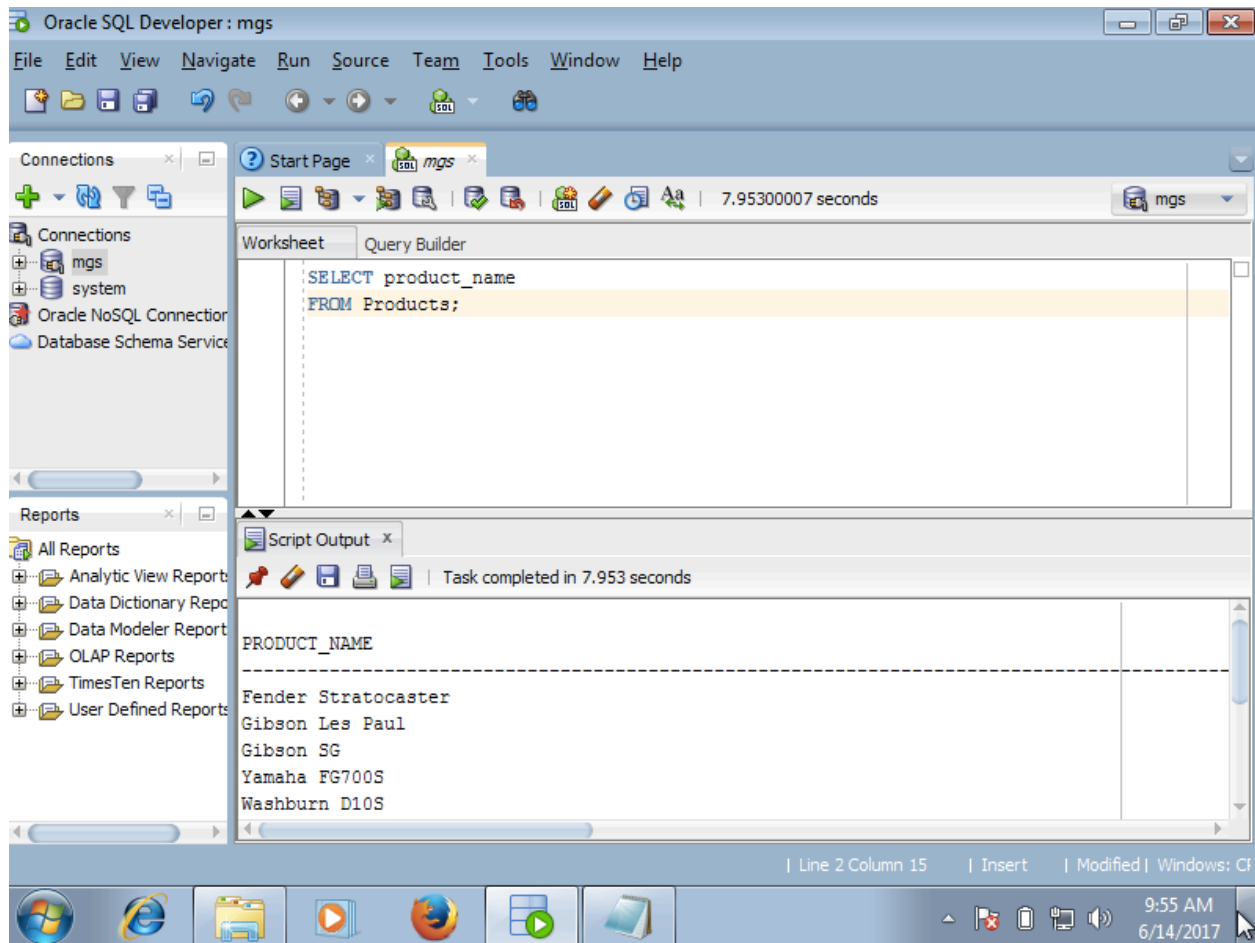


Domas Budrys CSCI-4430
