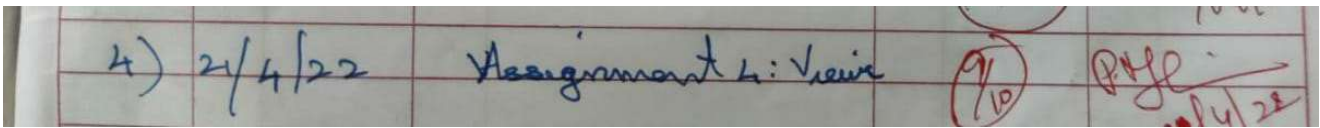**UCS1412 – DBMS Lab**
Assignment – 4

Name: Jayannthan P T   *Dept: CSE 'A'*   *Roll No.: 205001049*

Title: Views

Observation Correction:

Schema Diagram:

Blue_Flavor		
Operation	Operation on View	Reflection in Base Table
Insert	✘	✔
Update	PID, FOOD, PRICE	PID, FOOD, PRICE
Modify	✔	✔

Cheap_Food		
Operation	Operation on View	Reflection in Base Table
Insert	✔	✔
Update	PID, FLAVOR, FOOD, PRICE	PID, FLAVOR, FOOD, PRICE
Modify	✔	✔

Hot_Food		
Operation	Operation on View	Reflection in Base Table

Operation	Operation on View	Reflection in Base Table
Insert	✖	✖
Update	✖	✖
Modify	✖	✖

Pie_Food		
Operation	Operation on View	Reflection in Base Table
Insert	✖ (non-key preserved table)	✖
Update	✓	✓ (On ITEM_LIST)
Modify	✓	✓

Cheap_View		
Operation	Operation on View	Reflection in Base Table
Insert	✖ (cannot insert without price)	✖
Update	PID, FLAVOR, FOOD	PID, FLAVOR, FOOD
Modify	✓	✓

