

PART - 1

(a)

```
CREATE TABLE VENDORS5 AS  
SELECT *  
FROM VENDORS;
```

```
CREATE TABLE INVOICES5 AS  
SELECT *  
FROM INVOICES;
```

(b)

```
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_VENDORS  
FOREIGN KEY (VENDOR_ID)  
REFERENCES VENDORS5 (VENDOR_ID);
```

```
ALTER TABLE INVOICES5  
ADD CONSTRAINT INVOICES5_FK_TERMS  
FOREIGN KEY (TERMS_ID)  
REFERENCES TERMS (TERMS_ID);
```

(c)

```
-- QUESTION 1 : QUERY 3 : ALTERING TABLE INVOICES5 TO ADD NEW COLUMN BALANCE_DUE.
```

```
ALTER TABLE INVOICES5  
ADD BALANCE NUMBER(9,2);
```

(D)

```
-- QUESTION 1 : QUERY 4 : UPDATING TABLE INVOICES5 TO ADD DATA IN BALANCE_DUE COLUMN.
```

```
UPDATE INVOICES5  
SET BALANCE = INVOICES5.INVOICE_TOTAL- INVOICES5.PAYMENT_TOTAL - INVOICES5.CREDIT_TOTAL;
```

OR

UPDATE INVOICES5

SET BALANCE = INVOICE_TOTAL- PAYMENT_TOTAL - CREDIT_TOTAL;

PART - 2

(A)

-- QUESTION 2 : PART 1: INSERTING VALUES IN NEWLY CREAETD TABLE WITH USE OF VARIABLES.

DECLARE

X NUMBER;

Y NUMBER;

BEGIN

SELECT COUNT(*) INTO X FROM VENDORS5;

INSERT INTO VENDORS5

VALUES (X+2,'Advait Patel, LLC','420 Lotus Lane',NULL,'Glenview','IL',60025,8475084136,'Patel','Advait',3,620);

FOR I IN 1..3 LOOP

SELECT COUNT(*) INTO Y FROM INVOICES5 ;

INSERT INTO INVOICES5

VALUES (Y+1,X+2,'12345-OUT',SYSDATE,2550.58,0,0,3,SYSDATE+30,NULL,2550.58);

END LOOP;

COMMIT;

END;

/

OR

```
CREATE OR REPLACE PROCEDURE MYPROC
```

```
AS
```

```
VENDOR_ID_P NUMBER;
```

```
VENDOR_NAME_P VARCHAR2 (50) := 'Advait, LLC';
```

```
VENDOR_ADDRESS1_P VARCHAR2 (50) := '420';
```

```
VENDOR_ADDRESS2_P VARCHAR2 (50) := 'Lotus Lane';
```

```
VENDOR_CITY_P VARCHAR2 (50) := 'GLENVIEW';
```

```
VENDOR_STATE_P CHAR (2) := 'IL';
```

```
VENDOR_ZIP_CODE_P VARCHAR2 (20) := '60025';
```

```
VENDOR_PHONE_P VARCHAR (50) := '(847) 508-4136';
```

```
VENDOR_CONTACT_LAST_NAME_P VARCHAR2 (50) := 'Depaul';
```

```
VENDOR_CONTACT_FIRST_NAME_P VARCHAR2 (50) := 'University';
```

```
DEFAULT_TERMS_ID_P NUMBER := 5;
```

```
DEFAULT_ACCOUNT_NUMBER_P NUMBER := 507;
```

```
INVOICE_ID_P NUMBER;
```

```
INVOICE_NUMBER_P VARCHAR2 (50) := 'AP4200';
```

```
INVOICE_DATE_P DATE := SYSDATE;
```

INVOICE_TOTAL_P NUMBER (9, 2) := 1178.63;

PAYMENT_TOTAL_P NUMBER (9, 2) := 0;

CREDIT_TOTAL_P NUMBER (9, 2) := 0;

TERMS_ID_P NUMBER := 5;

INVOICE_DUE_DATE_P DATE;

PAYMENT_DATE_P DATE;

BALANCE_P NUMBER ;

i NUMBER := 1 ;

BEGIN

SELECT MAX (VENDOR_ID) + 1 INTO VENDOR_ID_P FROM VENDORS5;

INSERT INTO VENDORS5 VALUES (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_P);
```

```
WHILE i<4 LOOP
```

```
    SELECT MAX (INVOICE_ID) + 1 INTO INVOICE_ID_P FROM INVOICES5;
```

```
    SELECT SYSDATE + INTERVAL '30' DAY INTO INVOICE_DUE_DATE_P FROM DUAL;
```

```
    INSERT INTO INVOICES5 VALUES (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_P + 7000;
```

