

☐ 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:Main

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:Main/doc](#)

```

--[[
-- This module produces a link to a main article or articles. It implements
the
-- template {{main}}.
--
-- If the module is used in category or category talk space, it produces "The
-- main article for this category is xxx". Otherwise, it produces
-- "Main article: xxx".
--]]

local mHatnote = require('Module:Hatnote')
local mHatlist = require('Module:Hatnote list')
local mArguments -- lazily initialise
local p = {}

function p.main(frame)
    mArguments = require('Module:Arguments')
    local args = mArguments.getArgs(frame, {parentOnly = true})
    local pages = {}
    for k, v in pairs(args) do
        if type(k) == 'number' then
            local display = args['label ' .. k] or args['l' .. k]
            local page = display and
                string.format('%s|%', string.gsub(v, '|.*$', ''))
            pages[#pages + 1] = page
        end
    end
    if #pages == 0 and mw.title.getCurrentTitle().namespace == 0 then
        return mHatnote.makeWikitextError(
            'no page names specified',
            'Template:Main#Errors',
            args.category
        )
    end
    local options = {
        selfref = args.selfref
    }
    return p._main(pages, options)
end

function p._main(args, options)
    -- Get the list of pages. If no first page was specified we use the
current
    -- page name.
    local currentTitle = mw.title.getCurrentTitle()
    if #args == 0 then args = {currentTitle.text} end
    local firstPage = string.gsub(args[1], '|.*$', '')
    -- Find the pagetype.
    local pageType = mHatnote.findNamespaceId(firstPage) == 0 and
'article' or 'page'

```

```

-- Make the formatted link text
list = mHatlist.andList(args, true)
-- Build the text.
local isPlural = #args > 1
local mainForm
local curNs = currentTitle.namespace
if (curNs == 14) or (curNs == 15) then --category/talk namespaces
    mainForm = isPlural and
        'The main %ss for this [[Help:Categories|category]]
are %s'
        or
        'The main %s for this [[Help:Categories|category]] is
%s'
    else
        mainForm = isPlural and 'Main %ss: %s' or 'Main %s: %s'
    end
local text = string.format(mainForm, pageType, list)
-- Process the options and pass the text to the _rellink function in
-- [[Module:Hatnote]].
options = options or {}
local hnOptions = {
    selfref = options.selfref
}
return mHatnote._hatnote(text, hnOptions)
end

return p

```

Retrieved from "<https://www.bluegoldwiki.com/index.php?title=Module:Main&oldid=1798>"

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](#)