

## **Coding (or Script) - 4**

```

using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Web;
using System.Data.SqlClient;
using System.Configuration;
using System.IO;
using System.Net;
using BAI.Utilities;
using System.Web.Mail;
using System.Text.RegularExpressions;

```

```

/*****
 *
 *
 *   Created By: Ken Wu
 *
 *   Date: 03/28/06
 *
 *
 *   Description: Consume Web Service for netForum Registration
 *
 *   Deployment Files: \bin\AutoNetForumWebService.exe, \bin\AutoNetForumWebService.pdb
 *   (debug)
 *
 *   \bin\BAI.Utilities.dll
 *
 *   \Web References\com.leveragesoftware.services.web\Reference.map,
 *
 *   \Web References\com.leveragesoftware.services.web\
 *   invitationService.wsdl
 *
 *
 *
 *****/

```

```
namespace AutoNetForumWebService
```

```

{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Class1
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            Console.WriteLine(args[0]);
            string FName = "";
            string Lname = "";
            string E_mail = "";
            string Pname = " ";
            com.leveragesoftware.services.web.UserInfo user_exh = new com.
            leveragesoftware.services.web.UserInfo();
            com.leveragesoftware.services.web.UserInfo user_net = new com.
            leveragesoftware.services.web.UserInfo();
            bool flg_exh , flg_net;
            string msg_exh, msg_net;

```

```
com.leveragesoftware.services.web.Result result_exh = new com.
leveragesoftware.services.web.Result();
com.leveragesoftware.services.web.Result result_net = new com.
leveragesoftware.services.web.Result();
com.leveragesoftware.services.web.ProfileData ProfileData_exh = new com.
leveragesoftware.services.web.ProfileData();
com.leveragesoftware.services.web.ProfileData ProfileData_net = new com.
leveragesoftware.services.web.ProfileData();

// Consume the leveragesoftware Web Service
com.leveragesoftware.services.web.InvitationService svcNET = new com.
leveragesoftware.services.web.InvitationService();

SqlConnection myConnection = new SqlConnection("server=CARLA; database=
AdditionalData; User ID=netForumWebUser; Password=XXXXXXXXXX"; // netForumStaging

SqlCommand myCommand = new SqlCommand("dbo.client_bai_netForumReg_Sel",
myConnection);

myCommand.CommandType = CommandType.StoredProcedure;

try
{
// myCommand.Parameters.Add ( new SqlParameter("@EventCode", SqlDbType.
VarChar) ).Value = "6289";
myCommand.Parameters.Add ( new SqlParameter("@EventCode", SqlDbType.
VarChar) ).Value = Convert.ToString(args[0]);

myConnection.Open();

SqlDataReader readStr = myCommand.ExecuteReader();

while (readStr.Read())
{
if( !readStr.IsDBNull(0) && !readStr.IsDBNull(1) && !readStr.IsDBNull(
(2) && !readStr.IsDBNull(3) ) // First Name and Last Name and Email can not be NULL
{
E_mail = readStr.GetString(0); // Email
Fname = readStr.GetString(1); // First Name
Lname = readStr.GetString(2); // Last Name
Pname = readStr.GetString(3); // Plan Name

user_net.Email = E_mail;
user_net.ID = E_mail;
user_net.FirstName = Fname;
user_net.LastName = Lname;
user_net.PlanName = Pname;

// Add User To the System
result_net = svcNET.Add("networkingatbai", "8291a28082aalc81",
user_net);

flg_net = result_net.Success;
msg_net = result_net.Message;

if(flg_net == false) // Add funcation fail
{
AddToErrorLog(E_mail, Fname, Lname, Pname, flg_net, msg_net);
}
}
}
}
```

```

    }
    catch (Exception ex)
    {
        string EmailFrom;
        string EmailTo;
        EmailFrom = "www@bai.org";
        EmailTo = "www@bai.org, www@bai.org, www@bai.org, www@bai.org";
        SendMessage(EmailFrom, EmailTo, "Exception Thrown in
AutoNetForumWebService Application", ex.ToString(), 0);
    }
    finally
    {
        myConnection.Close();
    }
}

// ** netForum Registration Error Log
static void AddToErrorLog(string E_mail, string FName, string Lname, string Pname
, bool flg_net, string msg_net)
{
    SqlConnection myConnection = new SqlConnection("server=CARLA; database=
AdditionalData; User ID=netForumWebUser; Password=554617");

    SqlCommand myCommand = new SqlCommand("client_bai_ErrorLogNetForumExh_Ins",
myConnection);

    myCommand.CommandType = CommandType.StoredProcedure;

    myCommand.Parameters.Add ( new SqlParameter("@Email", SqlDbType.VarChar) ).
Value = Convert.ToString(E_mail);
    myCommand.Parameters.Add ( new SqlParameter("@FirstName", SqlDbType.VarChar)
).Value = Convert.ToString(FName);
    myCommand.Parameters.Add ( new SqlParameter("@LastName", SqlDbType.VarChar)
).Value = Convert.ToString(Lname);
    myCommand.Parameters.Add ( new SqlParameter("@PlanName", SqlDbType.VarChar)
).Value = Convert.ToString(Pname);
    myCommand.Parameters.Add ( new SqlParameter("@Flag", SqlDbType.Bit) ).Value
= Convert.ToBoolean(flg_net);
    myCommand.Parameters.Add ( new SqlParameter("@ErrorMessage", SqlDbType.
VarChar) ).Value = Convert.ToString(msg_net);
    myCommand.Parameters.Add ( new SqlParameter("@ErrorDate", SqlDbType.DateTime)
).Value = System.DateTime.Now;

    myConnection.Open();

    myCommand.ExecuteNonQuery();

    myConnection.Close();
}

static string SendMessage(string from, string recipients, string subject, string
body, int format)
{
    string retValue = "";

    try
    {
        string[] aryRecipients;
        char[] Separators = new Char[] {',', ';', '\t'};
        aryRecipients = recipients.Split(Separators);
    }
}

```



```
foreach(string recipient in aryRecipients)
{
    if((recipient != "") && (recipient != null))
    {
        retVal = SendMail(from, recipient, subject, body, format);
    }
}
catch(Exception e)
{
    retVal = "Error";
    throw new Exception(e.ToString());
}
return retVal;
}

// SendMail
static string SendMail(string from, string to, string subject, string body, int
format)
{
    string strErrorMessage = null;
    if(ValidateEmail(to))
    {
        try
        {
            MailMessage objMessage = new MailMessage();
            objMessage.From = from;
            objMessage.To = to;
            objMessage.Subject = subject;
            objMessage.Body = body;
            if(format == 0)
            {
                objMessage.BodyFormat = MailFormat.Text;
            }
            else
            {
                objMessage.BodyFormat = MailFormat.Html;
            }
            Smtplib.Send(objMessage);
            return "Sent";
        }
        catch (Exception e)
        {
            if(e.Message == "Could not access cdo_newmail_object")
            {
                strErrorMessage = "Failed. Mail server was not available. Please
try again later.";
            }
            else
            {
                strErrorMessage = "Failed. " + e.ToString() + ".";
            }
            return strErrorMessage;
        }
    }
    else
    {
        strErrorMessage = "Failed. The address was poorly formed.";
        return strErrorMessage;
    }
}
```

```
static bool ValidateEmail(string email)
{
    bool blnValidated = false;
    if(Regex.IsMatch(email, "@"))
    {
        blnValidated = true;
    }
    return blnValidated;
}
```

```
}
}
```