

Advanced Java Programming [IT404]

Open-ended Project

Submitted to Dr Akansha Singh

Ramanuj Sharma A2305218670

Simran Gogia A2305218675

Utkarsh A2305218717

6CSE12

Project Title: E-Commerce website using Servlets and JSP

Tech Stack: Java Servlets, MySQL, JSP, Bootstrap

Theory:

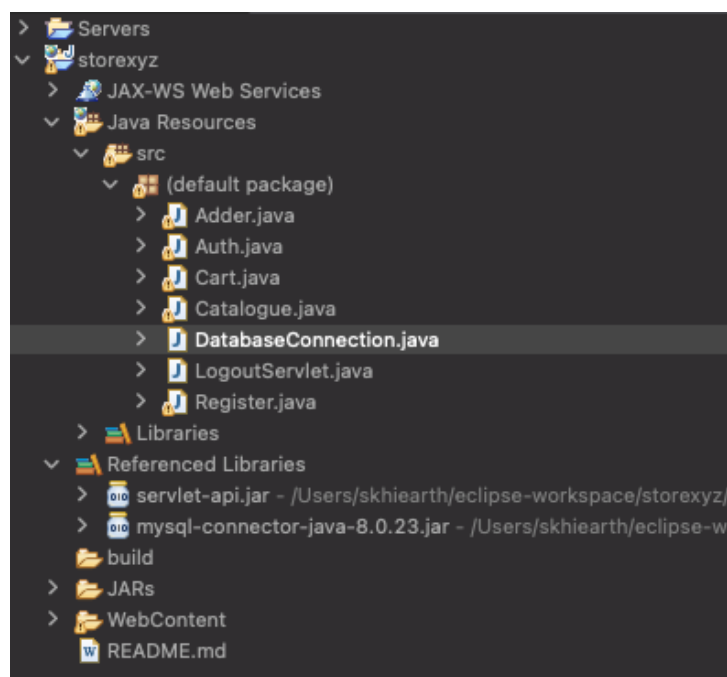
The project, titled *store.xyz*, is a mockup for an E-Commerce application built on Java and MySQL. We have used Eclipse as the IDE for Development.

Our project is on an E-Commerce website using Servlets and JSP. Apart from these, we have used Bootstrap for design and some custom CSS. On the main screen of our application, *store.xyz*, users can see all the products that are available in the catalogue. These are not statically typed, and are fetched from the SQL database using Servlet and rendered using iterative loops in JSP, which is then embedded in the main HTML page.

Source Code: The source code in its entirety has been uploaded on GitHub, which we also used for version control during the development stage. Some key source files are shown here.

GitHub Repository - <https://github.com/skhiearth/E-Commerce-Java-Servlets>

Project File Structure:



DatabaseConnection.java (<https://github.com/skhiearth/E-Commerce-Java-Servlets/blob/main/src/DatabaseConnection.java>)

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DatabaseConnection {
    protected static Connection initializeDatabase()
        throws SQLException, ClassNotFoundException {
        String dbURL = "jdbc:mysql://localhost:3306/";
        // Database name to access
        String dbName = "ECommerce";
        String dbUsername = "root";
        String dbPassword = "9810282699";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con = DriverManager.getConnection(dbURL + dbName, dbUsername, dbPassword);
        System.out.println("Connected database successfully...");
        return con;
    }
}
```

Catalogue.java (<https://github.com/skhiearth/E-Commerce-Java-Servlets/blob/main/src/Catalogue.java>)

```
import java.io.IOException;
import java.io.PrintWriter;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import java.sql.*;
import java.util.Scanner;
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

import java.util.*;
import javax.servlet.*;

@WebServlet("/Catalogue")
public class Catalogue extends HttpServlet {

    private ServletConfig config;

    //Setting JSP page
    String page="Catalogue.jsp";

    public Catalogue() {
        Statement stmt = null;

        try {
            Connection con = DatabaseConnection.initializeDatabase();
            stmt = con.createStatement();

            String sql = "CREATE TABLE IF NOT EXISTS CATALOGUE " +
                "(id INTEGER not NULL, " +
                " image VARCHAR(400) not NULL, " +
                " name VARCHAR(60) not NULL, " +
                " price DOUBLE not NULL, " +
                " description VARCHAR(1000) not NULL, " +
                " PRIMARY KEY ( id ))";

            stmt.executeUpdate(sql);

            System.out.println("Created table in given database...");
        } catch (Exception e) {
            System.out.println(e);
        }
    }

    public void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        PrintWriter out = response.getWriter();

