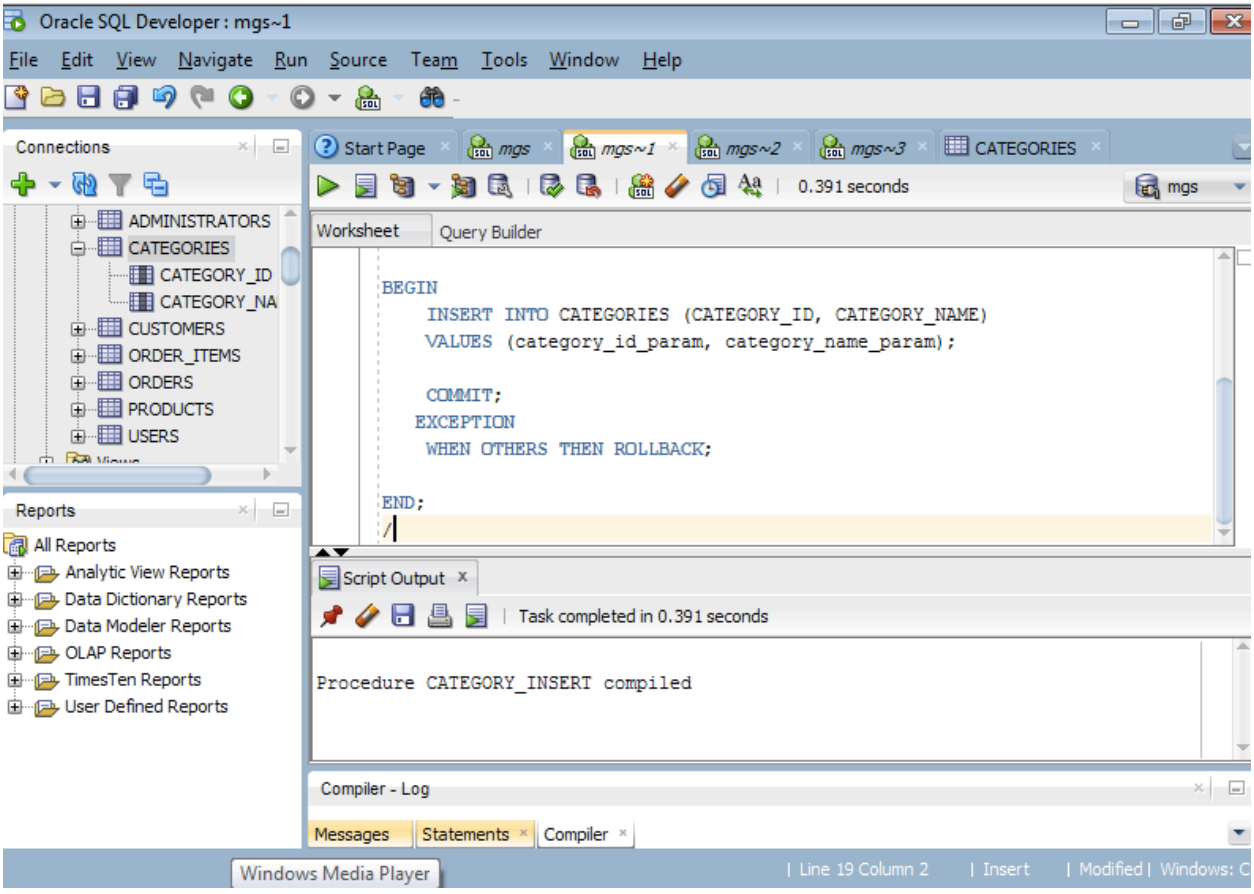
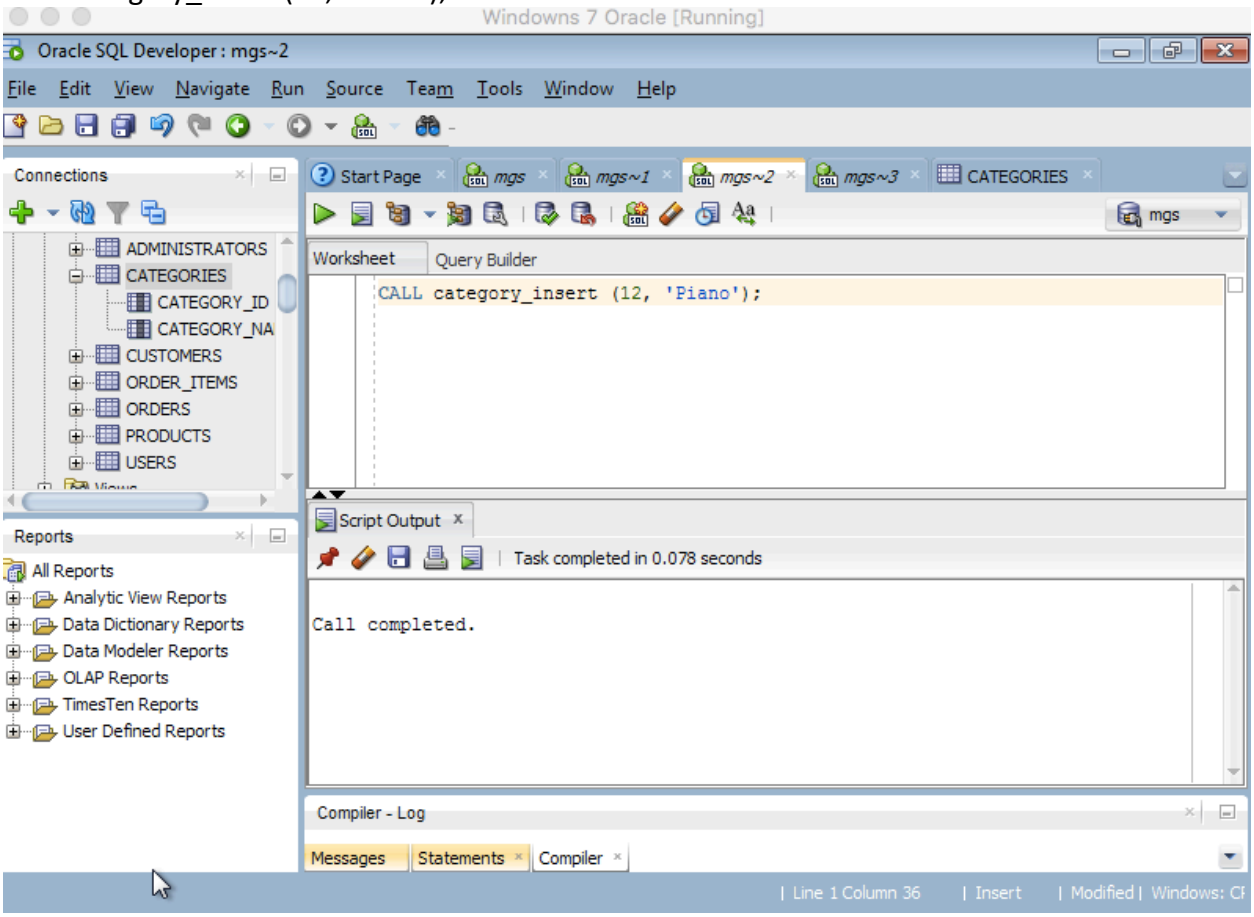
Domas Budrys Assignment 7 – CSCI 4430

1.

