

```

-- the following settings is assumed for a A4 landscape output
SET linesize 120
SET pagesize 35

-- set to output date in Malaysia date format
ALTER SESSION SET NLS_DATE_FORMAT = 'dd/mm/yyyy';

-- 01/06/2016 & 30/06/2016

-- the substitution variable v_startDate and v_endDate should be DATE data type
ACCEPT v_startDate DATE FORMAT 'dd/mm/yyyy' PROMPT ' Enter the start date: '
ACCEPT v_endDate   DATE FORMAT 'dd/mm/yyyy' PROMPT ' Enter the end date   : '

COL order_no    FORMAT 9999999 HEADING "Order No  "
COL order_date  FORMAT A11 HEADING "Order Date"
COL menu_id     FORMAT A7 HEADING "Menu ID"
COL descr       FORMAT A30 HEADING "Description"
COL quantity    FORMAT 99 HEADING "Qty"

-- large numbers(prices and quantities) should be separated by commas
COL price        FORMAT $9,999.99 HEADING "Price (RM)"
COL subtotal     FORMAT $99,9999.99 HEADING "Subtotal (RM)"

TTITLE CENTER 'Order List for ' _DATE -
RIGHT 'Page No: ' FORMAT 999 SQL.PNO SKIP 2
BREAK ON Order_no SKIP 2 ON order_date
COMPUTE SUM LABEL 'Total (RM): ' OF Subtotal ON Order_No

-- make your statement readable and easy to maintain
-- DO NOT write long lines of codes, break into multiple lines
-- Adjust alignment for ease of readability
SELECT FO.order_no,
       order_date,
       OL.menu_id,
       descr,
       quantity,
       OL.price,
       (quantity * OL.price) AS Subtotal
FROM   food_order FO
       join order_list OL ON FO.order_no = OL.order_no
       join menu       M  ON OL.menu_id  = M.menu_id
WHERE  order_date BETWEEN '&v_startDate' and '&v_endDate'
ORDER BY order_no;

-- Note: the v_startDate and v_endDate data needs to be processed as a string, therefore enclosed
in ' '.

CLEAR COLUMNS
CLEAR BREAKS
CLEAR COMPUTES
TTITLE OFF

```

```

/*
-- Example how to execute the script and sample output
-- assuming the above script is saved in D:\ drive and named Query1.txt

```

```
SQL> start d:\Query1.txt
```

```
Session altered. -- you have changed the DATE format from the Oracle default date format
```

```

Enter the start date: 01/06/2016 -- data for your v_startDate substitution variable, no need to enclosed with quote ' '
Enter the end date : 30/06/2016 -- data for your v_endDate substitution variable, no need to enclosed with quote ' '
old 13:      (order_date BETWEEN '&v_startDate' and '&v_endDate') -- system show how your variable is used
new 13:      (order_date BETWEEN '01/06/2016' and '30/06/2016') -- the select statement enclosed the data in quotes ' '

-- the old and new values are show to you for verification, useful for debugging your code
-- you can turn off the verification once you are satisfied that no further debugging is necessary
-- SET VERIFY OFF

```

Order List for 19/04/2023

Page No: 1

Order No	Order Date	Menu ID	Description	Qty	Price (RM)	Subtotal (RM)
1001	01/06/2016	M001	Sparkling Lime	2	\$3.90	\$7.80
		M091	Cappuccino	1	\$5.90	\$5.90
*****						-----
Total (RM)						\$13.70
1002	03/06/2016	M101	Crab stick and Fish Fingers	1	\$11.95	\$11.95
		M001	Sparkling Lime	1	\$3.90	\$3.90
		M091	Cappuccino	2	\$5.90	\$11.80
*****						-----
Total (RM)						\$27.65
1003	05/06/2016	M101	Crab stick and Fish Fingers	2	\$11.95	\$23.90
		M091	Cappuccino	2	\$5.90	\$11.80
*****						-----
Total (RM)						\$35.70

7 rows selected.

```
SQL>
```

```
*/
```