



Modul:Unicode data/documentation functions

Die Dokumentation für dieses Modul kann unter *Modul:Unicode data/documentation functions /Doku* erstellt werden

```
local p = {}

local output_mt = {}
function output_mt:insert(str)
    self.n = self.n + 1
    self[self.n] = str
end

function output_mt:insert_format(...)
    self:insert(string.format(...))
end

output_mt.join = table.concat
output_mt.__index = output_mt

local function Output()
    return setmetatable({ n = 0 }, output_mt)
end

function p.show_modules()
    local output = Output()

    output:insert [[

{| class="wikitable" style="text-align: center;"
|+ Character name data modules,<br>organized by first three digits of codepoint

    for i = -1, 0xF do
        if i >= 0 then
            output:insert_format('\n! %X', i)
        else
            output:insert '\n!'
        end
    end

    output:insert '\n|-'

    local prev = -1
    local row
    local found_module = false
    for i = 0, 0x10F do
        local first_two_digits = math.floor(i / 0x10)
        if first_two_digits ~= prev then
            if found_module then
                output:insert(row:join())
            end
            found_module = false

            row = Output{}
            row:insert_format("\n|-\n! %02Xx", first_two_digits)
            prev = first_two_digits
        end
    end
end
```



```
        row:insert '\n| '
        local name_module = ('Module:Unicode data/names/%03X'):format(i)
        if mw.title.new(name_module).exists then
            local first_cp = i * 0x1000
            row:insert_format("[[%s|U+%04X&ndash;<br>U+%04X]]",
                name_module, first_cp, first_cp + 0xFFF)
            found_module = true
        end
    end
    output:insert "\n|}"
    return output:join()
end
return p
```