Homework 6: Server-side Scripting

1. Objectives

- Get experience with the PHP programming language
- Get experience with the Sunlight Congress APIs
- Get experience using JSON parsers in PHP.

2. Description

In this exercise, you are asked to create a webpage that allows you to search for congress information using the Sunlight Congress APIs, and the results will be displayed in tabular format.

2.1. Description of the Search Form

A user first opens a page, called **congress.php (or any valid web page name)**, where he/she can select a congress database, choose chamber type (i.e., Senate or House) and enter a keyword. Also, the form should include a reference to *Sunlight Foundation* "e.g., Powered by Sunlight Foundation", linking to the web page: http://sunlightfoundation.com/

An example of the page is shown in Figure 1.

Figure 1: Initial Search Screen

Initially a user needs to select the options for Congress Database:

- The congress database can be one of the following options: Legislators, Committees, Bills, and Amendments.
- For each option, when selected, you should replace the text "Keyword*" with the respective text:
  - For Legislators: "State/Representative*"
  - For Committees: "Committee ID*"
  - For Bills: "Bill ID*"
  - For Amendments: "Amendment ID*"

The search form has two buttons:

- **Search** button:
  If the user clicks on the Search button without providing a value for all of the form fields, you should show an error message. The error message **should indicate which fields** are missing. An example of an error message is shown in Figure 2.

Figure 2: An Error Message when there is no input

Once the user provides all selections and required data and clicks on the Search button, your page should send a request to your web server for **congress.php** with the form data. You can use either GET or POST to transfer the form data to the web server. A PHP script will retrieve the data and use it to query the Sunlight Congress API Service.

Figure 3: An Example of Valid Input

- **Clear** button: This button must clear the result area and reset the form fields to their initial state (including the text "Keywords*"). The Clear operation should be done using a JavaScript function.

2.2. Displaying Search Results of Legislators

In this section, we outline how to use the form data to construct HTTP requests to the *Sunlight Congress* restful web service and display the result in the web page.

For the Legislators search, we use the /legislators API. This API returns a list of legislators matching a given input in JSON format. The returned output includes a set of fields for each legislator (e.g., name, state, chamber, phone number, social network).

The PHP script (i.e., **congress.php**) uses the input information (e.g., Chamber is house, State is WA) to construct a restful web service URL to retrieve all congress members matching the query:

```
https://congress.api.sunlightfoundation.com/legislators?chamber=house&state=WA&apikey=
YOUR_API_KEY_HERE
```

The /legislators API URL has "*chamber*" parameter (a.k.a. name) whose value is the one provided from the search form (i.e., senate or house). Another parameter is "*state*" whose value is a two-letter code (uppercase) for the US state (e.g., WA for the state of Washington). In the search form, the user should enter the full name of the state. The PHP script should map the state name into the corresponding uppercase two-letter code (e.g., Washington into WA). The mappings of states to 2-letter codes will be given in the Hints section later.

