**ASSIGNMENT 5**

1. Code a script with the SQL statements to:
   * Use CREATE TABLE ... AS ... to create copies of the vendors and invoices tables, named vendors5 and invoices5, respectively,
     1. **CREATE TABLE VENDORS5 AS (SELECT \* FROM VENDORS);**
     2. **CREATE TABLE INVOICES5 AS (SELECT \* FROM INVOICES);**
   * Use ALTER statements to replicate the primary key, **all** foreign keys, and **all** constraints on the new tables to match the original ones. Foreign keys should reference the copy when possible.

**ALTER TABLE VENDORS5 ADD CONSTRAINT VENDORS5\_PK PRIMARY KEY (VENDOR\_ID);**

**ALTER TABLE VENDORS5 ADD CONSTRAINT VENDORS5\_VENDOR\_NAME\_UQ UNIQUE (VENDOR\_NAME);**

**ALTER TABLE VENDORS5 ADD CONSTRAINT VENDORS5\_FK\_TERMS FOREIGN KEY (DEFAULT\_TERMS\_ID) REFERENCES TERMS (TERMS\_ID);**

**ALTER TABLE VENDORS5 ADD CONSTRAINT VENDORS5\_FK\_ACCOUNTS FOREIGN KEY (DEFAULT\_ACCOUNT\_NUMBER) REFERENCES GENERAL\_LEDGER\_ACCOUNTS (ACCOUNT\_NUMBER);**

**ALTER TABLE INVOICES5 ADD CONSTRAINT INVOICES5\_PK PRIMARY KEY (INVOICE\_ID);**

**ALTER TABLE INVOICES5 ADD CONSTRAINT INVOICES5\_FK\_TERMS FOREIGN KEY (VENDOR\_ID) REFERENCES VENDORS5 (VENDOR\_ID);**

**ALTER TABLE INVOICES5 ADD CONSTRAINT INVOICES5\_FK\_VENDORS FOREIGN KEY (TERMS\_ID) REFERENCES TERMS (TERMS\_ID);**

* + Use an ALTER statement to add a column, named Balance, to invoices5. The data type should be the same as the other currency values.

**ALTER TABLE INVOICES5 ADD BALANCE NUMBER (9, 2) DEFAULT 0;**

* + Use an UPDATE statement to populate the balance due, which should contain the difference between the invoice total and the sum of the payment and credit.

**UPDATE INVOICES5**

**SET BALANCE = (INVOICE\_TOTAL - (PAYMENT\_TOTAL + CREDIT\_TOTAL));**

1. Code a PL/SQL block (i.e. BEGIN ... END) with the statements to:
   * Add a new vendor and three (3) new invoices, with different totals, to the new tables. Added data should be in the same format as existing data.  
       
     The vendor name should be **'Your Name, LLC'**. (Not the literal "Your Name").  
       
     **Use 5 for the vendor, and invoice, terms.**  
       
     The invoice date should be set to the current date (system date, nothing hard-coded), and the due date should be the current date + 30 days.  
       
     The invoice total should include a cents amount, and the same value should be used for the balance due. Set the payment and credit to zero.  
       
     Add one (1) to the highest value in the table to set the key value for the new records. Do not hard-code any new key values - **use variables**.

