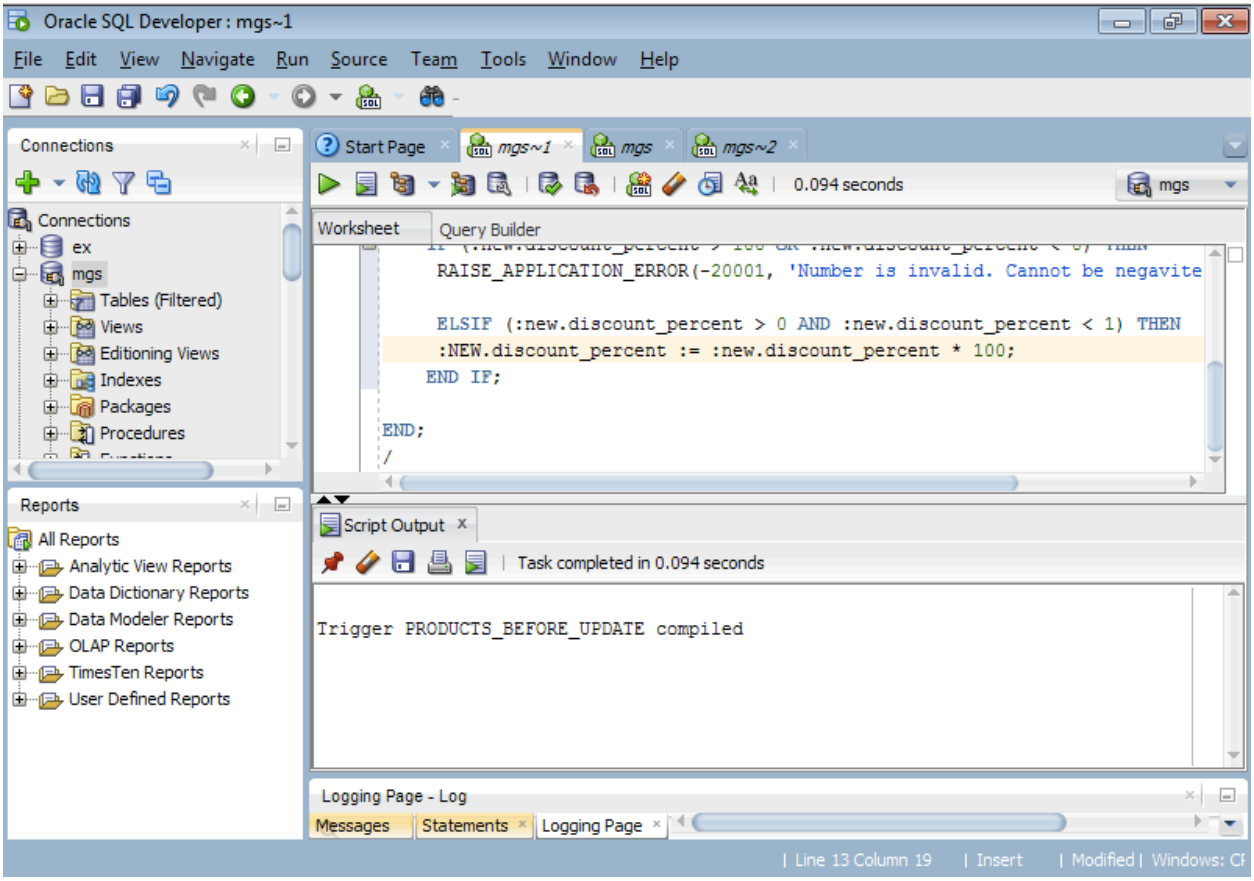
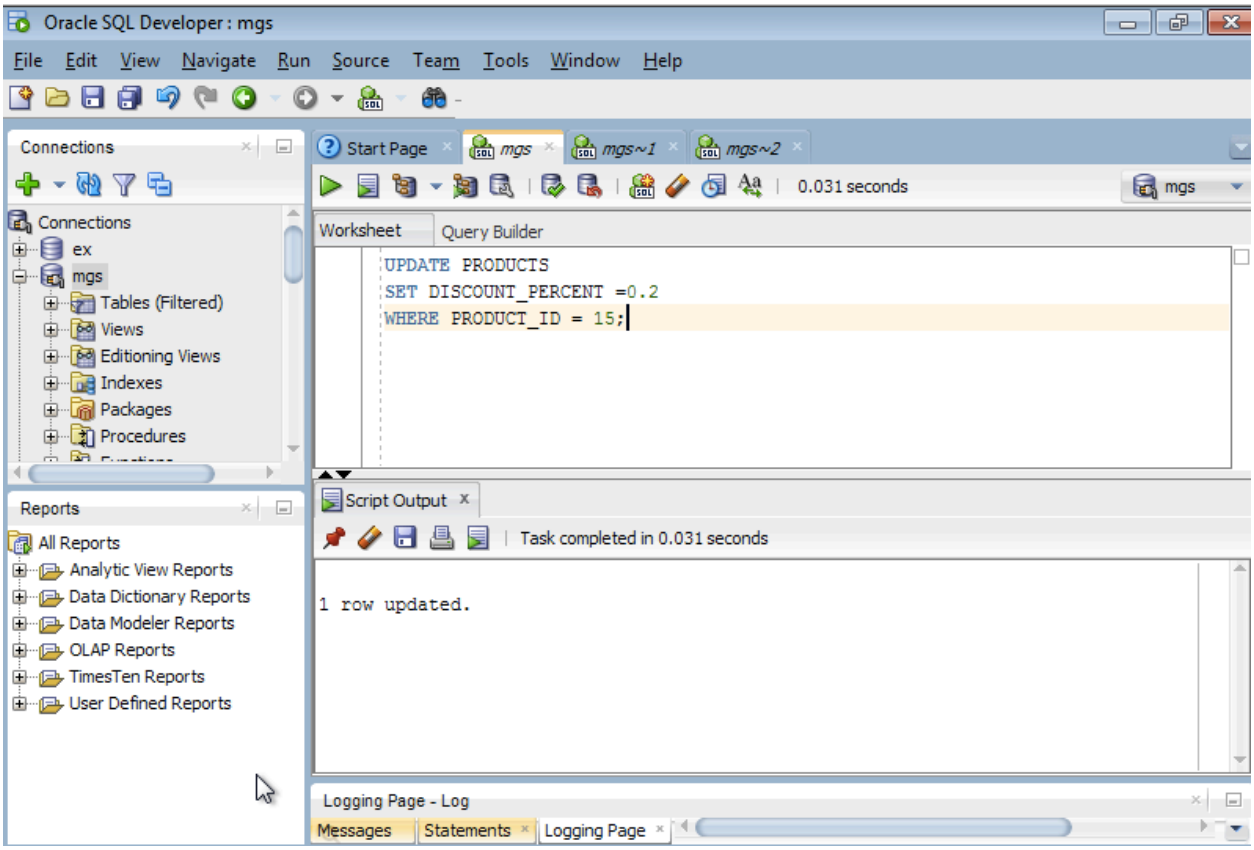
Budrys Assignment 8