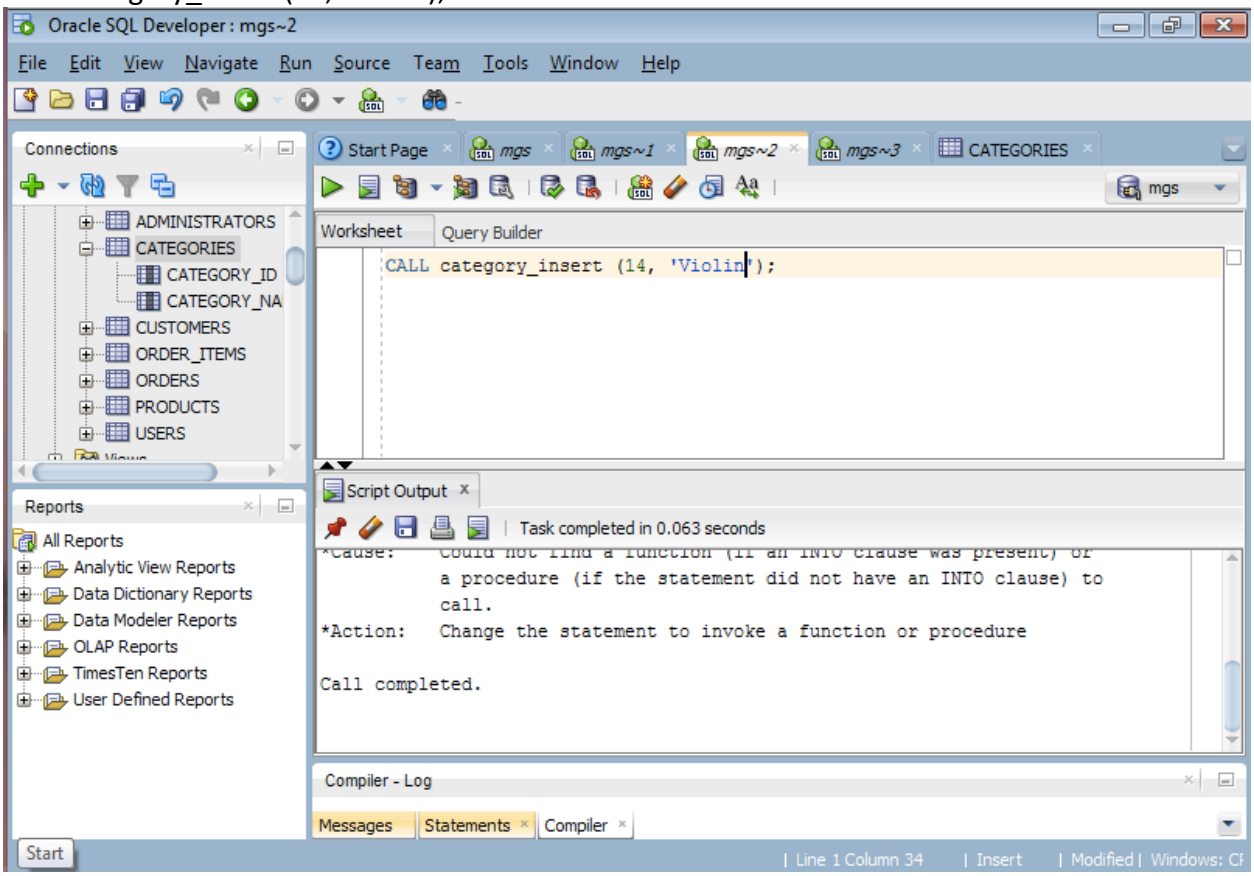
```
CREATE OR REPLACE PROCEDURE category_insert
(
   category_id_param   NUMBER,
   category_name_param VARCHAR2
)
AS
BEGIN
   INSERT INTO CATEGORIES (CATEGORY_ID, CATEGORY_NAME)
   VALUES (category_id_param, category_name_param);

   COMMIT;
  EXCEPTION
   WHEN OTHERS THEN ROLLBACK;
END;
/
```



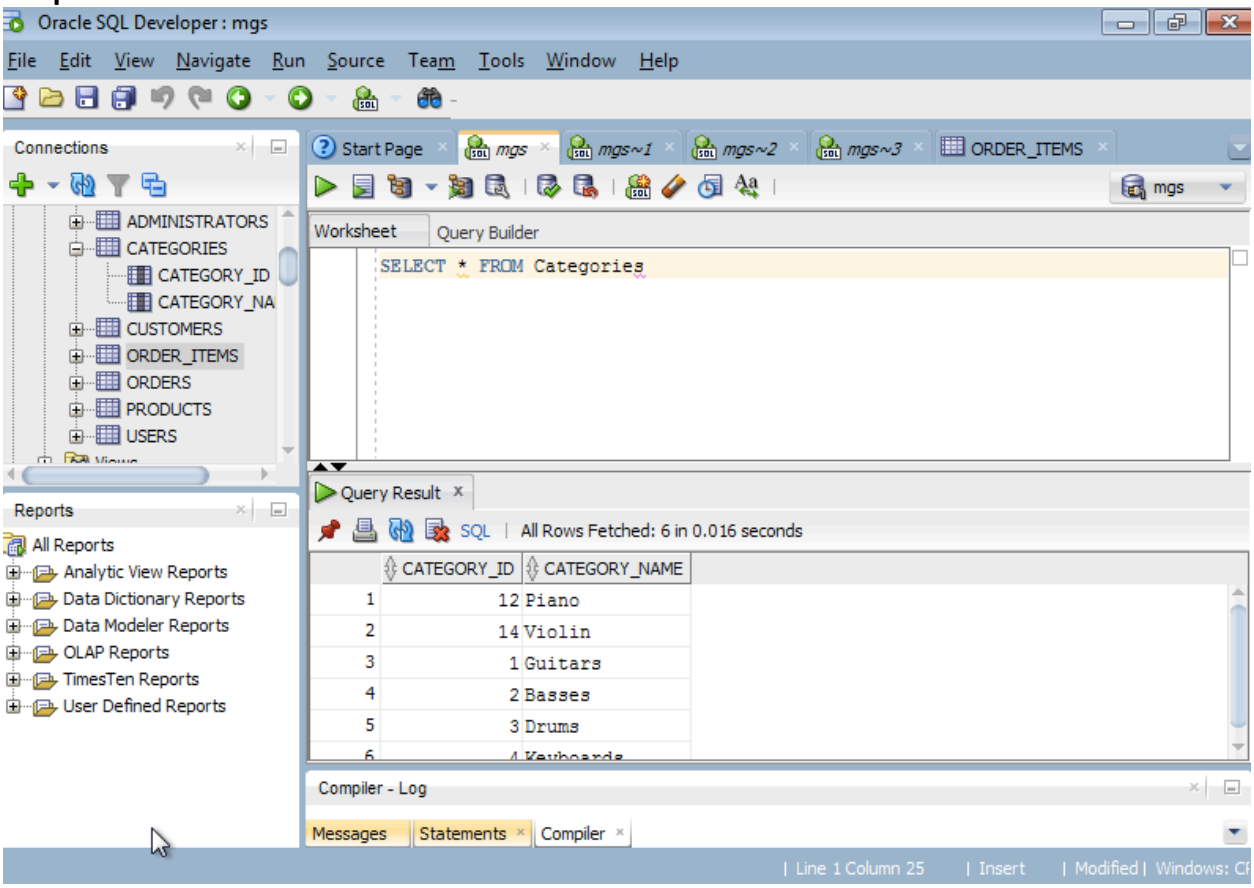
CALL category_insert (14, 'Violin');