Assignment 1

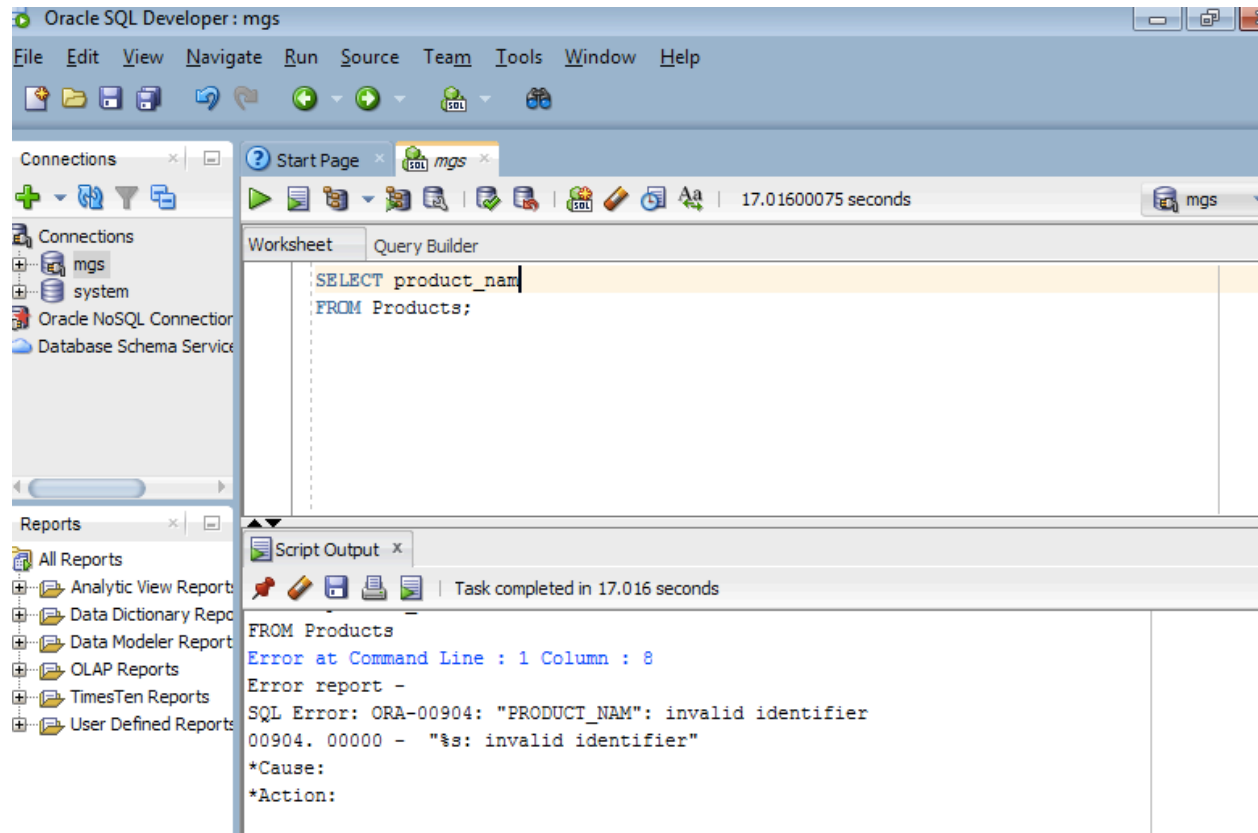
Chapter 2:

11.