```
        i := i + 1;
    END LOOP;
END;
/
EXECUTE MYPROC;
```

(B)

```
CREATE OR REPLACE PROCEDURE UPDATEPROC
AS
    VENDOR_ID_P NUMBER (9, 2);
BEGIN
    SELECT    MAX (VENDOR_ID) INTO VENDOR_ID_P
    FROM      INVOICES5;
```

```
UPDATE    INVOICES5
SET       CREDIT_TOTAL = INVOICE_TOTAL * 0.1
WHERE     VENDOR_ID = VENDOR_ID_P;
```

```
UPDATE    INVOICES5
SET       BALANCE = INVOICE_TOTAL - (PAYMENT_TOTAL + CREDIT_TOTAL)
WHERE     VENDOR_ID = VENDOR_ID_P;
END;
/
EXECUTE UPDATEPROC;
```

OR

```
UPDATE INVOICES5
SET CREDIT_TOTAL = (INVOICE_TOTAL*10)/100
WHERE VENDOR_ID = (SELECT VENDOR_ID FROM VENDORS5 WHERE VENDOR_NAME = 'Advait Patel, LLC');
```

UPDATE INVOICES5

SET BALANCE = (INVOICE_TOTAL - PAYMENT_TOTAL - CREDIT_TOTAL)

WHERE VENDOR_ID = (SELECT VENDOR_ID FROM VENDORS5 WHERE VENDOR_NAME = 'Advait Patel, LLC');

PART - 3

(A)

SET LINESIZE 1000

COLUMN INVOICE_TOTAL FORMAT \$999,999,999.99

COLUMN CREDIT_TOTAL FORMAT \$999,999,999.99

COLUMN PAYMENT_TOTAL FORMAT \$999,999,999.99

COLUMN BALANCE FORMAT \$999,999,999.99

COLUMN VENDOR_NAME FORMAT A20

```
SELECT INVOICES5.VENDOR_ID, VENDORS5.VENDOR_NAME, INVOICE_ID,  
REPLACE(TO_CHAR(INVOICE_DATE, 'Month dd, yyyy'), ' ', '') AS INVOICE_DATE ,  
REPLACE(TO_CHAR(INVOICE_DUE_DATE, 'Month dd, yyyy'), ' ', '') AS INVOICE_DUE_DATE , INVOICE_TOTAL, CREDIT_TOTAL, BALANCE  
FROM INVOICES5  
  
JOIN VENDORS5 ON VENDORS5.VENDOR_ID = INVOICES5.VENDOR_ID  
  
WHERE INVOICES5.VENDOR_ID = (SELECT MAX(VENDOR_ID) FROM VENDORS5);
```

(B)

----- QUESTION : 3 : PART 2 : ENSURING THE DATABASE IS RESTORED IN ITS INITIAL STATE. -----

-- AFTER USING DROP TABLE, THE DATABASE IS NOW IN ITS ORIGINAL STATE.

```
DROP TABLE INVOICES5;
```

```
DROP TABLE VENDORS5;
```

OUTPUTS

PART – 1

(A)

Table VENDORS5 created.

Table INVOICES5 created.

(B)

Table VENDORS5 altered.

Table VENDORS5 altered.

Table VENDORS5 altered.

Table VENDORS5 altered.

Table INVOICES5 altered.

Table INVOICES5 altered.

Table INVOICES5 altered.

Table INVOICES5 altered.

(C)

Table INVOICES5 altered.

(D)

114 rows updated.

OR

114 rows updated.

PART – 2

(A)

PL/SQL procedure successfully completed.

OR

Procedure MYPROC compiled

PL/SQL procedure successfully completed.

(B)

Procedure UPDATEPROC compiled

PL/SQL procedure successfully completed.

OR

3 rows updated.

3 rows updated.

PART – 3

(A)

VENDOR_ID	VENDOR_NAME	INVOICE_ID	INVOICE_DATE	INVOICE_DUE_DATE	INVOICE_TOTAL	CREDIT_TOTAL	BALANCE
-----	-----	-----	-----	-----	-----	-----	-----
126	Advait, LLC	121	October26,2015	November25,2015	\$1,178.63	\$117.86	\$1,060.77
126	Advait, LLC	122	October26,2015	November25,2015	\$8,178.63	\$817.86	\$7,360.77
126	Advait, LLC	123	October26,2015	November25,2015	\$15,178.63	\$1,517.86	\$13,660.77

(B)

Table INVOICES5 dropped.

Table VENDORS5 dropped.
