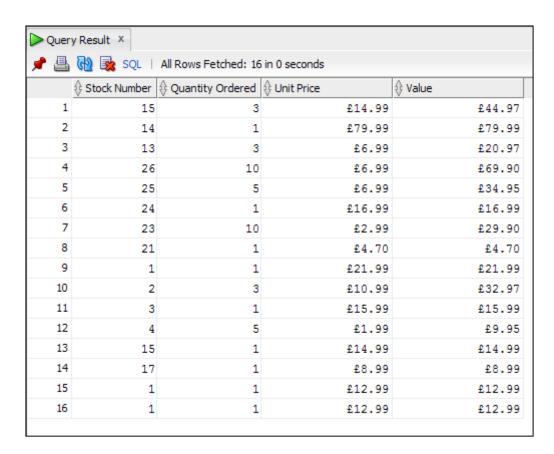# Exercise Four
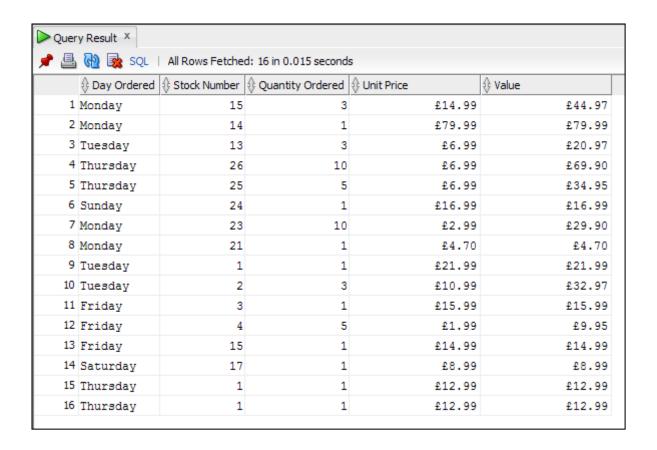
4.1 Using the Order Details table in the Worksheet, display the od_stock_number, od_quantity, od_price, in addition create a fourth column which represents the value of the order by multiplying od_unit_price by od_quantity. The output should be as follows … Save the SQL as c:\course\ex4_1.sql.

	Stock Number	Quantity Ordered	Unit Price	Value
1	15	3	£14.99	£44.97
2	14	1	£79.99	£79.99
3	13	3	£6.99	£20.97
4	26	10	£6.99	£69.90
5	25	5	£6.99	£34.95
6	24	1	£16.99	£16.99
7	23	10	£2.99	£29.90
8	21	1	£4.70	£4.70
9	1	1	£21.99	£21.99
10	2	3	£10.99	£32.97
11	3	1	£15.99	£15.99
12	4	5	£1.99	£9.95
13	15	1	£14.99	£14.99
14	17	1	£8.99	£8.99
15	1	1	£12.99	£12.99
16	1	1	£12.99	£12.99

Query Result — All Rows Fetched: 16 in 0 seconds

©Seer Computing Ltd

Using SQL with
SQL Developer (19.2) 192

Exercise Four
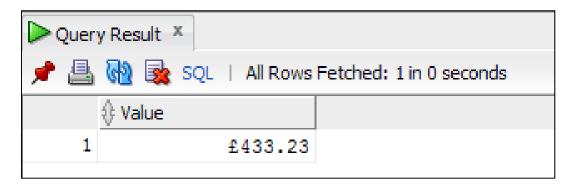
4.2 Amend the previous SQL to include the od_order_date, display only the day of the week the order was placed … Save the SQL as c:\course\ex4_2.sql.

	Day Ordered	Stock Number	Quantity Ordered	Unit Price	Value
1	Monday	15	3	£14.99	£44.97
2	Monday	14	1	£79.99	£79.99
3	Tuesday	13	3	£6.99	£20.97
4	Thursday	26	10	£6.99	£69.90
5	Thursday	25	5	£6.99	£34.95
6	Sunday	24	1	£16.99	£16.99
7	Monday	23	10	£2.99	£29.90
8	Monday	21	1	£4.70	£4.70
9	Tuesday	1	1	£21.99	£21.99
10	Tuesday	2	3	£10.99	£32.97
11	Friday	3	1	£15.99	£15.99
12	Friday	4	5	£1.99	£9.95
13	Friday	15	1	£14.99	£14.99
14	Saturday	17	1	£8.99	£8.99
15	Thursday	1	1	£12.99	£12.99
16	Thursday	1	1	£12.99	£12.99

Query Result × — All Rows Fetched: 16 in 0.015 seconds

©Seer Computing Ltd

4.3 Using the SUM function ascertain the Grand total of the orders. Hint: Remove all other columns from this query before running. Save as c:\course\ex4_3.sql

4.4 Include in the above, the average, minimum and maximum functions to give four results. Save as c:\course\ex4_4.sql

©Seer Computing Ltd