

COURSE PICKER DOCUMENTATION

QUANQUAN LIU, ELIZABETH SANTORELLA

1. PICKER SCRIPTS AND FUNCTIONS

1.1. **index.js.**

- `onLoad()`: Called to load the home screen of Picker, makes a table of the courses offered by department. List of departments is contained in `config.js` under the variable name "courses."
- `browseCourses()`: Upon clicking the button "Browse Courses" on the homepage, takes list of checked courses on the homepage and redirects page to `browse.html` with department numbers encoded in URI
- `browseIAPCourses()`: Redirects to IAP-browse page, you have to update the courses offered during IAP

1.2. **browse.js.**

- `onLoad()`: Loads Exhibit database and calls `onLoadHelper`.
- `onLoadHelper()`: Called from `onLoad()`. Calls `coursesFromURI()` and gets picked courses by calling `getExhibitSet("picked-classes")` (see `stored-data.js`). Gets the previously picked classes and converts classes to courses and removes duplicate courses. Loads all classes under each selected department and retrieves previously entered data from cookies and load the information in the mini-timegrid.
- (Helper function for `onLoadHelper`) `coursesFromURI()`: Returns list of course departments decoded from URI.
- (Helper function) `savedPickedClass()`: Gets picked classes from cookie and MySQL database, only gets classes from cookie since MySQL retrieval doesn't actually work since Picker does not operate on a MySQL database.
- (Helper function) `classesToCourses(classList)`: Input: list of classes. Returns: List of courses without duplicates. For example, an input of {"6.01", "18.03", "5", "6.042"} would return {"6", "18", "5"}.
- (Helper function) `addCourses(courseIDs, urls)`: Input: list of courses and url and adds each data file containing a different course to the urls. If the course is

inter-departmental, load a separate data file containing interdepartmental courses. Also loads data separately for HASS courses (necessary?). Marks input courses as loaded.

- `onAddMoreSelectChange()`: Controls menu bar "add more courses" in the top left corner of the `browse.html` page.
- `fillAddMoreSelect()`: Creates the drop-down list for `onAddMoreSelectChange()`.

1.3. **stored-data.js.**

- `writeCookie(cookieName, data)`: creates cookie with `cookieName` containing data set to expire in one week (default)
- `updateExhibitSections()`: update cookies
- `updateStoredDataFromExhibit()`: check for cookies

1.4. **class-related.js.**

- `enableClassList()`: Contains blurb to update pre-registration div but only active at certain times in the year.
- `getStoredSections()`: Get user stored information from MySQL database. (Currently not functional).

1.5. **mini-timegrid.js.**

- `updateMiniTimeGrid()`: Updates the mini-timegrid with newly picked sections
- `enableMiniTimegrid()`: Creates and updates the mini-timegrid

1.6. **load-and-process.js.**

- `loadURLs(urls, fDone, individualClasses)`: Loads the classes under each course.
- `processOfficialData(json, individualClasses)`: Load the classes from each course. If a class exists under more than one name (i.e. 6.042/18.042), the class will load under it's "master" name (i.e. 6.042).
- `processOfficialDataItem(item)`: Puts the information for a particular class in the correct format
- `processPrereqs(prereqs)`: Puts a string of prereqs into correct format