|  |
| --- |
| , RD Dep. |
| **Report Introduction to OLAP** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| REVISION HISTORY | | | | | |
| Ver. | Description of Change | Author | Date | Approved | |
| Name | Effective Date |
| 1.0 | Initial Version | Hanna Hul | 15-Nov-2017 |  |  |
|  |  |  |  |  |  |

Contents

[1. SQL for analysis 3](#_Toc498543804)

# SQL for analysis

1. Topic 2. Task 2 (please, make an example of usage GROUPING\_ID).

create synonym channels for sh.channels;

create synonym times for sh.times;

create synonym countries for sh.countries;

create synonym customers for sh.customers;

create synonym sales for sh.sales;

select DECODE(grouping\_id(times.calendar\_month\_desc),1,'GRAND TOTAL',times.calendar\_month\_desc) year\_month,

-- grouping\_id(times.calendar\_month\_desc) id\_,

DECODE(grouping\_id(times.calendar\_month\_desc,channels.channel\_desc,

countries.country\_name),3,'Total by Channels',channels.channel\_desc) channel,

DECODE(grouping\_id(times.calendar\_month\_desc,channels.channel\_desc,

countries.country\_name),1,concat(channels.channel\_desc, ' Total by States'),countries.country\_name) country,

round(max (amount\_sold)) max\_sales$,

round(min (amount\_sold)) min\_sales$,

round(sum(amount\_sold)) sales$

from sales, times, channels, customers, countries

where sales.time\_id = times.time\_id and

sales.channel\_id = channels.channel\_id and

sales.cust\_id = customers.cust\_id and

customers.country\_id = countries.country\_id and

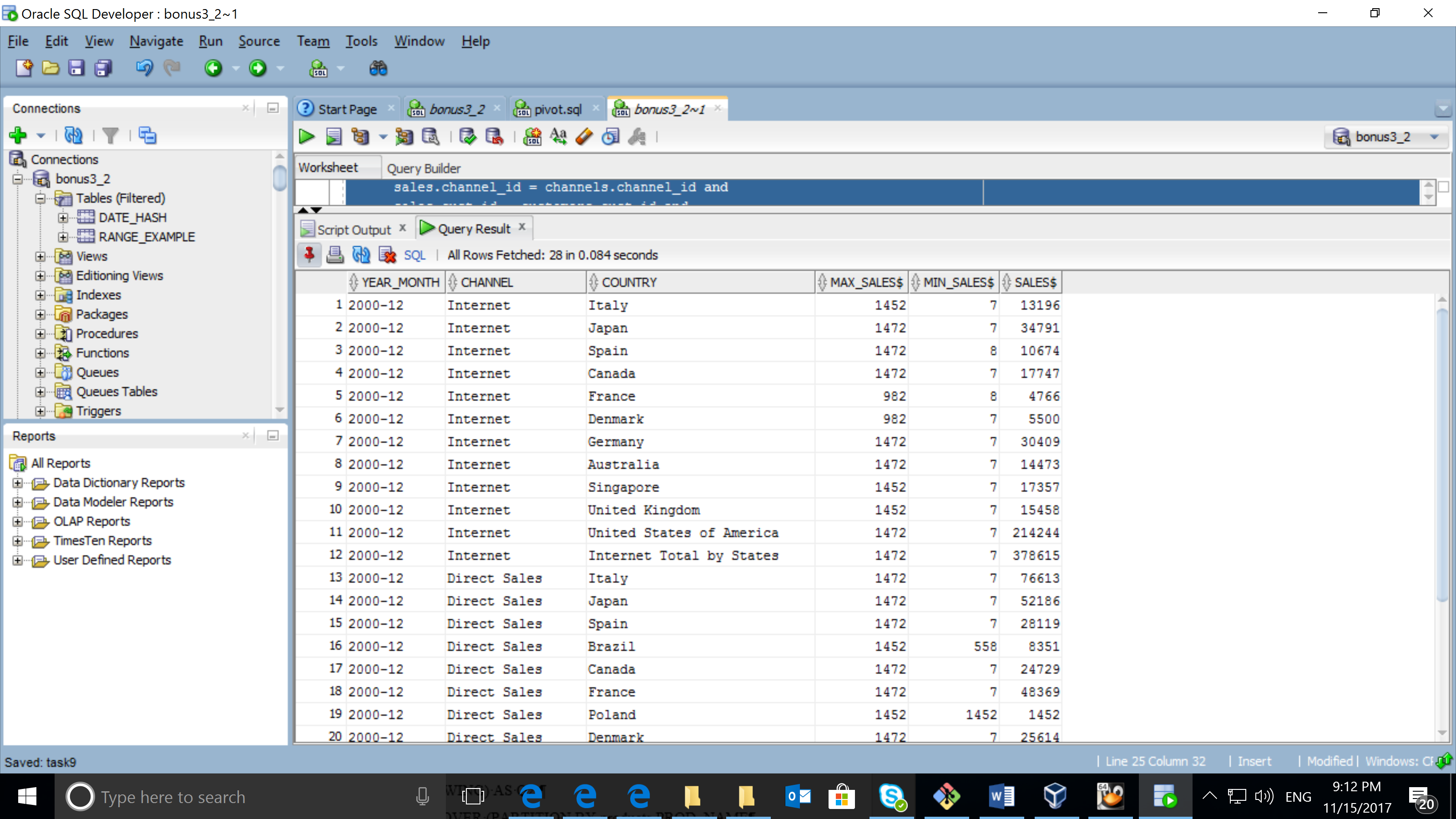
times.calendar\_month\_desc = '2000-12' and

