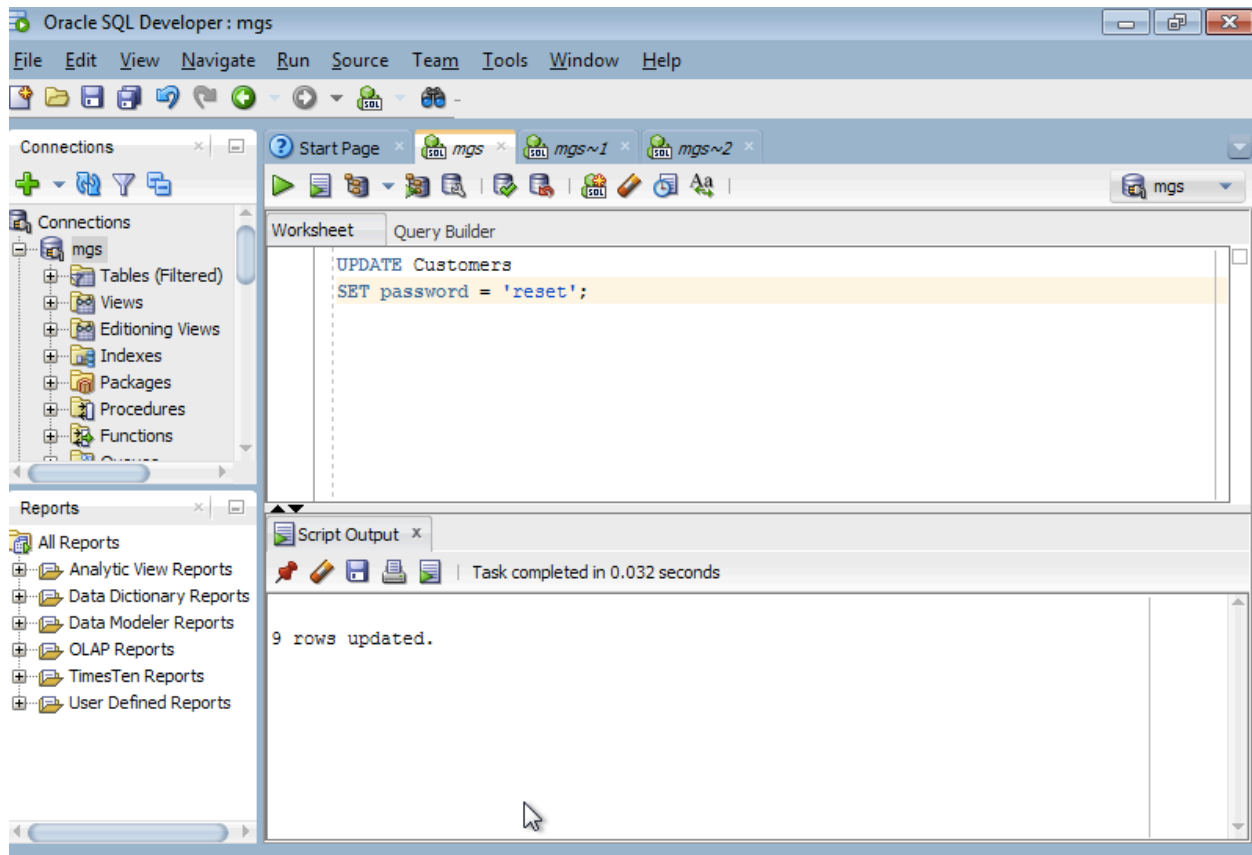




9.

UPDATE Customers

SET password = 'reset';



## CHAPTER 8

### 1.

SELECT list\_price, TO\_CHAR(list\_price,'\$99,999.99') AS list\_price  
FROM Products

The screenshot shows the Oracle SQL Developer interface. The main window displays a query in the SQL editor:

```
SELECT list_price, TO_CHAR(list_price,'$99,999.99') AS list_price  
FROM Products
```

Below the editor, the Query Result window shows the results of the query. The status bar indicates "All Rows Fetched: 10 in 0 seconds". The results are displayed in a table with two columns: LIST\_PRICE and LIST\_PRICE\_1.

	LIST_PRICE	LIST_PRICE_1
1	699	\$699.00
2	1199	\$1,199.00
3	2517	\$2,517.00
4	489.99	\$489.99
5	299	\$299.00
6	415	\$415.00
7	799.99	\$799.99

The bottom status bar shows the cursor position: "Line 4 Column 1 | Insert | Modified | Windows: CF".

2.

```
SELECT date_added,  
CAST(date_added AS VARCHAR2(9)) AS varchar_date  
FROM Products
```

The screenshot displays the Oracle SQL Developer interface. The main window is titled 'Oracle SQL Developer : mgs~1'. The 'Connections' pane on the left shows a tree view with 'mgs' and 'system' connections. The 'Query Builder' tab is active, showing the following SQL query:

```
SELECT date_added,  
CAST(date_added AS VARCHAR2(9)) AS varchar_date  
FROM Products
```

The 'Query Result' pane at the bottom shows the execution results. It indicates 'All Rows Fetched: 10 in 0.016 seconds'. The results are displayed in a table with two columns: 'DATE\_ADDED' and 'VARCHAR\_DATE'.

