

Inhaltsverzeichnis

- [1. Modul:Key/Doku](#)
- [2. Modul:Key](#)

Modul:Key/Doku

Dies ist die Dokumentationsseite für [Modul:Key](#)

[Vorlage:Uses TemplateStyles](#) Implements [Vorlage:TI](#)

Modul:Key

[Vorlage:Uses TemplateStyles](#) Implements [Vorlage:TI](#)

```
-- This module implements {{key press}}.
```

```
local kbdPrefix = '<kbd class="keyboard-key nowrap">'
```

```
local kbdSuffix = '</kbd>'
```

```
local keyText = {
    ['caps lock'] = '⌘ Caps Lock',
    ['[[caps lock]]'] = '⌘ [[Caps Lock]]',
    ['shift'] = '⇧ Shift',
    ['[[shift key|shift]]'] = '⇧ [[Shift key|Shift]]',
    ['enter'] = '↵ Enter',
    ['[[enter key|enter]]'] = '↵ [[Enter key|Enter]]',
    ['cmd'] = '⌘ Cmd',
    ['[[command key|cmd]]'] = '⌘ [[Command key|Cmd]]',
    ['command'] = '⌘ Command',
    ['[[command key|command]]'] = '⌘ [[Command key|Command]]',
    ['opt'] = '⌘ Opt',
    ['[[option key|opt]]'] = '⌘ [[Option key|Opt]]',
    ['option'] = '⌘ Option',
    ['[[option key|option]]'] = '⌘ [[Option key|Option]]',
    ['tab'] = 'Tab ⇧',
    ['[[tab key|tab]]'] = '[[Tab key|Tab]] ⇧',
    ['backspace'] = '← Backspace',
    ['[[backspace]]'] = '← [[Backspace]]',
    ['win'] = '⌘ Win',
    ['[[windows key|win]]'] = '⌘ [[Windows key|Win]]',
    ['menu'] = '☰ Menu',
    ['[[menu key|menu]]'] = '☰ [[Menu key|Menu]]',
```

```

['up'] = '↑',
[['arrow keys|up']] = '[[Arrow keys|↑]]',
['down'] = '↓',
[['arrow keys|down']] = '[[Arrow keys|↓]]',
['left'] = '←',
[['arrow keys|left']] = '[[Arrow keys|←]]',
['right'] = '→',
[['arrow keys|right']] = '[[Arrow keys|→]]',
['asterisk'] = '&#42;',
['hash'] = '&#35;',
[['#']] = '[[Number sign|#]]',
['colon'] = '&#58;',
[[':']] = '[[Colon (punctuation)|:]]',
['pipe'] = '&#124;',
[['|']] = '[[Pipe symbol|&#124;]]',
['semicolon'] = '&#59;',
[[';']] = '[[Semi-colon|&#59;]]',
['equals'] = '&#61;',

```

-- Left & right analog sticks.

```

['l up'] = 'L↑',
['l down'] = 'L↓',
['l left'] = 'L←',
['l right'] = 'L→',
['l ne'] = 'L↗',
['l se'] = 'L↘',
['l nw'] = 'L↖',
['l sw'] = 'L↙',

```

```

['r up'] = 'R↑',
['r down'] = 'R↓',
['r left'] = 'R←',
['r right'] = 'R→',
['r ne'] = 'R↗',
['r se'] = 'R↘',
['r nw'] = 'R↖',
['r sw'] = 'R↙',

```

-- PlayStation.

```

['ex'] = 'x',
['circle'] = '○',
['square'] = '□',
['triangle'] = '△',

```

-- Nintendo 64 and GameCube.

```

['c up'] = 'C↑',
['c down'] = 'C↓',
['c left'] = 'C←',
['c right'] = 'C→',
['c ne'] = 'C↗',
['c se'] = 'C↘',
['c nw'] = 'C↖',

```

```

    ['c sw'] = 'C↙',
}

local keyAlias = {
  -- ['alternate name for key (alias)'] = 'name for key used in key table'
  ['[[cmd key|cmd]]'] = '[[command key|cmd]]',
  ['[[cmd key|command]]'] = '[[command key|command]]',
  ['[[opt key|opt]]'] = '[[option key|opt]]',
  ['[[option key]]'] = '[[option key|option]]',
  ['[[opt key|option]]'] = '[[option key|option]]',
  ['[[win key|win]]'] = '[[windows key|win]]',
  ['*'] = 'asterisk',
  ['#'] = 'hash',
  [':'] = 'colon',
  [';'] = 'semicolon',
  ['l-up'] = 'l up',
  ['l-down'] = 'l down',
  ['l-left'] = 'l left',
  ['l-right'] = 'l right',
  ['l-ne'] = 'l ne',
  ['l-se'] = 'l se',
  ['l-nw'] = 'l nw',
  ['l-sw'] = 'l sw',
  ['r-up'] = 'r up',
  ['r-down'] = 'r down',
  ['r-left'] = 'r left',
  ['r-right'] = 'r right',
  ['r-ne'] = 'r ne',
  ['r-se'] = 'r se',
  ['r-nw'] = 'r nw',
  ['r-sw'] = 'r sw',
  ['ps x'] = 'ex',
  ['ps c'] = 'circle',
  ['ps s'] = 'square',
  ['ps t'] = 'triangle',
  ['c-up'] = 'c up',
  ['c-down'] = 'c down',
  ['c-left'] = 'c left',
  ['c-right'] = 'c right',
  ['c-ne'] = 'c ne',
  ['c-se'] = 'c se',
  ['c-nw'] = 'c nw',
  ['c-sw'] = 'c sw',
}

local Collection = {}
Collection.__index = Collection
do
  function Collection:add(item)
    if item ~= nil then
      self.n = self.n + 1
      self[self.n] = item
    end
  end
end

```

```

        end
    end
    function Collection:join(sep)
        return table.concat(self, sep)
    end
    function Collection:sort(comp)
        table.sort(self, comp)
    end
    function Collection.new()
        return setmetatable({n = 0}, Collection)
    end
end

local function keyPress(args)
    local chainNames = {
        'chain first',
        'chain second',
        'chain third',
        'chain fourth',
        'chain fifth',
        'chain sixth',
        'chain seventh',
        'chain eighth',
        'chain ninth',
    }
    local result = Collection.new()
    local chainDefault = args.chain or '+'
    for i, id in ipairs(args) do
        if i > 1 then
            result:add(args[chainNames[i - 1]] or chainDefault)
        end
        local lc = id:lower()
        local text = keyText[lc] or keyText[keyAlias[lc]] or id
        result:add(kbdPrefix .. text .. kbdSuffix)
    end
    return mw.getCurrentFrame():extensionTag{
        name = 'templatestyles', args = { src = 'Template:Key press/styles.css'}
    } .. result:join()
end

local function keypress(frame)
    -- Called by "{{key press|...}}".
    -- Using the template doubles the post-expand include size.
    return keyPress(frame:getParent().args)
end

local function press(frame)
    -- Called by "{{#invoke:key|press|...}}".
    return keyPress(frame.args)
end

return {

```

```
    keypress = keypress,  
    press = press,  
}
```