1.

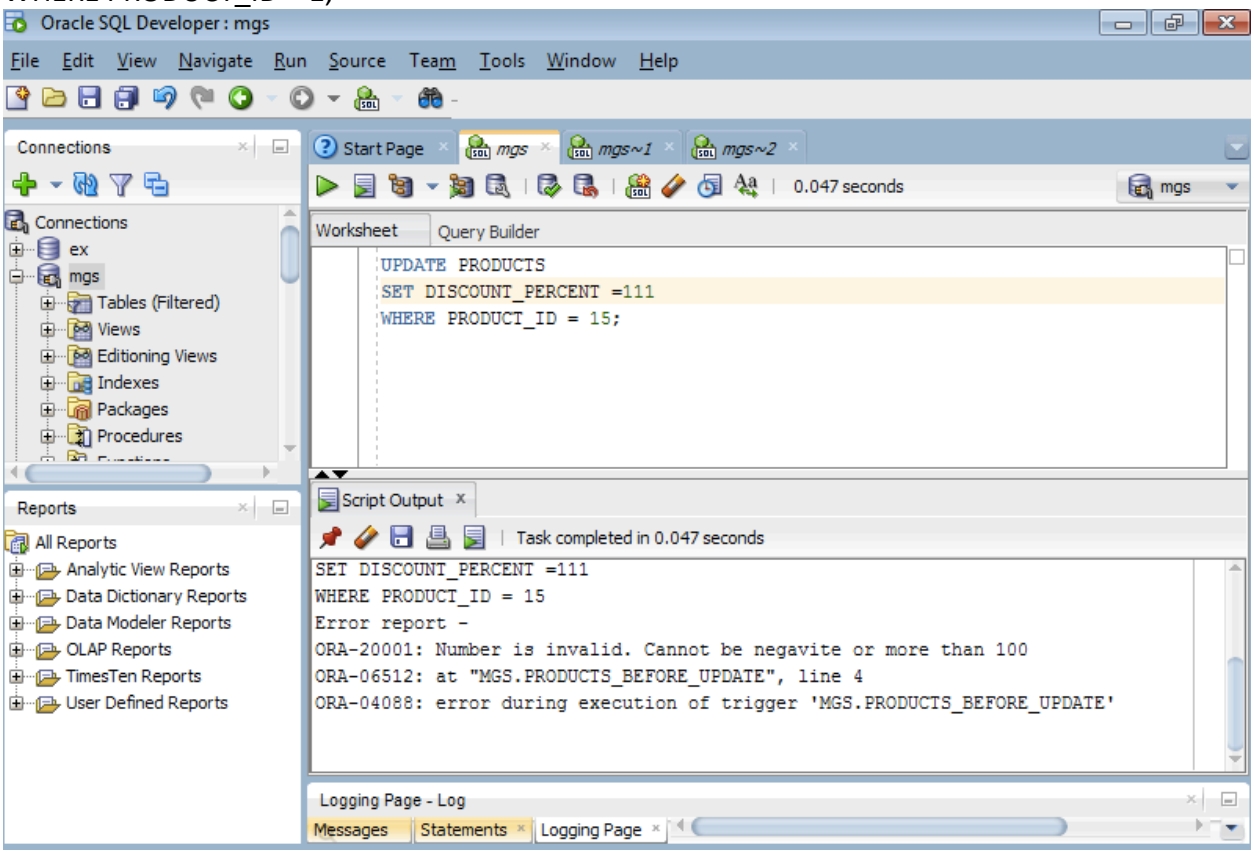
```
CREATE OR REPLACE TRIGGER products_before_update
BEFORE UPDATE OF discount_percent
ON Products
FOR EACH ROW

BEGIN

   IF (:new.discount_percent > 100 OR :new.discount_percent < 0) THEN
    RAISE_APPLICATION_ERROR(-20001, 'Number is invalid. Cannot be negavite or more than
100');

    ELSIF (:new.discount_percent > 0 AND :new.discount_percent < 1) THEN
    :NEW.discount_percent := :new.discount_percent * 100;
   END IF;

END;
/
```



