

Forum:[Hack Forums](#)

Thread: [TreeView Node Tag](#)

twistedweb - August 2, 2018, 6:00 am

For the life of me, I can not figure out how to use the Tag property with Treeview as I did with listview. Then again, there is a possibility I'm just burnt out at this point. To help explain my question, here is how I am currently using the Tag property for listview items.

After retrieval from database I call a method to fill each row, then set tag per item as seen below...

```
Code    private async Task FillRow(MyClass myClass)
        {

            ListViewItem item = await GetItems(myClass);
            this.listview1.Items.Add(item);
        }

private async Task<ListViewItem> GetItems(MyClass myClass)
{
    ListViewItem item = new ListViewItem(new string[] {
        myClass.Name,
        myClass.Description,
        myClass.MoreInfo
    });

    item.Tag = myClass;

    return item;
}
```

In the instance of a listview row being double clicked, I can then launch the editor as such...

```
Code private void OpenEditor()
{
    MyClass myClass = this.listview1.SelectedItems[0].Tag as MyClass;

    if (this.Editor != null && !this.Editor.IsDisposed)
    {
        // additional code for Update
    }
    else
    {
        // additional code for Insert
    }
}
```

Here is how I am currently utilizing the Treeview which has worked fine for 'navigation' up until this point, but now that I'm getting further along, the need to manipulate the potential 'navigation' items have become a necessity.

```
Code private void FillTree()
{
```

```

using (/* sqlconnection and connection string from ConfigurationManager */)
{
    using (SqlCommand cmd = new SqlCommand("MyStoredProcedure", sqlcon))
    {
        cmd.CommandType = CommandType.StoredProcedure;
        using (SqlDataAdapter da = new SqlDataAdapter(cmd))
        {
            DataTable dt = new DataTable();
            da.Fill(dt);

            treeView1.BeginUpdate();
            treeView1.Nodes.Clear();

            PopulateTreeView(treeView1.Nodes, 0, dt);

            treeView1.Select();
            treeView1.EndUpdate();
            // Expand All Nodes
            //treeView1.ExpandAll();
        }
    }
}

```

```

protected void PopulateTreeView(TreeNodeCollection parentNode, int parentID, DataTable dt)
{
    foreach (DataRow dr in dt.Rows)
    {
        if (Convert.ToInt32(dr["parentID"]) == parentID)
        {
            String key = dr["ID"].ToString();
            String text = dr["name"].ToString();

            TreeNodeCollection newParentNode = parentNode.Add(key, text).Nodes;
            PopulateTreeView(newParentNode, Convert.ToInt32(dr["ID"]), dt);
        }
    }
}

```

I guess ultimately my question is, am I able to implement Tag somewhere in here to store these values for an editor down the road? Or am I using the treeview in a terrible manner?

HFDone - August 2, 2018, 7:53 am

```

TreeNodeCollection newParentNode = parentNode.Add(key, text).Nodes;
// HERE YOU CAN LOOP THE NODES AND SET THE TAG
PopulateTreeView(newParentNode, Convert.ToInt32(dr["ID"]), dt);

```