Front End: Admin.js

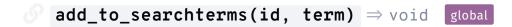
Documentation

O add_to_blacklist(module_id=undefined, SmObjId, PiqYear,
PiqSession, title, searchterm) ⇒ void global

Write table row for module in blacklist, append to blacklist table

Parameters

module_id : Number Default: undefined numerical part of the module CSS selector id, matches DB id SmObjId : Number course catalogue id • PiqYear : Number Required course catalogue year • PiqSession : Number course catalogue year title : String Required module title searchterm : String Required searchterm which found the module in the course catalogue



Write table row for searchterm and add to DOM.

Parameters

• id : Number

• searchterm : String

• term : String searchterm value add_to_suggestions(module_id=undefined, SmObjId, PiqYear, PiqSession, title, whitelisted, searchterm) ⇒ void global Write table row for module in suggestions, append to suggestions table **Parameters** • module_id : Number Default: undefined numerical part of the module CSS selector id, matches DB id • SmObjId : Number course catalogue id • PiqYear : Number course catalogue year • PiqSession : Number Required course catalogue year title : String Required module title • whitelisted : Boolean whitelisted the whitelist status

Required

add_to_whitelist(module_id=undefined, SmObjId, PiqYear, PiqSession, title, searchterm) ⇒ void global

Write table row for module in whitelist, append to whitelist table

Parameters

 \bigcirc checkUpdatedCookie() \Rightarrow void \bigcirc global

Check if the "updated_recently" cookie exists - if it doesn't, updateModules()

searchterm which found the module in the course catalogue

Convert session to human readable span element as string.

Parameters

• session : Number

session code, either 3, 4, 003, 004.

• year : Number

module year data.

Returns:

String — A span containing the humanreadable semester and

year.



delete_blacklisted_module(module_id) ⇒ void global

Delete module from the DB.

Parameters

• module_id : Number

Required

DB id for module to delete



delete_searchterm(id) ⇒ void global

Delete searchterm from the DB.

Parameters

• id : Number

DB id for searchterm to delete

 ${\it o}$ flag_in_suggestions(module_id, whitelisted) \Rightarrow void

Flag module as whitelisted or blacklisted in the suggestions, changing Status and (de)activates corresponding buttons.

Parameters

module_id : Number
 the numerical part of the CSS selector
 whitelisted : Boolean
 Required
 the whitelist status

Get cookie by name

Parameters

• cname: String

Required

name of the cookie

Returns: String — the value of the cookie

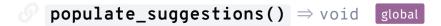
 \bigcirc populate_blacklist() \Rightarrow void global

Request blacklisted modules from server, add them to DOM.

 ${\mathscr O}$ populate_searchterms() \Rightarrow void ${\sf global}$

Request searchterms from server, add them to DOM.

Request studyprograms for selected semester from server, as well as studyprogamid_moduleids list, add them to global JS scope.



Request found modules from server, add them to DOM.

 \bigcirc populate_whitelist() \Rightarrow void \bigcirc global

Request whitelisted modules from server, add them to DOM.

post_module_to_db(module_id=undefined, SmObjId,
PiqYear, PiqSession, whitelisted, searchterm) ⇒ void
global

Flag tablerows for modules OF THE WHITELIST ONLY contained in the studyprogram input via #studyprogram_input.

Parameters

• module_id : Number Default: undefined

numerical part of the module CSS selector id.

• SmObjId : Number Required

course catalogue id.

• PiqYear : Number Required

course catalogue year.

• PiqSession : Number Required

course catalogue year.

whitelist status of the module to save.

• searchterm : String

Required

searchterm which found the module in the course catalogue



Remove searchterm from the DOM.

Parameters

• id : Number

Required

numerical part of CSS selector id for searchterm to delete, matches DB id.

Remove module from the DOM.

Parameters

• module_id : Number

Reauired

the numerical part of the CSS selector, matches DB id.

Save module from SmObjId input into the DB, using the selected semester or current if all selected.



Save searchterm from input into the DB.

SetCookie(cname, cvalue, exdays) ⇒ void global

Save a cookie which expires after exdays days.

Parameters

cname: String
 name of the cookie
 cvalue:
 value of the cookie
 exdays: Number
 days until cookie expires

 ${m \bigcirc}$ update_whitelist_status(module_id, whitelisted) \Rightarrow void ${m \bigcirc}$ global

Flag tablerows for modules OF THE WHITELIST ONLY contained in the studyprogram input via #studyprogram_input.

Parameters

module_id : Number
 numerical part of the module CSS selector id, matches the DB id for saved modules.

• whitelisted : Boolean Required

whitelist status of the module to save.





write_tr_prefix_for_list(module_id=undefined, SmObjId, PiqYear, PiqSession, title, searchterm) ⇒ String global

Generate generalizable part of table row for a given module - what differs are buttons and suffixes.

Parameters

module_id : Number

Default: undefined

numerical part of the module CSS selector id, mismatches DB id if in suggestions

• SmObjId : Number

course catalogue id

PiqYear : Number

Required

course catalogue year

• PiqSession : Number

course catalogue year

• title : String

module title

searchterm : String

searchterm which found the module in the course catalogue

String — String matching an opened tr DOM element, with td **Returns:** elements inside