PART A

```
CREATE TABLE productdim (
  productid     int identity not null,
  prod_sku      VARCHAR(15)  NOT NULL,
  prod_descript VARCHAR(255) NULL,
  prod_type     VARCHAR(255) NULL,
  prod_base     VARCHAR(255) NULL,
  prod_category VARCHAR(255) NULL,
  brand_id      NUMERIC(4,0)   NULL,
  brand_name    VARCHAR(100) NULL
);



CREATE TABLE employeedim (
  employeeid    int identity not null,
  emp_num       NUMERIC(6,0)  NOT NULL,
  emp_fname     VARCHAR(20) NULL,
  emp_lname     VARCHAR(25) NOT NULL,
  emp_email     VARCHAR(25) NOT NULL,
  emp_phone     VARCHAR(20) NULL,
  emp_hiredate  DATE         NOT NULL,
  emp_title     VARCHAR(45) NOT NULL,
  dept_num      NUMERIC(5,0)  NULL,
  dept_name     VARCHAR(50) NOT NULL
);


CREATE TABLE customerdim (
  customerid    int identity not null,
  cust_code     NUMERIC(38,0) NOT NULL,
  cust_fname    VARCHAR(20) NOT NULL,
  cust_lname    VARCHAR(20) NOT NULL,
  cust_street   VARCHAR(70) NULL,
  cust_city     VARCHAR(50) NULL,
  cust_state    CHAR(2)      NULL,
  cust_zip      CHAR(5)      NULL,

);

CREATE TABLE timedim (
  timeid        int identity not null,
  inv_date      DATE         NULL,
  month_value   int,
  quarter_value int,
  year_value    int

)


CREATE TABLE FACT
```

```
(
    TIMEID      INT NOT NULL,
    CUSTOMERID  INT NOT NULL,
    PRODUCTID   INT NOT NULL,
    EMPLOYEEID  INT NOT NULL,
    LINE_QTY    NUMERIC NOT NULL,
    LINE_PRICE  NUMERIC(8,2)
)


CREATE TABLE STAGE
(
    TIMEID      INT NULL,
    CUSTOMERID  INT NULL,
    PRODUCTID   INT NULL,
    EMPLOYEEID  INT NULL,
    INV_DATE    DATE NOT NULL,
    CUST_CODE   NUMERIC(38,0) NOT NULL,
    EMP_NUM     NUMERIC(6,0) NOT NULL,
    PROD_SKU    VARCHAR(15) NOT NULL,
    LINE_QTY    NUMERIC NOT NULL,
    LINE_PRICE  NUMERIC(8,2)
)

ALTER TABLE TIMEDIM
ADD CONSTRAINT PK_TIMEDIM PRIMARY KEY(TIMEID)

ALTER TABLE CUSTOMERDIM
ADD CONSTRAINT PK_CUSTOMERDIM PRIMARY KEY(CUSTOMERID)

ALTER TABLE PRODUCTDIM
ADD CONSTRAINT PK_PRODUCTDIM PRIMARY KEY(PRODUCTID)

ALTER TABLE EMPLOYEEDIM
ADD CONSTRAINT PK_EMPLOYEEDIM PRIMARY KEY(EMPLOYEEID)

ALTER TABLE FACT
ADD  CONSTRAINT PK_FACT PRIMARY KEY(TIMEID, CUSTOMERID, PRODUCTID, EMPLOYEEID),
     CONSTRAINT FK_FACT_TIMEDIM FOREIGN KEY(TIMEID) REFERENCES TIMEDIM,
     CONSTRAINT FK_FACT_PRODUCTDIM FOREIGN KEY(PRODUCTID) REFERENCES PRODUCTDIM,
     CONSTRAINT FK_FACT_EMPLOYEEDIM FOREIGN KEY (EMPLOYEEID) REFERENCES EMPLOYEEDIM,
     CONSTRAINT FK_FACT_CUSTOMERDIM FOREIGN KEY (CUSTOMERID) REFERENCES CUSTOMERDIM
```

```
PART B

ALTER PROCEDURE [dbo].[TEN]
AS
BEGIN
    ALTER TABLE FACT