CALL category_insert (12, 'Piano');

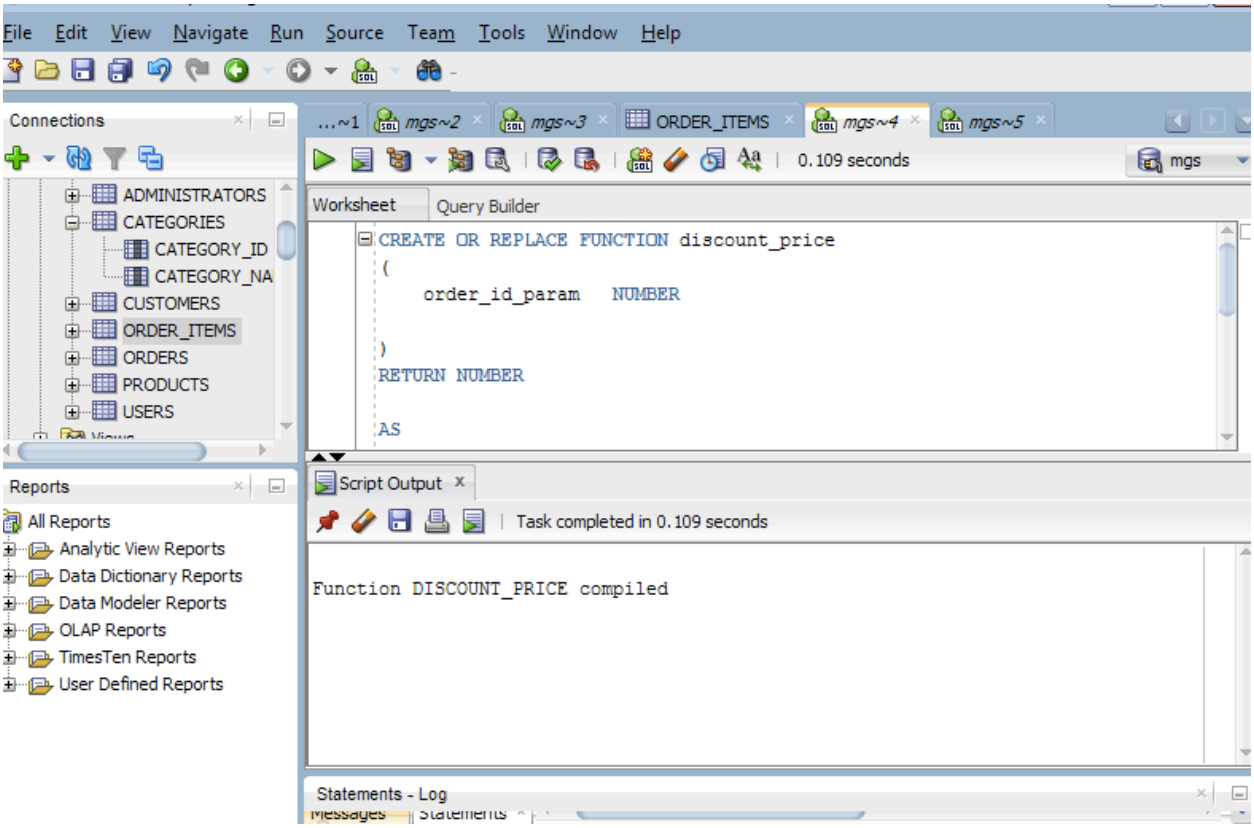


Output:

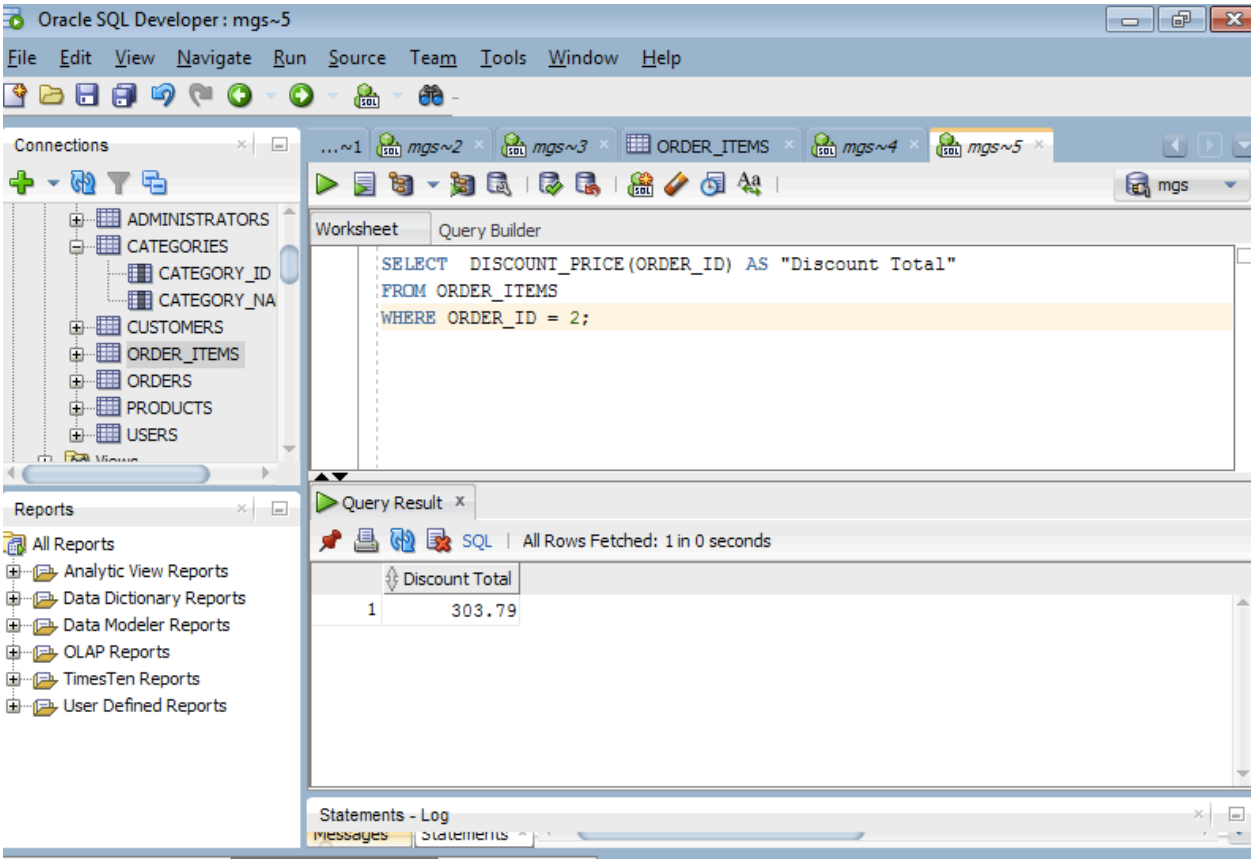

Oracle SQL Developer : mgs
File Edit View Navigate Run Source Team Tools Window Help
Connections
ADMINISTRATORS
CATEGORIES
CATEGORY_ID
CATEGORY_NA
CUSTOMERS
ORDER_ITEMS
ORDERS
PRODUCTS
USERS
Reports
All Reports
Analytic View Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
mgs
mgs~1
mgs~2
mgs~3
ORDER_ITEMS
mgs
Worksheet
Query Builder
SELECT * FROM Categories
Query Result
SQL | All Rows Fetched: 6 in 0.016 seconds
CATEGORY_ID CATEGORY_NAME
1 12 Piano
2 14 Violin
3 1 Guitars
4 2 Basses
5 3 Drums
Compiler - Log
Messages
Statements
Compiler
Line 1 Column 25
Insert
Modified

2.

```
CREATE OR REPLACE FUNCTION discount_price
(
   order_id_param   NUMBER
)
RETURN NUMBER
AS
   order_id_var    NUMBER;

BEGIN
   SELECT ITEM_PRICE - DISCOUNT_AMOUNT  AS "TotalSOme "
   INTO order_id_var
   FROM ORDER_ITEMS
   WHERE ORDER_ID = order_id_param;

RETURN  order_id_var;
END;
```

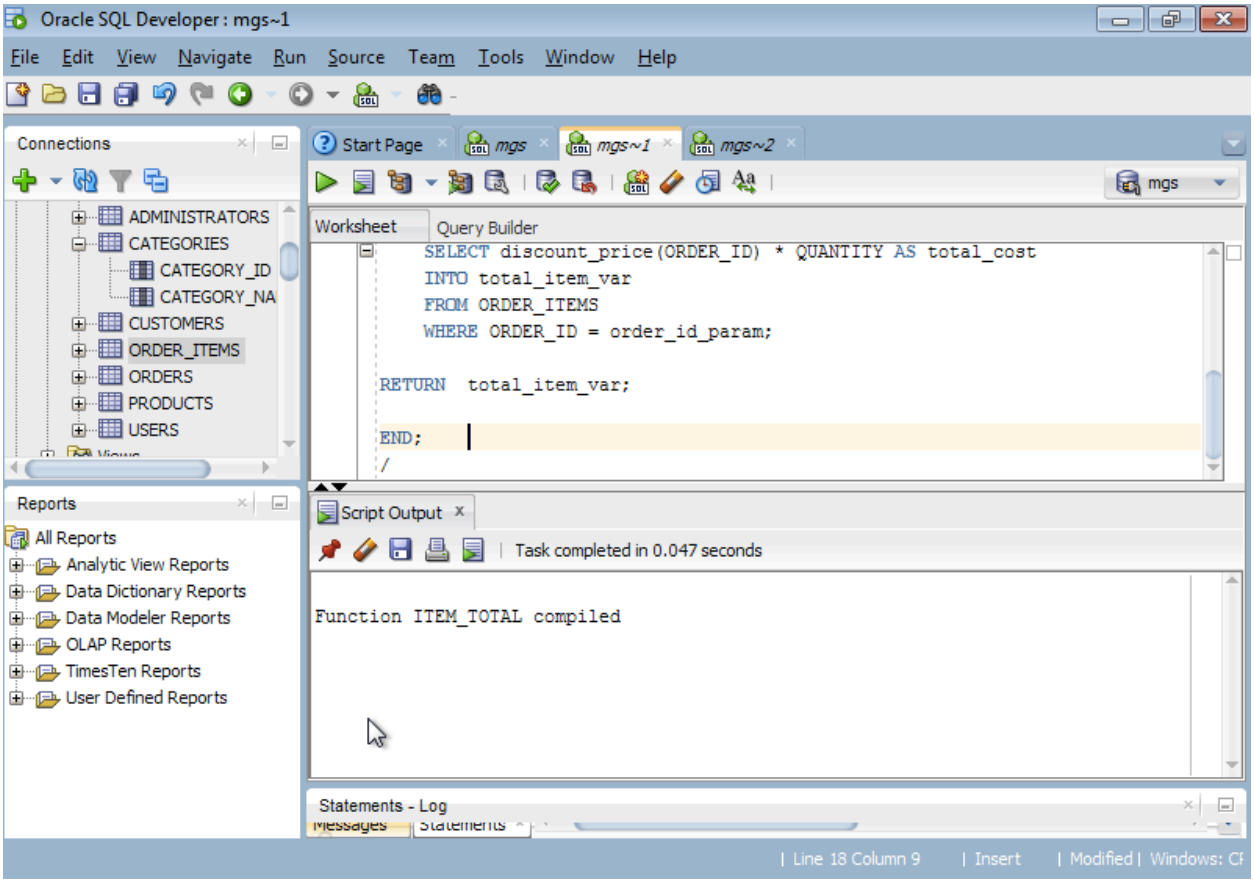


```
SELECT  DISCOUNT_PRICE(ORDER_ID) AS "Discount Total"
FROM ORDER_ITEMS
WHERE ORDER_ID = 2;
```

