

## Battle.Net

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;
using Microsoft.VisualBasic;

namespace BattleNetUserData
{
    public partial class frmBattleNet : Form
    {
        // this is the database connection string that will be used throughout this
        program
        string sqlConn = @"Data Source=ZERABANE-PC\SQLEXPRESS;Initial
        Catalog=BattleNet;Integrated Security=True";
        SqlConnection dbConn;

        public frmBattleNet()
        {
            InitializeComponent();

            // try to initialise the database
            try
            {
                dbConn = new SqlConnection(sqlConn);
            }
            catch (System.Exception ex)
            {
                MessageBox.Show(ex.Message);
            }
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            // this makes the tabs for diablo and starcraft unusable until a username
            has been searched
            tabDiablo.Enabled = false;
            tabStarcraft.Enabled = false;
        }
        // this button will populate the fields depending on the username typed in
        (use DMoney)
        private void btnSearch_Click(object sender, EventArgs e)
        {
            // the button is disables until search so that you dont encounter an error
            while trying to add a key to nobodys account
            btnAddGameKey.Enabled = true;
            dbConn.Open();
            try
            {
                //this will reun a Select statement that will only show the users data
                that matches the name typed into the search bar
            }
        }
    }
}
```

```

        string sqlUser = @"select * from tblUser where username = '" +
txtUsernameSearch.Text + "'";

        SqlCommand sqlCmd = new SqlCommand(sqlUser, dbConn);
        SqlDataReader reader = sqlCmd.ExecuteReader();
        reader.Read();

        //filling the textboxes with the data from tblUsers
        txtUserBattleTag.Text = reader[0].ToString();
        txtUserDiablo.Text = reader[2].ToString();
        txtUserStarcraft.Text = reader[3].ToString();
        txtUserFriendsCount.Text = reader[4].ToString();
        reader.Close();

        //this will fill in the data for the user if they have diablo 3
        if (txtUserDiablo.Text == "True")
        {
            tabDiablo.Enabled = true;

            string sqlDiablo = @"select u.username, d.diabloID,
d.NumberOfChars ,d.favouriteChar, d.HighestLevel, d.goldAmount, d.clanName from
tblDiablo3 d join tblUser u on u.battleTag = d.battleTag where u.username = '" +
txtUsernameSearch.Text + "'";

            SqlCommand sqlDiabloCmd = new SqlCommand(sqlDiablo, dbConn);
            SqlDataReader readerDiablo = sqlDiabloCmd.ExecuteReader();
            readerDiablo.Read();

            txtDiabloUsername.Text = readerDiablo[0].ToString();
            txtDiabloID.Text = readerDiablo[1].ToString();
            txtDiabloNumChars.Text = readerDiablo[2].ToString();
            txtDiabloFavChar.Text = readerDiablo[3].ToString();
            txtDiabloHighestLevel.Text = readerDiablo[4].ToString();
            txtDiabloGold.Text = readerDiablo[5].ToString();
            txtDiabloClanName.Text = readerDiablo[6].ToString();

            readerDiablo.Close();
        }
        //this will fill in the data for the user if they have starcraft 2
        if (txtUserStarcraft.Text == "True")
        {
            tabStarcraft.Enabled = true;

            string sqlStarcraft = @"select
u.username,s.StarcraftID,s.ExpansionLevel,s.PreferredRace,s.clanName,s.averageApm,s.win
s,s.losses from tblStarcraft2 s join tblUser u on u.battleTag = s.BattleTag where
u.UserName = '" + txtUsernameSearch.Text + "'";