Spool File:

```
SQL> @Z:assn4_views.sql
SQL> -- Consider the schema used in the Assignment3.
SQL>
SQL> -- Create the following views and perform DML operations on it. Classify whether the
view is updatable or not.
SQL>
SQL> -- 1. Create a view named Blue_Flavor, which display the product details (product id,
SQL> -- food, price) of Blueberry flavor.
SQL>
SQL> DROP VIEW Blue_Flavor;

View dropped.

SQL> -- Creating view
SQL> CREATE VIEW Blue_Flavor
  2  AS
  3     SELECT PID, FOOD, PRICE
```

```
     4    FROM PRODUCTS
     5    WHERE FLAVOR='Blueberry';

View created.

SQL>
SQL> -- DESC Blue_Flavor;
SQL>
SQL> --displaying items in Blue_Flavor
SQL> SELECT *
   2  FROM Blue_Flavor;

PID              FOOD                      PRICE
--------------   --------------------   ----------
90-BLU-11        Tart                        3.25
51-BLU           Danish                      1.15

SQL>
SQL> --viewing the updatable columns in Blue_Flavor
SQL> SELECT COLUMN_NAME, UPDATABLE
   2  FROM USER_UPDATABLE_COLUMNS
   3  WHERE
   4  TABLE_NAME='BLUE_FLAVOR';

COLUMN_NAME                     UPD
-----------------------------   ---
PID                             YES
FOOD                            YES
PRICE                           YES

SQL>
SQL> --  CREATING A SAVEPOINT
SQL> SAVEPOINT VIEW1;

Savepoint created.

SQL>
SQL> --inserting a tuple in products with flavour="Blueberry"
SQL> INSERT INTO PRODUCTS
   2  VALUES
   3    ('20-Ch', 'Blueberry', 'Chocolate', 15);

1 row created.

SQL>
SQL>
SQL> --displaying items in Blue_Flavor after adding tuple in products
SQL> SELECT *
   2  FROM Blue_Flavor;

PID              FOOD                      PRICE
--------------   --------------------   ----------
90-BLU-11        Tart                        3.25
51-BLU           Danish                      1.15
```

```
20-Ch          Chocolate                    15

SQL>
SQL> --updating a tuple in Blue_Flavor
SQL> UPDATE Blue_Flavor
  2  SET PRICE=PRICE*1.15
  3  WHERE PID='20-Ch';

1 row updated.

SQL>
SQL> --displaying items in Blue_Flavor after updating
SQL> SELECT *
  2  FROM Blue_Flavor
  3  WHERE PID ='20-Ch';

PID             FOOD                  PRICE
--------------- --------------------- ----------
20-Ch           Chocolate                 17.25

SQL>
SQL> --displaying items in base table after updating
SQL> SELECT *
  2  FROM PRODUCTS
  3  WHERE PID ='20-Ch';

PID             FLAVOR              FOOD                 PRICE
--------------- ------------------- -------------------- ----------
20-Ch           Blueberry           Chocolate                17.25

SQL>
SQL>
SQL> --deleting a tuple in Blue_Flavor
SQL> DELETE FROM Blue_Flavor WHERE PID='20-Ch';

1 row deleted.

SQL>
SQL> --displaying items in Blue_Flavor after deleting
SQL> SELECT *
  2  FROM Blue_Flavor
  3  WHERE PID ='20-Ch';

no rows selected

SQL>
SQL> --displaying items in base table after deleting
SQL> SELECT *
  2  FROM PRODUCTS
  3  WHERE PID ='20-Ch';

no rows selected

SQL>
```

```
SQL> --ROLLINGBACK TO SAVEPOINT
SQL> ROLLBACK
  2  TO VIEW1;

Rollback complete.

SQL>
SQL> --SAVING PROGRESS
SQL> COMMIT;

Commit complete.

SQL>
SQL> -- 2. Create a view named Cheap_Food, which display the details (product id, flavor,
SQL> -- food, price) of products with price lesser than $1. Ensure that, the price of
these
SQL> -- food(s) should never rise above $1 through view.
SQL>
SQL> DROP VIEW Cheap_Food;

View dropped.

SQL> -- Creating view
SQL> CREATE VIEW Cheap_Food
  2  AS
  3    SELECT PID, FLAVOR, FOOD, PRICE
  4    FROM PRODUCTS
  5    WHERE PRICE < 1
  6  WITH CHECK OPTION;

View created.

SQL>
SQL> -- DESC Cheap_Food;
SQL>
SQL> -- Displaying the contents of the Cheap_Food
SQL> SELECT *
  2  FROM Cheap_Food;

PID             FLAVOR               FOOD                 PRICE
--------------- -------------------- -------------------- ----------
70-LEM          Lemon                Cookie                     .79