        Connection con = null;

        ResultSet rs;

        response.setContentType("text/html");

        List dataList = new ArrayList();

        try {
            Class.forName("com.mysql.jdbc.Driver");

            con = DatabaseConnection.initializeDatabase();

            String sql = "select * from CATALOGUE";
            Statement s = con.createStatement();
            s.executeQuery (sql);
            rs = s.getResultSet();

            while (rs.next ()) {
                dataList.add(rs.getString("image"));
                dataList.add(rs.getString("name"));
                dataList.add(rs.getInt("price"));
                dataList.add(rs.getString("description"));
                dataList.add(rs.getInt("id"));
            }

            rs.close ();
            s.close ();
        } catch (Exception e) {
            System.out.println("Exception is :"+e);
        }

        request.setAttribute("data",dataList);
        RequestDispatcher dispatcher = request.getRequestDispatcher(page);
        if (dispatcher != null){
            dispatcher.forward(request, response);
        }
    }
}
```

Auth.java (<https://github.com/skhiearth/E-Commerce-Java-Servlets/blob/main/src/Auth.java>)

Cart.java

(<https://github.com/skhiearth/E-Commerce-Java-Servlets/blob/main/src/Cart.java>)

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import java.sql.*;
import java.util.Scanner;
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

@WebServlet("/Auth")
public class Auth extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public Auth() {
        Statement stmt = null;
    }

    try {
        Connection con = DatabaseConnection.initializeDatabase();
        stmt = con.createStatement();

        String sql = "CREATE TABLE IF NOT EXISTS USERS" +
            " (username VARCHAR(60) not NULL, " +
            " password VARCHAR(60) not NULL, " +
            " name VARCHAR(60) not NULL, " +
            " contact VARCHAR(60) not NULL, " +
            " PRIMARY KEY (username));";

        stmt.executeUpdate(sql);
        System.out.println("Created table in given database...");
    } catch (Exception e) {
        System.out.println(e);
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        response.getWriter().append("Served at: ").append(request.getContextPath());
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        try {
            String username = request.getParameter("username");
            String password = request.getParameter("pass");

            Boolean login = false;

            System.out.println("Username: " + username);

            Statement stmt = null;

            try {
                Connection con = DatabaseConnection.initializeDatabase();
                stmt = con.createStatement();

                String selector = "SELECT username, password FROM USERS";
                ResultSet rs = stmt.executeQuery(selector);

                if (rs.next() == false) {
                    System.out.println("Incorrect username or password.");
                    PrintWriter out=response.getWriter();
                    String someMessage = "Incorrect username or password.";
                    out.println("<script type='text/javascript'>");
                    out.println("alert(' " + someMessage + " ');");
                    out.println("</script>");
                    response.sendRedirect("signin.html");
                    login = false;
                } else {
                    while(rs.next()){
                        String usr = rs.getString("username");
                        String pass = rs.getString("password");

                        if(usr.equals(username) && pass.equals(password)){
                            System.out.println("Logged in with user: " + username);
                            System.out.println("Successfully Logged In!");

                            HttpSession session=request.getSession();
                            session.setAttribute("username", username);

                            response.sendRedirect("index.jsp");
                            login = true;
                            break;
                        } else {
                            login = false;
                            PrintWriter out=response.getWriter();
                            out.println("<script type='text/javascript'>");
                            out.println("alert('User or password incorrect');");
                            out.println("location='signin.html';");
                            out.println("</script>");
                            response.sendRedirect("signin.html");
                        }
                    }

                    PrintWriter out=response.getWriter();
                    out.println("<script type='text/javascript'>");
                    out.println("alert('User or password incorrect');");
                    out.println("location='signin.html';");
                    out.println("</script>");
                    response.sendRedirect("signin.html");
                }

            } catch (Exception e) {
                System.out.println(e);
                PrintWriter out=response.getWriter();
                out.println("<script type='text/javascript'>");
                out.println("alert('User or password incorrect');");
                out.println("location='signin.html';");
                out.println("</script>");
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import java.sql.*;
import java.util.Scanner;
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

import java.util.*;
import javax.servlet.*;

@WebServlet("/Cart")
public class Cart extends HttpServlet{

    private ServletConfig config;

    //Setting JSP page
    String page="Cart.jsp";

    public void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        PrintWriter out = response.getWriter();

        Connection con = null;

        ResultSet rs;

        response.setContentType("text/html");

        List dataList = new ArrayList();
        List<Integer> ids = new ArrayList();

        try {
            Class.forName("com.mysql.jdbc.Driver");

            con = DatabaseConnection.initializeDatabase();

            String sql = "select * from CART";
            Statement s = con.createStatement();
            s.executeQuery (sql);
            rs = s.getResultSet();

            HttpSession session=request.getSession(false);

            String name = null;

            if(session!=null){
                name=(String)session.getAttribute("username");
                System.out.println("Username: " + name);
                if(name != null){
                    while (rs.next ()) {
                        String username = rs.getString("username");

                        if(username.equals(name)){
                            dataList.add(rs.getInt("id"));
                            ids.add(rs.getInt("id"));
                        }
                    }

                    System.out.println(ids);
                    Set<Integer> uniqueIds = new HashSet<Integer>(ids);
                    for (Integer id : uniqueIds) {
                        System.out.println(id);

                        Statement stat = null;

                        stat = con.createStatement();
                        sql = "SELECT * FROM CATALOGUE";

                        rs = stat.executeQuery(sql);
                        while(rs.next()){
                            //Retrieve by column name
                            int _id = rs.getInt("id");
                            double price = rs.getDouble("price");
                            String _name = rs.getString("name");

                            if(_id == id) {
                                dataList.add(id);
                                dataList.add(_name);

                                int occurrences = Collections.frequency(ids, id);
                                dataList.add(occurrences);

                                dataList.add(price);
                                int totalPrice = (int)price * occurrences;
                                dataList.add(totalPrice);
                            }
                        }

                        System.out.println(dataList);
                    }
                } else {
                    RequestDispatcher dispatcher = request.getRequestDispatcher("error.jsp");
                    if (dispatcher != null){
                        dispatcher.forward(request, response);
                    }
                }
            } else {
                PrintWriter out=response.getWriter();
                out.println("<script type='text/javascript'>");
                out.println("alert('User or password incorrect');");
                out.println("location='signin.html';");
                out.println("</script>");
                request.getRequestDispatcher("index.jsp").include( request, response);
            }

            rs.close ();
            s.close ();
        } catch (Exception e) {
            System.out.println("Exception is :"+e);
            out.println("<script type='text/javascript'>");
            out.println("alert('You need to be logged in!');");
            out.println("location='index.jsp';");
            out.println("</script>");
        }

        request.setAttribute("data",dataList);
        RequestDispatcher dispatcher = request.getRequestDispatcher(page);
        if (dispatcher != null){
            dispatcher.forward(request, response);
        }
    }
}
```