            SqlCommand sqlStarcraftCmd = new SqlCommand(sqlStarcraft, dbConn);
            SqlDataReader readerStarcraft = sqlStarcraftCmd.ExecuteReader();
            readerStarcraft.Read();

            txtStarcraftUsername.Text = readerStarcraft[0].ToString();
            txtStarcraftID.Text = readerStarcraft[1].ToString();
            txtStarcraftExpansion.Text = readerStarcraft[2].ToString();
            txtStarcraftRace.Text = readerStarcraft[3].ToString();
            txtStarcraftClanName.Text = readerStarcraft[4].ToString();
            txtStarcraftAverageapm.Text = readerStarcraft[5].ToString();
            txtStarcraftWins.Text = readerStarcraft[6].ToString();
            txtStarcraftLosses.Text = readerStarcraft[7].ToString();

            readerStarcraft.Close();
        }
    }
}

```

```

    }

}
catch (IndexOutOfRangeException ior)
{
    MessageBox.Show(ior.Message);
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message);
}

dbConn.Close();

}
// this button will add a new player to the database
private void btnAddPlayer_Click(object sender, EventArgs e)
{
    //rand num generator will make a new number for the users battltag and will
    be added to the end of there name to create the battletag
    Random tagGen = new Random();
    int newTag = tagGen.Next(0, 9999);

    string newBattleTag = txtUsernameSearch.Text + "#" + newTag.ToString();
    string sqlAddUser = "insert Into
tblUser(battleTag,Username,diablo3,starcraft2,friendsCount) values ('"+
newBattleTag+"','"+txtUsernameSearch.Text+"','"+"0','"+"0','"+"0"+"')";

    // the iff statement will show all the boxes needed for a new gamekey if
    the user wishes to claim a key
    SqlCommand sqlCmdAddUser = new SqlCommand(sqlAddUser, dbConn);
    if(MessageBox.Show("Would you like to add a game key?", "Claim Game Key",
MessageBoxButtons.YesNo)==DialogResult.Yes)
    {
        lblEnterSerial.Visible = true;
        txtSerialKey.Visible = true;
        btnAcceptInput.Visible = true;
        cmbGameSelect.Visible = true;
    }
    else
    {
    }

    dbConn.Open();

    sqlCmdAddUser.ExecuteNonQuery();

    dbConn.Close();
}
// this will delete an entire user including theyre starcraft and diablo data
if it available
private void btnDeleteUser_Click(object sender, EventArgs e)
{
    string username = txtUsernameSearch.Text;

```

```

        string sqlDeleteUser = "Delete From tblStarcraft2 where battleTag like '"
+username+"%' " + " Delete From tblDiablo3 where battleTag like '" + username+ "%'" + "
Delete from tblUser where username = '" + username + "'";

        SqlCommand sqlCmdDeleteUser = new SqlCommand(sqlDeleteUser, dbConn);

        //after slecting to add a game key it will ask if the user is sure
        dbConn.Open();
        if (MessageBox.Show("Would you like to add a game key?", "Claim Game Key",
        MessageBoxButtons.YesNo) == DialogResult.Yes)
        {
            sqlCmdDeleteUser.ExecuteNonQuery();
        }
        dbConn.Close();
    }
    //will remove a key depending on if the user selects yes or no and also
    deleting related data
    private void btnRemoveKey_Click(object sender, EventArgs e)
    {

        string username = txtUsernameSearch.Text;

        string sqlDiabloDeleteKey = " Delete From tblDiablo3 where battleTag like
'" + username + "%' " + " update tbluser set diablo3 = 0 where battleTag like '" +
username + "%'";

        SqlCommand sqlCmdDiabloDeleteKey = new SqlCommand(sqlDiabloDeleteKey,
        dbConn);

        //will ask the user if theyre sure, if yes it will remove the key
        dbConn.Open();
        if (MessageBox.Show("Would youlike to revoke your game key?", "Revoke Game
Key", MessageBoxButtons.YesNo) == DialogResult.Yes)
        {
            sqlCmdDiabloDeleteKey.ExecuteNonQuery();
            txtDiabloUsername.Clear();
            txtDiabloID.Clear();
            txtDiabloNumChars.Clear();
            txtDiabloFavChar.Clear();
            txtDiabloHighestLevel.Clear();
            txtDiabloGold.Clear();
            txtDiabloClanName.Clear();
        }

        dbConn.Close();
    }
    //this Button will show a basic form f all the users in the database
    private void btnVeiwAllUsers_Click(object sender, EventArgs e)
    {
        // the if is for functionality so that the button is reused to also serve
        as a close button for the report
        if (btnVeiwAllUsers.Text == "Veiw All Users")
        {
            dbConn.Open();
            txtViewUsers.Clear();
            string sqlShowUsers = @"select battleTag, Username, diablo3,
starcraft2, friendsCount from tblUser";

