1. Develop a UDF to receive a product number and returns the total number product sold/

```
create or replace function udf_total_product(p_prod_number int)
return int
as
v_prod_sold int;
v_prod_no int;
cursor c1 is
select ol.product_no ,count(o.order_no) from orders  o join orderline ol
on o.order_no=ol.order_no
group by ol.product_no;
begin
open c1;
    loop
    fetch c1 into v_prod_no,v_prod_sold;
    exit when c1%notfound;

    if v_prod_no=p_prod_number then
    return v_prod_sold;
    end if;
    end loop;
    close c1;
end;
/
-- select udf total product(120) from dual;
```

```
ript Output
Task completed in 0.032 seconds

tion UDF_TOTAL_PRODUCT compiled
```

Develop a UDF to receive a product number and year, and returns the total number product sold for that year

```
-- select udf_total_product_yearly(180,2001) from dual
create or replace function udf_total_product_yearly(p_prod_number int,p_year int)
return int
as
v_prod_sold int;
v_year int;
v_prod_no int;
cursor c1 is
select ol.product_no ,extract(year from o.order_date),count(o.order_no)  from orders  o join orderline ol
on o.order_no=ol.order_no
join returnprod rp on ol.orderline_no=rp.orderline_no
group by ol.product_no,extract(year from o.order_date)
having ol.product_no = p_prod_number  ;

begin
```

Script Output | Query Result | SQL | All Rows Fetched: 1 in 0 seconds

	UDF_TOTAL_PRODUCT_YEARLY(180,2001)
1	4

Develop a UDF that will receive a product number, a month, a year, and returns the sales for that year

```
select ol.product_no ,extract(year from o.order_date) ,(sum(o.subtotal)-sum(nvl(amount_refunded,0 ))) as sale
from orders  o join orderline ol
on o.order_no=ol.order_no
join returnprod rp on ol.orderline_no=rp.orderline_no
group by ol.product_no,extract(year from o.order_date)
having ol.product_no=p_prod_no;

begin
open c1;

  loop
    fetch c1 into v_prod_no,v_year,v_sales;
    exit when c1%notfound;
     if v_prod_no=p_prod_no and v_year=p_year then
    return v_sales;
    else return null;
    end if;
    end loop;
    close c1;
    exception WHEN others then
    return null;
end;
```

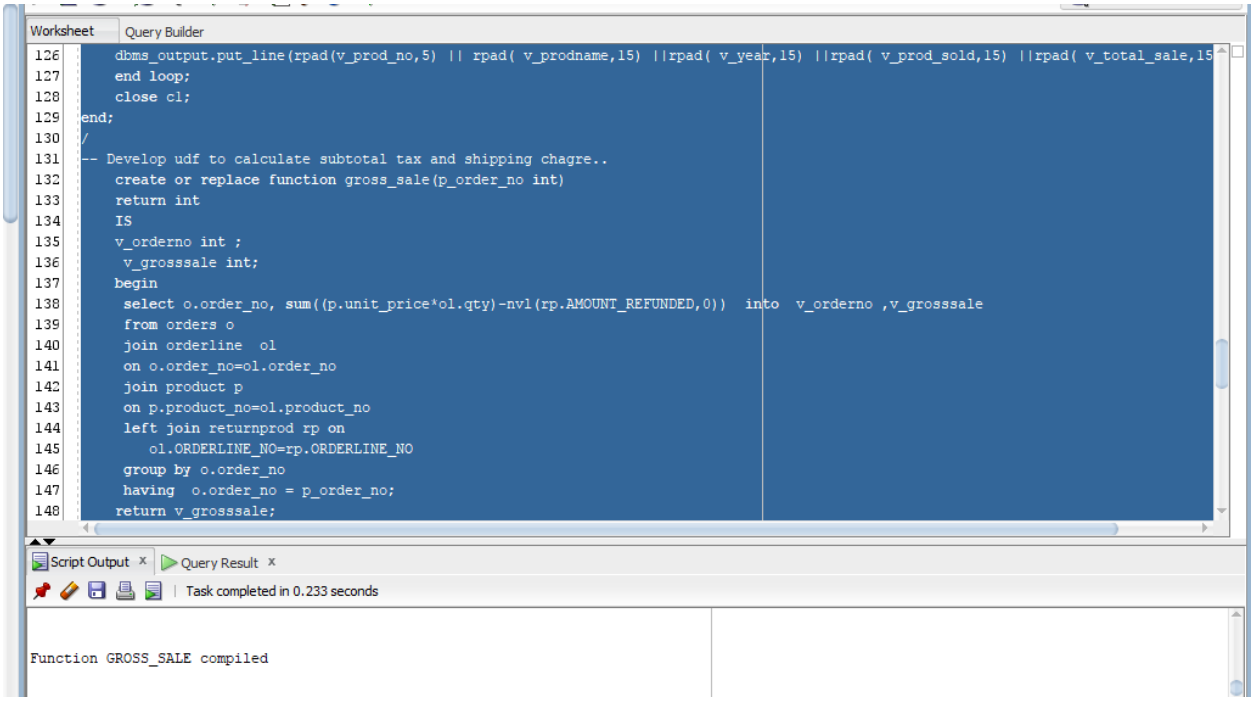
Script Output | Query Result | Task completed in 0.032 seconds

```
Function UDF_TOTAL_PRODUCT_DETAI compiled
```