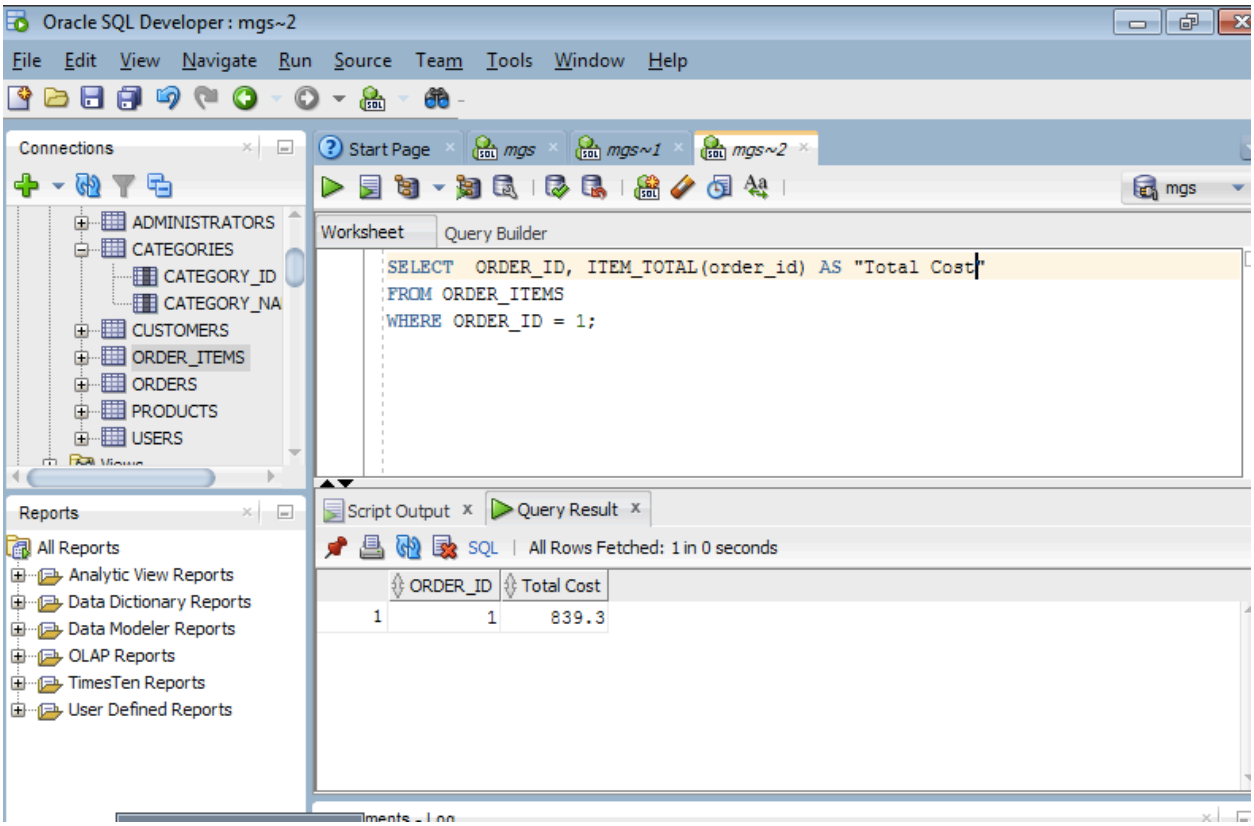


3.

```
CREATE OR REPLACE FUNCTION item_total
(
   order_id_param   NUMBER
)
RETURN NUMBER
AS
   total_item_var   NUMBER;

BEGIN
   SELECT discount_price(ORDER_ID) * QUANTITY AS total_cost
   INTO total_item_var
   FROM ORDER_ITEMS
   WHERE ORDER_ID = order_id_param;

RETURN  total_item_var;
END;
/
```

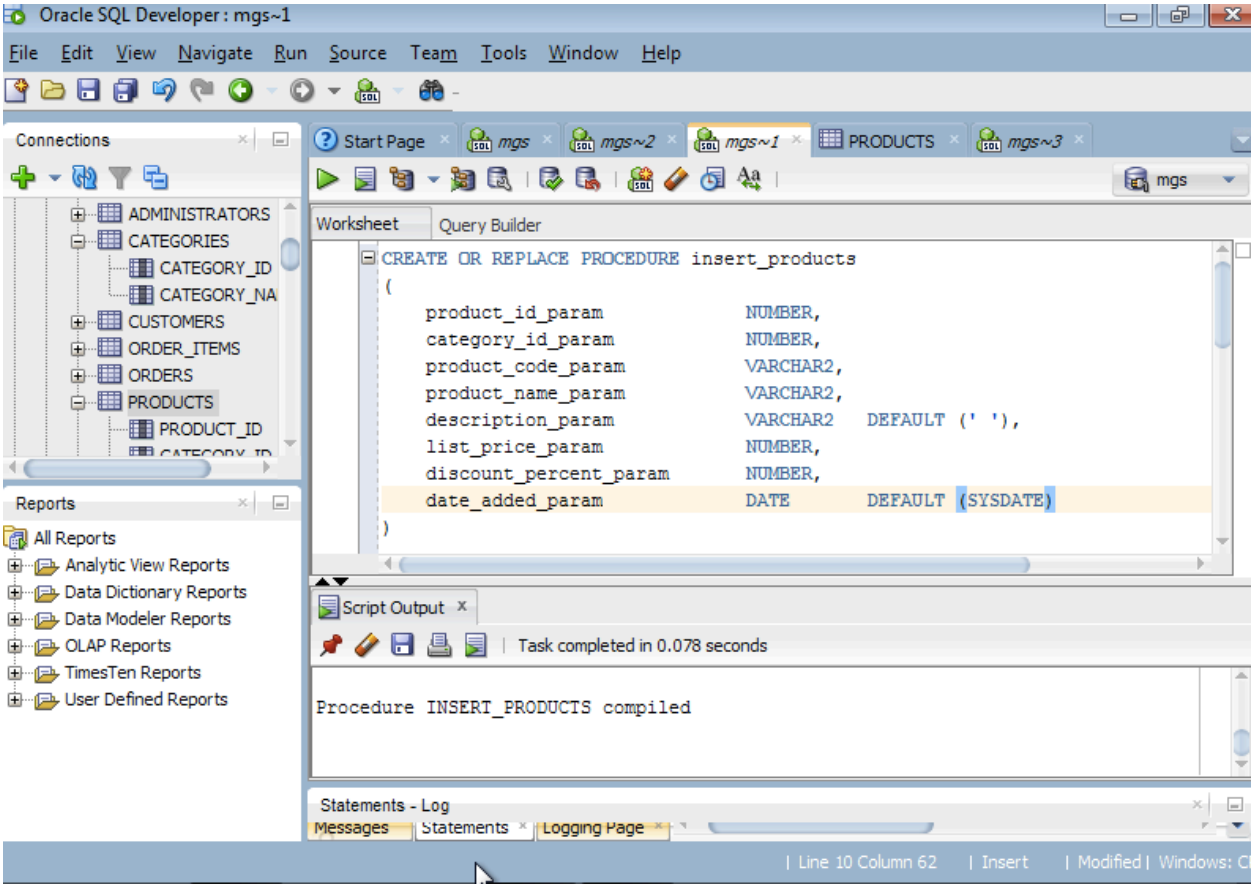


```
SELECT  ORDER_ID, ITEM_TOTAL(order_id) AS "Total Cost"
FROM ORDER_ITEMS
WHERE ORDER_ID = 1;
```