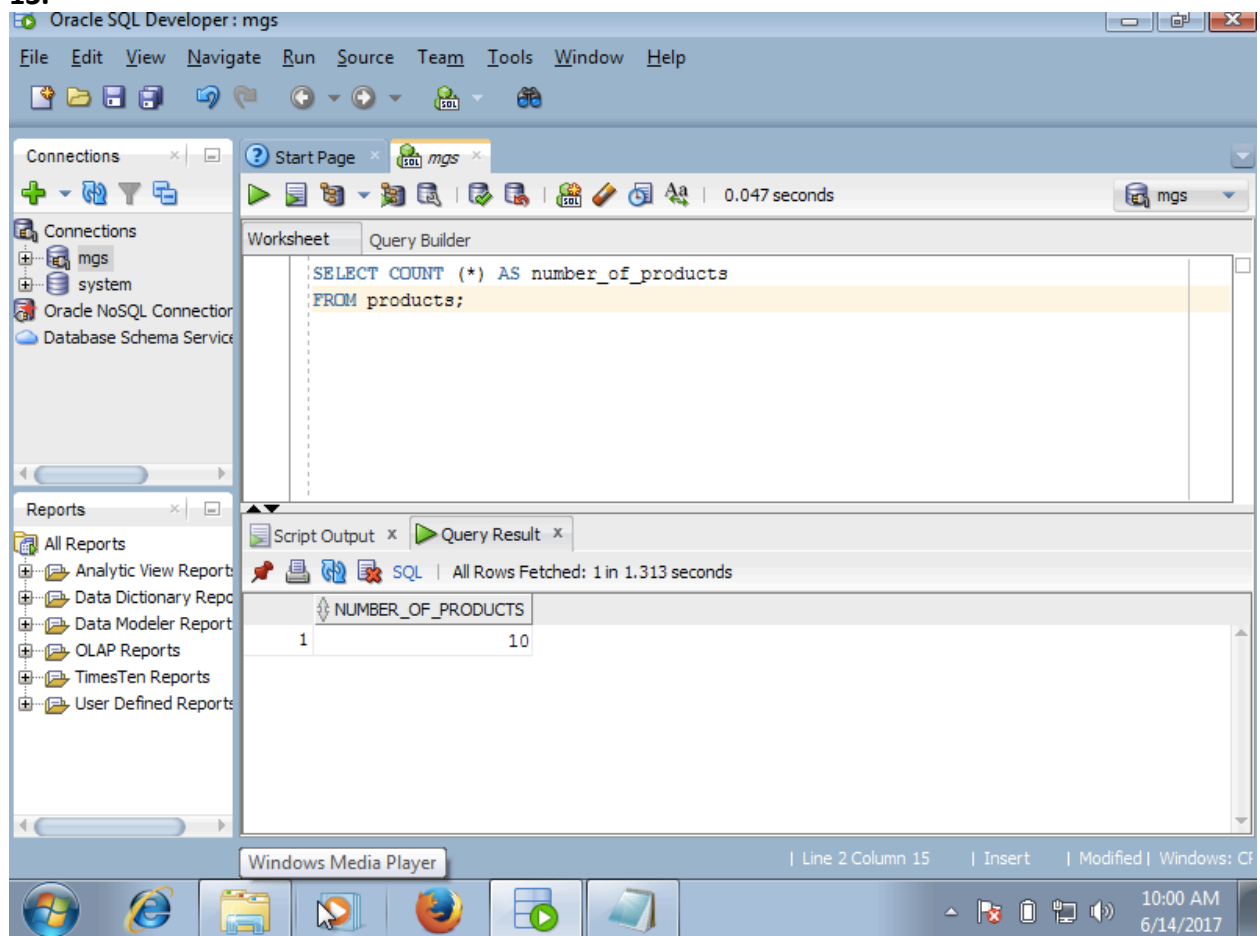
SELECT product_name
FROM Products;

12.



SELECT product_nam
FROM Products;

13.



```
SELECT COUNT (*) AS number_of_products
FROM products;
```

18.

The screenshot shows the Oracle SQL Developer interface. The main window displays a SQL script in the 'Worksheet' tab. The script consists of two queries. The first query is selected and highlighted in yellow:

```
SELECT product_name, list_price, date_added
FROM products
ORDER BY product_name;
```

The second query is also visible:

```
SELECT COUNT(*) AS number_of_products,
       MAX(list_price) AS most_expensive_product,
       MIN(date_added) AS oldest_product
FROM products;
```

