

שאלה 1

(א)

```

DELIMITER //
CREATE PROCEDURE myPrecuder(IN prodName VARCHAR(50))
BEGIN
    SELECT p.ProductName, cu.CompanyName, EXTRACT(YEAR FROM o.OrderDate) AS year,
        ROUND(sum(CASE WHEN EXTRACT(QUARTER FROM o.OrderDate) = 1 THEN od.Quantity*od.UnitPrice ELSE 0 END))
    AS quarter1,
        ROUND(sum(CASE WHEN EXTRACT(QUARTER FROM o.OrderDate) = 2 THEN od.Quantity*od.UnitPrice ELSE 0 END))
    AS quarter3,
        ROUND(sum(CASE WHEN EXTRACT(QUARTER FROM o.OrderDate) = 3 THEN od.Quantity*od.UnitPrice ELSE 0 END))
    AS quarter3,
        ROUND(sum(CASE WHEN EXTRACT(QUARTER FROM o.OrderDate) = 4 THEN od.Quantity*od.UnitPrice ELSE 0 END))
    AS quarter4
    FROM orders o
    INNER JOIN `order details` od -- join to get each order price (by order id)
    INNER JOIN products p -- join to get product name (by product id)
    INNER JOIN customers cu -- join to get customer company name (by customer id)
    ON o.OrderID = od.OrderID AND p.ProductID = od.ProductID AND o.CustomerID = cu.CustomerID AND p.ProductName =
    prodName
    GROUP BY p.ProductName, cu.CompanyName, EXTRACT(YEAR FROM o.OrderDate)
    ORDER BY p.ProductName, cu.CompanyName;
END //
DELIMITER ;

```

(ב)

```

1  import java.sql.*;
2  import java.util.Scanner;
3
4  public class c_sql {
5      public static void main(String[] args) {
6          try {
7              Class.forName("com.mysql.jdbc.Driver");
8              try{Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/northwind",
9                  user: "root", password: " ");
10                 Scanner myObj = new Scanner(System.in); // Create a Scanner object
11                 System.out.println("Please Enter Product Name:");
12                 String input_name = myObj.nextLine(); // Read user input
13                 PreparedStatement pstmt = null;
14                 ResultSet rs = null;
15                 String query = "CALL myPrecuder(?)";
16                 pstmt = con.prepareStatement(query);
17                 pstmt.setString(1, input_name);
18                 rs = pstmt.executeQuery();
19                 int numColumns = rs.getMetaData().getColumnCount();
20                 while (rs.next()) {
21                     for (int col = 1; col <= numColumns; col++) {
22                         System.out.print(rs.getString(col) + " ");
23                     }
24                     System.out.println();
25                 }
26             } catch (Exception ex) {
27                 System.out.println("Error");
28             }
29         } catch (Exception ex) {ex.printStackTrace();}
30     }
31 }
32

```

שאלה 2

טריגר זה יעבוד עבור product רנדומלי שבחרנו (80).

```
DELIMITER //
CREATE TRIGGER new_order
AFTER INSERT ON orders
FOR EACH ROW
BEGIN -- using BEGIN/END to execute multiple actions
    INSERT INTO `order details` (OrderID, ProductID, UnitPrice, Quantity, Discount) -- STEP 1: add the order to
`order details`
    VALUES (new.OrderID, 80, 5, 1, 0);
    UPDATE products -- STEP 2: update products.UnitsInStock column
    SET UnitsInStock = UnitsInStock-1
    WHERE `ProductID` = 80;
END //
DELIMITER ;
```

שאלה 3

```
SELECT doctor_name, salary,
FROM Doctor d
JOIN Queue_Summary q
ON (d.doctor_id = q.doctor_id AND q.date = '2020-04-20' AND q.num_of_patients > 4)
```

השאלתה מחזירה רק את הדוקטורים והמשכורת שלהם אשר בתאריך 20-04-2020 היה להם יותר מ-4 מטפלים.

שאלה 4

(א)

נתונה הרלציה: $R(A, B, C)$

A	B	C
a1	b2	c1
a1	b1	c2
a2	b2	c1
נובע גם:		
a1	b1	c1
a1	b2	c2

1NF - ניתן לראות כי 1NF מתקיים מכיוון שכל ערך קיים אך ורק בעמודה שלו (ערכים אטומים).

ניתן לראות כי אף עמודה או זוג עמודות לא גוררת עמודה אחרת:

$\{b2, c1\} \rightarrow a2$ וגם $\{b2, c1\} \rightarrow a1$ לכן CB איננו $CandidateKey (CK)$

$\{a1, c1\} \rightarrow b2$ וגם $\{a1, c1\} \rightarrow b1$ לכן AC איננו CK

$\{a1, b1\} \rightarrow c2$ וגם $\{a1, b1\} \rightarrow c1$ ולכן AB איננו CK

ולכן $2NF, 3NF, 3.5NF$ מתקיימות מכיוון שאין גרירה.

4NF – לא מתקיים מכיוון שיש תלות רב ערכית לדוגמה:

$\{a1, b1\} \rightarrow c2$ ו- $\{a1, b2\} \rightarrow c1$ מקיים: $\{a1, b1\} \rightarrow c1$

(ב)

$R(A, B, C, D, E)$

1. $AB \rightarrow CD$

2. $AC \rightarrow E$

3. $D \rightarrow A$

4. $BC \rightarrow DE$

CK	Non-Prime	NF
AB BC BD	E (נמצא רק הימין לגרירה)	NF2 (בגלל שורה 2)