	DATE_ADDED	VARCHAR_DATE
1	30-OCT-11	30-OCT-11
2	05-DEC-11	05-DEC-11
3	04-FEB-12	04-FEB-12
4	01-JUN-12	01-JUN-12
5	30-JUL-12	30-JUL-12
6	30-JUL-12	30-JUL-12
7	01-JUN-12	01-JUN-12

The status bar at the bottom indicates 'Line 4 Column 1 | Insert | Modified | Windows: CF'.

3.

```
SELECT list_price, discount_percent,  
ROUND (list_price *(discount_percent *0.01), 2) AS "Discount Amount"  
FROM Products
```

The screenshot shows the Oracle SQL Developer interface. The 'Connections' pane on the left lists 'mgs' and 'system'. The 'Query Builder' tab is active, displaying the SQL query: `SELECT list_price, discount_percent, ROUND (list_price *(discount_percent *0.01), 2) AS "Discount Amount" FROM Products`. The 'Query Result' pane below shows the execution results with 10 rows fetched in 0 seconds. The results are displayed in a table with columns: LIST\_PRICE, DISCOUNT\_PERCENT, and Discount Amount.

	LIST_PRICE	DISCOUNT_PERCENT	Discount Amount
1	699	30	209.7
2	1199	30	359.7
3	2517	52	1308.84
4	489.99	38	186.2
5	299	0	0
6	415	39	161.85
7	799.99	30	240

4.

```
SELECT TO_CHAR(order_date, 'YYYY') AS "Order Year",  
TO_CHAR(order_date, 'MON-DD-YYYY') AS "Order Date",  
TO_CHAR(order_date, 'HH:MI AM') AS "Order Time",  
TO_CHAR(order_date, 'MM/DD/YY HH:MI:SS') AS "Order Date and Time"  
FROM ORDERS;
```

The screenshot displays the Oracle SQL Developer interface. The main window shows a worksheet with the following SQL query:

```
SELECT TO_CHAR(order_date, 'YYYY') AS "Order Year",  
TO_CHAR(order_date, 'MON-DD-YYYY') AS "Order Date",  
TO_CHAR(order_date, 'HH:MI AM') AS "Order Time",  
TO_CHAR(order_date, 'MM/DD/YY HH:MI:SS') AS "Order Date and Time"  
FROM ORDERS;
```

Below the worksheet, the 'Query Result' tab is active, showing the results of the query. The status bar indicates 'All Rows Fetched: 9 in 0.016 seconds'. The results are displayed in a table with the following columns: Order Year, Order Date, Order Time, and Order Date and Time. The table contains 7 rows of data.

	Order Year	Order Date	Order Time	Order Date and Time
1	2012	MAR-28-2012	09:40 AM	03/28/12 09:40:28
2	2012	MAR-28-2012	11:23 AM	03/28/12 11:23:20
3	2012	MAR-29-2012	09:44 AM	03/29/12 09:44:58
4	2012	MAR-30-2012	03:22 PM	03/30/12 03:22:31
5	2012	MAR-31-2012	05:43 AM	03/31/12 05:43:11
6	2012	MAR-31-2012	06:37 PM	03/31/12 06:37:22
7	2012	APR-01-2012	11:11 PM	04/01/12 11:11:12

5.

```
SELECT card_number, LENGTH(card_number) AS "Card Number Length",  
SUBSTR(card_number, 13) AS "Last 4 Digits",  
REPLACE(card_number, card_number, 'XXXX-XXXX-XXX-' || SUBSTR(card_number, 13, LEN  
LENGTH(card_number))) AS "XXXX-.... Format"
```

FROM Orders

The screenshot shows the Oracle SQL Developer interface. The main window displays a SQL query in the Worksheet tab. The query is as follows:

```
SELECT card_number, LENGTH(card_number) AS "Card Number Length",  
SUBSTR(card_number, 13) AS "Last 4 Digits",  
REPLACE(card_number, card_number, 'XXXX-XXXX-XXX-' || SUBSTR(card_number, 13, LEN  
LENGTH(card_number))) AS "XXXX-.... Format"  
  
FROM Orders
```

The Query Result tab shows the results of the query. The results are displayed in a table with the following columns: CARD\_NUMBER, Card Number Length, Last 4 Digits, and XXXX-.... Format. The table contains 7 rows of data.

	CARD_NUMBER	Card Number Length	Last 4 Digits	XXXX-.... Format
1	4111111111111111	16	1111	XXXX-XXXX-XXX-1111
2	4012888888881881	16	1881	XXXX-XXXX-XXX-1881
3	4111111111111111	16	1111	XXXX-XXXX-XXX-1111
4	378282246310005	16	005	XXXX-XXXX-XXX-005
5	4111111111111111	16	1111	XXXX-XXXX-XXX-1111
6	6011111111111117	16	1117	XXXX-XXXX-XXX-1117
7	5555555555554444	16	4444	XXXX-XXXX-XXX-4444

6.

```
SELECT order_id, order_date,  
order_date + INTERVAL '2'DAY AS approx_ship_date,  
ship_date,  
SUBSTR (LTRIM (TO_TIMESTAMP(ship_date, 'DD.MM.YYYY') - TO_TIMESTAMP(order_date,  
'DD.MM.YYYY'), '+0000000'), 1, 1) AS days_to_ship  
FROM Orders
```

```
WHERE TO_CHAR(order_date, 'MM-YY') = '05-12';
```

