

☐ 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:Shortcut

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:Shortcut/doc](#)

```

-- This module implements {{shortcut}}.

-- Set constants
local CONFIG_MODULE = 'Module:Shortcut/config'

-- Load required modules
local checkType = require('libraryUtil').checkType
local yesno = require('Module:Yesno')

local p = {}

local function message(msg, ...)
    return mw.message.newRawMessage(msg, ...):plain()
end

local function makeCategoryLink(cat)
    return string.format('[[%s:%s]]', mw.site.namespaces[14].name, cat)
end

function p._main(shortcuts, options, frame, cfg)
    checkType('_main', 1, shortcuts, 'table')
    checkType('_main', 2, options, 'table', true)
    options = options or {}
    frame = frame or mw.getCurrentFrame()
    cfg = cfg or mw.loadData(CONFIG_MODULE)
    local isCategorized = yesno(options.category) ~= false

    -- Validate shortcuts
    for i, shortcut in ipairs(shortcuts) do
        if type(shortcut) ~= 'string' or #shortcut < 1 then
            error(message(cfg['invalid-shortcut-error'], i), 2)
        end
    end

    -- Make the list items. These are the shortcuts plus any extra lines
    such
    -- as options.msg.
    local listItems = {}
    for i, shortcut in ipairs(shortcuts) do
        listItems[i] = frame:expandTemplate{
            title = 'No redirect',
            args = {shortcut}
        }
    end
    table.insert(listItems, options.msg)

    -- Return an error if we have nothing to display
    if #listItems < 1 then
        local msg = cfg['no-content-error']
        msg = string.format('<strong class="error">%s</strong>', msg)
        if isCategorized and cfg['no-content-error-category'] then

```

```

        msg = msg .. makeCategoryLink(cfg['no-content-error-
category'])
    end
    return msg
end

local root = mw.html.create()
root:wikitext(frame:extensionTag{ name = 'templatestyles', args = {
src = 'Shortcut/styles.css'} })
-- Anchors
local anchorDiv = root
    :tag('div')
        :addClass('module-shortcutanchordiv')
for i, shortcut in ipairs(shortcuts) do
    local anchor = mw.uri.anchorEncode(shortcut)
    anchorDiv:tag('span'):attr('id', anchor)
end

-- Shortcut heading
local shortcutHeading
do
    local nShortcuts = #shortcuts
    if nShortcuts > 0 then
        shortcutHeading = message(cfg['shortcut-heading'],
nShortcuts)
        shortcutHeading = frame:preprocess(shortcutHeading)
    end
end

-- Shortcut box
local shortcutList = root
    :tag('div')
        :addClass('module-shortcutboxplain plainlist
noprnt')
        :attr('role', 'note')
if shortcutHeading then
    shortcutList
        :tag('div')
            :addClass('module-shortcutlist')
            :wikitext(shortcutHeading)
end
local list = shortcutList:tag('ul')
for i, item in ipairs(listItems) do
    list:tag('li'):wikitext(item)
end
return tostring(root)
end

function p.main(frame)
    local args = require('Module:Arguments').getArgs(frame, {
        wrappers = 'Template:Shortcut'
    })

```

```

    })

    -- Separate shortcuts from options
    local shortcuts, options = {}, {}
    for k, v in pairs(args) do
        if type(k) == 'number' then
            shortcuts[k] = v
        else
            options[k] = v
        end
    end

    -- Compress the shortcut array, which may contain nils.
    local function compressArray(t)
        local nums, ret = {}, {}
        for k in pairs(t) do
            nums[#nums + 1] = k
        end
        table.sort(nums)
        for i, num in ipairs(nums) do
            ret[i] = t[num]
        end
        return ret
    end

    shortcuts = compressArray(shortcuts)

    return p._main(shortcuts, options, frame)
end

return p

```

Retrieved from "<https://www.bluegoldwiki.com/index.php?title=Module:Shortcut&oldid=1812>"

Namespaces

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

Variants

This page was last edited on 19 February 2020, at 09:51.

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](#)