70-W            Walnut               Cookie                     .79


SQL>
SQL> --viewing the updatable columns in Cheap_Food
SQL> SELECT COLUMN_NAME, UPDATABLE
  2  FROM USER_UPDATABLE_COLUMNS
  3  WHERE
  4  TABLE_NAME='CHEAP_FOOD';

COLUMN_NAME                    UPD
------------------------------ ---
PID                            YES
```

```
FLAVOR                          YES
FOOD                            YES
PRICE                           YES

SQL>
SQL> --  CREATING A SAVEPOINT
SQL> SAVEPOINT VIEW2;

Savepoint created.

SQL>
SQL>
SQL> --inserting a tuple in Cheap_Food
SQL>
SQL> insert INTO Cheap_Food
  2  values
  3    ('20-Ch1', 'Mango', 'Chocolate', 0.75);

1 row created.

SQL>
SQL>
SQL> --inserting a tuple in Cheap_Food with price>1
SQL>
SQL> insert INTO Cheap_Food
  2  values
  3    ('20-Ch2', 'Man', 'Choco', 1.75);
insert INTO Cheap_Food
            *
ERROR at line 1:
ORA-01402: view WITH CHECK OPTION where-clause violation


SQL>
SQL>
SQL> --displaying items in Cheap_Food after insertion
SQL> SELECT *
  2  FROM Cheap_Food
  3  WHERE PID='20-Ch1';

PID             FLAVOR               FOOD                 PRICE
--------------  -------------------- -------------------- ----------
20-Ch1          Mango                Chocolate                  .75

SQL>
SQL> --displaying items in base table after insertion
SQL> SELECT *
  2  FROM PRODUCTS
  3  WHERE PID='20-Ch1';

PID             FLAVOR               FOOD                 PRICE
--------------  -------------------- -------------------- ----------
20-Ch1          Mango                Chocolate                  .75
```

```
SQL>
SQL> --updating a tuple in Cheap_Food
SQL> UPDATE Cheap_Food
  2  SET PRICE=PRICE+0.1
  3  WHERE PID='20-Ch1';

1 row updated.

SQL>
SQL> --displaying items in Cheap_Food after updating
SQL> SELECT *
  2  FROM Cheap_Food
  3  WHERE PID ='20-Ch1';

PID             FLAVOR               FOOD                      PRICE
--------------- -------------------- -------------------- ----------
20-Ch1          Mango                Chocolate                   .85

SQL>
SQL> --displaying items in base table after updating
SQL> SELECT *
  2  FROM PRODUCTS
  3  WHERE PID='20-Ch1';

PID             FLAVOR               FOOD                      PRICE
--------------- -------------------- -------------------- ----------
20-Ch1          Mango                Chocolate                   .85

SQL>
SQL>
SQL> --deleting a tuple in Cheap_Food
SQL> DELETE FROM Cheap_Food WHERE PID='20-Ch1';

1 row deleted.

SQL>
SQL> --displaying items in Cheap_Food after deleting
SQL> SELECT *
  2  FROM Cheap_Food
  3  WHERE PID ='20-Ch1';

no rows selected

SQL>
SQL>
SQL> --displaying items in base table after deleting
SQL> SELECT *
  2  FROM PRODUCTS
  3  WHERE PID='20-Ch1';

no rows selected

SQL>
SQL>
```

```
SQL>
SQL>
SQL> --ROLLINGBACK TO SAVEPOINT
SQL> ROLLBACK
  2  TO VIEW2;

Rollback complete.

SQL>
SQL> --SAVING PROGRESS
SQL> COMMIT;

Commit complete.

SQL>
SQL> -- 3. Create a view called Hot_Food that show the product id and its quantity WHERE
the
SQL> -- same product is ordered more than once in the same receipt.
SQL>
SQL> DROP VIEW Hot_Food;

View dropped.

SQL>
SQL> -- Creating view
SQL> CREATE VIEW Hot_Food
  2  (
  3     pid,
  4     count
  5  )
  6  AS
  7     SELECT ITEM, COUNT(ITEM)
  8     FROM ITEM_LIST
  9     GROUP BY RNO,ITEM
 10     HAVING COUNT(*)>1;

View created.

SQL>
SQL>
SQL> -- DESC Hot_Food;
SQL>
SQL> --displaying items in Hot_Food
SQL> SELECT *
  2  FROM Hot_Food;

PID                     COUNT
--------------- ----------
70-R                        2
90-APR-PF                   2
50-APP                      2
51-ATW                      2
90-ALM-I                    2
90-BER-11                   2
```

```
90-PEC-11                 2
70-M-CH-DZ                2
46-11                     2
70-M-CH-DZ                2
