```
using System;
using System.Data;
using System.Collections.Generic;
using System.Web.UI;
using System.Web.UI.WebControls;
using Oracle.DataAccess.Client;

namespace WebApplication3
{
    public partial class Reports : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
            {
                GridView1.Visible = false;
                GridView2.Visible = false;
            }
        }

        // ========================
        // SEARCH BUTTON
        // ========================
        protected void Search_book(object sender, EventArgs e)
        {
            string report = Book_report.Text.Trim();

            if (report == "Inventory List")
                LoadInventory();
            else if (report == "In/Out Register")
                LoadInOutRegister();
```

```
    else if (report == "Book yet to be returned")
        LoadPendingBooks();
    else if (report == "Lost Book List")
        LoadLostBooks();
    else
        ScriptManager.RegisterClientScriptBlock(
            this, GetType(), "msg",
            "alert('Please select valid report')", true);
}

// ========================
// INVENTORY
// ========================
private void LoadInventory()
{
    GridView1.Visible = false;
    GridView2.Visible = true;

    string query =
        "SELECT BOOKID, SUBJECT, AUTHOR, PUBLISHER, STATUS " +
        "FROM BOOK_MASTER ORDER BY SUBJECT";

    LoadData(query, null, GridView2);
}

// ========================
// IN / OUT REGISTER
// ========================
private void LoadInOutRegister()
{
    GridView1.Visible = true;
```

```
GridView2.Visible = false;

DateTime? fromDate = ParseDate(from_date.Text);
DateTime? toDate = ParseDate(to_date.Text);

string query =
   "SELECT b.BOOKID, b.SUBJECT, i.IT_DATE AS ISSUE_DATE, " +
   "u.FULLNAME, r.RT_DATE AS RETURN_DATE " +
   "FROM ISSUE_BOOKS i " +
   "JOIN ISSUE_BOOKS r ON i.T_ID = r.T_ID " +
   "JOIN BOOK_MASTER b ON b.BOOKID = i.BOOK_ID " +
   "JOIN HASH_USER u ON u.EMPLOYEEID = i.USER_ID " +
   "WHERE i.ACTIVITY='Issue' AND r.ACTIVITY='Returned'";

List<OracleParameter> p = new List<OracleParameter>();

if (fromDate.HasValue)
{
   query += " AND i.IT_DATE >= :fromDate";
   p.Add(new OracleParameter(":fromDate", fromDate.Value));
}

if (toDate.HasValue)
{
   query += " AND i.IT_DATE <= :toDate";
   p.Add(new OracleParameter(":toDate", toDate.Value));
}

query += " ORDER BY b.SUBJECT";

LoadData(query, p.ToArray(), GridView1);
```

```
}

// ========================
// PENDING BOOKS
// ========================
private void LoadPendingBooks()
{
    GridView1.Visible = true;
    GridView2.Visible = false;

    string query =
        "SELECT i.BOOK_ID AS BOOKID, b.SUBJECT, i.IT_DATE AS ISSUE_DATE, " +
        "u.FULLNAME, 'Not Returned Yet' AS RETURN_DATE " +
        "FROM ISSUE_BOOKS i " +
        "JOIN BOOK_MASTER b ON b.BOOKID=i.BOOK_ID " +
        "JOIN HASH_USER u ON u.EMPLOYEEID=i.USER_ID " +
        "WHERE i.AVAILABILITY=1 " +
        "ORDER BY b.SUBJECT";

    LoadData(query, null, GridView1);
}

// ========================
// LOST BOOKS
// ========================
private void LoadLostBooks()
{
    GridView1.Visible = true;
    GridView2.Visible = false;

    string query =
```

```
        "SELECT b.BOOKID, b.SUBJECT, i.IT_DATE AS ISSUE_DATE, " +
        "u.FULLNAME, 'Lost' AS RETURN_DATE " +
        "FROM BOOK_MASTER b " +
        "LEFT JOIN ISSUE_BOOKS i ON i.BOOK_ID = b.BOOKID " +
        "LEFT JOIN HASH_USER u ON u.EMPLOYEEID = i.USER_ID " +
        "WHERE b.STATUS = 'Lost' " +
        "ORDER BY b.SUBJECT";

    LoadData(query, null, GridView1);
}

// ========================
// COMMON DATA METHOD
// ========================
private void LoadData(string query, OracleParameter[] parameters, GridView grid)
{
    DbConnection db = new DbConnection();
    OracleCommand cmd = null;
    OracleDataAdapter da = null;
    DataTable dt = new DataTable();

    try
    {
        cmd = new OracleCommand(query, db.GetConnection());

        if (parameters != null)
            cmd.Parameters.AddRange(parameters);

        da = new OracleDataAdapter(cmd);
        da.Fill(dt);
```

```
            grid.DataSource = dt;
            grid.DataBind();
        }
        catch (Exception ex)
        {
            Response.Write(ex.Message);
        }
        finally
        {
            db.CloseConnection();
        }
    }

    // ========================
    // DATE PARSER
    // ========================
    private DateTime? ParseDate(string text)
    {
        DateTime dt;
        if (DateTime.TryParse(text, out dt))
            return dt;
        return null;
    }
  }
}
```