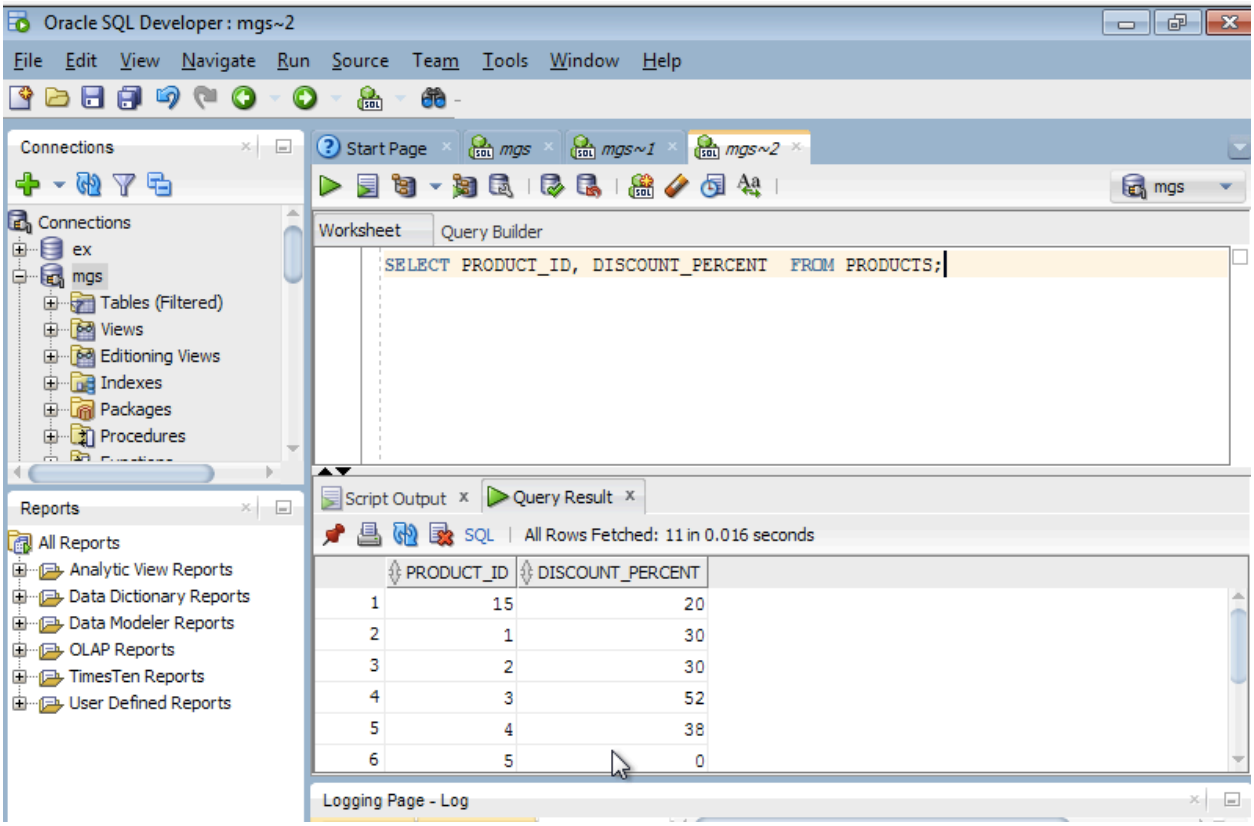
```
UPDATE PRODUCTS
SET DISCOUNT_PERCENT =0.2
WHERE PRODUCT_ID = 15;
```



UPDATE PRODUCTS
SET DISCOUNT_PERCENT =111
WHERE PRODUCT_ID = 1;