channels.channel\_desc IN ('Internet','Direct Sales')

group by rollup (times.calendar\_month\_desc,

channels.channel\_desc,

countries.country\_name);



1. Topic 3. Task 2.

CREATE SYNONYM products FOR sh.products;

CREATE SYNONYM times FOR sh.times;

CREATE SYNONYM countries FOR sh.countries;

CREATE SYNONYM customers FOR sh.customers;

CREATE SYNONYM sales FOR sh.sales;

SELECT DISTINCT

NVL(products.prod\_name,'TOTAL') prod\_name,

NTH\_VALUE(SUM(AMOUNT\_SOLD),1) OVER (PARTITION BY products.PROD\_NAME

ORDER BY times.CALENDAR\_QUARTER\_NUMBER ASC RANGE BETWEEN UNBOUNDED

PRECEDING AND UNBOUNDED FOLLOWING) AS Q1,

NTH\_VALUE(SUM(AMOUNT\_SOLD),2) OVER (PARTITION BY PRoducts.PROD\_NAME

ORDER BY Times.CALENDAR\_QUARTER\_NUMBER ASC RANGE BETWEEN UNBOUNDED

PRECEDING AND UNBOUNDED FOLLOWING) AS Q2,

NTH\_VALUE(SUM(AMOUNT\_SOLD),3) OVER (PARTITION BY products.PROD\_NAME

ORDER BY Times.CALENDAR\_QUARTER\_NUMBER ASC RANGE BETWEEN UNBOUNDED

PRECEDING AND UNBOUNDED FOLLOWING) AS Q3,

NTH\_VALUE(SUM(AMOUNT\_SOLD),4) OVER (PARTITION BY products.PROD\_NAME

ORDER BY Times.CALENDAR\_QUARTER\_NUMBER ASC RANGE BETWEEN UNBOUNDED

PRECEDING AND UNBOUNDED FOLLOWING) AS Q4,