            SqlCommand sqlCmdShowUser = new SqlCommand(sqlShowUsers, dbConn);
            SqlDataReader reader = sqlCmdShowUser.ExecuteReader();

```

```

        txtViewUsers.AppendText(String.Format("{0,-20}{1,-20}{2,-15}{3,-20}{4,-20}\n", "Battle Tag", "Username", "Diablo3", "Starcraft2", "Friends count"));
txtViewUsers.AppendText("=====\n");

        while (reader.Read())
        {
            txtViewUsers.AppendText(String.Format("{0,-20}{1,-20}{2,-20}{3,-20}{4,-20}\n", reader["battleTag"], reader["username"], reader["Diablo3"], reader["Starcraft2"], reader["friendscount"]));
        }

        txtViewUsers.Visible = true;
        btnVeiwAllUsers.Text = "Close Window";

        reader.Close();
    }

    else
    {
        txtViewUsers.Visible = false;
        btnVeiwAllUsers.Text = "Veiw All Users";
    }
    dbConn.Close();
}
//will show a message box to delete the key from the user database and remove all related data
private void btnStarcraftRemoveKey_Click(object sender, EventArgs e)
{
    string username = txtUsernameSearch.Text;

    string sqlstarcraftDeleteKey = " Delete From tblstarcraft2 where battleTag like '" + username + "%' " + " update tbluser set starcraft2 = 0 where battleTag like '" + username + "%'";

    SqlCommand sqlCmdstarcraftDeleteKey = new SqlCommand(sqlstarcraftDeleteKey, dbConn);

    //will ask the user if theyre sure, if yes it will remove the key
    dbConn.Open();
    if (MessageBox.Show("Would youlike to revoke your game key?", "Revoke Game Key", MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
        sqlCmdstarcraftDeleteKey.ExecuteNonQuery();
        txtStarcraftUsername.Clear();
        txtStarcraftID.Clear();
        txtStarcraftExpansion.Clear();
        txtStarcraftRace.Clear();
        txtStarcraftClanName.Clear();
        txtStacraftAverageapm.Clear();
        txtStarcraftWins.Clear();
        txtStarcraftLosses.Clear();
    }
    dbConn.Close();
}
//will ask the user if they want to add a game key or not and then change the data according to the selection
private void btnAddGameKey_Click(object sender, EventArgs e)
{

```

```

        lblEnterSerial.Visible = true;
        txtSerialKey.Visible = true;
        btnAcceptInput.Visible = true;
        cmbGameSelect.Visible = true;
    }
    //accepting input from adding a new gamekey
    private void btnAcceptInput_Click(object sender, EventArgs e)
    {
        string game = " ";
        string table = " ";
        string sqlGame = " ";

        string username = txtUsernameSearch.Text;

        // rand num gen for the players new game id
        Random gameID = new Random();
        long newGameID = gameID.Next(10000000,99999999);
        // the selects are set up in a way that depending on the game the user
        chooses the sql will change to fit that form
        if (cmbGameSelect.Text == "Diablo 3")
        {
            game = "Diablo3";
            table = "tblDiablo3";
            sqlGame = " ('" + txtUserBattleTag.Text + "','" + newGameID + "','0," +
            "'null'" + "','0,0,'null')";
        }

        else if(cmbGameSelect.Text == "Starcraft 2")
        {
            game = "starcraft2";
            table = "tblstarcraft2";
            sqlGame = " ('" + txtUserBattleTag.Text + "','" + newGameID + "','Wings
            of Liberty','null','null',0,0,0)";
        }