**CREATE OR REPLACE PROCEDURE VENDORS5\_INVOICES5\_NEW\_RECORDS**

**AS**

**P\_VENDOR\_ID NUMBER;**

**P\_VENDOR\_NAME VARCHAR2 (50):= 'Bhanumathi, LLC';**

**P\_VENDOR\_ADDRESS1 VARCHAR2 (50):= 'Jackson';**

**P\_VENDOR\_ADDRESS2 VARCHAR2 (50):= 'APT# 203';**

**P\_VENDOR\_CITY VARCHAR2 (50):= 'Chicago';**

**P\_VENDOR\_STATE CHAR (2):= 'IL';**

**P\_VENDOR\_ZIP\_CODE VARCHAR2 (20):= '60616';**

**P\_VENDOR\_PHONE VARCHAR (50):= '(123) 456-8596';**

**P\_VENDOR\_CONTACT\_LAST\_NAME VARCHAR2 (50):= 'AAA';**

**P\_VENDOR\_CONTACT\_FIRST\_NAME VARCHAR2 (50):= 'BBB';**

**P\_DEFAULT\_TERMS\_ID NUMBER:= 5;**

**P\_DEFAULT\_ACCOUNT\_NUMBER NUMBER:= 100;**

**P\_INVOICE\_ID NUMBER;**

**P\_INVOICE\_NUMBER VARCHAR2 (50):= 'A1000';**

**P\_INVOICE\_DATE DATE:= SYSDATE;**

**P\_INVOICE\_TOTAL NUMBER (9, 2):= 100.10;**

**P\_PAYMENT\_TOTAL NUMBER (9, 2):= 0;**

**P\_CREDIT\_TOTAL NUMBER (9, 2):= 0;**

**P\_TERMS\_ID NUMBER:= 5;**

**P\_INVOICE\_DUE\_DATE DATE;**

**P\_PAYMENT\_DATE DATE;**

**P\_BALANCE NUMBER:= 0;**

**i NUMBER :=1 ;**

**BEGIN**

**SELECT MAX (VENDOR\_ID) + 1 INTO P\_VENDOR\_ID FROM VENDORS5;**

**INSERT INTO VENDORS5 VALUES (P\_VENDOR\_ID,**

**P\_VENDOR\_NAME,**

**P\_VENDOR\_ADDRESS1,**

**P\_VENDOR\_ADDRESS2,**

**P\_VENDOR\_CITY,**

**P\_VENDOR\_STATE,**

**P\_VENDOR\_ZIP\_CODE,**

**P\_VENDOR\_PHONE,**

**P\_VENDOR\_CONTACT\_LAST\_NAME,**

**P\_VENDOR\_CONTACT\_FIRST\_NAME,**

**P\_DEFAULT\_TERMS\_ID,**

**P\_DEFAULT\_ACCOUNT\_NUMBER);**

**WHILE i<4 LOOP**

**SELECT MAX (INVOICE\_ID) + 1 INTO P\_INVOICE\_ID FROM INVOICES5;**

**SELECT SYSDATE + INTERVAL '30' DAY INTO P\_INVOICE\_DUE\_DATE FROM DUAL;**

**INSERT INTO INVOICES5 VALUES (P\_INVOICE\_ID,**

**P\_VENDOR\_ID,**

**P\_INVOICE\_NUMBER,**

**P\_INVOICE\_DATE,**

**P\_INVOICE\_TOTAL,**

**P\_PAYMENT\_TOTAL,**

**P\_CREDIT\_TOTAL,**

**P\_TERMS\_ID,**

**P\_INVOICE\_DUE\_DATE,**

**P\_PAYMENT\_DATE,**

**P\_BALANCE);**

**P\_INVOICE\_TOTAL:= P\_INVOICE\_TOTAL + 2000;**

**i := i + 1;**

**END LOOP;**

**END;**

**/**

**EXECUTE VENDORS5\_INVOICES5\_NEW\_RECORDS;**

* Update the credit in the new invoices for the new vendor, setting it to 10% of the invoice total (rounding to 2 decimal places), and also updating the balance due amount.

**CREATE OR REPLACE PROCEDURE UPDATE\_CREDIT\_BAL**

**AS**

**P\_VENDOR\_ID NUMBER (9, 2);**

**BEGIN**

**SELECT MAX (VENDOR\_ID) INTO P\_VENDOR\_ID**

**FROM INVOICES5;**

**UPDATE INVOICES5**

**SET CREDIT\_TOTAL = INVOICE\_TOTAL \* 0.1**

**WHERE VENDOR\_ID = P\_VENDOR\_ID;**

**UPDATE INVOICES5**

**SET BALANCE = INVOICE\_TOTAL – (PAYMENT\_TOTAL + CREDIT\_TOTAL)**

**WHERE VENDOR\_ID = P\_VENDOR\_ID;**

**END;**

**/**

**EXECUTE UPDATE\_CREDIT\_BAL;**

1. Code the SQL statements to:
   * List the added records: vendor id & name, invoice id, 2 dates, and 3 currency values (omit payment).  
       
     Both dates should be displayed with the full month name, e.g. May 10, 2015, with no extra spaces after the name.

