import $ from 'jquery'

const NAME = 'DirectChat'

const DATA\_KEY = 'lte.directchat'

const EVENT\_KEY = `.${DATA\_KEY}`

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

const EVENT\_TOGGLED = `toggled${EVENT\_KEY}`

const SELECTOR\_DATA\_TOGGLE = '[data-widget="chat-pane-toggle"]'

const SELECTOR\_DIRECT\_CHAT = '.direct-chat'

const CLASS\_NAME\_DIRECT\_CHAT\_OPEN = 'direct-chat-contacts-open'

class DirectChat {

constructor(element) {

this.\_element = element

}

toggle() {

$(this.\_element).parents(SELECTOR\_DIRECT\_CHAT).first().toggleClass(CLASS\_NAME\_DIRECT\_CHAT\_OPEN)

$(this.\_element).trigger($.Event(EVENT\_TOGGLED))

}

// Static

static \_jQueryInterface(config) {

return this.each(function () {

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

if (!data) {

data = new DirectChat($(this))

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

} 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()

}

})

}

}

$(document).on('click', SELECTOR\_DATA\_TOGGLE, function (event) {

if (event) {

event.preventDefault()

}

DirectChat.\_jQueryInterface.call($(this), 'toggle')

})

$.fn[NAME] = DirectChat.\_jQueryInterface

$.fn[NAME].Constructor = DirectChat

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

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

return DirectChat.\_jQueryInterface

}

export default DirectChat

Footer