    DROP CONSTRAINT PK_FACT,
         CONSTRAINT FK_FACT_TIMEDIM,
         CONSTRAINT FK_FACT_PRODUCTDIM,
         CONSTRAINT FK_FACT_EMPLOYEEDIM,
         CONSTRAINT FK_FACT_CUSTOMERDIM

    ALTER TABLE TIMEDIM
    DROP    CONSTRAINT PK_TIMEDIM

    ALTER TABLE CUSTOMERDIM
    DROP CONSTRAINT PK_CUSTOMERDIM

    ALTER TABLE PRODUCTDIM
    DROP CONSTRAINT PK_PRODUCTDIM

    ALTER TABLE EMPLOYEEDIM
    DROP CONSTRAINT PK_EMPLOYEEDIM


    TRUNCATE TABLE FACT
    TRUNCATE TABLE CUSTOMERDIM
    TRUNCATE TABLE EMPLOYEEDIM
    TRUNCATE TABLE TIMEDIM
    TRUNCATE TABLE PRODUCTDIM

    ALTER TABLE TIMEDIM
    ADD CONSTRAINT PK_TIMEDIM PRIMARY KEY(TIMEID)

    ALTER TABLE CUSTOMERDIM
    ADD CONSTRAINT PK_CUSTOMERDIM PRIMARY KEY(CUSTOMERID)

    ALTER TABLE PRODUCTDIM
    ADD CONSTRAINT PK_PRODUCTDIM PRIMARY KEY(PRODUCTID)

    ALTER TABLE EMPLOYEEDIM
    ADD CONSTRAINT PK_EMPLOYEEDIM PRIMARY KEY(EMPLOYEEID)

    ALTER TABLE FACT
    ADD  CONSTRAINT PK_FACT PRIMARY KEY(TIMEID, CUSTOMERID, PRODUCTID, EMPLOYEEID),
         CONSTRAINT FK_FACT_TIMEDIM FOREIGN KEY(TIMEID) REFERENCES TIMEDIM,
         CONSTRAINT FK_FACT_PRODUCTDIM FOREIGN KEY(PRODUCTID) REFERENCES PRODUCTDIM,
         CONSTRAINT FK_FACT_EMPLOYEEDIM FOREIGN KEY (EMPLOYEEID) REFERENCES
           EMPLOYEEDIM,
         CONSTRAINT FK_FACT_CUSTOMERDIM FOREIGN KEY (CUSTOMERID) REFERENCES
           CUSTOMERDIM
```

```
    INSERT INTO TIMEDIM
    SELECT  DISTINCT INV_DATE, MONTH(INV_DATE), YEAR(INV_DATE), DATEPART(QUARTER,
      INV_DATE)
    FROM    LGINVOICE

    INSERT INTO CUSTOMERDIM
    SELECT  CUST_CODE, CUST_FNAME, CUST_LNAME, CUST_STREET, CUST_CITY, CUST_STATE,
      CUST_ZIP
    FROM    LGCUSTOMER

    INSERT INTO EMPLOYEEDIM
    SELECT  E.EMP_NUM, E.EMP_FNAME, E.EMP_LNAME, E.EMP_EMAIL, E.EMP_PHONE,
      E.EMP_HIREDATE, E.EMP_TITLE, E.DEPT_NUM, D.DEPT_NAME
    FROM    LGEMPLOYEE E INNER JOIN LGDEPARTMENT D ON E.DEPT_NUM = D.DEPT_NUM

    INSERT INTO PRODUCTDIM
    SELECT  P.PROD_SKU, P.PROD_DESCRIPT, P.PROD_TYPE, P.PROD_BASE, P.PROD_CATEGORY,
      P.BRAND_ID, B.BRAND_NAME
    FROM    LGPRODUCT P INNER JOIN LGBRAND B ON P.BRAND_ID = B.BRAND_ID

    INSERT INTO STAGE (INV_DATE, CUST_CODE, EMP_NUM, PROD_SKU, LINE_QTY, LINE_PRICE)
    SELECT  I.INV_DATE, I.CUST_CODE, I.EMPLOYEE_ID, L.PROD_SKU, SUM(L.LINE_QTY), AVG
      (L.LINE_PRICE)
    FROM    LGINVOICE I INNER JOIN LGLINE L ON I.INV_NUM = L.INV_NUM
    GROUP BY I.INV_DATE, I.CUST_CODE, I.EMPLOYEE_ID, L.PROD_SKU
END
```

```
PART C

--What are the top 5 products in terms of sales (total quantity * price)?