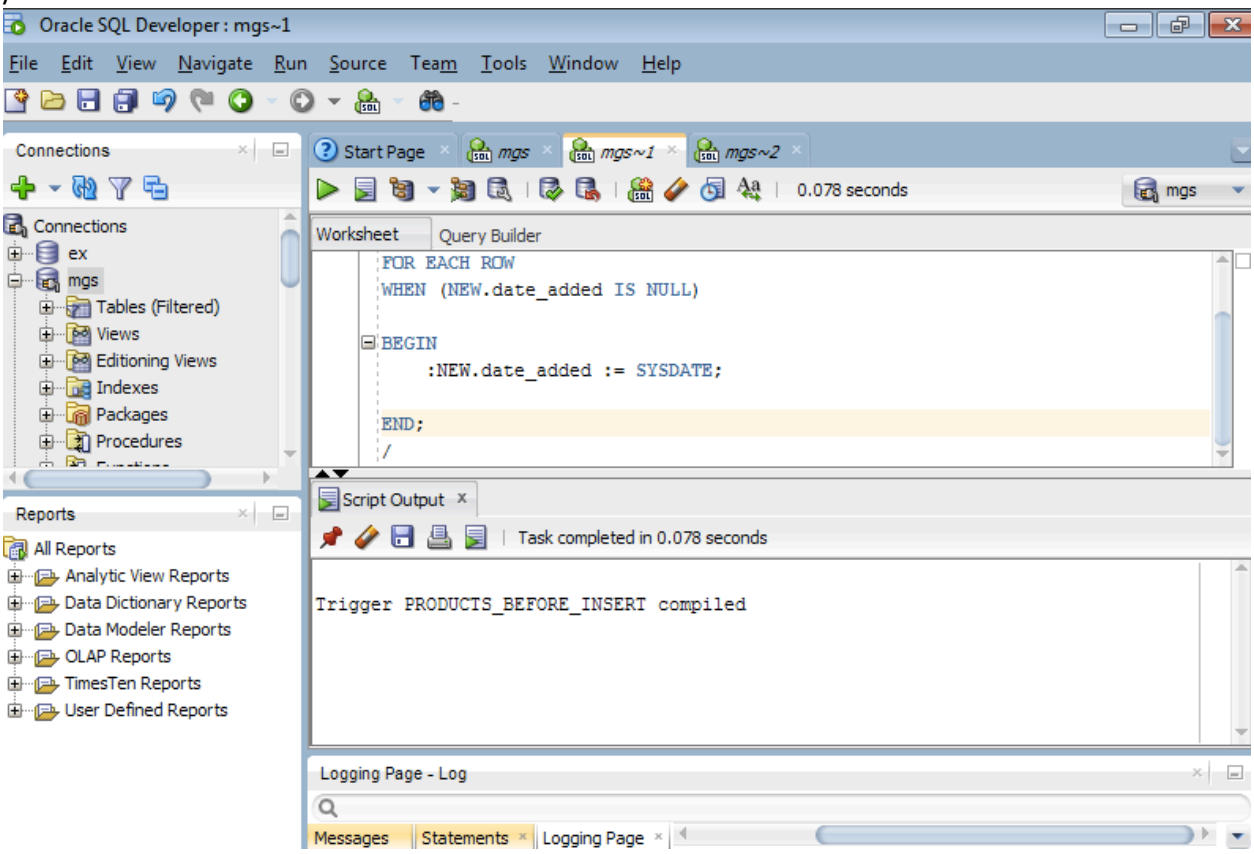


SELECT PRODUCT_ID, DISCOUNT_PERCENT FROM PRODUCTS;

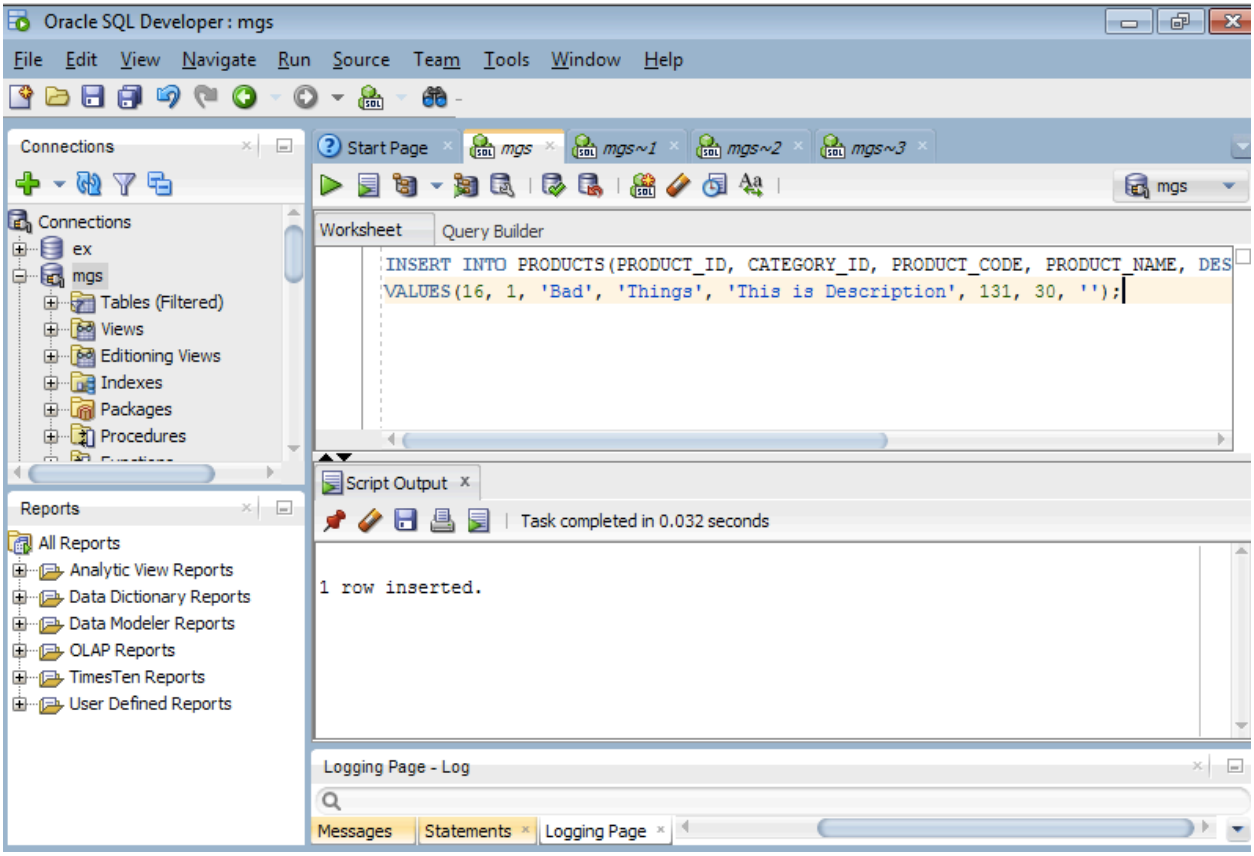


2.