**SET LINESIZE 999**

**COLUMN VENDOR\_NAME FORMAT A15**

**COLUMN INVOICE\_TOTAL FORMAT A15**

**COLUMN CREDIT\_TOTAL FORMAT A15**

**COLUMN BALANCE FORMAT A15**

**SELECT V.VENDOR\_ID,**

**V.VENDOR\_NAME,**

**I.INVOICE\_ID,**

**TO\_CHAR (I.INVOICE\_DATE,'fmMONTHDD, YYYY’) AS INVOICE\_DATE,**

**TO\_CHAR (I.INVOICE\_DUE\_DATE, 'fmMONTHDD, YYYY’) AS INVOICE\_DUE\_DATE,**

**TO\_CHAR (I.INVOICE\_TOTAL, '$999,999,99.99') AS INVOICE\_TOTAL,**

**TO\_CHAR (I.CREDIT\_TOTAL, '$999,999,99.99') AS CREDIT\_TOTAL,**

**TO\_CHAR (I.BALANCE, '$999,999,99.99') AS BALANCE**

**FROM VENDORS5 V**

**JOIN INVOICES5 I**

**ON V.VENDOR\_ID = I.VENDOR\_ID**

**WHERE V.VENDOR\_ID = (SELECT MAX (VENDOR\_ID) FROM VENDORS5);**

**OUTPUT:**

VENDOR\_ID VENDOR\_NAME INVOICE\_ID INVOICE\_DATE INVOICE\_DUE\_DATE INVOICE\_TOTAL CREDIT\_TOTAL BALANCE\_DUE

---------- --------------- ---------- ---------------- ---------------- --------------- --------------- ---------------

124 Bhanumathi, LLC 115 OCTOBER26,2015 NOVEMBER25,2015 $1,00.10 $10.01 $90.09

124 Bhanumathi, LLC 116 OCTOBER26,2015 NOVEMBER25,2015 $21,00.10 $2,10.01 $18,90.09

124 Bhanumathi, LLC 117 OCTOBER26,2015 NOVEMBER25,2015 $41,00.10 $4,10.01 $36,90.09

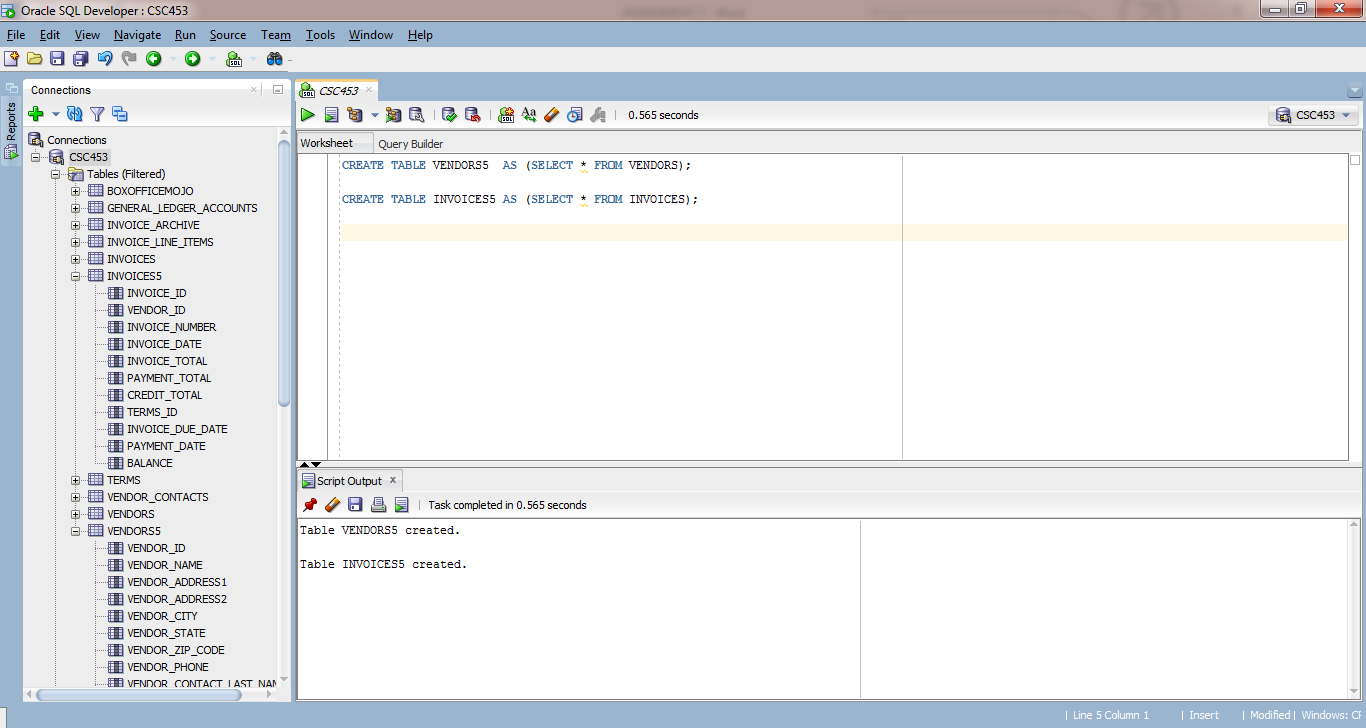
* + Ensure the database is restored to its initial state, dropping created objects.