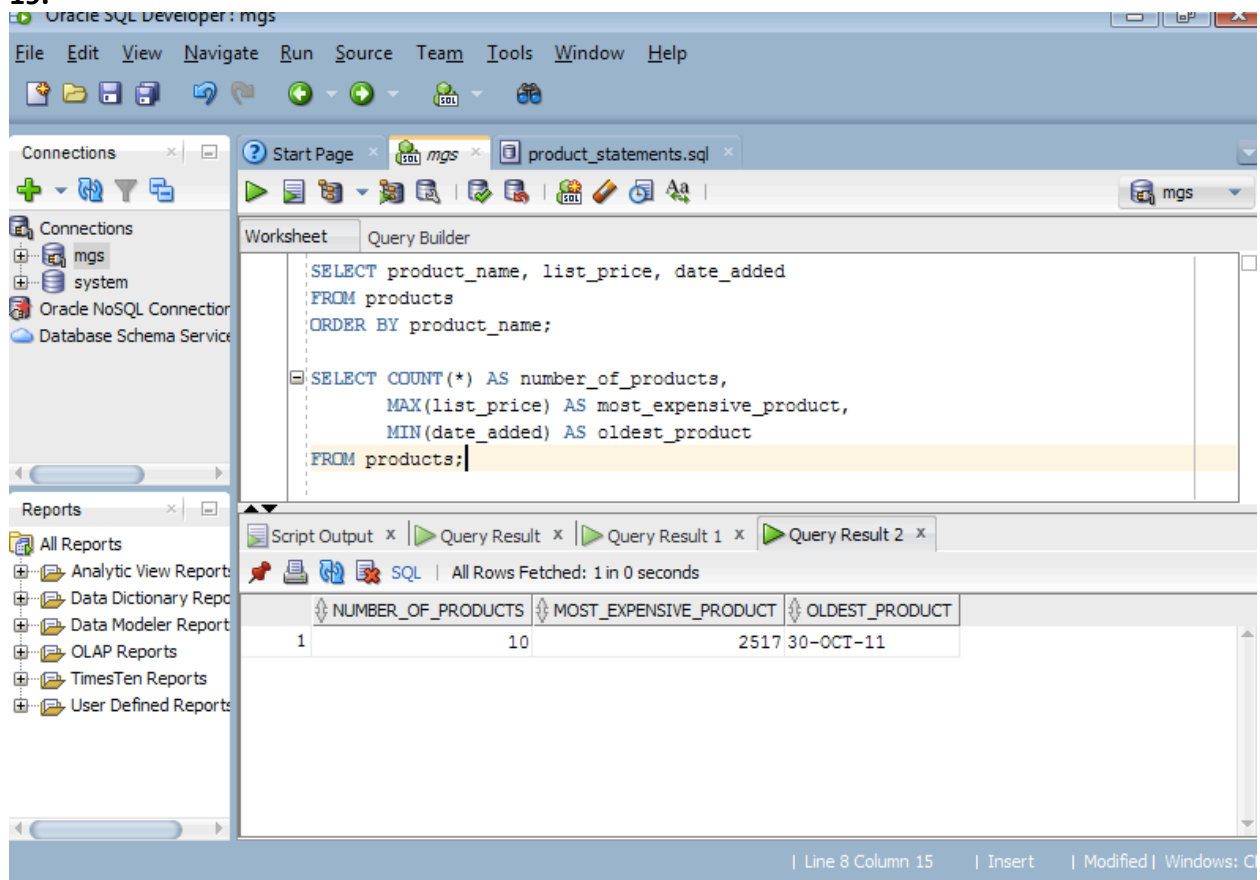
Below the script, the 'Query Result' tab shows the results of the first query. The results are displayed in a table with three columns: PRODUCT_NAME, LIST_PRICE, and DATE_ADDED. The table contains 7 rows of data.

	PRODUCT_NAME	LIST_PRICE	DATE_ADDED
1	Fender Precision	799.99	01-JUN-12
2	Fender Stratocaster	699.30	30-OCT-11
3	Gibson Les Paul	1199.05	05-DEC-11
4	Gibson SG	2517.04	04-FEB-12
5	Hofner Icon	499.99	30-JUL-12
6	Ludwig 5-piece Drum Set with Cymbals	699.99	30-JUL-12
7	Rodriguez Caballero 11	415.30	03-JUL-12

The status bar at the bottom indicates 'Line 3 Column 23', 'Insert' mode, and the date '10:05 AM 6/14/2017'.

```
SELECT product_name, list_price, date_added
FROM products
ORDER BY product_name;
```

19.



```
SELECT COUNT(*) AS number_of_products,
       MAX(list_price) AS most_expensive_product,
       MIN(date_added) AS oldest_product
FROM products;
```

Chapter 3:

1.

The screenshot shows the Oracle SQL Developer interface. The main window displays a query in the 'Query Builder' tab:

```
SELECT product_code, product_name, list_price, discount_percent
FROM products
ORDER BY LIST_PRICE DESC;
```