90-CHR-11                 2


PID                     COUNT
--------------- ----------
90-BLU-11                 2
50-CHS                    2
70-M-CH-DZ                2
70-R                      2
90-APP-11                 2
70-MAR                    2
50-APR                    2
51-BC                     2
50-ALM                    2


20 rows selected.

SQL>
SQL> --viewing the updatable columns in Hot_Food
SQL> SELECT COLUMN_NAME, UPDATABLE
  2  FROM USER_UPDATABLE_COLUMNS
  3  WHERE
  4  TABLE_NAME='HOT_FOOD';

COLUMN_NAME                    UPD
------------------------------ ---
PID                            NO
COUNT                          NO

SQL> -- This table isnt updatable because we have used aggregate functions (group by
class)
SQL>
SQL>
SQL> --  CREATING A SAVEPOINT
SQL> SAVEPOINT VIEW3;

Savepoint created.

SQL>
SQL> --inserting a tuple in Cheap_Food
SQL> INSERT INTO Hot_Food
  2  VALUES('50-AB', 3);
INSERT INTO Hot_Food
*
ERROR at line 1:
ORA-01733: virtual column not allowed here



SQL>
SQL>
SQL> -- updating item in Hot_Food
```

```
SQL> UPDATE Hot_Food
  2   SET ITEM='54-SD9-34JS'
  3   WHERE ITEM='46-11';
WHERE ITEM='46-11'
      *
ERROR at line 3:
ORA-00904: "ITEM": invalid identifier


SQL>
SQL> --updating quantity in Hot_Food
SQL> UPDATE Hot_Food
  2   SET QUANTITY=4
  3   WHERE ITEM='46-11';
WHERE ITEM='46-11'
      *
ERROR at line 3:
ORA-00904: "ITEM": invalid identifier


SQL>
SQL> --deleting a row in Hot_Food
SQL> DELETE FROM Hot_Food WHERE ITEM='46-11';
DELETE FROM Hot_Food WHERE ITEM='46-11'
                              *
ERROR at line 1:
ORA-00904: "ITEM": invalid identifier


SQL>
SQL>
SQL> -- making changes in base table to see reflection
SQL> INSERT INTO PRODUCTS
  2   VALUES('9-B', 'ORANGE', 'JUICE', 9);

1 row created.

SQL> INSERT INTO RECEIPTS
  2   VALUES(12345, '9-JUN-22', 21);

1 row created.

SQL> INSERT INTO ITEM_LIST
  2   VALUES(12345, 98, '9-B');

1 row created.

SQL> INSERT INTO ITEM_LIST
  2   VALUES(12345, 99, '9-B');

1 row created.

SQL>
SQL> SELECT *
```

```
  2  from Hot_food;

PID                   COUNT
-------------- ----------
70-R                     2
90-APR-PF                2
50-APP                   2
51-ATW                   2
90-ALM-I                 2
90-BER-11                2
90-PEC-11                2
70-M-CH-DZ               2
46-11                    2
70-M-CH-DZ               2
90-CHR-11                2

PID                   COUNT
-------------- ----------
90-BLU-11                2
9-B                      2
50-CHS                   2
70-M-CH-DZ               2
70-R                     2
90-APP-11                2
70-MAR                   2
50-APR                   2
51-BC                    2
50-ALM                   2

21 rows selected.

SQL> -- note the changes in counts
SQL> -- making changes in the base table reflects in the view but not vice versa
SQL>
SQL> --ROLLINGBACK TO SAVEPOINT
SQL> ROLLBACK
  2  TO VIEW3;

Rollback complete.

SQL>
SQL> --SAVING PROGRESS
SQL> COMMIT;

Commit complete.

SQL>
SQL>
SQL> -- 4. Create a view named Pie_Food that will display the details (customer lname, flavor,
SQL> -- receipt number and date, ordinal) who had ordered the Pie food with receipt details.
SQL>
SQL> DROP VIEW Pie_Food;
```

```
View dropped.

SQL>
SQL> -- Creating view
SQL> CREATE VIEW Pie_Food
  2  AS
  3    SELECT LNAME, FLAVOR, I.RNO, RDATE, ORDINAL
  4    FROM CUSTOMERS C, RECEIPTS R, ITEM_LIST I, PRODUCTS P
  5    WHERE C.CID=R.CID AND R.RNO=I.RNO AND I.ITEM=P.PID AND FOOD='Pie';

View created.

SQL>
SQL> --displaying items in Pie_Food
SQL> SELECT *
  2  FROM Pie_Food;

LNAME                          FLAVOR                      RNO RDATE
------------------------------ -------------------- ---------- ---------
   ORDINAL
----------
JULIET                         Apple                     39685 28-OCT-07
        4