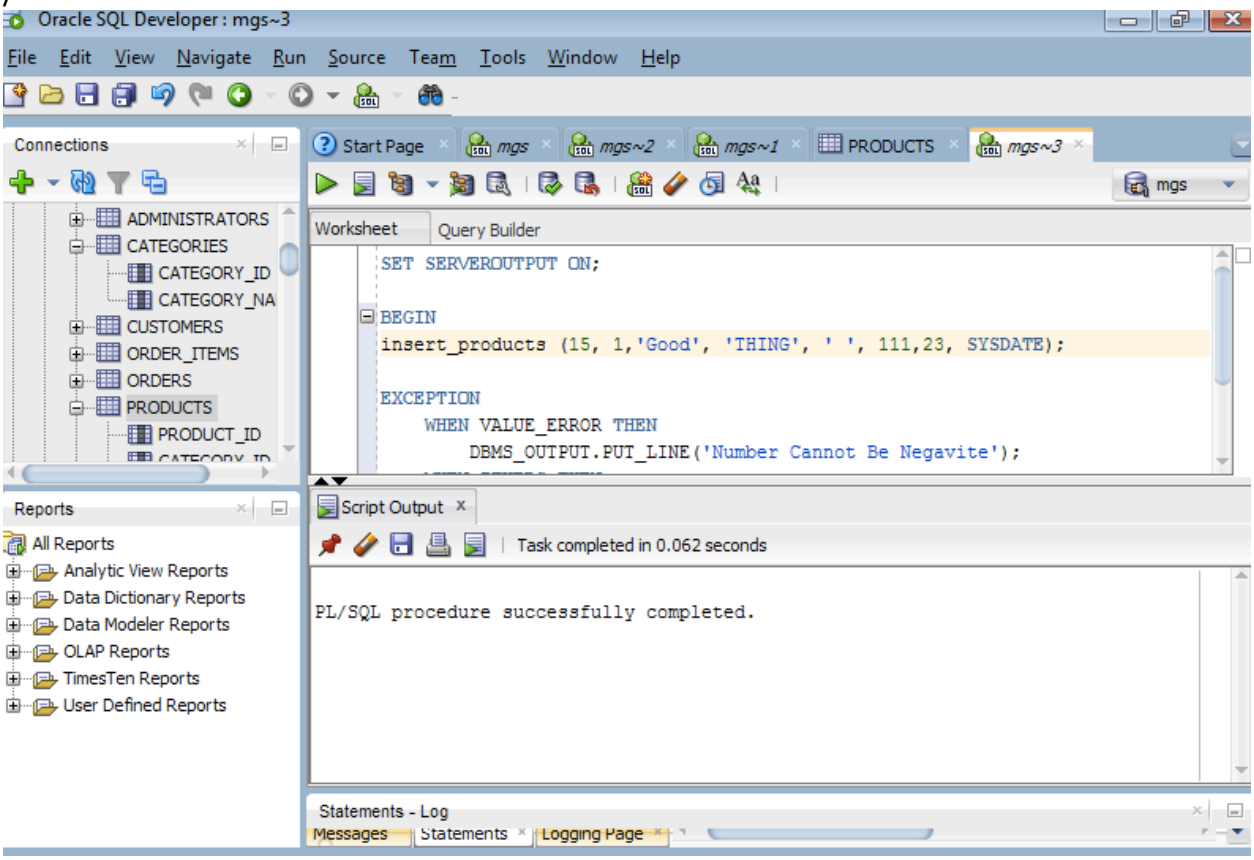


4.

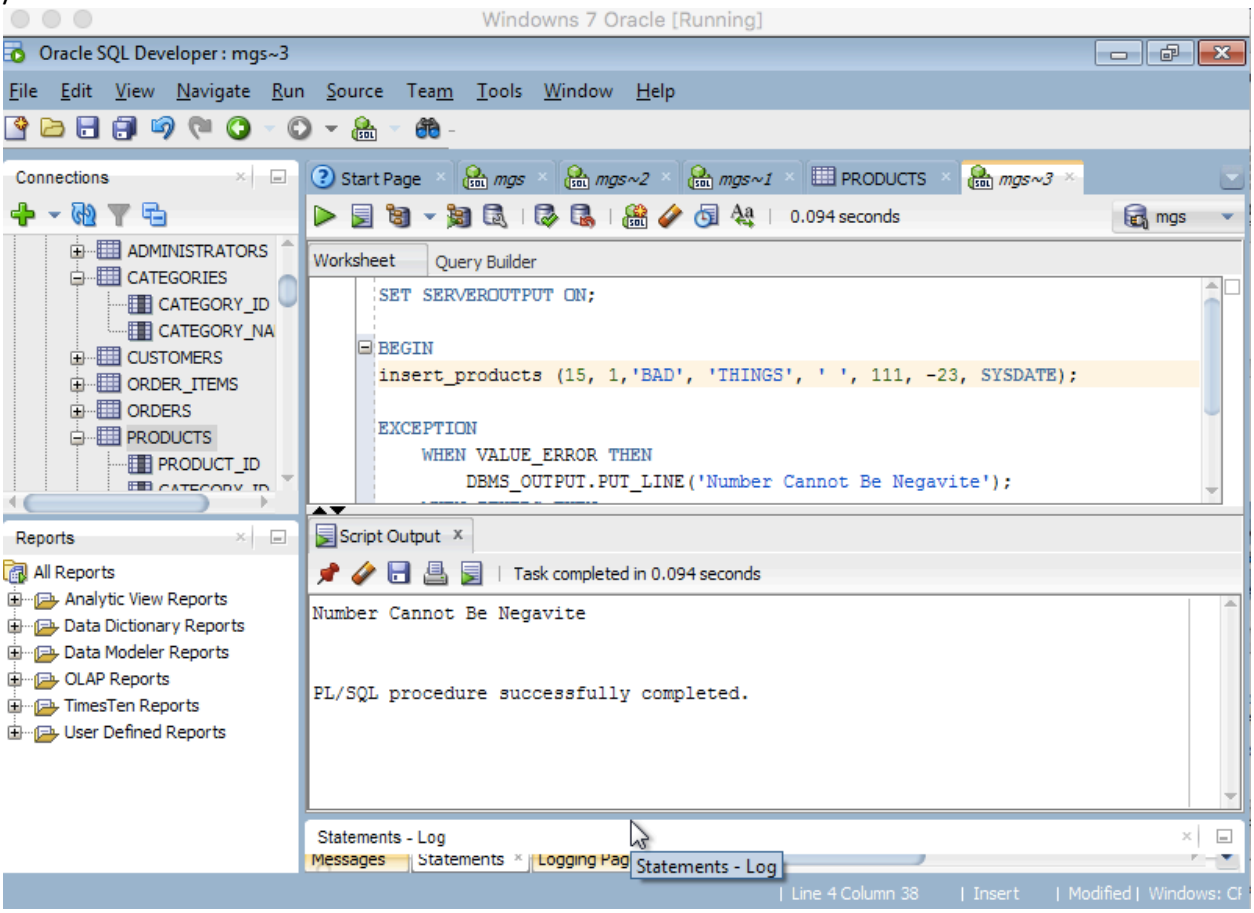
```
CREATE OR REPLACE PROCEDURE insert_products
(
   product_id_param           NUMBER,
   category_id_param          NUMBER,
   product_code_param          VARCHAR2,
   product_name_param          VARCHAR2,
   description_param          VARCHAR2   DEFAULT (' '),
   list_price_param           NUMBER,
   discount_percent_param      NUMBER,
   date_added_param           DATE      DEFAULT (SYSDATE)
)

AS

BEGIN

   IF
      list_price_param < 0 THEN
      RAISE VALUE_ERROR;
   END IF;

    IF
      discount_percent_param < 0 THEN
      RAISE VALUE_ERROR;
   END IF;


   INSERT INTO PRODUCTS(PRODUCT_ID, CATEGORY_ID, PRODUCT_CODE, PRODUCT_NAME,
                    DESCRIPTION, LIST_PRICE, DISCOUNT_PERCENT, DATE_ADDED)

   VALUES(product_id_param, category_id_param, product_code_param,
product_name_param,
         description_param, list_price_param, discount_percent_param, date_added_param);

COMMIT;
END;
/
```