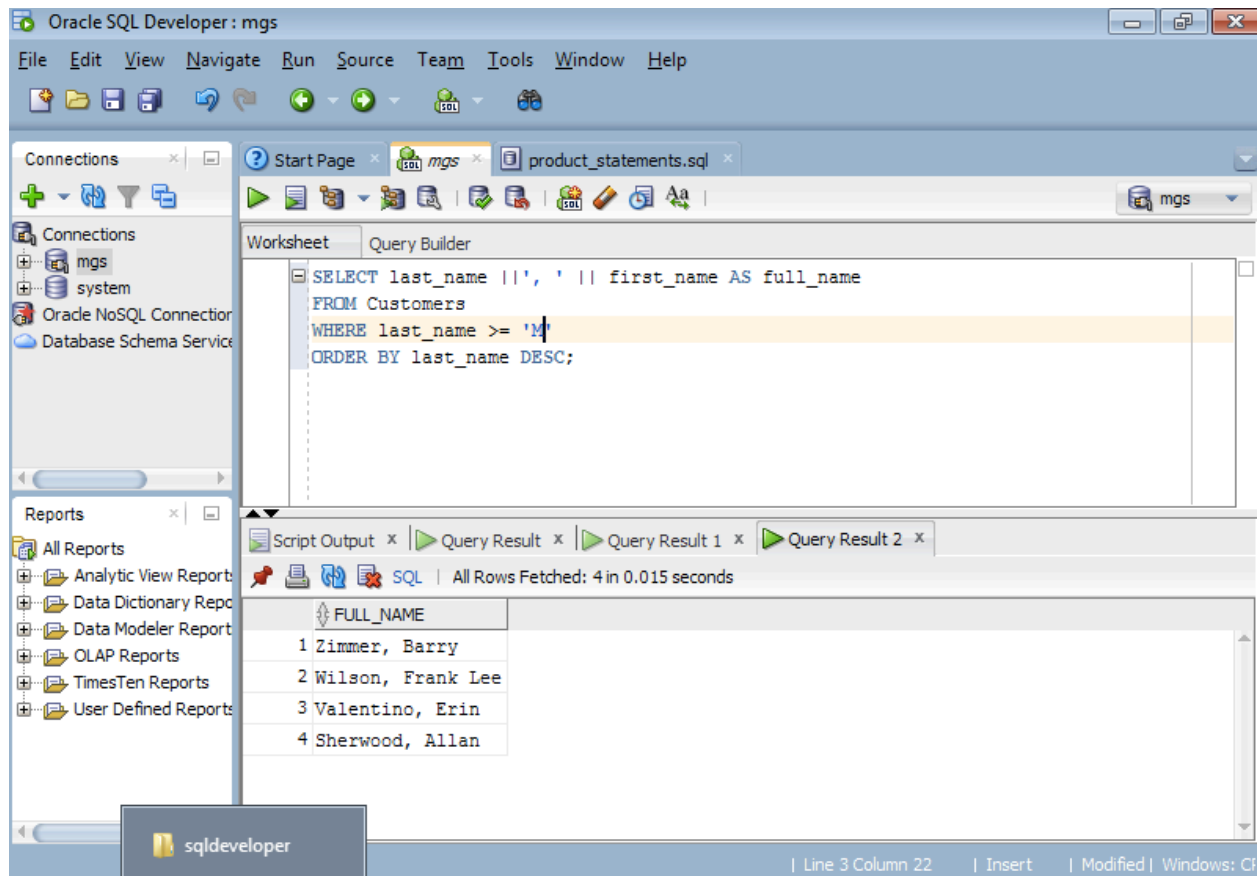
The 'Connections' pane on the left shows the 'mgs' connection selected. The 'Reports' pane on the left shows 'All Reports' selected. The 'Script Output' pane at the bottom shows the query results:

	PRODUCT_CODE	PRODUCT_NAME	LIST_PRICE	DISCOUNT_PERCENT
1	sg	Gibson SG	2517	52
2	les_paul	Gibson Les Paul	1199	30
3	tama	Tama 5-Piece Drum Set with Cymbals	799.99	15
4	precision	Fender Precision	799.99	30
5	ludwig	Ludwig 5-piece Drum Set with Cymbals	699.99	30
6	strat	Fender Stratocaster	699	30
7	hofner	Hofner Icon	499.99	25

The status bar at the bottom indicates 'Line 3 Column 26', 'Insert', 'Modified', and 'Windows: CF'.