JULIET                         Apple                     66227 10-OCT-07
        2

TRAVIS                         Apple                     48647 09-OCT-07
        2


LNAME                          FLAVOR                      RNO RDATE
------------------------------ -------------------- ---------- ---------
   ORDINAL
----------
JOSETTE                        Apple                     87454 21-OCT-07
        1

JOSETTE                        Apple                     47353 12-OCT-07
        2

RUPERT                         Apple                     53376 30-OCT-07
        3


LNAME                          FLAVOR                      RNO RDATE
------------------------------ -------------------- ---------- ---------
   ORDINAL
----------
CUC                            Apple                     50660 18-OCT-07
        2

KIP                            Apple                     11548 21-OCT-07
```

```
            2

RAYFORD                        Apple                        29226 26-OCT-07
            2


LNAME                          FLAVOR                        RNO RDATE
------------------------------ -------------------- ---------- ---------
    ORDINAL
----------
RAYFORD                        Apple                        51991 17-OCT-07
            1

ARIANE                         Apple                        39109 02-OCT-07
            1

ARIANE                         Apple                        44798 04-OCT-07
            3


LNAME                          FLAVOR                        RNO RDATE
------------------------------ -------------------- ---------- ---------
    ORDINAL
----------
CHARLENE                       Apple                        98806 15-OCT-07
            3


13 rows selected.

SQL>
SQL> --viewing the updatable columns in Pie_Food
SQL> SELECT COLUMN_NAME, UPDATABLE
  2  FROM USER_UPDATABLE_COLUMNS
  3  WHERE
  4  TABLE_NAME='PIE_FOOD';

COLUMN_NAME                    UPD
------------------------------ ---
LNAME                          NO
FLAVOR                         NO
RNO                            YES
RDATE                          NO
ORDINAL                        YES


SQL>
SQL> -- THIS VIEW ISNT UPDATABLE BECAUSE WE HAVE JOINED MULTIPLE TABLE AND THE NOT NULL
COLUMNS FROM PRODUCTS , CUSTOMERS (PK) ARENT A PART OF THE VIEW
SQL>
SQL> --  CREATING A SAVEPOINT
SQL> SAVEPOINT VIEW4;

Savepoint created.
```

```
SQL>
SQL>
SQL> --inserting a tuple in Pie_Food
SQL> INSERT INTO Pie_Food
  2  VALUES('Tony', 'Blueberry', 51991, '9-Jun-21', 98);
INSERT INTO Pie_Food
*
ERROR at line 1:
ORA-01779: cannot modify a column which maps to a non key-preserved table


SQL>
SQL>
SQL> --displaying items in Pie_Food after inserting
SQL> SELECT *
  2  FROM Pie_Food
  3  WHERE LNAME='Steve';

no rows selected

SQL>
SQL>
SQL> --displaying items in base table after inserting
SQL> SELECT LNAME, FLAVOR, R.RNO, RDATE, ORDINAL
  2  FROM CUSTOMERS C, RECEIPTS R, ITEM_LIST I, PRODUCTS P
  3  WHERE C.CID=R.CID AND R.RNO=I.RNO AND I.ITEM=P.PID AND FOOD='Pie' AND LNAME='Tony';

no rows selected

SQL>
SQL>
SQL> --updating a tuple in Pie_Food
SQL> UPDATE Pie_Food SET LNAME='Steve' WHERE RNO=51991;
UPDATE Pie_Food SET LNAME='Steve' WHERE RNO=51991
                 *
ERROR at line 1:
ORA-01779: cannot modify a column which maps to a non key-preserved table


SQL>
SQL> -- updating RNO
SQL> UPDATE Pie_Food SET RNO=51991 WHERE LNAME='Steve';
UPDATE Pie_Food SET RNO=51991 WHERE LNAME='Steve'
                 *
ERROR at line 1:
ORA-01779: cannot modify a column which maps to a non key-preserved table


SQL>
SQL> -- updating ORDINAL
SQL> UPDATE Pie_Food SET ORDINAL=99 WHERE LNAME='Steve';

0 rows updated.
```

```
SQL>
SQL> -- updating FLAVOUR
SQL> UPDATE Pie_Food SET FLAVOR='NEW_FLAVOUR' WHERE LNAME='Steve';
UPDATE Pie_Food SET FLAVOR='NEW_FLAVOUR' WHERE LNAME='Steve'
                   *
ERROR at line 1:
ORA-01779: cannot modify a column which maps to a non key-preserved table