LogoutServlet.java (<https://github.com/skhiearth/E-Commerce-Java-Servlets/blob/main/src/LogoutServlet.java>)

```
import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

/**
 * Servlet implementation class LogoutServlet
 */
@WebServlet("/LogoutServlet")
public class LogoutServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public LogoutServlet() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        request.getRequestDispatcher("index.jsp").include(request, response);

        HttpSession session=request.getSession();
        session.invalidate();
        response.sendRedirect("index.jsp");
    }

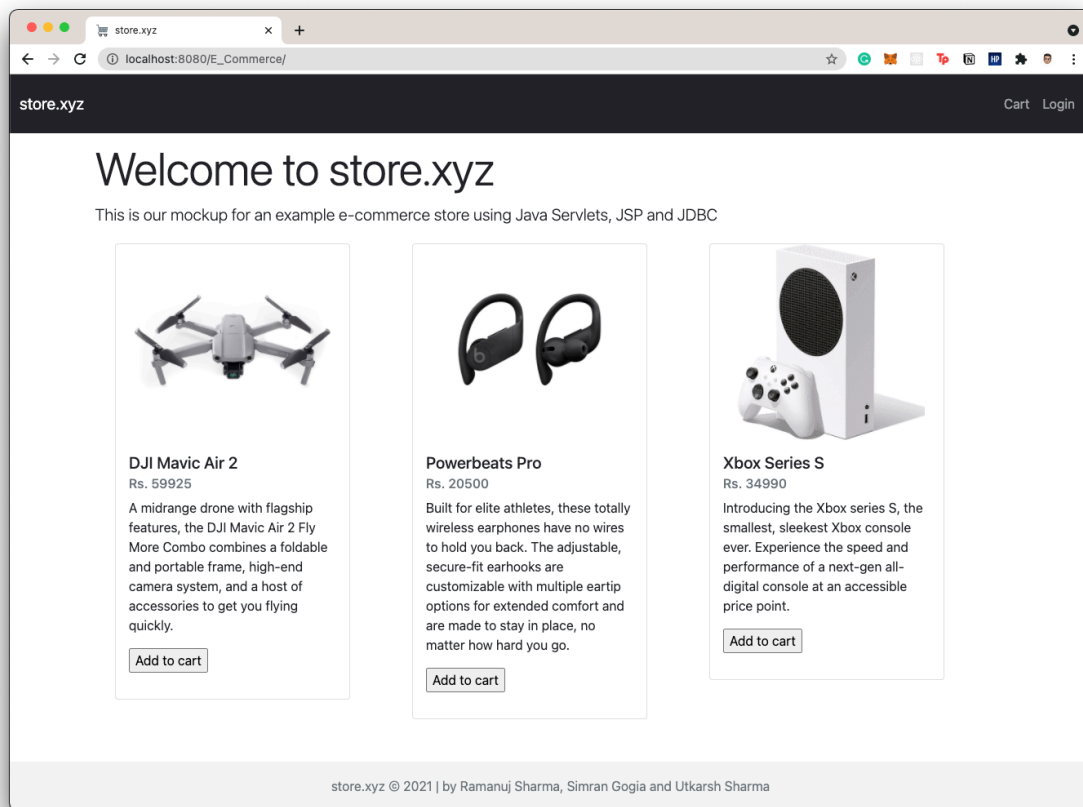
    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);
    }
}
```