DROP TABLE INVOICES5;

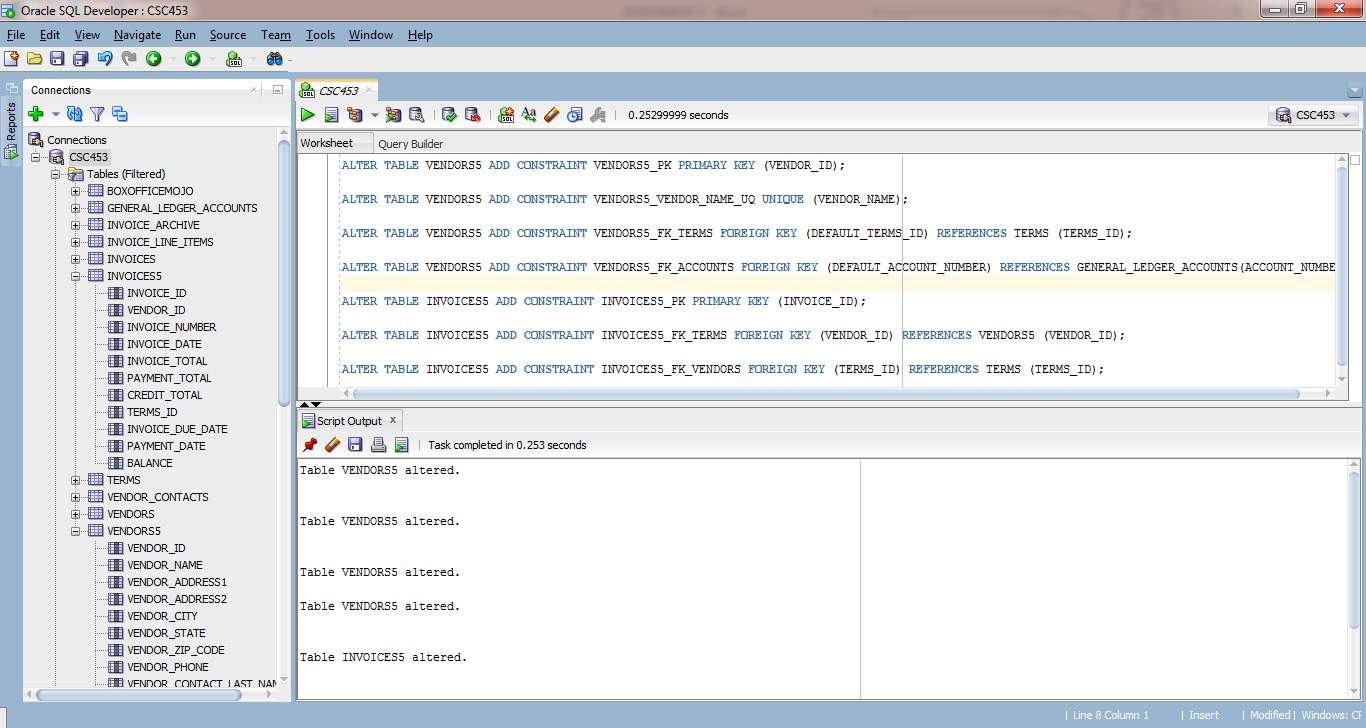
DROP TABLE VENDORS5;