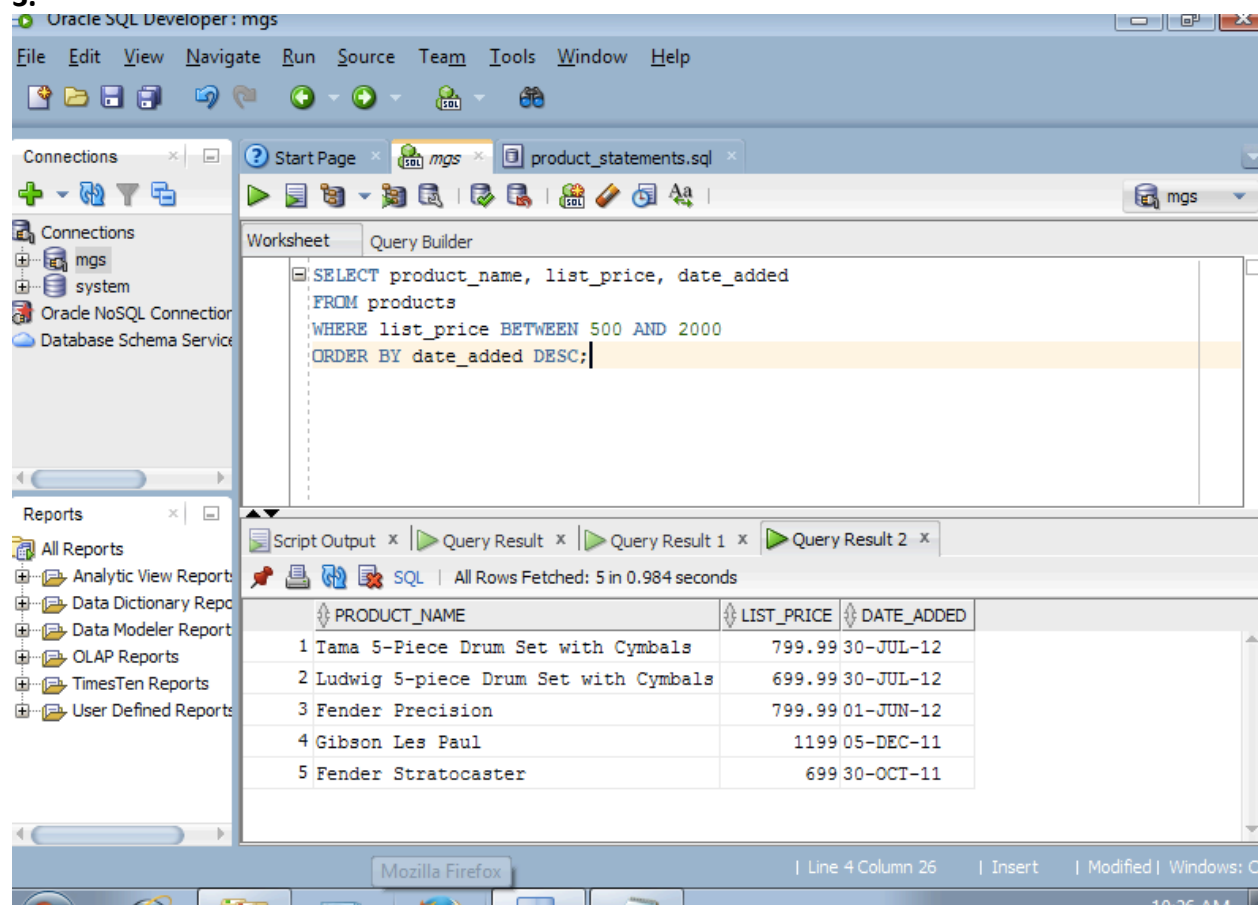
```
SELECT product_code, product_name, list_price, discount_percent
FROM products
ORDER BY LIST_PRICE DESC;
```

2.



```
SELECT last_name || ', ' || first_name AS full_name
FROM Customers
WHERE last_name >= 'M'
```

3.



```
SELECT product_name, list_price, date_added  
FROM products  
WHERE list_price BETWEEN 500 AND 2000  
ORDER BY date_added DESC;
```


4.

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 product_name, list_price, discount_percent,
       (discount_percent * 0.01) * list_price AS discount_amount,
       list_price - (discount_percent * 0.01) * list_price AS discount_price
FROM products
WHERE ROWNUM <= 5
ORDER BY discount_price DESC;
```

The Results pane shows the output of the query, displaying 5 rows of data. The columns are: PRODUCT_NAME, LIST_PRICE, DISCOUNT_PERCENT, DISCOUNT_AMOUNT, and DISCOUNT_PRICE. The data is sorted by DISCOUNT_PRICE in descending order.

	PRODUCT_NAME	LIST_PRICE	DISCOUNT_PERCENT	DISCOUNT_AMOUNT	DISCOUNT_PRICE
1	Gibson SG	2517	52	1308.84	1208.16
2	Gibson Les Paul	1199	30	359.7	839.3
3	Fender Stratocaster	699	30	209.7	489.3
4	Yamaha FG700S	489.99	38	186.1962	303.7938
5	Washburn D10S	299	0	0	299

