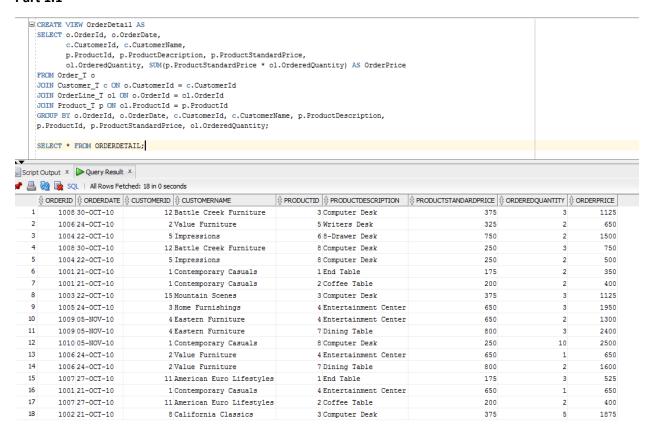
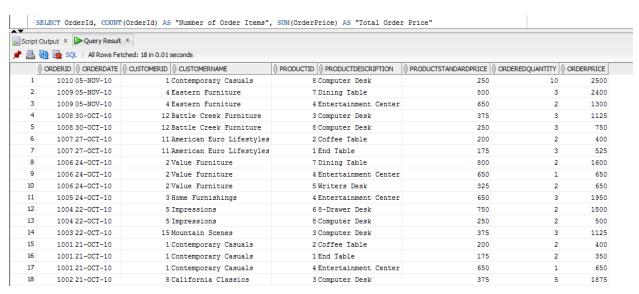
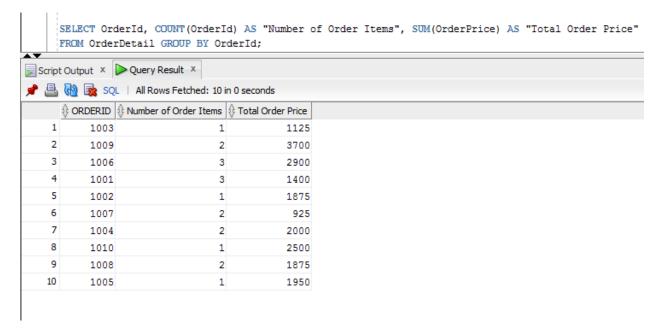
Part 1.1



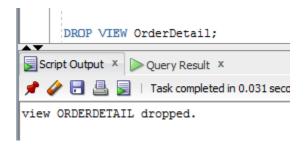
Part 1.2



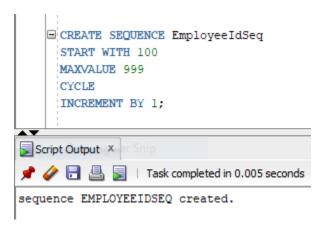
Part 1.3



Part 1.4



Part 2.1



Part 2.2

Part 2.3

```
CREATE VIEW EmployeeIN AS

SELECT EmployeeId, EmployeeName, EmployeeCity, EmployeeState

FROM Employee_T

WHERE EmployeeState = 'IN'

WITH CHECK OPTION;

Script Output × Query Result ×

Query Result ×

Task completed in 0.025 seconds

view EMPLOYEEIN created.
```

Part 2.4

Part 2.5

```
UPDATE EmployeeIN SET EmployeeState = 'CA' WHERE EmployeeName = 'Brandon Young';

Script Output X Query Result X

Query Result X

Task completed in 0.037 seconds

Error starting at line: 46 in command -

UPDATE EmployeeIN SET EmployeeState = 'CA' WHERE EmployeeName = 'Brandon Young'

Error report -

SQL Error: ORA-01402: view WITH CHECK OPTION where-clause violation

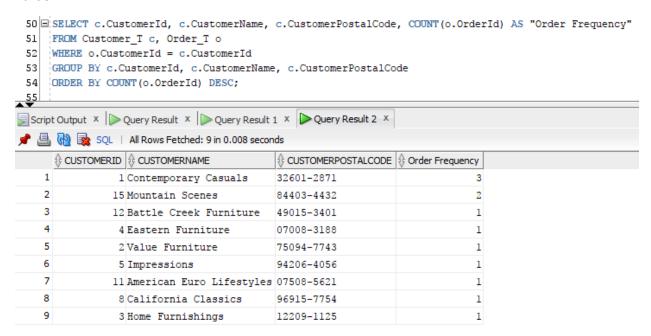
01402. 00000 - "view WITH CHECK OPTION where-clause violation"

*Cause:

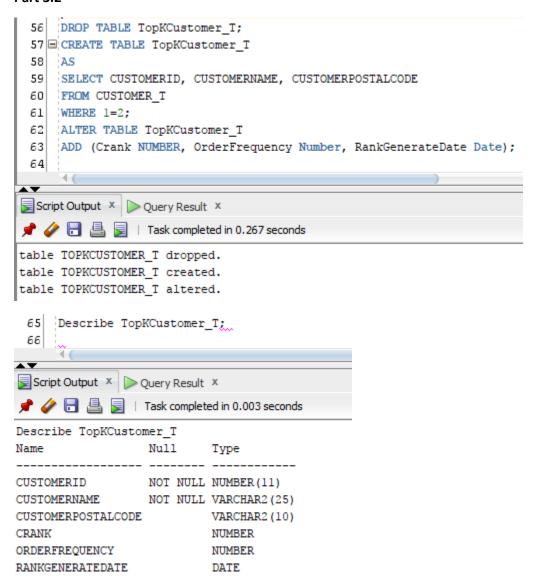
*Action:
```

Because of the WITH CHECK OPTION the Oracle Database prohibits any changes to EmployeeIN that would produce rows that are not included in the subquery.

Part 3.1



Part 3.2



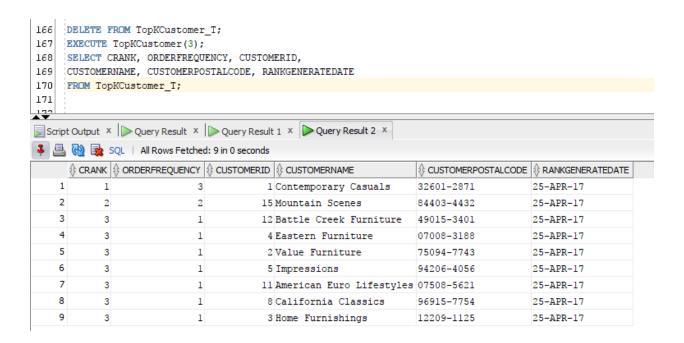
Part 3.3

```
129 SET SERVEROUTPUT ON
130 CREATE OR REPLACE PROCEDURE TopKCustomer(k NUMBER) AS
      rowCount NUMBER;
131
      ci CHAR(9);
132
133
     cn VARCHAR2 (40);
      cp VARCHAR2 (40);
134
135
      cr NUMBER:=1;
      ofq VARCHAR2(40);
136
137
      prevofq VARCHAR2(40):=NULL;
138
      CURSOR cl IS SELECT c.CustomerId, c.CustomerName, c.CustomerPostalCode, COUNT(o.OrderId) AS "Order Frequency"
139
140
    FROM Customer_T c, Order_T o
141
      WHERE o.CustomerId = c.CustomerId
142
      GROUP BY c.CustomerId, c.CustomerName, c.CustomerPostalCode
      ORDER BY COUNT(o.OrderId) DESC;
143
144 BEGIN
      SELECT Count(*) INTO rowCount FROM Customer T;
145
146
       OPEN C1;
147 E LOOP
        FETCH cl INTO ci, cn, cp, ofq;
148
         IF (prevofq = NULL) OR (ofq != prevofq) THEN
149
150
         cr:=cr+1;
        END IF;
151
        EXIT WHEN cl%NOTFOUND OR cr > k;
152
153
         INSERT INTO TopkCustomer_T (CustomerId, CustomerName, CustomerPostalCode, CRank, OrderFrequency, RankGenerateDate)
        VALUES (ci, cn, cp, cr, ofq, SYSDATE);
154
155
       prevofq:=ofq;
156
      END LOOP;
157
      CLOSE C1;
158 END TopKCustomer;
159
Script Output X Duery Result X
📌 🤌 🔡 📕 | Task completed in 0 seconds
PROCEDURE TOPKCUSTOMER compiled
```

```
160 DELETE FROM TopKCustomer T;
161 EXECUTE TopKCustomer(2);
162 SELECT CRANK, ORDERFREQUENCY, CUSTOMERID,
163 CUSTOMERNAME, CUSTOMERPOSTALCODE, RANKGENERATEDATE
164 | FROM TopKCustomer T;
Script Output X Query Result X
🗸 🖺 🙌 🗽 SQL | All Rows Fetched: 2 in 0.234 seconds
      ♦ CRANK |♦ ORDERFREQUENCY |♦ CUSTOMERID |♦ CUSTOMERNAME

⊕ CUSTOMERPOSTALCODE | ⊕ RANKGENERATEDATE

    1
                                          1 Contemporary Casuals 32601-2871
                                                                                      25-APR-17
    2
                                         15 Mountain Scenes
                                                                 84403-4432
                                                                                      25-APR-17
```



Part 4.1

```
112 GREATE TABLE AuditProduct_T

113 (ProductID number(11),

114 ProductLineId number(11),

115 ProductDescription varchar2(50),

116 ProductFinish varchar2(20),

117 productStandardPrice number(6,2),

118 ChangeDate date,

119 ChangeStatus varchar2(10) Constraint Status_C

120 CHECK (ChangeStatus IN ('Insert', 'Update', 'Delete')));

121

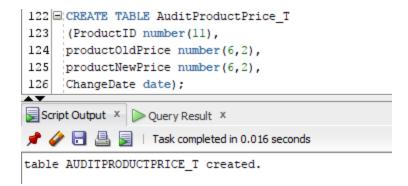
122

Script Output X Query Result X

ProductFinish varchar2(10) Constraint Status_C

CHECK (ChangeStatus IN ('Insert', 'Update', 'Delete')));
```

table AUDITPRODUCT_T created.

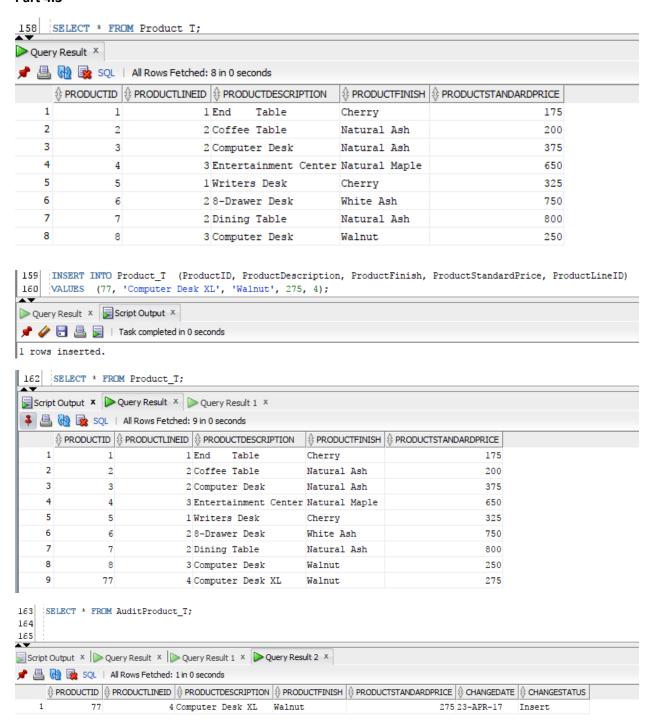


Part 4.2

```
128 CREATE OR REPLACE TRIGGER AuditProductTrig
129 AFTER INSERT OR UPDATE OR DELETE ON Product T
      REFERENCING NEW AS NewTuple OLD AS OldTuple
130
131
      FOR EACH ROW
132
133 BEGIN
134 CASE
135
        WHEN INSERTING THEN
136 🖃
         INSERT INTO AuditProduct_T (ProductId, ProductLineId, ProductDescription,
137
         ProductFinish, ProductStandardPrice, ChangeDate, ChangeStatus)
          VALUES (:NewTuple.ProductId, :NewTuple.ProductLineId, :NewTuple.ProductDescription, :NewTuple.ProductFinish,
138
139
          :NewTuple.ProductStandardPrice, SYSDATE, 'Insert');
       WHEN DELETING THEN
140
         INSERT INTO AuditProduct_T (ProductId, ProductLineId, ProductDescription,
142
          ProductFinish, ProductStandardPrice, ChangeDate, ChangeStatus)
          VALUES (:OldTuple.ProductId, :OldTuple.ProductLineId, :OldTuple.ProductDescription, :OldTuple.ProductFinish,
143
144
          :OldTuple.ProductStandardPrice, SYSDATE, 'Delete');
145
       WHEN UPDATING THEN
146 □
         INSERT INTO AuditProduct_T (ProductId, ProductLineId, ProductDescription,
147
          ProductFinish, ProductStandardPrice, ChangeDate, ChangeStatus)
148
          VALUES (:NewTuple.ProductId, :NewTuple.ProductFinish,
          :NewTuple.ProductStandardPrice, SYSDATE, 'Update');
149
150
         INSERT INTO AuditProductPrice_T (ProductId, ProductOldPrice, ProductNewPrice, ChangeDate)
151
          VALUES (:NewTuple.ProductId, :OldTuple.ProductStandardPrice, :NewTuple.ProductStandardPrice, SYSDATE);
152
       END CASE;
       END:
153
Script Output X Query Result X
📌 🧼 🔡 볼 📘 | Task completed in 0 seconds
```

TRIGGER AUDITPRODUCTTRIG compiled

Part 4.3



```
UPDATE Product_T SET ProductStandardPrice = 300 WHERE ProductId = 77;

165
166
167

Script Output × Query Result × Query Result 1 × Query Result 2 ×

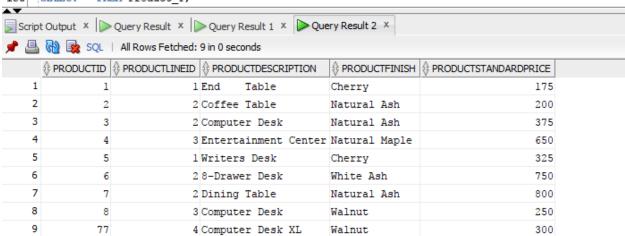
165
166
167

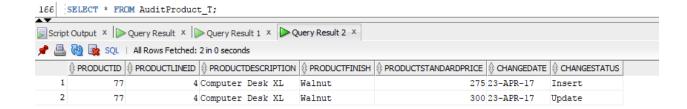
Task completed in 0.015 seconds

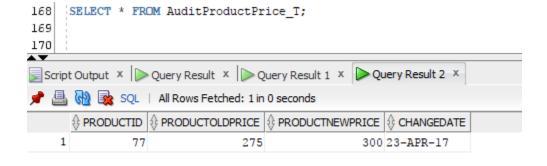
1 rows updated.

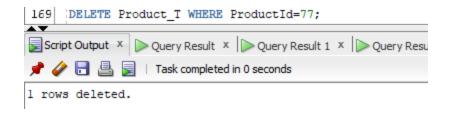
165
SELECT * FROM Product_T;

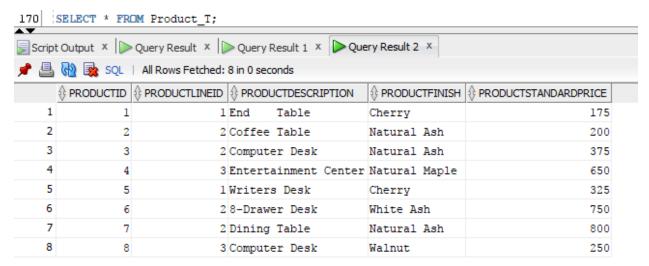
Script Output × Query Result × Query Result 1 × Query Result 2 ×
```

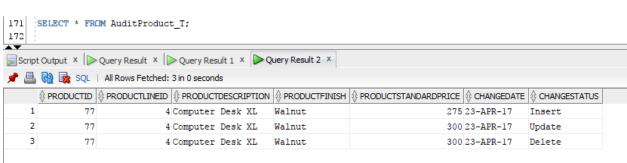












Part 5

Enter product finish: Natural Ash			
Order Frequency	Product Id	Description	Finish
4	3	Computer Desk	Natural Ash
2	7	Dining Table	Natural Ash
2	2	Coffee Table	Natural Ash
Enter product finis	h: Cherry		
Order Frequency	Product Id	Description	Finish
2	1	End Table	Cherry
1	5	Writers Desk	Cherry

```
package huntmj01.cs364.hw4;
import oracle.jdbc.*;
import oracle.jdbc.pool.OracleDataSource;
import java.sql.*;
import java.util.Scanner;
public class Main {
      public static void main(String[] args) throws SQLException {
            // make the connection to the database
            OracleDataSource ods = new OracleDataSource();
            ods.setUser("SYSTEM");
            ods.setPassword("admin");
            ods.setURL("jdbc:oracle:thin:@//localhost:1521/xe");
            Connection conn = ods.getConnection();
            // retrieve the user input
            String input;
            Scanner keyboard = new Scanner(System.in);
            System.out.print("Enter product finish: ");
            input = keyboard.nextLine();
            System.out.println("");
            // write and execute the query
            Statement stmt = conn.createStatement();
            ResultSet rs =
            stmt.executeQuery("SELECT COUNT(o.OrderId) AS \"Order Frequency\",
            p.ProductId, p.ProductDescription, p.ProductFinish "
            + "FROM Product_T p, Order_T o, OrderLine_T ol "
            + "WHERE p.ProductId = ol.ProductId AND o.OrderId = ol.OrderId "
            + "AND p.ProductFinish LIKE " + "'"+ input + "' "
            + "GROUP BY p.ProductId, p.ProductDescription, p.ProductFinish"
            + "ORDER BY COUNT(o.OrderId) DESC");
            // print header to the console
            System.out.printf("%-24s %-24s %-24s %-24s\n", "Order Frequency",
            "Product Id", " Description", "Finish");
            System. out. printf("%-24s %-24s %-24s %-24s \n", "-----",
            "-----", "-----", " ------
            ");
            // loop through the result set
            while(rs.next()){
                   // print out the row
                   System.out.println(String.format("%-24s %-24s %-24s %-24s\n",
                   rs.getString(1), rs.getString(2), rs.getString(3),
                   rs.getString(4)));
            }
            // close the the keyboard and <u>db</u> connection
            keyboard.close();
             conn.close();
      }
}
```