index.jsp (<https://github.com/skhiearth/E-Commerce-Java-Servlets/blob/main/WebContent/index.jsp>)

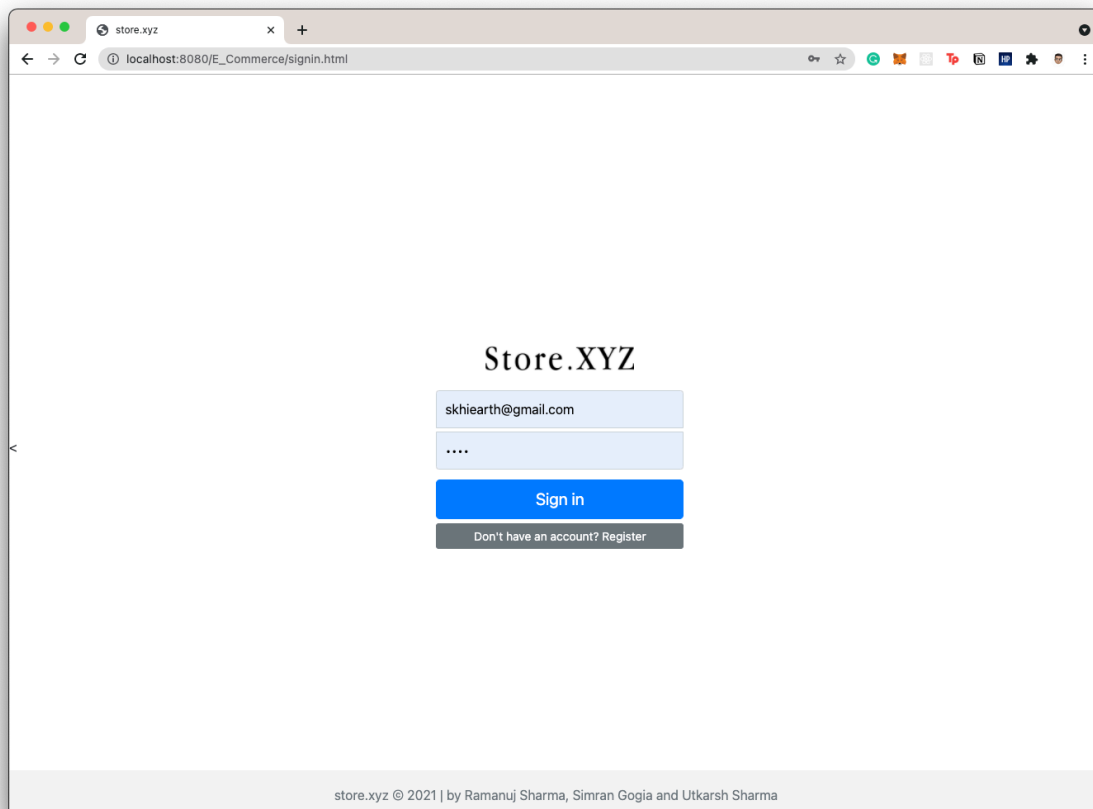
Full Source code on GitHub (<https://github.com/skhiearth/E-Commerce-Java-Servlets>)