```
SELECT product_name, list_price, discount_percent,
       (discount_percent * 0.01) * list_price AS discount_amount,
       list_price - (discount_percent * 0.01) * list_price AS discount_price
FROM products
WHERE ROWNUM <= 5
ORDER BY discount_price DESC;
ORDER BY last_name DESC;
```

5.

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

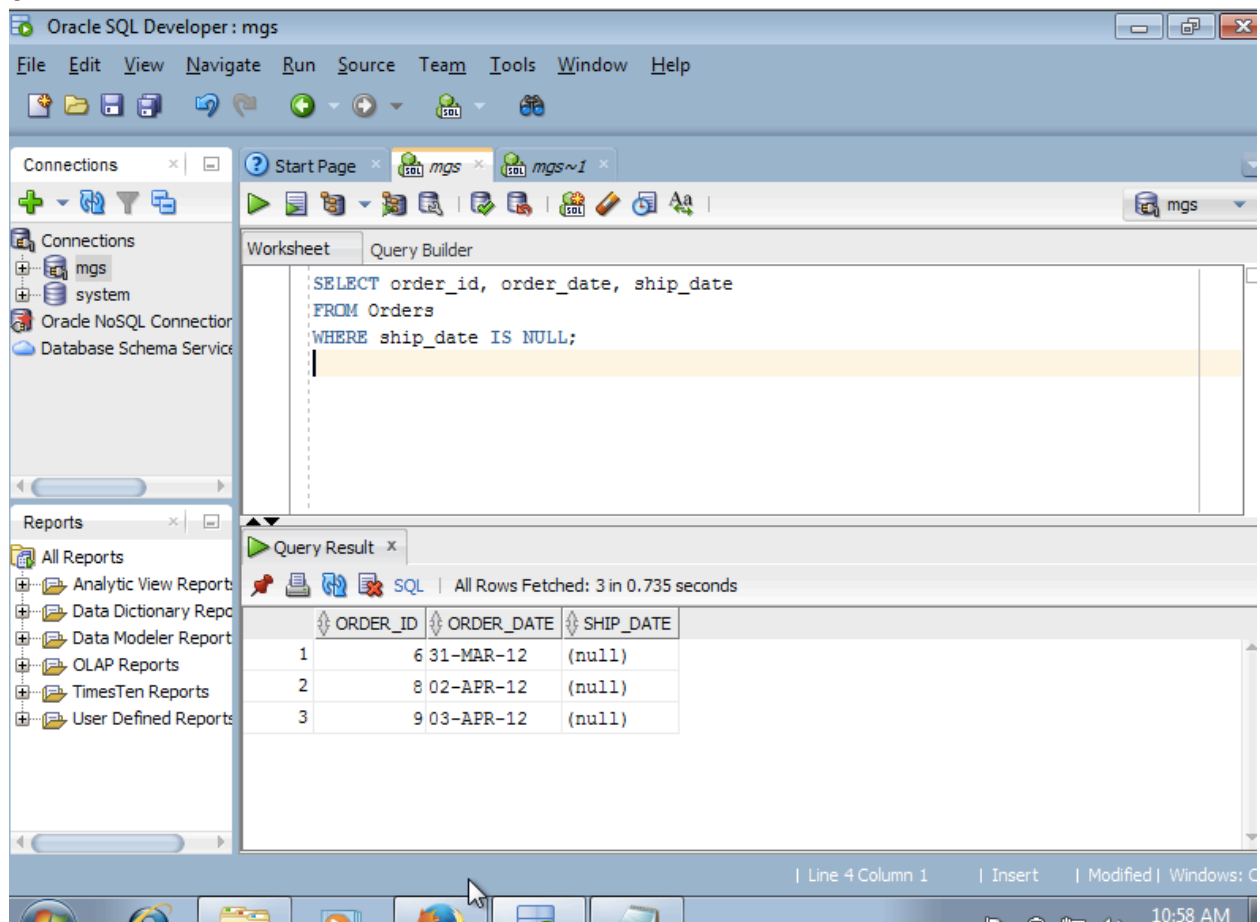
```
SELECT item_id, item_price, discount_amount, quantity,
       item_price * quantity AS price_total,
       discount_amount * quantity AS discount_total,
       (item_price - discount_amount * quantity) * quantity AS item_total
FROM Order_Items
WHERE (item_price - discount_amount * quantity) * quantity > 500
ORDER BY item_total DESC;
```

The Query Result window shows the following data:

ITEM_ID	ITEM_PRICE	DISCOUNT_AMOUNT	QUANTITY	PRICE_TOTAL	DISCOUNT_TOTAL	ITEM_TOTAL
1	3	2517	1	2517	1308.84	1208.16
2	5	1199	2	2398	719.4	959.6
3	1	1199	1	1199	359.7	839.3
4	11	799.99	1	799.99	120	679.99
5	9	799.99	1	799.99	240	559.99

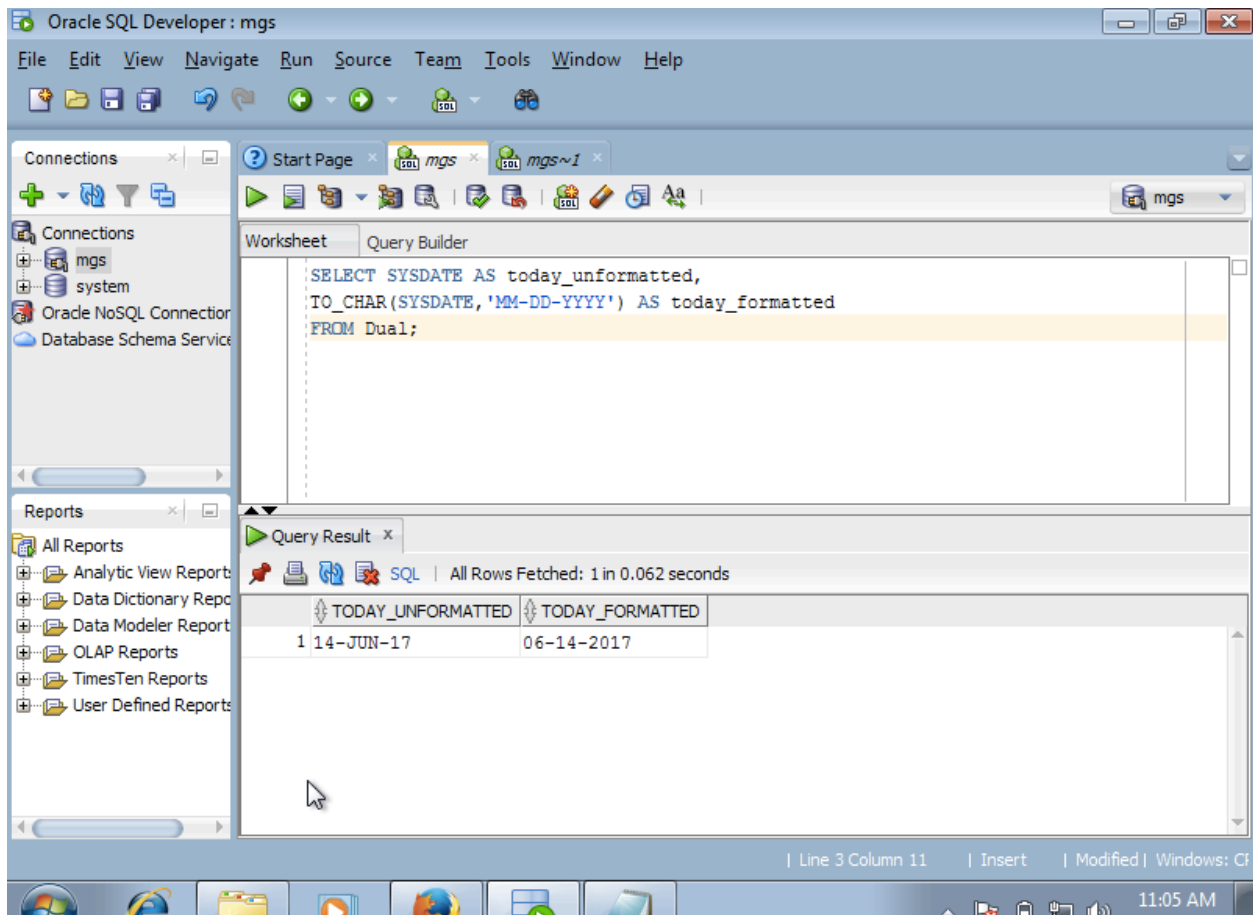
```
SELECT item_id, item_price, discount_amount, quantity,
       item_price * quantity AS price_total,