        if (txtSerialKey.Text.Length == 13)
        {
            // game will set the the game key in usertable to the desired one and
            give it a positive value it will inser into the "table" than is changed depending on
            the one chosed
            //the final bit of sql is different for each game
            string sqlAddkey = "update tblUser set " + game + " = 1 where username
            like '" + username + "' " + "insert into " + table + " values " + sqlGame;
            SqlCommand sqlCmdAddKey = new SqlCommand(sqlAddkey, dbConn);

            dbConn.Open();

            sqlCmdAddKey.ExecuteNonQuery();

            dbConn.Close();
            txtSerialKey.Clear();
        }
        else
        {
            MessageBox.Show("that is not a valid key please reenter", "Problem?");
            txtSerialKey.Clear();
        }
    }

```

```

    }
    //shows a report depending on the character chose
    private void btnShowChars_Click(object sender, EventArgs e)
    {
        // the if is for functionality so that the button is reused to also serve
        as a close button for the report
        if (btnShowChars.Text == "Show only selected Character")
        {
            dbConn.Open();

            //the sql will only show the players with a favourite char as the user
            has chosen
            txtViewUsers.Clear();
            string sqlShowUser = @"select
battleTag,favouriteChar,Highestlevel,clanName from tblDiablo3 where favouriteChar = '"
+ cmbFavChar.Text + "'";

            SqlCommand sqlCmdShowUser = new SqlCommand(sqlShowUser, dbConn);
            SqlDataReader reader = sqlCmdShowUser.ExecuteReader();

            //this will append the text in the box for a new clear report
            txtViewUsers.AppendText(String.Format("{0,-20}{1,-20}{2,-20}{3,-
20}\n", "BattleTag", "FavouriteChar", "highestLevel", "clanName"));

            txtViewUsers.AppendText("=====\n");

            while (reader.Read())
            {
                txtViewUsers.AppendText(String.Format("{0,-20}{1,-20}{2,-20}{3,-
20}\n", reader["battleTag"], reader["favouriteChar"], reader["HighestLevel"],
reader["clanName"]));
            }

            txtViewUsers.Visible = true;
            btnShowChars.Text = "Close Window";

            reader.Close();
        }

        else
        {
            txtViewUsers.Visible = false;
            btnShowChars.Text = "Show only selected Character";
        }
        dbConn.Close();
    }
    //shows report for users under certain constraints
    private void btnShowReport_Click(object sender, EventArgs e)
    {
        // the if is for functionality so that the button is reused to also serve
        as a close button for the report
        if (btnShowReport.Text == "Show Report")
        {
            dbConn.Open();

            //the sql will show users that have a better win , apm or more losses
            than the user has typed in
            txtViewUsers.Clear();

```

```

        string sqlShowReport = @"select
battleTag,preferredRace,averageapm,wins,losses from tblstarcraft2 where
"+cmbPlayerSort.Text+" > '" + txtStarcraftOrder.Text + "'";

        SqlCommand sqlCmdShowReport = new SqlCommand(sqlShowReport, dbConn);
        SqlDataReader reader = sqlCmdShowReport.ExecuteReader();

        //this will append the text in the box for a new clear report
        txtViewUsers.AppendText(String.Format("{0,-20}{1,-20}{2,-20}{3,-
20}{3,-20}\n", "BattleTag", "Best Race", "Apm,", "Wins", "Losses"));

        txtViewUsers.AppendText("=====\n");

        while (reader.Read())
        {
            txtViewUsers.AppendText(String.Format("{0,-20}{1,-20}{2,-20}{3,-
20}{3,-20}\n", reader["battleTag"], reader["preferredRace"], reader["averageApm"],
reader["Wins"], reader["Losses"]));
        }

        txtViewUsers.Visible = true;
        btnShowReport.Text = "Close Window";

        reader.Close();
    }

    else
    {
        txtViewUsers.Visible = false;
        btnShowReport.Text = "Show Report";
    }
    dbConn.Close();
}

}
}

```