

make-syllabus.py

Micah Sherr, 2013
Georgetown University
msherr@cs.georgetown.edu

Released under the GPLv3 License (see LICENSE file)

Summary

make-syllabus.py parses a YAML file that contains a description of a course syllabus and outputs HTML and iCalendar versions of the syllabus. The HTML output is suitable for including in a course webpage; the iCalendar file could be used by students to “subscribe” via Google Calendar, Apple Calendar, or any other calendar system that supports iCalendar subscriptions.

Technical Stuff

make-syllabus.py should run on MacOSX and Linux.

Installation

Do the following:

- `mkdir -p ~/env/makesyllabus`
- `virtualenv ~/env/makesyllabus`
- `source ~/env/makesyllabus/bin/activate`
- `pip install python-dateutil`
- `pip install pyaml`
- `pip install jinja2`
- `pip install icalendar`

Usage

To use make-syllabus, make sure that you have the correct python environment set up. First, do:

```
% source ~/env/makesyllabus/bin/activate
```

Then, for usage instructions, do

```
% python make-syllabus.py -h
```

Current usage

```
usage: make-syllabus.py [-h] [--holidays HOLIDAYS] --start START --end END
                        --days DAYS --schedule SCHEDULE [--template TEMPLATE]
                        [--ical ICAL] [--header HEADER] [--footer FOOTER]
                        [--starttime STARTTIME] [--endtime ENDTIME]
                        [--course COURSE]
```

Syllabus generator

optional arguments:

-h, --help	show this help message and exit
--holidays HOLIDAYS	list of holidays, separated by ":"; use # to define ranges
--start START	start date
--end END	end date
--days DAYS	days of week (0-6, where Monday is 0 and Sunday is 0)
--schedule SCHEDULE	schedule file
--template TEMPLATE	jinja2 template file (must specify either this or --ical)
--ical ICAL	icalendar output file (must specify either this or --template)
--header HEADER	header file to insert before template
--footer FOOTER	footer file to insert after template
--starttime STARTTIME	the time at which the class begins; useful for iCalendar output
--endtime ENDTIME	the time at which the class ends; useful for iCalendar output
--course COURSE	some identifier which will precede the summary in iCalendar output

Examples

Example command-line usage Produce an iCalendar output file for the syllabus:

```
% python make-syllabus.py --start "january 8, 2014" --end "april 28, 2014" \
--days 02 --schedule schedules/cosc755-spring2014.yaml \
--ical cosc755-spring2014.ics --starttime 3:30pm --endtime 4:45pm \
--holidays "1/20/2014:2/17/2014:3/8/2014#3/16/2014:4/17/2014#4/21:2014" \
--course COSC755
```

Note that in the above example, the substring “3/8/2014#3/16/2014” is used to denote the dates between 3/8 and 3/16 (INCLUSIVE).

Produce html output file for the syllabus (writes to STDOUT):

```
% python make-syllabus.py --start "january 8, 2014" --end "april 28, 2014" \
--days 02 --schedule schedules/cosc755-spring2014.yaml \
--starttime 3:30pm --endtime 4:45pm \
--holidays "1/20/2014:2/17/2014:3/8/2014#3/16/2014:4/17/2014#4/21:2014" \
--course COSC755 --header templates/html/header.html \
--footer templates/html/footer.html --template templates/html/syllabus.html
```

Example schedule Schedules are written in Yaml. Each class is written as a top-level Yaml stanza. For example, the following describes three classes:

```
# COSC755: Surveillance and Censorship
# Spring 2014

-
  description: Course introduction
  note: Project assigned
-
  description: Law and policy
  readings:
    -
      title: "The System of Foreign Intelligence Surveillance Law"
      url: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=586616
-
  description: State-sponsored espionage
  readings:
    -
      title: Snowden leaks
    -
      title: "APT1: Exposing One of China's Cyber Espionage Units"
      url: http://intelreport.mandiant.com/Mandiant_APT1_Report.pdf
```

see the `schedules/` directory for additional (and more complete) examples.

Example template The following is an example HTML template. Templates are processed using the jinja2 template processor.

```
<TR>
{% if noclass == True %}
  <TD></TD>
  <TD>{{ lec_date }}</TD>
  <TD COLSPAN="2" align="CENTER"><I>No class</I></TD>
  <TD></TD>
{% else %}
  <TD>{{ lec_num }}</TD>
```

```

<TD>{{ lec_date }}</TD>
<TD>{{ description }}</TD>
<TD>{% for reading in readings %}
    {% if reading.url is defined %}
        <li><a href="{{ reading.url }}">{{ reading.title }}</a></li>
    {% else %}
        <li>{{ reading.title }}</li>
    {% endif %}
{% endfor %}</TD>
<TD>{{ note }}</TD>
{% endif %}
</TR>

```

See the templates/ directory for more complete examples.