       discount_amount * quantity AS discount_total,
       (item_price - discount_amount * quantity) * quantity AS item_total
FROM Order_Items
WHERE (item_price - discount_amount * quantity) * quantity > 500
ORDER BY item_total DESC;
```

6.



```
SELECT order_id, order_date, ship_date
FROM Orders
WHERE ship_date IS NULL;
```

7.



```
SELECT SYSDATE AS today_unformatted,  
TO_CHAR(SYSDATE,'MM-DD-YYYY') AS today_formatted  
FROM Dual;
```

8.

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

```
SELECT '$100' AS price,  
       '7%' AS tax_rate,  
       100 * 0.07 AS tax_amount,  
       100 * 0.07 + 100 AS total  
FROM DUAL;
```

Below the query, the Query Result window shows the results of the query. The status bar indicates "All Rows Fetched: 1 in 0 seconds". The results are displayed in a table with the following columns: PRICE, TAX_RATE, TAX_AMOUNT, and TOTAL.

	PRICE	TAX_RATE	TAX_AMOUNT	TOTAL
1	\$100	7%	7	107

The bottom status bar shows "Line 5 Column 11 | Insert | Modified | Windows: CF" and the system clock displays "11:12 AM".

```
SELECT '$100' AS price,  
       '7%' AS tax_rate,  
       100 * 0.07 AS tax_amount,  
       100 * 0.07 + 100 AS total  
FROM DUAL;
```