SELECT  PROD_DESCRIPT
FROM    PRODUCTDIM
WHERE   PRODUCTID IN
        (SELECT TOP 5 PRODUCTID
         FROM   FACT
         GROUP BY PRODUCTID
         ORDER BY SUM(LINE_QTY*LINE_PRICE) DESC
        )

--List the names of employees who have sold the most products in terms of amount of
  sales (total of quantity * price).
SELECT  E.EMP_LNAME
FROM    EMPLOYEEDIM E INNER JOIN FACT F ON E.EMPLOYEEID = F.EMPLOYEEID
GROUP BY E.EMPLOYEEID, E.EMP_LNAME
HAVING  SUM(F.LINE_QTY*F.LINE_PRICE) =
        (SELECT TOP 1 SUM(F.LINE_QTY*F.LINE_PRICE)
         FROM   EMPLOYEEDIM E INNER JOIN FACT F ON E.EMPLOYEEID = F.EMPLOYEEID
         GROUP BY E.EMPLOYEEID, E.EMP_LNAME
         ORDER BY SUM(F.LINE_QTY*F.LINE_PRICE) DESC
        )

--List the total amount of sales by customer city and brand name.
SELECT  C.CUST_CITY, P.BRAND_NAME, SUM(F.LINE_QTY*F.LINE_PRICE) AS TOTAL
FROM    CUSTOMERDIM C INNER JOIN FACT F ON C.CUSTOMERID = F.CUSTOMERID
        INNER JOIN PRODUCTDIM P ON F.PRODUCTID = P.PRODUCTID
GROUP BY C.CUSTOMERID, P.PRODUCTID, C.CUST_CITY, P.BRAND_NAME
ORDER BY SUM(F.LINE_QTY*F.LINE_PRICE) DESC

--List the customer names of customers and the top 5 products each of these customers
  have bought.
DECLARE CUSTOMER_CURSOR CURSOR FOR
        SELECT  CUSTOMERID, CUST_LNAME
        FROM    CUSTOMERDIM
        ORDER BY CUST_LNAME
DECLARE @CUSTOMERID INT
DECLARE @CUST_LNAME VARCHAR(20)

OPEN CUSTOMER_CURSOR
FETCH NEXT FROM CUSTOMER_CURSOR INTO @CUSTOMERID, @CUST_LNAME
WHILE(@@FETCH_STATUS = 0)
BEGIN
    SELECT 'CUSTOMER NAME = ' + @CUST_LNAME
    SELECT  TOP 5 '         ' + P.PROD_DESCRIPT AS [TOP PRODUCT]
    FROM    PRODUCTDIM P INNER JOIN FACT F ON P.PRODUCTID = F.PRODUCTID
    WHERE   F.CUSTOMERID = @CUSTOMERID
    GROUP BY P.PRODUCTID, P.PROD_DESCRIPT
    ORDER BY    SUM(F.LINE_QTY*F.LINE_PRICE) DESC
    FETCH NEXT FROM CUSTOMER_CURSOR INTO @CUSTOMERID, @CUST_LNAME
END
```

```
CLOSE CUSTOMER_CURSOR
DEALLOCATE CUSTOMER_CURSOR
```