**OUTPUT:**

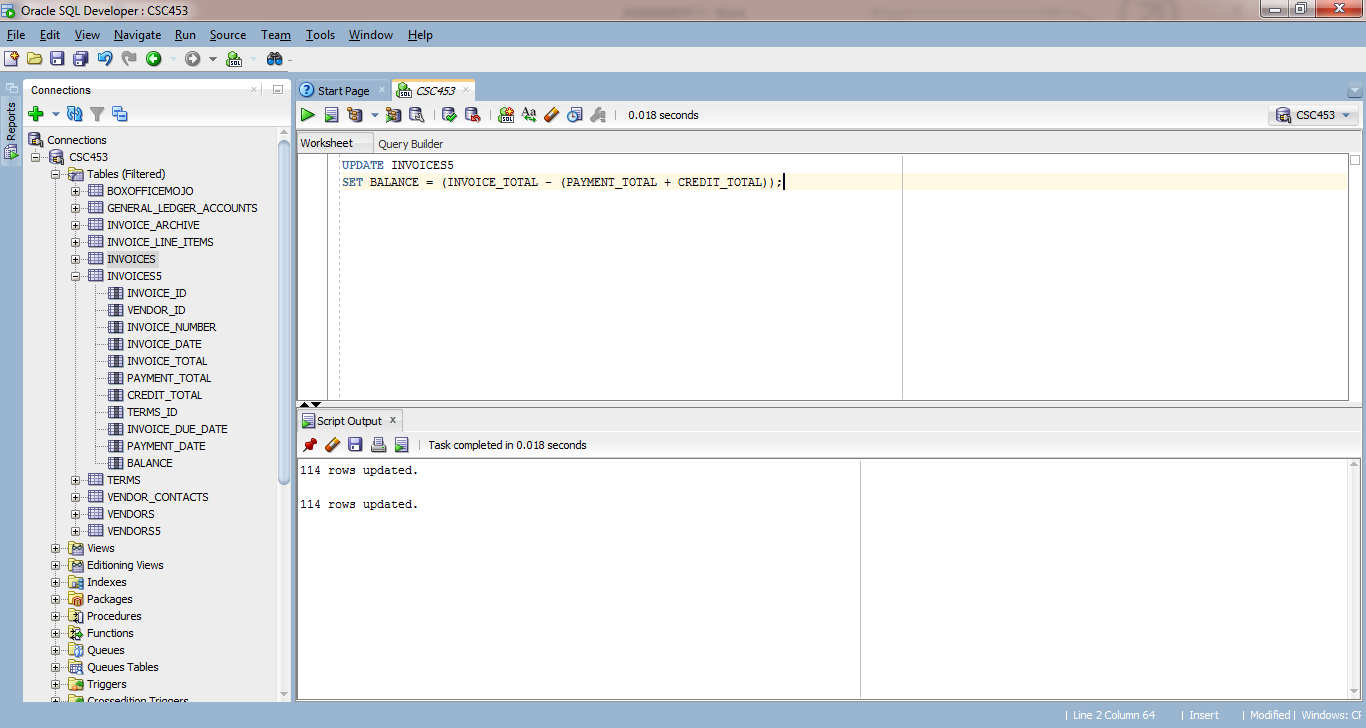
**Q1.A**