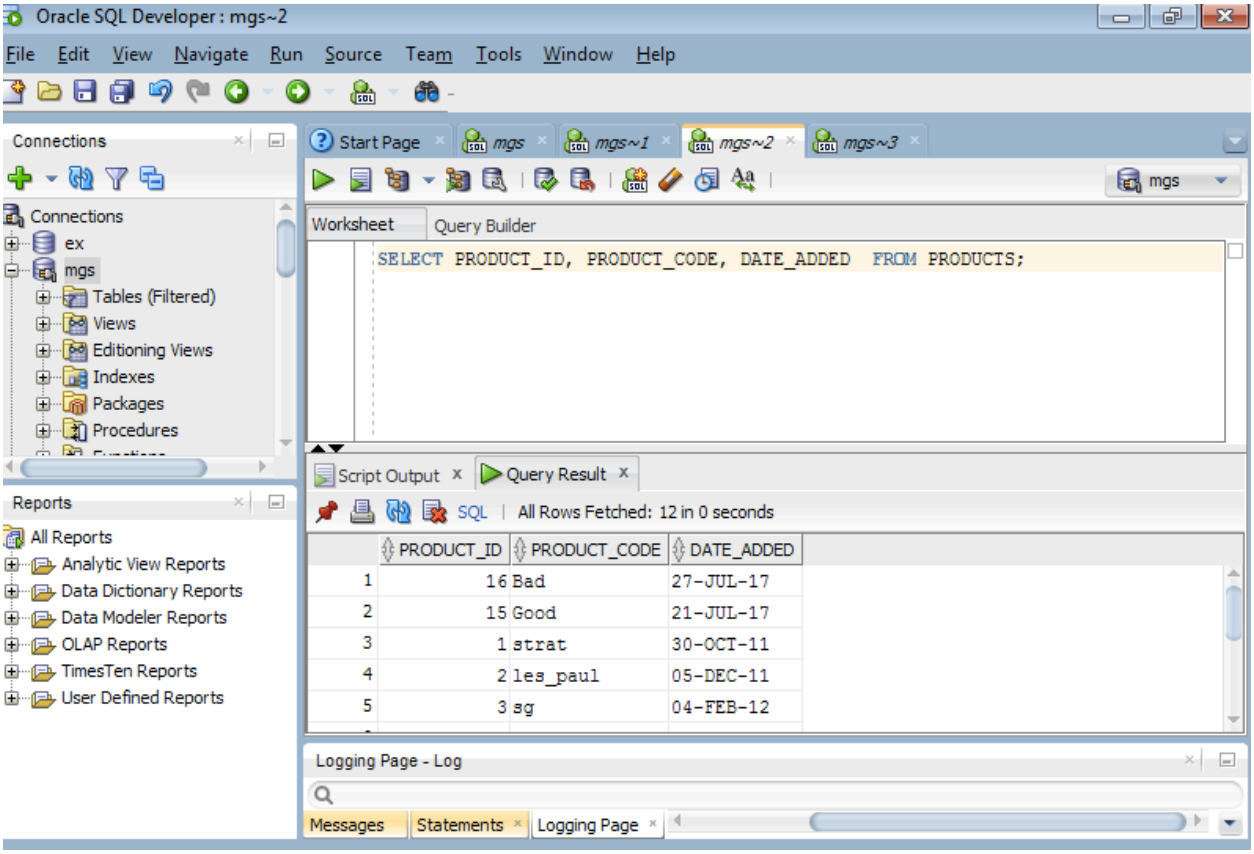
```
CREATE OR REPLACE TRIGGER products_before_insert
BEFORE INSERT ON Products
FOR EACH ROW
WHEN (NEW.date_added IS NULL)

BEGIN
   :NEW.date_added := SYSDATE;

END;
/
```



INSERT INTO PRODUCTS(PRODUCT_ID, CATEGORY_ID, PRODUCT_CODE, PRODUCT_NAME, DESCRIPTION, LIST_PRICE, DISCOUNT_PERCENT, DATE_ADDED)
VALUES(16, 1, 'Bad', 'Things', 'This is Description', 131, 30, '');

