import $ from 'jquery'

const NAME = 'Layout'

const DATA\_KEY = 'lte.layout'

const JQUERY\_NO\_CONFLICT = $.fn[NAME]

const SELECTOR\_HEADER = '.main-header'

const SELECTOR\_MAIN\_SIDEBAR = '.main-sidebar'

const SELECTOR\_SIDEBAR = '.main-sidebar .sidebar'

const SELECTOR\_CONTENT = '.content-wrapper'

const SELECTOR\_CONTROL\_SIDEBAR\_CONTENT = '.control-sidebar-content'

const SELECTOR\_CONTROL\_SIDEBAR\_BTN = '[data-widget="control-sidebar"]'

const SELECTOR\_FOOTER = '.main-footer'

const SELECTOR\_PUSHMENU\_BTN = '[data-widget="pushmenu"]'

const SELECTOR\_LOGIN\_BOX = '.login-box'

const SELECTOR\_REGISTER\_BOX = '.register-box'

const SELECTOR\_PRELOADER = '.preloader'

const CLASS\_NAME\_SIDEBAR\_COLLAPSED = 'sidebar-collapse'

const CLASS\_NAME\_SIDEBAR\_FOCUSED = 'sidebar-focused'

const CLASS\_NAME\_LAYOUT\_FIXED = 'layout-fixed'

const CLASS\_NAME\_CONTROL\_SIDEBAR\_SLIDE\_OPEN = 'control-sidebar-slide-open'

const CLASS\_NAME\_CONTROL\_SIDEBAR\_OPEN = 'control-sidebar-open'

const CLASS\_NAME\_IFRAME\_MODE = 'iframe-mode'

const Default = {

scrollbarTheme: 'os-theme-light',

scrollbarAutoHide: 'l',

panelAutoHeight: true,

panelAutoHeightMode: 'min-height',

preloadDuration: 200,

loginRegisterAutoHeight: true

}

class Layout {

constructor(element, config) {

this.\_config = config

this.\_element = element

}

// Public

fixLayoutHeight(extra = null) {

const $body = $('body')

let controlSidebar = 0

if ($body.hasClass(CLASS\_NAME\_CONTROL\_SIDEBAR\_SLIDE\_OPEN) || $body.hasClass(CLASS\_NAME\_CONTROL\_SIDEBAR\_OPEN) || extra === 'control\_sidebar') {

controlSidebar = $(SELECTOR\_CONTROL\_SIDEBAR\_CONTENT).outerHeight()

}

const heights = {

window: $(window).height(),

header: $(SELECTOR\_HEADER).length > 0 ? $(SELECTOR\_HEADER).outerHeight() : 0,

footer: $(SELECTOR\_FOOTER).length > 0 ? $(SELECTOR\_FOOTER).outerHeight() : 0,

sidebar: $(SELECTOR\_SIDEBAR).length > 0 ? $(SELECTOR\_SIDEBAR).height() : 0,

controlSidebar

}

const max = this.\_max(heights)

let offset = this.\_config.panelAutoHeight

if (offset === true) {

offset = 0

}

const $contentSelector = $(SELECTOR\_CONTENT)

if (offset !== false) {

if (max === heights.controlSidebar) {

$contentSelector.css(this.\_config.panelAutoHeightMode, (max + offset))

} else if (max === heights.window) {

$contentSelector.css(this.\_config.panelAutoHeightMode, (max + offset) - heights.header - heights.footer)

} else {

$contentSelector.css(this.\_config.panelAutoHeightMode, (max + offset) - heights.header)

}

if (this.\_isFooterFixed()) {

$contentSelector.css(this.\_config.panelAutoHeightMode, parseFloat($contentSelector.css(this.\_config.panelAutoHeightMode)) + heights.footer)

}

}

if (!$body.hasClass(CLASS\_NAME\_LAYOUT\_FIXED)) {

return

}

if (typeof $.fn.overlayScrollbars !== 'undefined') {

$(SELECTOR\_SIDEBAR).overlayScrollbars({

className: this.\_config.scrollbarTheme,

sizeAutoCapable: true,

scrollbars: {

autoHide: this.\_config.scrollbarAutoHide,

clickScrolling: true

}

})

