

Replacement of Wiki.Tree to set compatibility with Html standard (replacing pageid tag with standard name tag)

and added security checking to avoid anonymous user authentication with private pages

```
wiki.tree(path : str, depth : num?, reverse : bool? false, pageidtag : string? "pageid", secure: bool? true) :
xml
```

```
var mPars = {
  sPath: __request.args.path ?? args.path ?? null,
  nDepth: __request.args.depth ?? args.depth ?? null,
  bReverse: __request.args.reverse ?? args.reverse ?? false,
  sPageIdTag: __request.args.pageidtag ?? args.pageidtag ?? "pageid",
  bSecure: __request.args.secure ?? args.secure ?? true
};

var xhWikiTree=Wiki.Tree{ path: mPars.sPath, depth: mPars.nDepth,
reverse: mPars.bReverse };
var shWikiTree="";
if (mPars.bSecure)
{
  //var xhWikiTree=WikiTree(sPageTitle);
  //Web.Html("<br />");Web.Text("WikiTree Html: "..xhWikiTree);
  var slWikiTree=Xml.List{doc: xhWikiTree, xpath: "//ul/li", xml: true };
  //Web.Html("<br />");Web.Text("#"..#slWikiTree);
  var shWikiTreeSec="",shNoPerItems="";
  foreach (var sItem in slWikiTree)
  {
    //Web.Text("#"..__index..": ");Web.Pre(Json.Format(sItem));
    //Web.Html("<br />");Web.Text("#"..__index..": "..sItem);
    var sIniTagPageId=" pageid=\"";
    var sEndTagPageId="\"";
    var nIniPosPageId=String.IndexOf(sItem,sIniTagPageId);
    var
sBackPageId=String.SubStr(sItem,nIniPosPageId+String.Length(sIniTagPageId));
    var nEndPosPageId=String.IndexOf(sBackPageId,sEndTagPageId);
    var sItemPageId=String.SubStr(sBackPageId,0,nEndPosPageId);
    var sItemOperSec=Wiki.Template("Template:@himikel/GetPageSecOpers",{
pageid: sItemPageId });
    /*
    var sIniTagTitle=" title=\"";
    var sEndTagTitle="\"";
    var nIniPosTitle=String.IndexOf(sItem,sIniTagTitle);
    var
sBackTitle=String.SubStr(sItem,nIniPosTitle+String.Length(sIniTagTitle));
    var nEndPosTitle=String.IndexOf(sBackTitle,sEndTagTitle);
    var sItemTitle=String.SubStr(sBackTitle,0,nEndPosTitle);
    var sItemOperSec=Wiki.Template("Template:@himikel/GetPageSecOpers",{
title: sItemTitle });
```

```

*/
var bItemBrowse=(String.IndexOf(sItemOperSec,"BROWSE")>=0);
//Web.Html("<br />");Web.Text("item#".."__index..": "..sItemPageId.."
OperSec: "..sItemOperSec.." Browse="..bItemBrowse);
//Web.Html("<br />");Web.Text("title#".."__index..": "..sItemTitle.."
OperSec: "..sItemOperSec.." Browse="..bItemBrowse);
var bInWikiTree=(String.IndexOf(shWikiTreeSec,sItem)>=0);
var bInNoPerItems=(String.IndexOf(shNoPerItems,sItem)>=0);
if (bItemBrowse)
{
  if ((!bInWikiTree)&&(!bInNoPerItems))
  {
    //var mItemPage=Wiki.GetPage(Num.Int(sItemPageId));
    //let shWikiTreeSec..="<li>"..Web.Link{ uri: mItemPage.Uri, text:
mItemPage.Title}.."</li>";
    let shWikiTreeSec..=sItem;
  }//if
}
else
  let shNoPerItems..=sItem;
}//fe
let shWikiTreeSec="<div
class=\"wiki-tree\"><ul>"..shWikiTreeSec.."</ul></div>";
//Web.Html("<br />");Web.Text("WikiTreeSec Html: "..shWikiTreeSec);
//Web.Html(shWikiTreeSec);
let shWikiTree=shWikiTreeSec;
}//if
else
  let shWikiTree=Xml.Text{ doc: xhWikiTree, xml: true };
if (mPars.sPageIdTag!="pageid")
  let shWikiTree=String.Replace(shWikiTree,"pageid",mPars.sPageIdTag);
//Web.Pre(shWikiTree);
Web.Html(shWikiTree);
//Web.Xml(xhWikiTree);

```