Oracle SQL Developer : mgs~1
File Edit View Navigate Run Source Team Tools Window Help
Connections
ADMINISTRATORS
CATEGORIES
CATEGORY_ID
CATEGORY_NA
CUSTOMERS
ORDER_ITEMS
ORDERS
PRODUCTS
PRODUCT_ID
Reports
All Reports
Analytic View Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
mgs
mgs~2
mgs~1
PRODUCTS
mgs~3
mgs
Worksheet
Query Builder
CREATE OR REPLACE PROCEDURE insert_products
(
product_id_param NUMBER,
category_id_param NUMBER,
product_code_param VARCHAR2,
product_name_param VARCHAR2,
description_param VARCHAR2 DEFAULT (' '),
list_price_param NUMBER,
discount_percent_param NUMBER,
date_added_param DATE DEFAULT (SYSDATE)
)
Script Output
Task completed in 0.078 seconds
Procedure INSERT_PRODUCTS compiled
Statements - Log
Messages
Statements
Logging Page
Line 10 Column 62
Insert
Modified

```
SET SERVEROUTPUT ON;

BEGIN
insert_products (15, 1,'Good', 'THINGS', ' ', 111, 23, SYSDATE);

EXCEPTION
   WHEN VALUE_ERROR THEN
      DBMS_OUTPUT.PUT_LINE('Number Cannot Be Negavite');
   WHEN OTHERS THEN
      DBMS_OUTPUT.PUT_LINE('This is unexpected Error');

END;
/
```

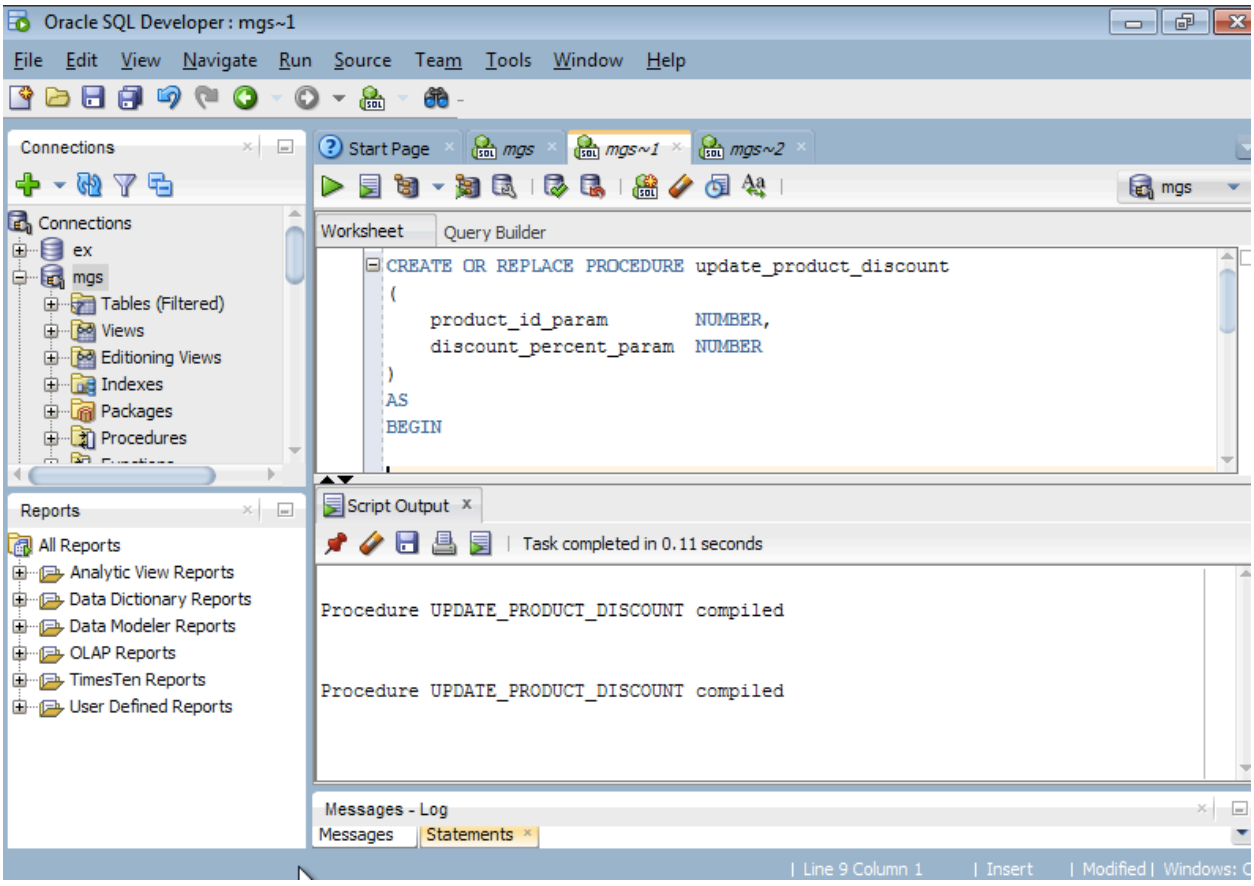


```
SET SERVEROUTPUT ON;

BEGIN
insert_products (15, 1,'BAD', 'THINGS', ' ', 111, -23, SYSDATE);

EXCEPTION
   WHEN VALUE_ERROR THEN
      DBMS_OUTPUT.PUT_LINE('Number Cannot Be Negavite');
   WHEN OTHERS THEN
      DBMS_OUTPUT.PUT_LINE('This is unexpected Error');

END;
/
```