} else {

$(SELECTOR\_SIDEBAR).css('overflow-y', 'auto')

}

}

fixLoginRegisterHeight() {

const $body = $('body')

const $selector = $(`${SELECTOR\_LOGIN\_BOX}, ${SELECTOR\_REGISTER\_BOX}`)

if ($body.hasClass(CLASS\_NAME\_IFRAME\_MODE)) {

$body.css('height', '100%')

$('.wrapper').css('height', '100%')

$('html').css('height', '100%')

} else if ($selector.length === 0) {

$body.css('height', 'auto')

$('html').css('height', 'auto')

} else {

const boxHeight = $selector.height()

if ($body.css(this.\_config.panelAutoHeightMode) !== boxHeight) {

$body.css(this.\_config.panelAutoHeightMode, boxHeight)

}

}

}

// Private

\_init() {

// Activate layout height watcher

this.fixLayoutHeight()

if (this.\_config.loginRegisterAutoHeight === true) {

this.fixLoginRegisterHeight()

} else if (this.\_config.loginRegisterAutoHeight === parseInt(this.\_config.loginRegisterAutoHeight, 10)) {

setInterval(this.fixLoginRegisterHeight, this.\_config.loginRegisterAutoHeight)

}

$(SELECTOR\_SIDEBAR)

.on('collapsed.lte.treeview expanded.lte.treeview', () => {

this.fixLayoutHeight()

})

$(SELECTOR\_MAIN\_SIDEBAR)

.on('mouseenter mouseleave', () => {

if ($('body').hasClass(CLASS\_NAME\_SIDEBAR\_COLLAPSED)) {

this.fixLayoutHeight()

}

})

$(SELECTOR\_PUSHMENU\_BTN)

.on('collapsed.lte.pushmenu shown.lte.pushmenu', () => {

setTimeout(() => {

this.fixLayoutHeight()

}, 300)

})

$(SELECTOR\_CONTROL\_SIDEBAR\_BTN)

.on('collapsed.lte.controlsidebar', () => {

this.fixLayoutHeight()

})

.on('expanded.lte.controlsidebar', () => {

this.fixLayoutHeight('control\_sidebar')

})

$(window).resize(() => {

this.fixLayoutHeight()

})

setTimeout(() => {

$('body.hold-transition').removeClass('hold-transition')

}, 50)

setTimeout(() => {

const $preloader = $(SELECTOR\_PRELOADER)

if ($preloader) {

$preloader.css('height', 0)

setTimeout(() => {

$preloader.children().hide()

}, 200)

}

}, this.\_config.preloadDuration)

}

\_max(numbers) {

// Calculate the maximum number in a list

let max = 0

Object.keys(numbers).forEach(key => {

if (numbers[key] > max) {

max = numbers[key]

}

})

return max

}

\_isFooterFixed() {

return $(SELECTOR\_FOOTER).css('position') === 'fixed'

}

// Static

static \_jQueryInterface(config) {

return this.each(function () {

let data = $(this).data(DATA\_KEY)

const \_config = $.extend({}, Default, typeof config === 'object' ? config : $(this).data())

if (!data) {

data = new Layout($(this), \_config)

$(this).data(DATA\_KEY, data)

data.\_init()

} else if (typeof config === 'string') {

if (typeof data[config] === 'undefined') {

throw new TypeError(`No method named "${config}"`)

}

data[config]()

} else if (typeof config === 'undefined') {

data.\_init()

}

})

}

}

$(window).on('load', () => {

Layout.\_jQueryInterface.call($('body'))

})

$(`${SELECTOR\_SIDEBAR} a`)

.on('focusin', () => {

$(SELECTOR\_MAIN\_SIDEBAR).addClass(CLASS\_NAME\_SIDEBAR\_FOCUSED)

})

.on('focusout', () => {

$(SELECTOR\_MAIN\_SIDEBAR).removeClass(CLASS\_NAME\_SIDEBAR\_FOCUSED)

})

$.fn[NAME] = Layout.\_jQueryInterface

$.fn[NAME].Constructor = Layout

$.fn[NAME].noConflict = function () {

$.fn[NAME] = JQUERY\_NO\_CONFLICT

return Layout.\_jQueryInterface

}

export default Layout

Footer