SQL>
SQL>
SQL> --displaying items in Pie_Food after updating
SQL>
SQL> SELECT *
  2  from Pie_Food
  3  WHERE LNAME='Steve';

no rows selected


SQL>
SQL> --displaying items in base table after updating
SQL> SELECT LNAME, FLAVOR, R.RNO, RDATE, ORDINAL
  2  FROM CUSTOMERS C, RECEIPTS R, ITEM_LIST I, PRODUCTS P
  3  WHERE C.CID=R.CID AND R.RNO=I.RNO AND I.ITEM=P.PID AND FOOD='Pie' AND LNAME='Steve';

no rows selected


SQL>
SQL> -- deleting a tuple in view
SQL> SELECT *
  2  FROM Pie_Food
  3  WHERE LNAME='Steve';

no rows selected


SQL>
SQL> --displaying items in Pie_Food after deleting
SQL>
SQL> SELECT *
  2  from Pie_Food
  3  WHERE LNAME='Steve';

no rows selected


SQL>
SQL> --displaying items in base table after deleting
SQL> SELECT LNAME, FLAVOR, R.RNO, RDATE, ORDINAL
  2  FROM CUSTOMERS C, RECEIPTS R, ITEM_LIST I, PRODUCTS P
  3  WHERE C.CID=R.CID AND R.RNO=I.RNO AND I.ITEM=P.PID AND FOOD='Pie' AND LNAME='Steve';

no rows selected


SQL>
SQL>
```

```
SQL> --ROLLINGBACK TO SAVEPOINT
SQL> ROLLBACK
  2  TO VIEW4;

Rollback complete.

SQL>
SQL> --SAVING PROGRESS
SQL> COMMIT;

Commit complete.

SQL>
SQL> -- 5. Create a view Cheap_View from Cheap_Food that shows only the product id, flavor
SQL> -- and food.
SQL>
SQL> -- Creating view
SQL> CREATE VIEW Cheap_View
  2  AS
  3    SELECT PID, FLAVOR, FOOD
  4    FROM CHEAP_FOOD;

View created.

SQL>
SQL> --displaying items in Cheap_View
SQL> SELECT *
  2  FROM Cheap_View;

PID             FLAVOR               FOOD
--------------- -------------------- --------------------
70-LEM          Lemon                Cookie
70-W            Walnut               Cookie

SQL>
SQL> --viewing the updatable columns in Cheap_View
SQL> SELECT COLUMN_NAME, UPDATABLE
  2  FROM USER_UPDATABLE_COLUMNS
  3  WHERE
  4  TABLE_NAME='CHEAP_VIEW';

COLUMN_NAME                    UPD
------------------------------ ---
PID                            YES
FLAVOR                         YES
FOOD                           YES

SQL>
SQL>
SQL> --  CREATING A SAVEPOINT
SQL> SAVEPOINT NEW_VIEW;

Savepoint created.
```

```
SQL>
SQL> --inserting a tuple in Cheap_View
SQL> insert INTO Cheap_food
  2  VALUES('20-Ch', 'Mango', 'Chocolate');
insert INTO Cheap_food
            *
ERROR at line 1:
ORA-00947: not enough values


SQL>
SQL> --displaying items in Cheap_View after inserting
SQL> SELECT *
  2  from Cheap_View
  3  WHERE PID='20-Ch';

no rows selected


SQL>
SQL> --displaying items in Cheap_Food after inserting
SQL> SELECT *
  2  from Cheap_Food
  3  WHERE PID='20-Ch';

no rows selected


SQL>
SQL> --updating a tuple in Pie_Food
SQL>
SQL> UPDATE Cheap_View SET PID='20-Ch1' WHERE PID='20-Ch';

0 rows updated.

SQL> UPDATE Cheap_View SET FLAVOR='NEW_FLAVOUR' WHERE PID='20-Ch1';

0 rows updated.

SQL> UPDATE Cheap_View SET FOOD='New_Food' WHERE PID='20-Ch1';

0 rows updated.

SQL>
SQL> --displaying items in Cheap_View after updating
SQL> SELECT *
  2  from Cheap_View
  3  WHERE PID='20-Ch1';

no rows selected


SQL>
SQL> --displaying items in Cheap_Food after updating
SQL> SELECT *
  2  from Cheap_Food
  3  WHERE PID='20-Ch1';
```

```
no rows selected

SQL>
SQL>
SQL> DELETE FROM Cheap_View WHERE PID='20-Ch1';

0 rows deleted.

SQL> DELETE FROM Cheap_View WHERE FOOD='New_Food';

0 rows deleted.

SQL>
SQL>
SQL> --displaying items in Cheap_View after deleting
SQL> SELECT *
  2  from Cheap_View
  3  WHERE PID='20-Ch1';

no rows selected

SQL>
SQL> --displaying items in Cheap_Food after deleting
SQL> SELECT *
  2  from Cheap_Food
  3  WHERE PID='20-Ch1';

no rows selected

SQL>
SQL>
SQL> --ROLLINGBACK TO SAVEPOINT
SQL> ROLLBACK
  2  TO NEW_VIEW;

Rollback complete.

SQL>
SQL> --SAVING PROGRESS
SQL> COMMIT;

Commit complete.

SQL>
SQL>
SQL>
SQL> -- 6. Drop the view Cheap_View
SQL>
SQL> DROP VIEW Cheap_View;

View dropped.

SQL>
```

```
SQL> spool off;
```

```
SQL> spool off;
```