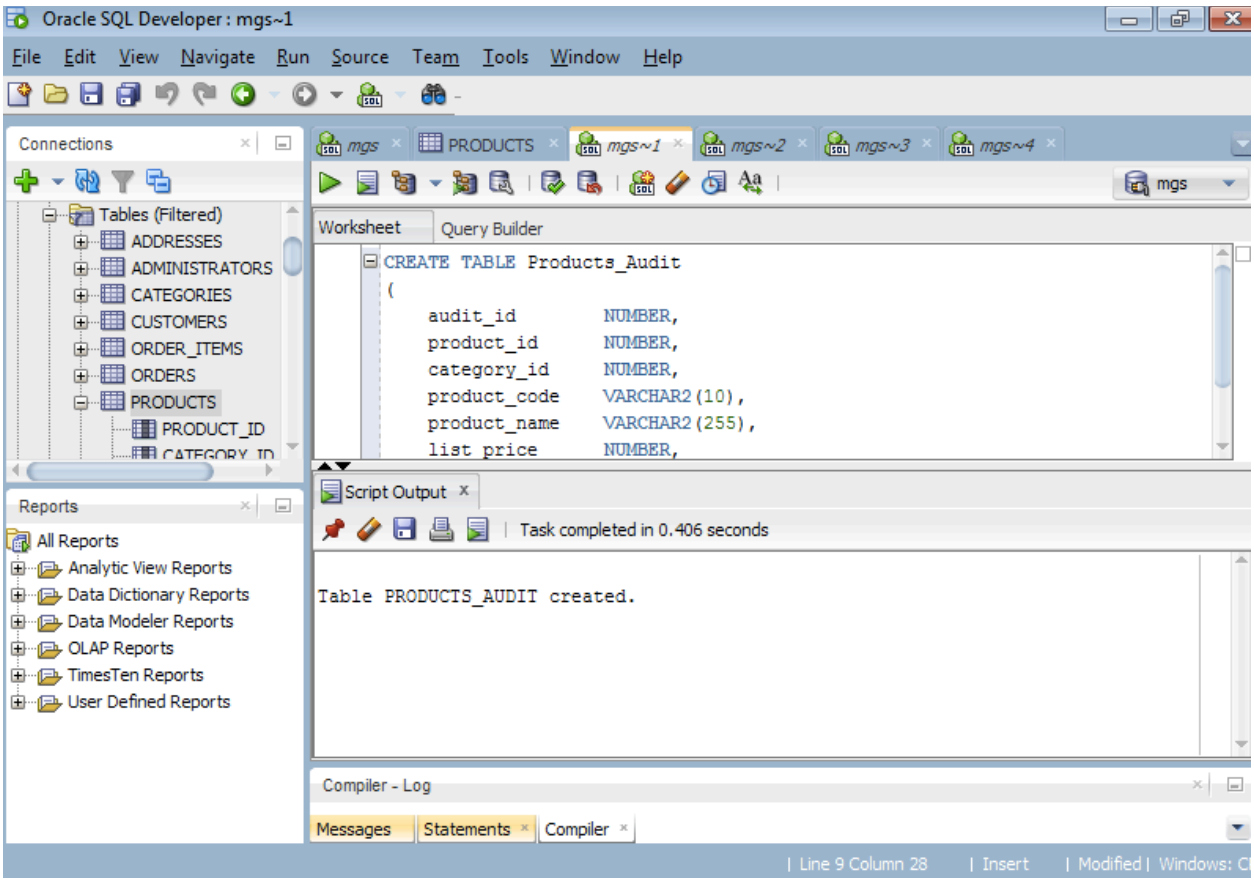


SELECT PRODUCT_ID, PRODUCT_CODE, DATE_ADDED FROM PRODUCTS;

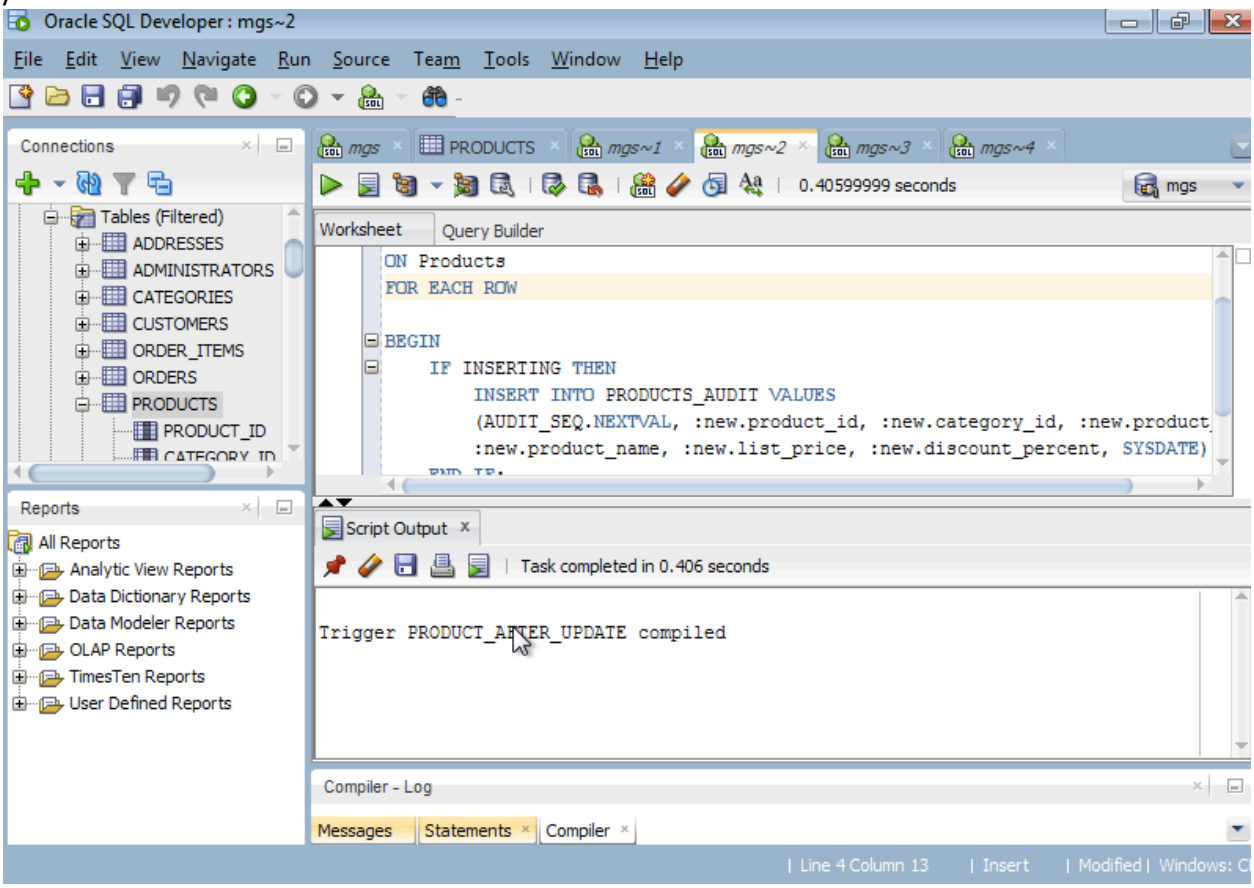


3.

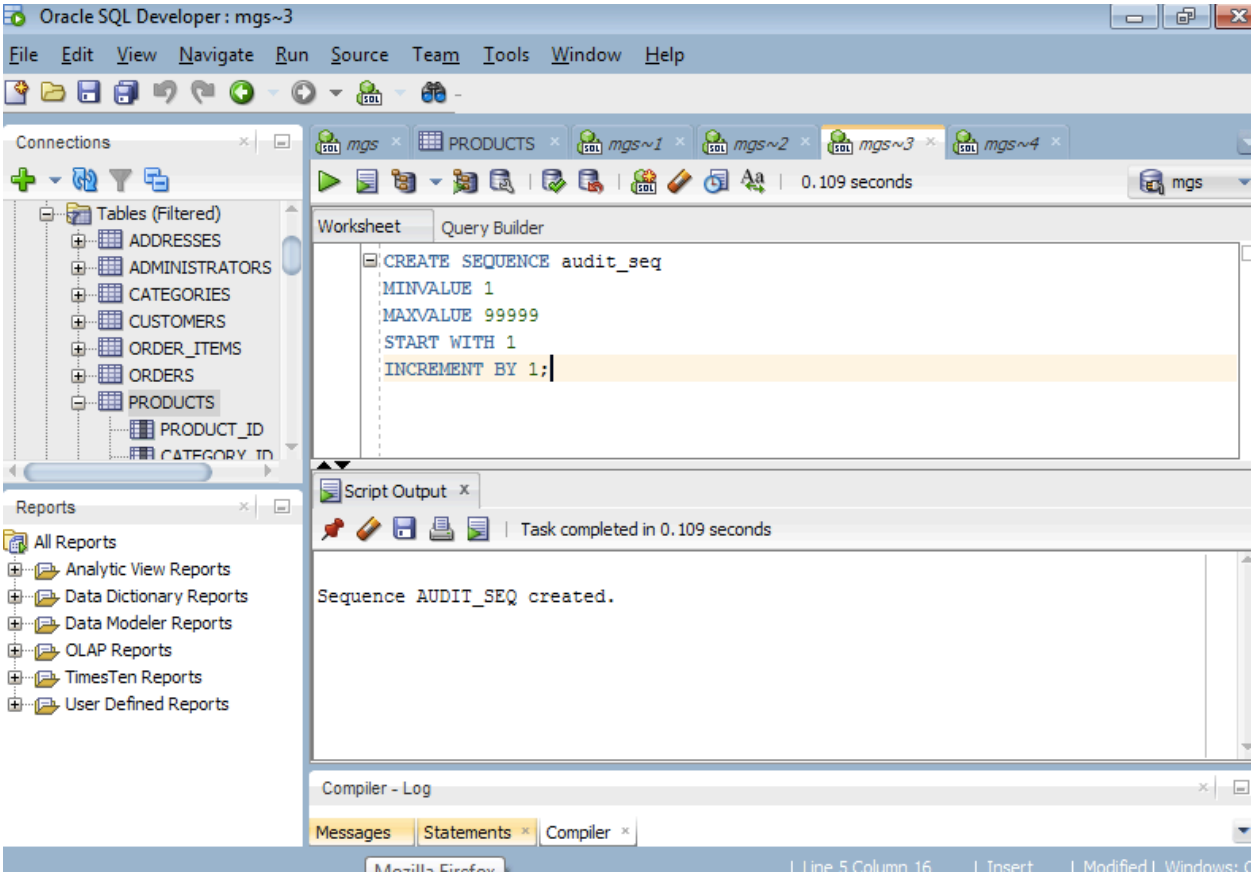
```
CREATE TABLE Products_Audit
(
   audit_id       NUMBER,
   product_id     NUMBER,
   category_id    NUMBER,
   product_code   VARCHAR2(10),
   product_name   VARCHAR2(255),
   list_price     NUMBER,
   discount_percent NUMBER,
   date_updated     DATE
);
```



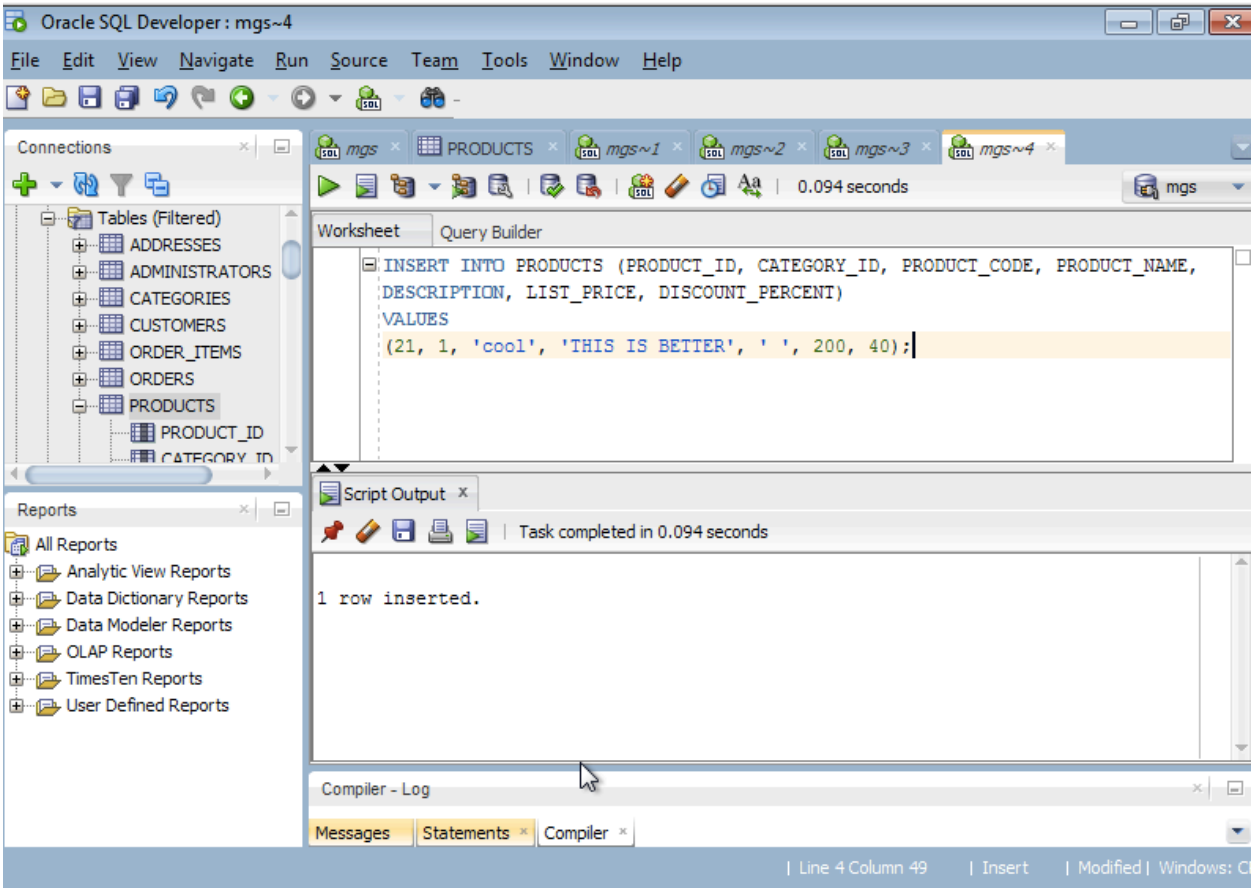
```
CREATE OR REPLACE TRIGGER product_after_update
AFTER INSERT OR UPDATE
ON Products
FOR EACH ROW

BEGIN
   IF INSERTING THEN
      INSERT INTO PRODUCTS_AUDIT VALUES
      (AUDIT_SEQ.NEXTVAL, :new.product_id, :new.category_id, :new.product_code,
      :new.product_name, :new.list_price, :new.discount_percent, SYSDATE);

   ELSIF UPDATING THEN
   INSERT INTO PRODUCTS_AUDIT VALUES
   (AUDIT_SEQ.NEXTVAL, :new.product_id, :new.category_id, :new.product_code,
      :new.product_name, :new.list_price, :new.discount_percent, SYSDATE);

   END IF;
END;
/
```



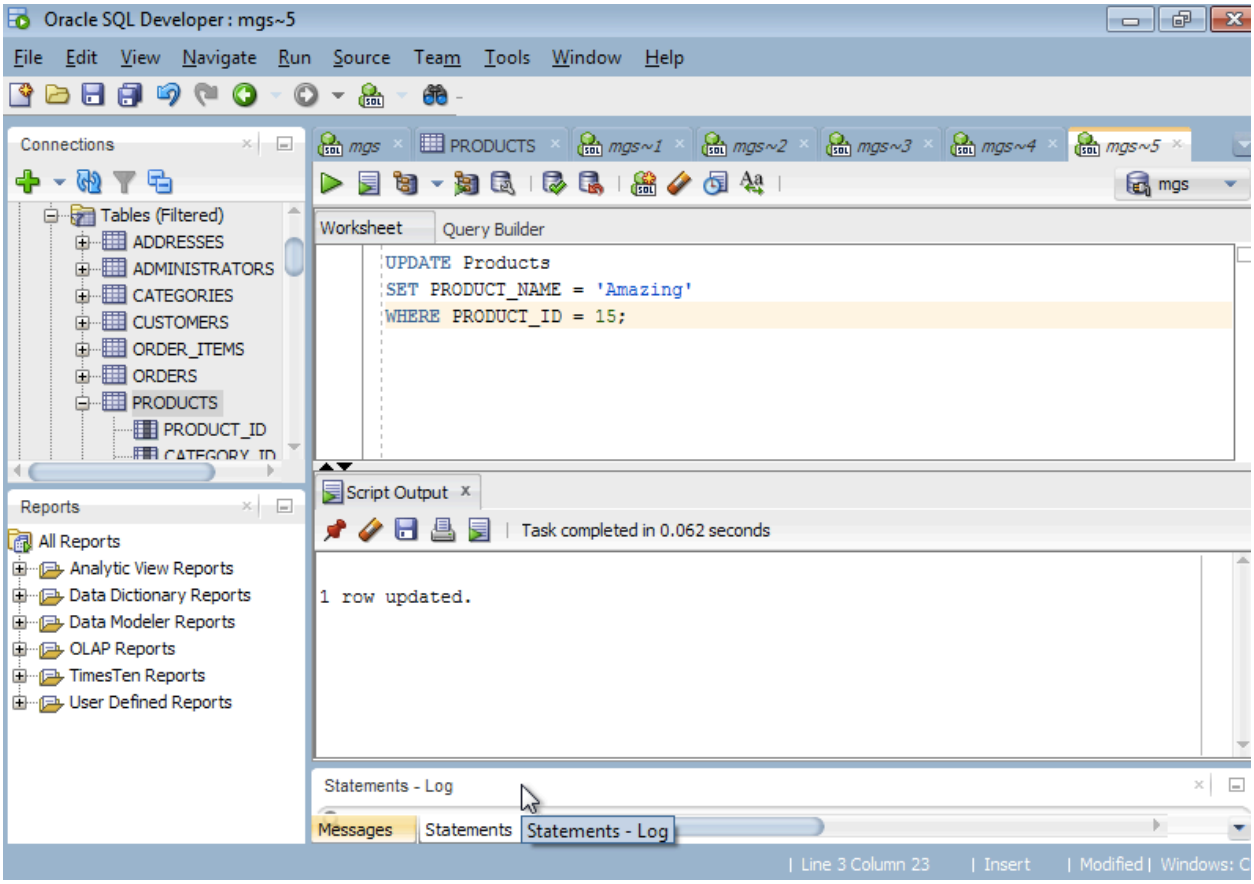
```
CREATE SEQUENCE audit_seq
MINVALUE 1
MAXVALUE 99999
START WITH 1
INCREMENT BY 1;
```



INSERT INTO PRODUCTS (PRODUCT_ID, CATEGORY_ID, PRODUCT_CODE, PRODUCT_NAME, DESCRIPTION, LIST_PRICE, DISCOUNT_PERCENT)
VALUES
(21, 1, 'cool', 'THIS IS BETTER', ' ', 200, 40);



```
UPDATE Products
SET PRODUCT_NAME = 'Amazing'
WHERE PRODUCT_ID = 15;
```



SELECT * FROM PRODUCTS_AUDIT