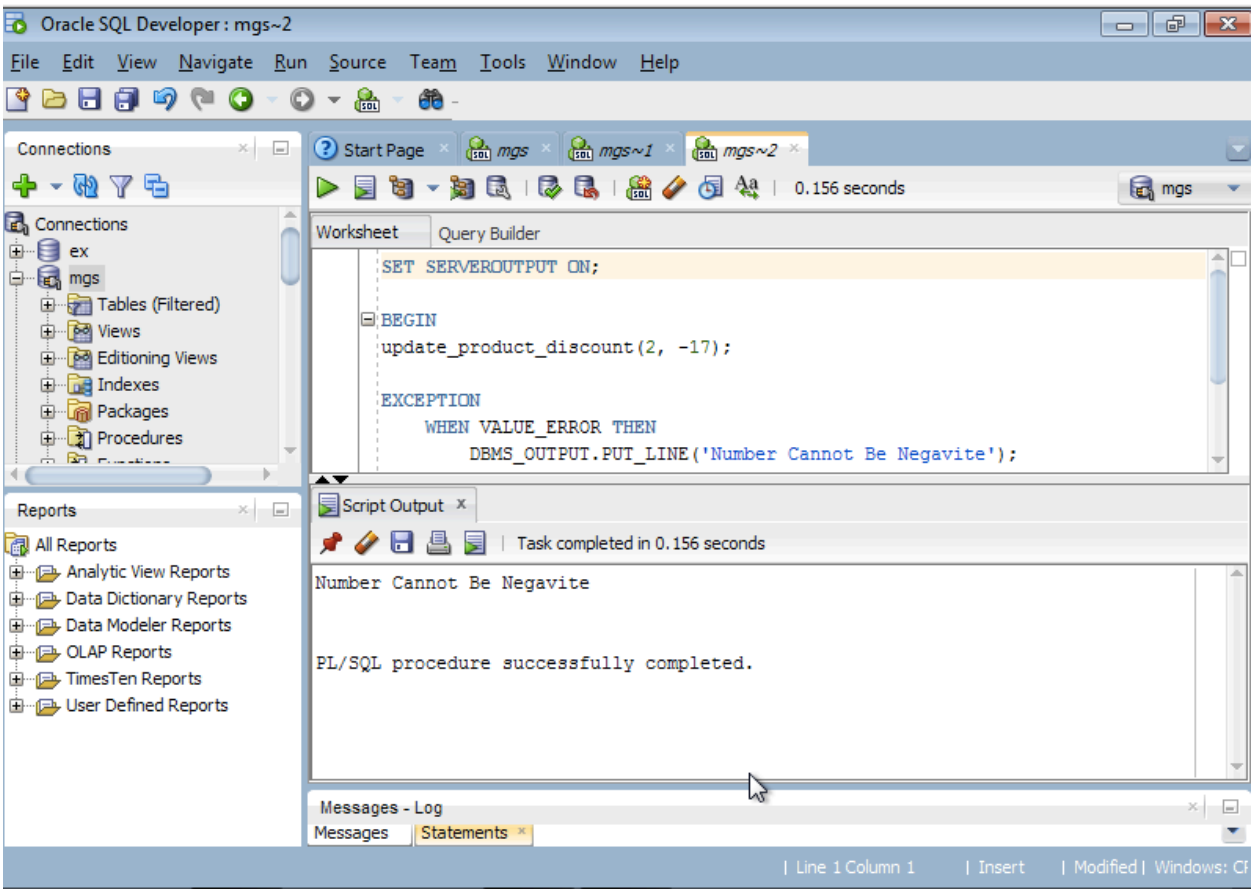


5.

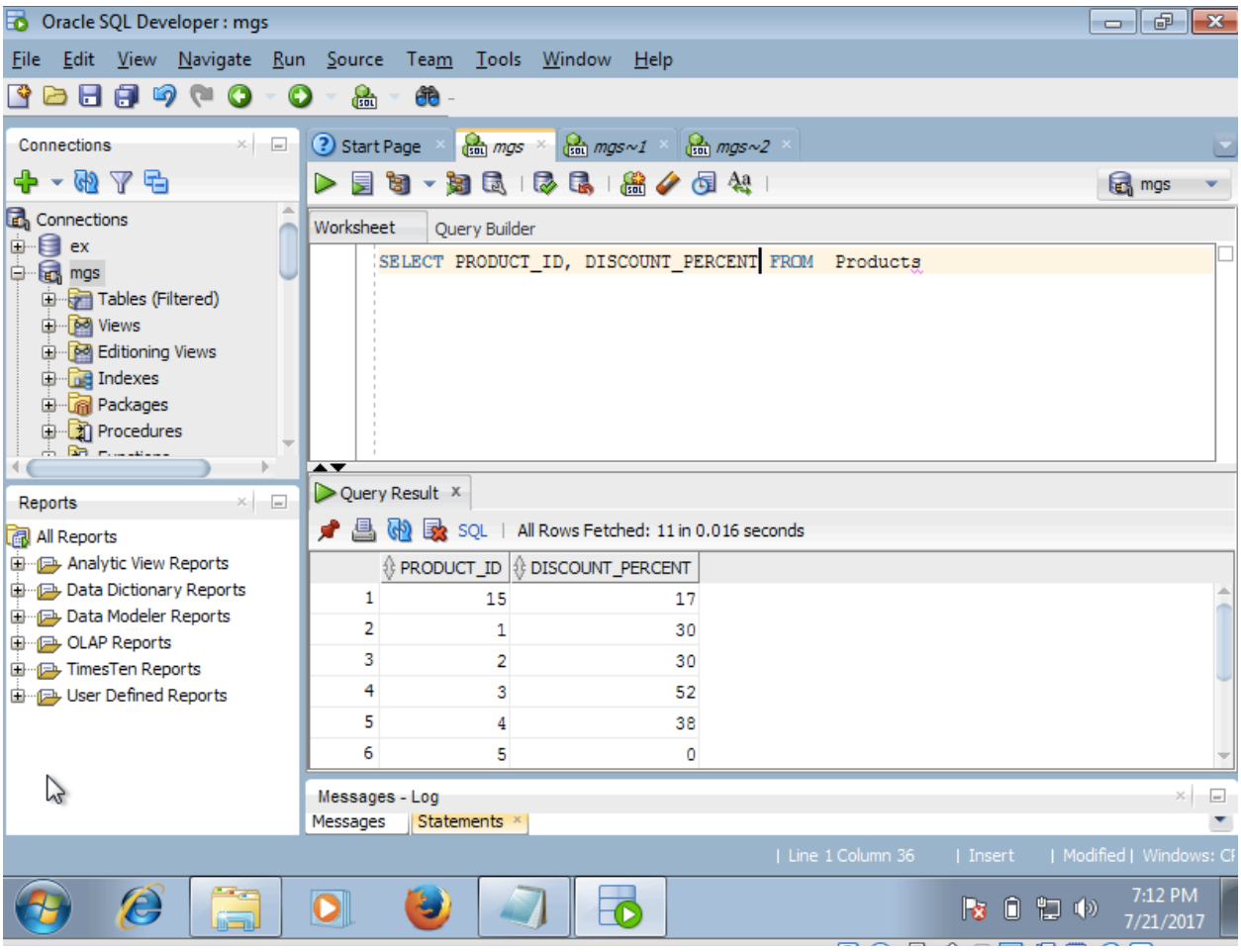
```
CREATE OR REPLACE PROCEDURE update_product_discount
(
   product_id_param       NUMBER,
   discount_percent_param  NUMBER
)
AS
BEGIN

   IF
      discount_percent_param < 0 THEN
      RAISE VALUE_ERROR;
   END IF;

   UPDATE PRODUCTS
   SET discount_percent = discount_percent_param
   WHERE product_id = product_id_param;

 COMMIT;
END;
/
```


Oracle SQL Developer : mgs~1
File Edit View Navigate Run Source Team Tools Window Help
Connections
Connections
ex
mgs
Tables (Filtered)
Views
Editioning Views
Indexes
Packages
Procedures
Reports
All Reports
Analytic View Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
mgs
mgs~1
mgs~2
mgs
Worksheet
Query Builder
CREATE OR REPLACE PROCEDURE update_product_discount
(
product_id_param NUMBER,
discount_percent_param NUMBER
)
AS
BEGIN
Script Output
Task completed in 0.11 seconds
Procedure UPDATE_PRODUCT_DISCOUNT compiled
Procedure UPDATE_PRODUCT_DISCOUNT compiled
Messages - Log
Messages
Statements
Line 9 Column 1
Insert
Modified

```
SET SERVEROUTPUT ON;

BEGIN
update_product_discount(2, -17);

EXCEPTION
   WHEN VALUE_ERROR THEN
      DBMS_OUTPUT.PUT_LINE('Number Cannot Be Negavite');
   WHEN OTHERS THEN
      DBMS_OUTPUT.PUT_LINE('This is unexpected Error');

END;
/
```



```
SET SERVEROUTPUT ON;

BEGIN
update_product_discount(15, 20);

EXCEPTION
   WHEN VALUE_ERROR THEN
      DBMS_OUTPUT.PUT_LINE('Number Cannot Be Negavite');
   WHEN OTHERS THEN
      DBMS_OUTPUT.PUT_LINE('This is unexpected Error');

END;
/
```



Updated Products Table: