

Forum:[Hack Forums](#)

Thread: [\[TUT\] VB.net - Listview Search Function \(MySql\) \[TUT\]](#)

CPAManager - April 27, 2012, 4:45 am

Listview Search Function (MySql)

This tutorial shows you how to make a search function in your listview control with MySql.

Demonstration on how it works:

Download MySql.Data.dll

[Click here to download the *.dll file!](#)

Now add the dll to your references!

So to start with create a new project and add the following controls:

1 Textbox

1 Combobox

2 Labels

Now make it look like this:

[img]<http://dl.dropbox.com/u/48593307/Billeder/tut1.png>[img]

Add all the rows that you have in your database to the Combobox items.

Note: On my project this search form is one for itself, and then it searches on another form, so you have to make this under the same form as where your listview is!

Below 'Public Class (YourProjectName)' add the following code:

PHP Code

```
<?php
```

```
Dim ds As New DataSet
```

```
    Dim strSQL As String
```

```
    Dim cmd As New MySqlCommand
```

```
    Dim dr As MySqlDataReader
```

```
    Dim a, b, c, d, f, g, h, i, j, k
```

```
    Dim ConnString As String =
```

```
"SERVER = localhost; USERID = root; PASSWORD = root; DATABASE = dbname"
```

```
    Dim CONNECTION As New MySqlConnection(ConnString)
```

Add as many letters as your listview columns has.

After this double click on the textbox, make sure that it handles TextChanged! Now add this code:

PHP Code

```
<?php
```

```
Private Sub TextBox1_TextChanged(sender As System.Object, e As System.EventArgs)  
Handles TextBox1.TextChanged
```

```
    strSQL = "SELECT * from dbname where " & ComboBox1.Text & " like '%" & TextBox1.Text & "%"  
    CONNECTION.Open()  
    cmd = New MySqlCommand(strSQL, CONNECTION)  
    dr = cmd.ExecuteReader()
```

```
    ListView1.Items.Clear()
```

```
    Do While dr.Read()
```

```
        a = (dr.Item("column1").ToString())  
        b = (dr.Item("column2").ToString())  
        c = (dr.Item("column3").ToString())  
        d = (dr.Item("column4").ToString())  
        f = (dr.Item("column5").ToString())  
        g = (dr.Item("column6").ToString())  
        h = (dr.Item("column7").ToString())
```

```
        Dim lv As ListViewItem = ListView1.Items.Add(a)  
        lv.SubItems.Add(b)  
        lv.SubItems.Add(c)  
        lv.SubItems.Add(d)  
        lv.SubItems.Add(f)  
        lv.SubItems.Add(g)  
        lv.SubItems.Add(h)
```

```
    Loop  
    dr.Close()  
    CONNECTION.Close()  
    cmd.Dispose()
```

```
End Sub
```

Edit the column1 and the others to your database column names!

I hope that you learned something from this! ?

Take care!

BlackLayer - April 27, 2012, 5:00 am

Nice coding programmer, try raise new coding ?

CPAManager - April 27, 2012, 5:04 am

(Apr 27, 2012 05:00 AM)Visual Studio Wrote: [\[CITE\]](#) Nice coding programmer, try raise new coding ?
Thanks, Im recording a video now on how it works ?

Fake. - April 27, 2012, 5:09 am

Sorry, but what is this line?

Code Dim a, b, c, d, f, g, h, i, j, k

CPAManager - April 27, 2012, 11:51 am

(Apr 27, 2012 05:09 AM)Fake. Wrote: [\[CITE\]](#) Sorry, but what is this line?

Code Dim a, b, c, d, f, g, h, i, j, k

It's the listview columns, lets say it have 10 columns, then you make 10 diffrent letters. ?

splendidshan - August 29, 2012, 2:47 am

i have a problem with this "%" syntax in my code. what is the other way to run the program without using it. thank you.

0xDEAD10CC - September 1, 2012, 11:38 pm

There's much better ways for adding the SubItems... Use the ListViewItem constructor...

ListViewItem(items As String())

Something like:

Code ListView1.Items.Add(New ListViewItem(Enumerable.Range(1, 7).Select(Function(i)
dr.Item(String.Format("column{0}", i)).ToString).ToArray))

EDIT:

(Apr 27, 2012 05:09 AM)Fake. Wrote: [\[CITE\]](#) Sorry, but what is this line?

Code Dim a, b, c, d, f, g, h, i, j, k

Implicitly typed letter variables lol... Not good.

Occulta - September 1, 2012, 11:40 pm

[img]<http://cdn.memegenerator.net/instances/400x/26062287.jpg>[/img]

How I feel about the OP.

Meerkat.? - September 2, 2012, 12:02 am

Why use MySQL for searching in a listview?

Code Dim FoundItem as ListViewItem = ListView.FindItemWithText()Hurr durr

0xDEAD10CC - September 2, 2012, 12:15 am

(Sep 2, 2012 12:02 AM)Mavamaarten~ Wrote: [\[CITE\]](#) Why use MySQL for searching in a listview?

Code Dim FoundItem as ListViewItem = ListView.FindItemWithText()Hurr durr

You have to be able to add the items back after you clear them out from the search filtering. Unless you want to keep a variable with the full collection of ListViewItems as reference the whole time?

picogenkaku - September 30, 2012, 7:09 am

What's the use of the DataSet when it is not used in the code? :)

mhaanne27 - December 29, 2012, 7:00 am

thank you very much.. ^^

it's helpful..

System[X] - December 29, 2012, 8:34 am

Why use MySql.Data.dll ?

It would be a lot better to simply make a small php file to handle the data sent and received from the mysql database not to mention a lot more secure.

Thanks

vbdotnet09 - December 29, 2012, 11:30 am

nice sharing brother :) keep it up :) this tutorial may help somebody :D

CPAManager - December 29, 2012, 12:16 pm

(Dec 29, 2012 09:34 AM)System[X] Wrote: [\[CITE\]](#) Why use MySql.Data.dll ?

It would be a lot better to simply make a small php file to handle the data sent and received from the mysql database not to mention a lot more secure.

Thanks

Made this along time ago, when i havent learned PHP yet. But now Im using PHP to connect to MySQL databases :)

Xyzo - December 29, 2012, 12:38 pm

thanks for making a tut on this, this is really usefull :)

System[X] - December 29, 2012, 12:38 pm

(Dec 29, 2012 01:16 PM)DaDuck Wrote: [\[CITE\]](#) Made this along time ago, when i havent learned PHP yet. But now Im using PHP to connect to MySQL databases :)
Great :D.

A php file to send and receive data from mysql is pretty easy and loads of examples on google.

Also here's my little tut on mysql to vb.net <https://hackforums.net/showthread.php?tid=2980504>
very easy to follow and understand

Thanks