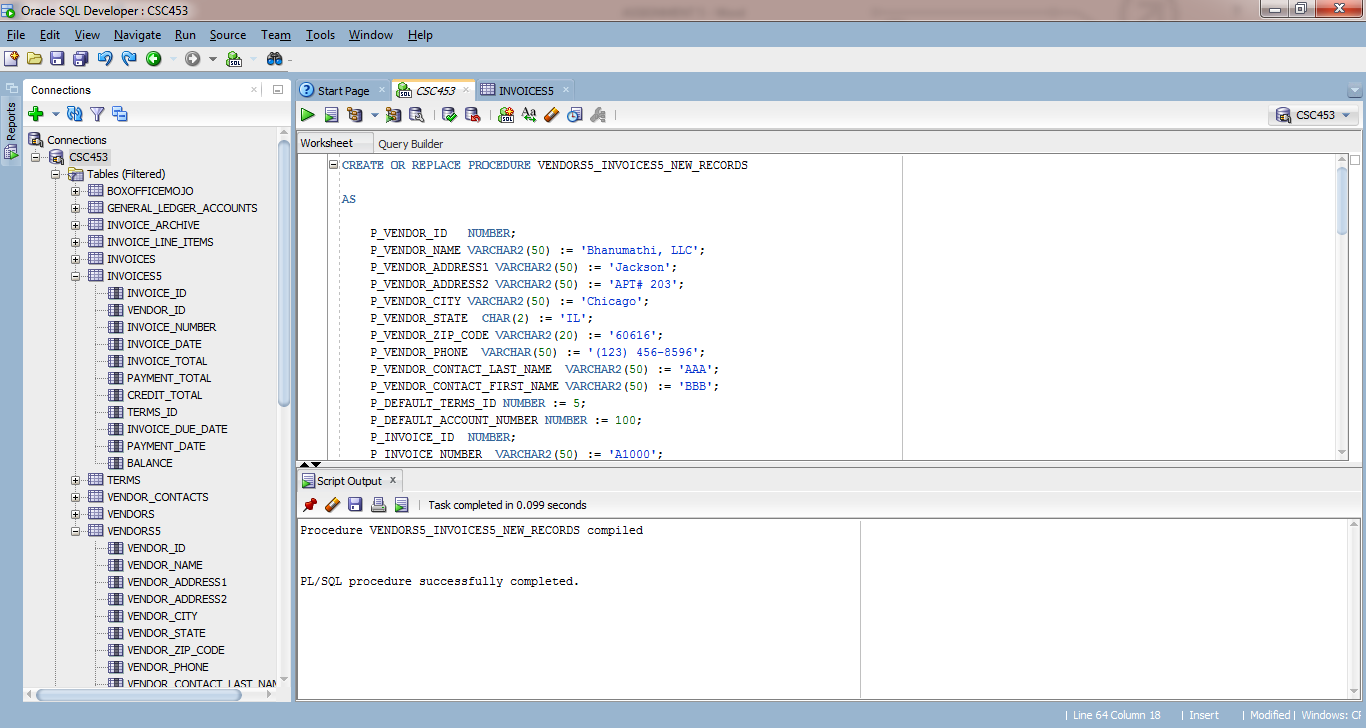
**Q1.B**



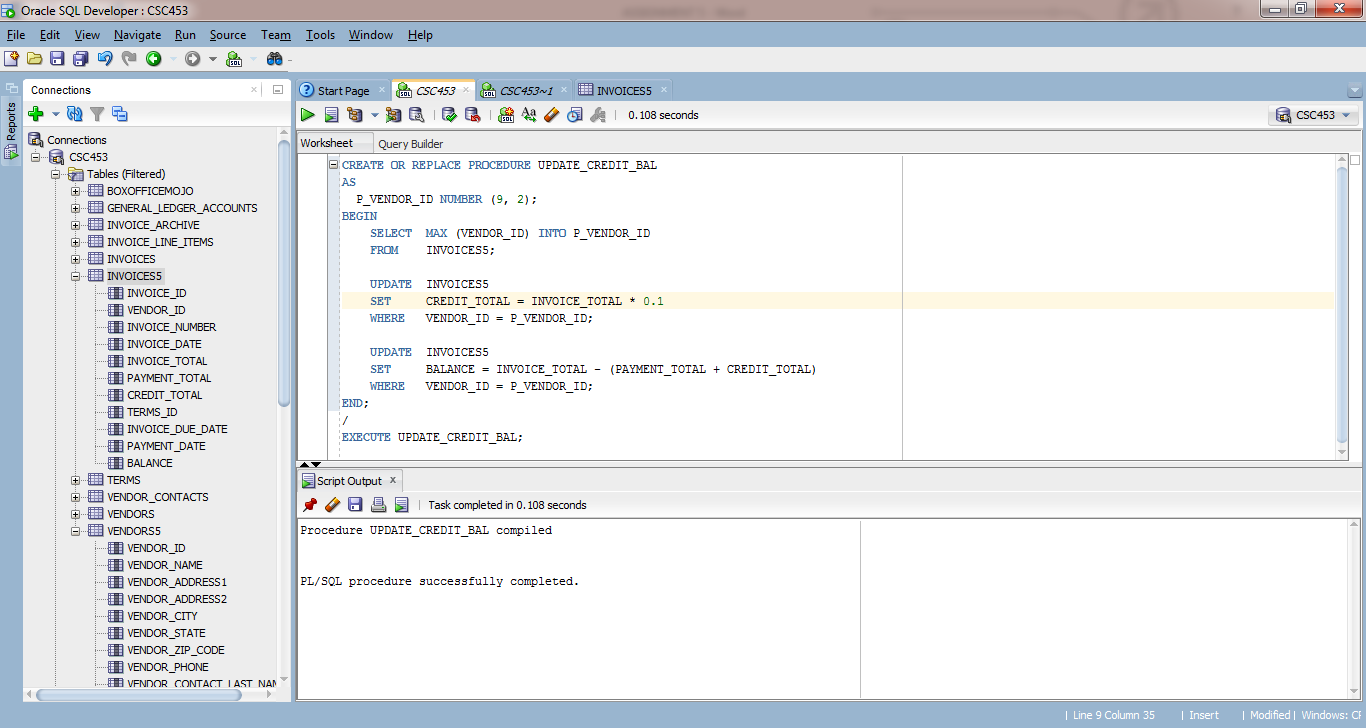