Outputs:

http://localhost:8080/E_Commerce/



http://localhost:8080/E_Commerce/signin.html



http://localhost:8080/E_Commerce/register.html

Store.XYZ

Simran Gogia

9999999999

simran@gmail.com

Register

Already have an account? Sign in

store.xyz © 2021 | by Ramanuj Sharma, Simran Gogia and Utkarsh Sharma

http://localhost:8080/E_Commerce/CartOne.jsp

store.xyz Logout

Your Cart

ID	Product Name	Quantity	Price per piece	Total Price
1	Powerbeats Pro	1	20500.0	20500
2	Xbox Series S	1	34990.0	34990

Place Order

store.xyz © 2021 | by Ramanuj Sharma, Simran Gogia and Utkarsh Sharma

Database Snapshots:

ECOMMERCE Database

The screenshot shows the phpMyAdmin interface for the 'ecommerce' database. The left sidebar displays the database structure with 'ecommerce' selected, showing sub-databases like 'CART', 'CATALOGUE', and 'USERS'. The main panel shows the 'Structure' tab for the 'ecommerce' database. It lists three tables: 'CART', 'CATALOGUE', and 'USERS'. Below the table list, there is a 'Create table' section with a 'Name' field and a 'Number of columns' field set to 4. The 'Console' at the bottom is empty.

Table	Action	Rows	Type	Collation	Size	Overhead
CART	Browse Structure Search Insert Empty Drop	7	InnoDB	utf8mb4_0900_ai_ci	16.0 KiB	-
CATALOGUE	Browse Structure Search Insert Empty Drop	6	InnoDB	utf8mb4_0900_ai_ci	16.0 KiB	-
USERS	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_0900_ai_ci	16.0 KiB	-
3 tables	Sum	16	InnoDB	utf8mb4_0900_ai_ci	48.0 KiB	0 B

CATALOGUE Table

The screenshot shows the phpMyAdmin interface for the 'ecommerce' database, specifically the 'CATALOGUE' table. The left sidebar shows the database structure with 'ecommerce' selected. The main panel shows the 'Browse' tab for the 'CATALOGUE' table. It displays a table with 6 rows (rows 0-5) and 5 columns: 'id', 'image', 'name', 'price', and 'description'. The table contains data for various products like 'DJI Mavic Air 2', 'Powerbeats Pro', 'Xbox Series S', 'Bose Noise Cancelling 700', 'Mac Mini (M1, 256GB, 8GB)', and 'Canon EOS 90D'. Below the table, there are options for 'Query results operations' including 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'.

id	image	name	price	description
0	https://static.bhphoto.com/images/images2500x2500/...	DJI Mavic Air 2	59925	A midrange drone with flagship features, the DJI M...
1	https://store.storeimages.cdn-apple.com/4982/as-im...	Powerbeats Pro	20500	Built for elite athletes, these totally wireless e...
2	https://encrypted-tbn1.gstatic.com/shopping?q=tbn:...	Xbox Series S	34990	Introducing the Xbox series S, the smallest, sleek...
3	https://rukminim1.flixcart.com/image/832/832/k5vcy...	Bose Noise Cancelling 700	31050	Nothing in the environment will come in the way of...
4	https://store.storeimages.cdn-apple.com/4668/as-im...	Mac Mini (M1, 256GB, 8GB)	64900	The Apple M1 chip takes our most versatile, do-it-...
5	https://www.reliancedigital.in/medias/Canon-EOS-90...	Canon EOS 90D	124990	The EOS 90D shoots high quality 4K 30p/25p videos ...

USERS Table

The screenshot shows the phpMyAdmin interface for the 'eCommerce' database. The 'USERS' table is selected, and its structure is displayed. The table has four columns: 'username', 'password', 'name', and 'contact'. The data is as follows:

username	password	name	contact
ramanuj@gmail.com	aaa	Ramanuj	1000010000
simran@gmail.com	password	Simran Gogia	9999999999
skhiearth@gmail.com	pass	Utkarsh	9810282699

The interface also shows a 'Query results operations' section with options to print, copy to clipboard, export, display chart, and create view.

CART Table

The screenshot shows the phpMyAdmin interface for the 'eCommerce' database. The 'CART' table is selected, and its structure is displayed. The table has two columns: 'username' and 'id'. The data is as follows:

username	id
skhiearth@gmail.com	0
skhiearth@gmail.com	1
skhiearth@gmail.com	0
skhiearth@gmail.com	1
skhiearth@gmail.com	0
simran@gmail.com	1
simran@gmail.com	2

The interface also shows a 'Query results operations' section with options to print, copy to clipboard, export, display chart, and create view.