SUM(SUM(AMOUNT\_SOLD)) OVER (PARTITION BY products.PROD\_NAME) AS YEAR\_SUM

FROM

sales, times, products, customers, countries

WHERE

sales.time\_id = times.time\_id

AND sales.prod\_id = products.prod\_id

AND sales.cust\_id = customers.cust\_id

AND customers.country\_id = countries.country\_id

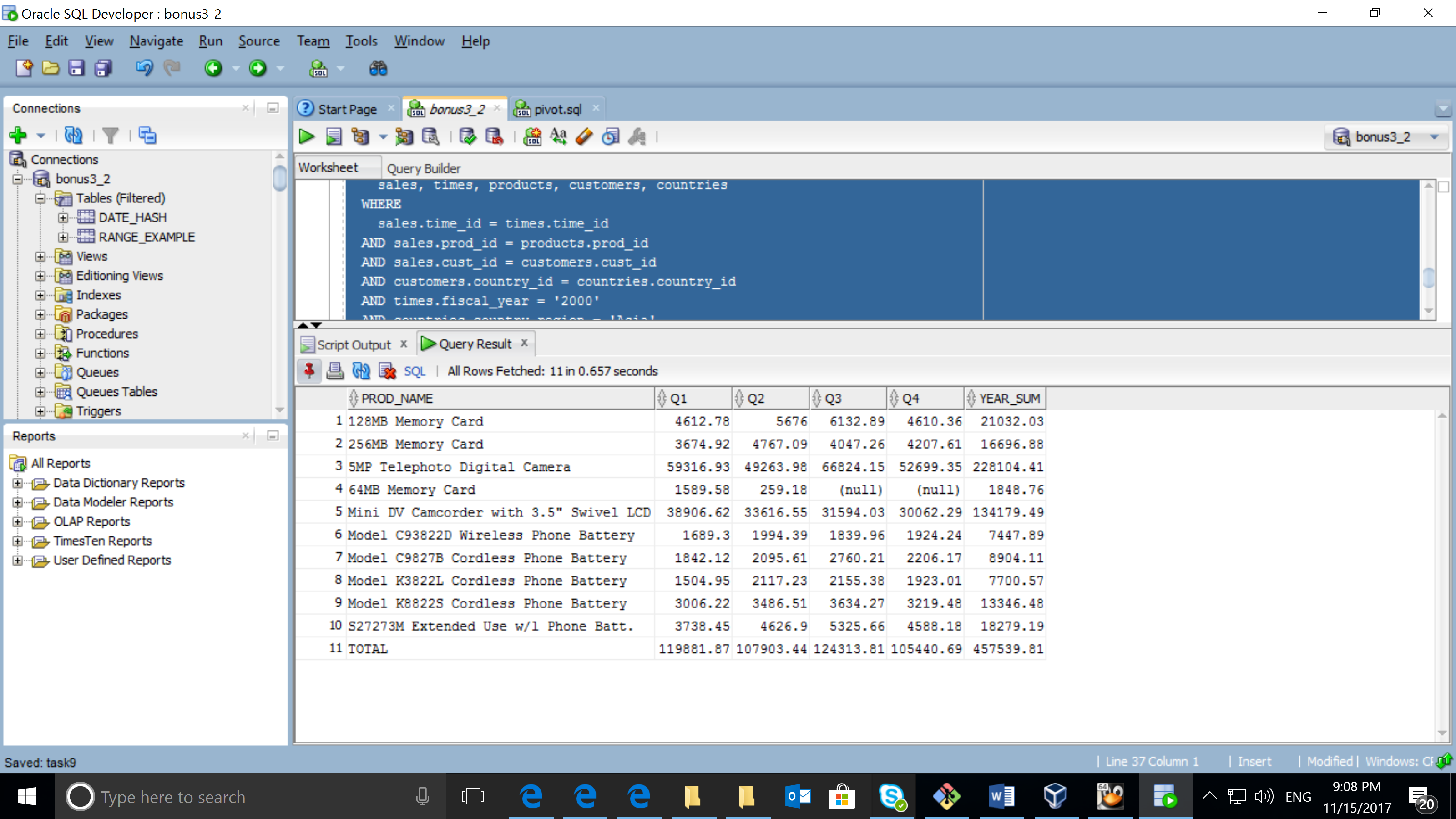
AND times.fiscal\_year = '2000'

AND countries.country\_region = 'Asia'

AND products.prod\_category = 'Photo'

GROUP BY ROLLUP(prod\_name), times.calendar\_quarter\_number

ORDER BY prod\_name nulls last;



1. Your personal example of PIVOT

create view SALES\_VIEW as

select PROD\_NAME PRODUCT,

COUNTRY\_NAME COUNTRY,

CHANNEL\_ID CHANNEL,

SUBSTR(CALENDAR\_QUARTER\_DESC, 6,2) QUARTER,

SUM(AMOUNT\_SOLD) AMOUNT\_SOLD,

SUM(QUANTITY\_SOLD) QUANTITY\_SOLD

from sh.SALES, sh.TIMES, sh.CUSTOMERS, sh.COUNTRIES, sh.PRODUCTS

where sh.SALES.TIME\_ID = sh.TIMES.TIME\_ID and

sh.SALES.PROD\_ID = sh.PRODUCTS.PROD\_ID and

sh.SALES.CUST\_ID = sh.CUSTOMERS.CUST\_ID and

sh.CUSTOMERS.COUNTRY\_ID = sh.COUNTRIES.COUNTRY\_ID

group by PROD\_NAME, COUNTRY\_NAME, CHANNEL\_ID, SUBSTR(CALENDAR\_QUARTER\_DESC, 6, 2) ;