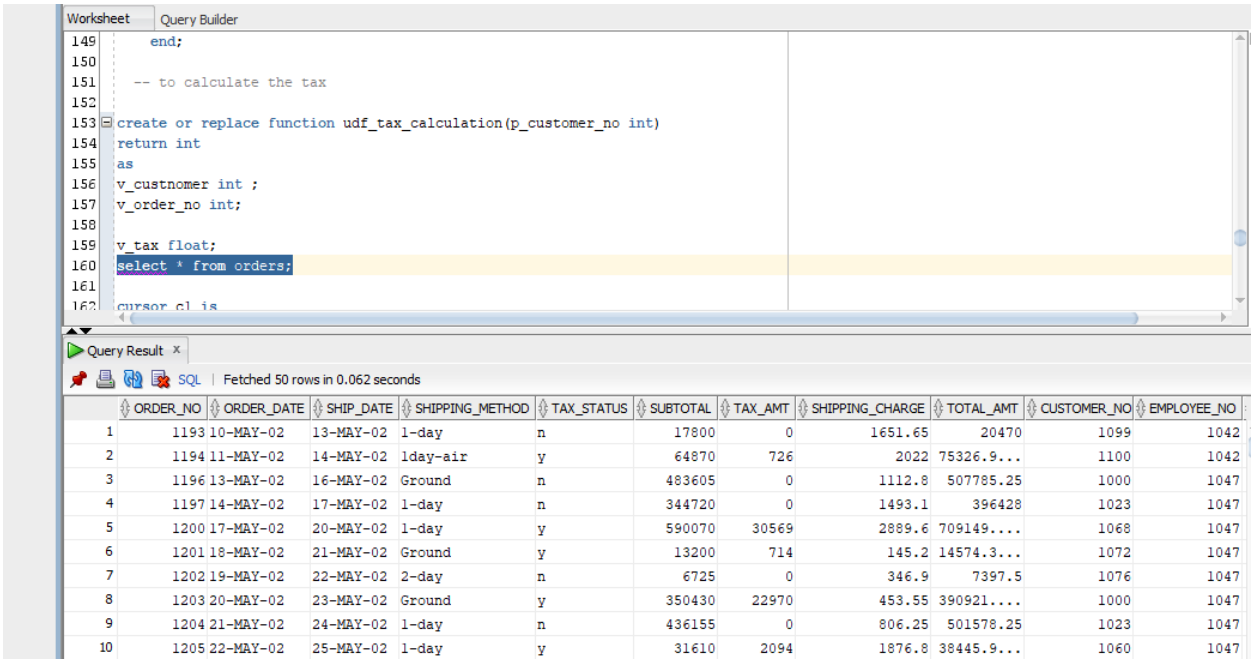
Develop a UDF that will receive a product number, a month, a year, and returns the sales for that year

Develop a SQL program to call these UDF and produce the following report

Product#, product_name, year, Number_Product_sold, total_sale



The shipping charge has been update and then on selecting the select * form orders;



	ORDER_NO	ORDER_DATE	SHIP_DATE	SHIPPING_METHOD	TAX_STATUS	SUBTOTAL	TAX_AMT	SHIPPING_CHARGE	TOTAL_AMT	CUSTOMER_NO	EMPLOYEE_NO
1	1193	10-MAY-02	13-MAY-02	1-day	n	17800	0	1651.65	20470	1099	1042
2	1194	11-MAY-02	14-MAY-02	1day-air	y	64870	726	2022	75326.9...	1100	1042
3	1196	13-MAY-02	16-MAY-02	Ground	n	483605	0	1112.8	507785.25	1000	1047
4	1197	14-MAY-02	17-MAY-02	1-day	n	344720	0	1493.1	396428	1023	1047
5	1200	17-MAY-02	20-MAY-02	1-day	y	590070	30569	2889.6	709149....	1068	1047
6	1201	18-MAY-02	21-MAY-02	Ground	y	13200	714	145.2	14574.3...	1072	1047
7	1202	19-MAY-02	22-MAY-02	2-day	n	6725	0	346.9	7397.5	1076	1047
8	1203	20-MAY-02	23-MAY-02	Ground	y	350430	22970	453.55	390921....	1000	1047
9	1204	21-MAY-02	24-MAY-02	1-day	n	436155	0	806.25	501578.25	1023	1047
10	1205	22-MAY-02	25-MAY-02	1-day	y	31610	2094	1876.8	38445.9...	1060	1047

Tax calculation

The product no is the parameter and it will reutn the toatal tax for that product/

```
152
153 create or replace function udf_tax_calculation(p_customer_no int)
154 return int
155 as
156 v_custnomer int ;
157 v_order_no int;
158 v_tax float;
159 cursor c1 is
160 select o.order_no,c.customer_no,sum(t.tax_rate*o.subtotal) as tax_amount from customer c
161 join orders o
162 on o.customer_no=c.customer_no
163 join tax t
164 on c.state=t.state
165 group by  o.order_no,c.customer_no
166 HAVING c.customer_no =p_customer_no ;
167 begin
168 open c1;
169     loop
170     fetch c1 into v_order_no,v_custnomer,v_tax;
171     exit when c1%notfound;
172    return v_tax;
```

Script Output | Query Result

SQL | All Rows Fetched: 1 in 0.016 seconds

	UDF_TAX_CALCULATION(1099)
1	199