﻿using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Drawing;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using CrystalDecisions.CrystalReports.Engine;

using System.Data.SqlClient;

namespace Liaqat\_Gym\_Management\_System

{

public partial class Report : UserControl

{

public Report()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

if (cmbReport.SelectedIndex == 0)

{

AttendenceReport();

}

else

{

CertificateReport();

}

}

private void CertificateReport()

{

string query = "SELECT ";

string from = fromdate.Value.ToShortDateString();

string to = todate.Value.ToShortDateString();

if (chkRank.Checked == false && chkSerialNumber.Checked == false && chkAmount.Checked == false && chkclub.Checked == false && chkIssueDate.Checked == false)

{

MessageBox.Show("Please Select Report Field", "Field is Missing", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

else

{

if (chkclub.Checked)

{

query = query + " c.Club\_Name as 'Club' , ";

}

if (chkRank.Checked)

{

query = query + " cc.Rank ,";

}

if (chkSerialNumber.Checked)

{

query = query + " SerialNumber ,";

}

if (chkAmount.Checked)

{

query = query + " Amount ,";

}

if (chkIssueDate.Checked)

{

query = query + " IssueDate as 'Issue Date' ,";

}

query = query.Substring(0, query.Length - 1);

query += " from ClubCertification as cc Inner Join Clubs as c On c.Club\_Id = cc.Club Where Convert(date,[IssueDate]) >= '" + from + "' And Convert(date,[IssueDate]) <= '" + to + "'";

if (!string.IsNullOrEmpty(txtClub.Text.ToString().Trim()))

{

query += " And c.Club\_Name = '" + txtClub.Text.Trim().ToString() + "'";

}

query += " Order by IssueDate desc";

ReportDocument crystalReport = new ReportDocument();

myReportViewer.DisplayGroupTree = true;

myReportViewer.ToolPanelView = CrystalDecisions.Windows.Forms.ToolPanelViewType.None;

//crystalReport.Load(System.Windows.Forms.Application.StartupPath + "\\MyReport.rpt");

//crystalReport.Load(Server.MapPath("~/CustomerReport.rpt"));

CertificationsReport myreport = new CertificationsReport();

CrystalDecisions.CrystalReports.Engine.TextObject fromdateReport;

fromdateReport = myreport.ReportDefinition.ReportObjects["FromDate"] as TextObject;

fromdateReport.Text = fromdate.Value.ToShortDateString();

CrystalDecisions.CrystalReports.Engine.TextObject todateReport;

todateReport = myreport.ReportDefinition.ReportObjects["ToDate"] as TextObject;

todateReport.Text = todate.Value.ToShortDateString();

myreport.SummaryInfo.ReportTitle = "Certificate Report";

crystalReport = myreport;

DataSet dsCustomers = GetData(query, crystalReport);

myreport.SetDataSource(dsCustomers);

myReportViewer.ReportSource = myreport;

myreport.Refresh();

}

}

private void AttendenceReport()

{

string query = "SELECT ";

string from = fromdate.Value.ToShortDateString();

string to = todate.Value.ToShortDateString();

if (chbReg.Checked == false &&chbName.Checked == false && chbPhone.Checked == false && chbDate.Checked == false)

{

MessageBox.Show("Please Select Report Field", "Field is Missing", MessageBoxButtons.OK, MessageBoxIcon.Asterisk);

}

else

{

if (chbReg.Checked)

{

query = query + " a.regno as 'Reg #' ,";

}

if (chbName.Checked)

{

query = query + " name as 'Name' ,";

}

if (chbPhone.Checked)

{

query = query + " phone as 'Contact' ,";

}

if (chbDate.Checked)

{

query = query + " date as 'Date & Time' ";

}

query = query.Substring(0, query.Length - 1);

query += " FROM Attendence a Inner Join [user] as u On a.regno = u.regno Where Convert(date,[date]) >= '" + from + "' And Convert(date,[date]) <= '" + to + "'";

if (!string.IsNullOrEmpty(txtMember.Text.ToString().Trim()))

{

query += " And name = '" + txtMember.Text.Trim().ToString()+ "' Or phone like '%"+ txtMember.Text.Trim().ToString()+ "%' Or nic Like '%" + txtMember.Text.Trim().ToString() + "%'" ;

}

ReportDocument crystalReport = new ReportDocument();

myReportViewer.DisplayGroupTree = true;

myReportViewer.ToolPanelView = CrystalDecisions.Windows.Forms.ToolPanelViewType.None;

//crystalReport.Load(System.Windows.Forms.Application.StartupPath + "\\MyReport.rpt");

//crystalReport.Load(Server.MapPath("~/CustomerReport.rpt"));

ReportDocument reportDocument = new ReportDocument();

AttendenceReport myreport = new AttendenceReport();

CrystalDecisions.CrystalReports.Engine.TextObject fromdateReport;

fromdateReport = myreport.ReportDefinition.ReportObjects["FromDate"] as TextObject;

fromdateReport.Text = fromdate.Value.ToShortDateString();

CrystalDecisions.CrystalReports.Engine.TextObject todateReport;

todateReport = myreport.ReportDefinition.ReportObjects["ToDate"] as TextObject;

todateReport.Text = todate.Value.ToShortDateString();

myreport.SummaryInfo.ReportTitle = "Attendence Report";

reportDocument = myreport;

DataSet dsCustomers = GetData(query, reportDocument);

myreport.SetDataSource(dsCustomers);

myReportViewer.ReportSource = myreport;

myreport.Refresh();

}

}

private ReportDataSet GetData(string query, ReportDocument crystalReport)

{

//string conString = "Data Source=DESKTOP-75F694L;Initial Catalog=Gym;Integrated Security=True";

SqlCommand cmd = new SqlCommand(query);

using (SqlConnection con = Connection.Sql\_Connection())

{

ReportDataSet dsCustomers = new ReportDataSet();

cmd.Connection = con;

con.Open();

using (SqlDataReader sdr = cmd.ExecuteReader())

{

//Get the List of all TextObjects in Section2.

List<TextObject> textObjects = crystalReport.ReportDefinition.Sections["Section2"].ReportObjects.OfType<TextObject>().ToList();

for (int i = 0; i < textObjects.Count; i++)

{

//Set the name of Column in TextObject.

textObjects[i].Text = string.Empty;

if (sdr.FieldCount > i)

{

textObjects[i].Text = sdr.GetName(i);

}

}

//Load all the data rows in the Dataset.

while (sdr.Read())

{

DataRow dr = dsCustomers.Tables[0].Rows.Add();

for (int i = 0; i < sdr.FieldCount; i++)

{

dr[i] = sdr[i];

}

}

}

con.Close();

return dsCustomers;

}

}

private void panel1\_Paint(object sender, PaintEventArgs e)

{

}

private void label5\_Click(object sender, EventArgs e)

{

}

private void cmbReport\_SelectedIndexChanged(object sender, EventArgs e)

{

if(cmbReport.SelectedIndex == 0)

{

pnlCertificateFields.Hide();

pnlCertificateSearch.Hide();

pnlAttendecneFields.Show();

pnlAttendenceSearch.Show();

}

else

{

pnlCertificateFields.Show();

pnlCertificateSearch.Show();

pnlAttendecneFields.Hide();

pnlAttendenceSearch.Hide();

}

}

private void Report\_Load(object sender, EventArgs e)

{

cmbReport.SelectedIndex = 0;

}

}

}