

Toggle menu  
Blue Gold Program Wiki

## Navigation

- [Main page](#)
- [Recent changes](#)
- [Random page](#)
- [Help about MediaWiki](#)

## Tools

- [What links here](#)
- [Related changes](#)
- [Special pages](#)
- [Permanent link](#)
- [Page information](#)

## Personal tools

- [Log in](#)

## personal-extra

Toggle search

Search

Random page

## Views

- [View](#)
- [View source](#)
- [History](#)
- [PDF Export](#)

## Actions

# Module:Lua banner

From Blue Gold Program Wiki

The printable version is no longer supported and may have rendering errors. Please update your browser bookmarks and please use the default browser print function instead.

*Documentation for this module may be created at [Module:Lua banner/doc](#)*

```

-- This module implements the {{lua}} template.
local yesno = require('Module:Yesno')
local mList = require('Module:List')
local mTableTools = require('Module:TableTools')
local mMessageBox = require('Module:Message box')

local p = {}

function p.main(frame)
    local origArgs = frame:getParent().args
    local args = {}
    for k, v in pairs(origArgs) do
        v = v:match('^%s*(.)%s*$')
        if v ~= '' then
            args[k] = v
        end
    end
    return p._main(args)
end

function p._main(args)
    local modules = mTableTools.compressSparseArray(args)
    local box = p.renderBox(modules)
    local trackingCategories = p.renderTrackingCategories(args, modules)
    return box .. trackingCategories
end

function p.renderBox(modules)
    local boxArgs = {}
    if #modules < 1 then
        boxArgs.text = '<strong class="error">Error: no modules
specified</strong>'
    else
        local moduleLinks = {}
        for i, module in ipairs(modules) do
            moduleLinks[i] = string.format('[[:%s]]', module)
            local maybeSandbox = mw.title.new(module ..
'/sandbox')
            if maybeSandbox.exists then
                moduleLinks[i] = moduleLinks[i] ..
string.format(' ( [[:%s|sandbox]])', maybeSandbox.fullText)
            end
        end
        local moduleList = mList.makeList('bulleted', moduleLinks)
        local title = mw.title.getCurrentTitle()
        if title.subpageText == "doc" then
            title = title.basePageTitle
        end
        if title.contentModel == "Scribunto" then
            boxArgs.text = 'This module depends on the following
other modules:' .. moduleList
        end
    end
end

```

```

else
    boxArgs.text = 'This template uses
[[Wikipedia:Lua|Lua]]:\n' .. moduleList
end
end
boxArgs.type = 'notice'
boxArgs.small = true
boxArgs.image = '[[File:Lua-logo-nolabel.svg|30px|alt=|link=]]'
return mMessageBox.main('mbox', boxArgs)
end

function p.renderTrackingCategories(args, modules, titleObj)
    if yesno(args.nocat) then
        return ''
    end

    local cats = {}

    -- Error category
    if #modules < 1 then
        cats[#cats + 1] = 'Lua templates with errors'
    end

    -- Lua templates category
    titleObj = titleObj or mw.title.getCurrentTitle()
    local subpageBlacklist = {
        doc = true,
        sandbox = true,
        sandbox2 = true,
        testcases = true
    }
    if not subpageBlacklist[titleObj.subpageText] then
        local protCatName
        if titleObj.namespace == 10 then
            local category = args.category
            if not category then
                local categories = {
                    ['Module:String'] = 'Lua String-based
templates',
                    ['Module:Math'] = 'Templates based on
the Math Lua module',
                    ['Module:BaseConvert'] = 'Templates
based on the BaseConvert Lua module',
                    ['Module:Citation'] = 'Lua-based
citation templates'
                }
                categories['Module:Citation/CS1'] =
categories['Module:Citation']
                category = modules[1] and
categories[modules[1]]
                category = category or 'Lua-based templates'
            end
        end
    end
end

```

```

        end
        cats[#cats + 1] = category
        protCatName = "Templates using under-protected Lua
modules"
    elseif titleObj.namespace == 828 then
        protCatName = "Modules depending on under-protected
modules"
    end
    if not args.noprotcat and protCatName then
        local protLevels = {
            autoconfirmed = 1,
            extendedconfirmed = 2,
            templateeditor = 3,
            sysop = 4
        }
        local currentProt
        if titleObj.id ~= 0 then
            -- id is 0 (page does not exist) if am
            previewing before creating a template.
            currentProt =
            titleObj.protectionLevels["edit"][1]
        end
        if currentProt == nil then currentProt = 0 else
            currentProt = protLevels[currentProt] end
        for i, module in ipairs(modules) do
            if module ~= "WP:libraryUtil" then
                local moduleProt =
                mw.title.new(module).protectionLevels["edit"][1]
                if moduleProt == nil then moduleProt
                = 0 else moduleProt = protLevels[moduleProt] end
                if moduleProt < currentProt then
                    cats[#cats + 1] = protCatName
                    break
                end
            end
        end
    end
end
end
end
end
end
for i, cat in ipairs(cats) do
    cats[i] = string.format('[[Category:%s]]', cat)
end
return table.concat(cats)
end

return p

```

Retrieved from "[https://www.bluegoldwiki.com/index.php?title=Module:Lua\\_banner&oldid=3596](https://www.bluegoldwiki.com/index.php?title=Module:Lua_banner&oldid=3596)"

## Namespaces

- [Module](#)
- [Discussion](#)

## Variants

This page was last edited on 15 November 2020, at 11:23.

## Blue Gold Program Wiki

The wiki version of the Lessons Learnt Report of the Blue Gold program, documents the experiences of a technical assistance (TA) team working in a development project implemented by the Bangladesh Water Development Board (BWDB) and the Department of Agricultural Extension (DAE) over an eight+ year period from March 2013 to December 2021. The wiki lessons learnt report (LLR) is intended to complement the BWDB and DAE project completion reports (PCRs), with the aim of recording lessons learnt for use in the design and implementation of future interventions in the coastal zone.

- [Privacy policy](#)
- [About Blue Gold Program Wiki](#)
- [Disclaimers](#)

Developed and maintained by Big Blue Communications for Blue Gold Program



[Blue Gold Program Wiki](#)