**Q1.D**



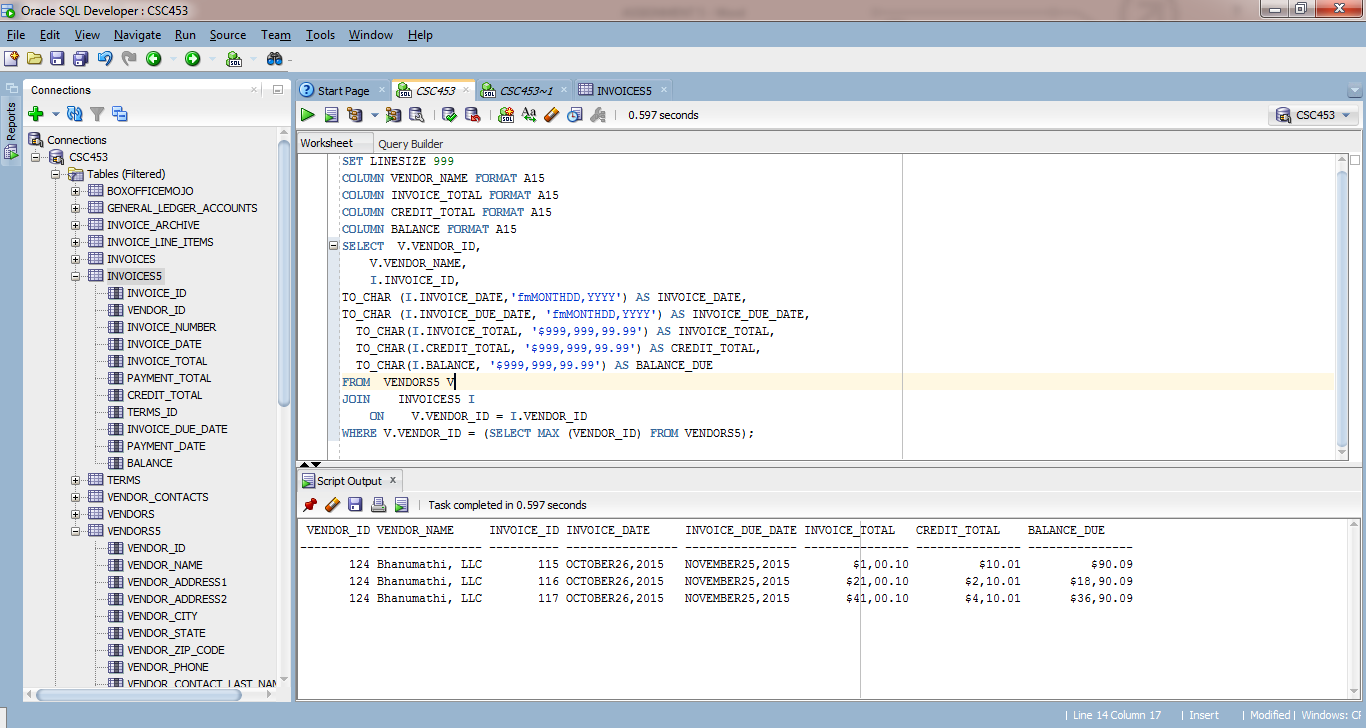
**Q2.A**



**Q2.B**



**Q3.A**



**Q3.B**