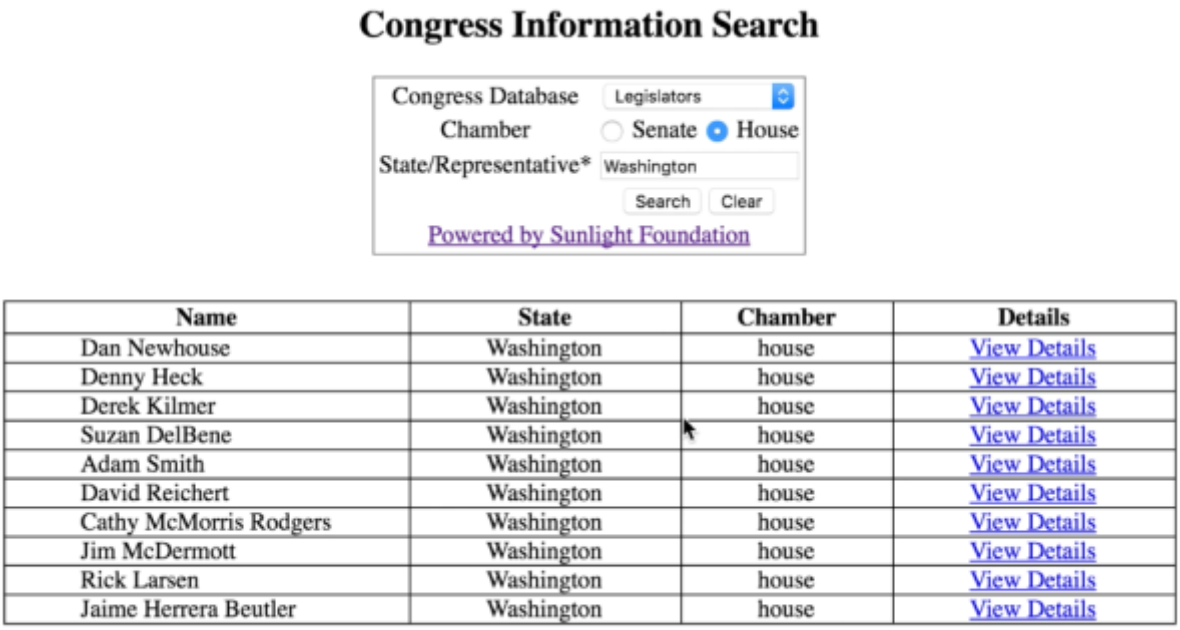
Figure 4 shows an example of the returned response of the /legislators API.

```
{
  "results":[
    {
      "bioguide_id":"N000189",
      "birthday":"1955-07-10",
      "chamber":"house",
      "contact_form":null,
      "crp_id":"N00036403",
      "district":4,
      "facebook_id":"RepNewhouse",
      "fax":"202-225-3251",
      "fec_ids":[
        "H4WA04104"
      ],
      "first_name":"Dan",
      "gender":"M",
      "govtrack_id":"412660",
      "in_office":true,
      "last_name":"Newhouse",
      "leadership_role":null,
      "middle_name":null,
      "name_suffix":null,
      "nickname":null,
      "oc_email":"Rep.Newhouse@opencongress.org",
      "ocd_id":"ocd-division/country:us/state:wa/cd:4",
      "office":"1641 Longworth House Office Building",
      "party":"R",
      "phone":"202-225-5816",
      "state":"WA",
      "state_name":"Washington",
      "term_end":"2017-01-03",
      "term_start":"2015-01-06",
      "thomas_id":"02275",
      "title":"Rep",
      "twitter_id":"RepNewhouse",
      "website":"https://newhouse.house.gov"
    }
  ],
  "count":1,
  "page":{
    "count":1.
    "per_page":20,
    "page":1
  }
}
```


Figure 4: A Sample Response of the /legislators API

The PHP script (i.e., **congress.php**) should parse the returned JSON-formatted object and extract the necessary fields. After extracting the data, the PHP script should display the data in a tabular format below the search form. A sample output is shown in Figure 5. The legislator name, state

and chamber should be displayed in the result table. Also a "Details" column should be listed in this table.



Name	State	Chamber	Details
Dan Newhouse	Washington	house	View Details
Denny Heck	Washington	house	View Details
Derek Kilmer	Washington	house	View Details
Suzan DelBene	Washington	house	View Details
Adam Smith	Washington	house	View Details
David Reichert	Washington	house	View Details
Cathy McMorris Rodgers	Washington	house	View Details
Jim McDermott	Washington	house	View Details
Rick Larsen	Washington	house	View Details
Jaime Herrera Beutler	Washington	house	View Details

Figure 5: An Example of Search Result

If the API service returns an empty result set, the page should display "***The API returned zero results for the request.***" instead of a result table, below the Search form.

Figure 6: Example of Empty result set

When the search result contains at least one record, you need to map the data extracted from the API result to render the result table as follows.

Table 1: Mapping the result from Legislator API into HTML Table

HTML Table Column	API service response
Name	The value of "first_name" "last_name" attributes
State	The value of "state_name" attribute
Chamber	The value of "chamber" attribute
View Details	The link uses "bioguide_id" attribute

The Details column contains a "View Details" link. When clicking on the "View Details" link of a certain legislator, the PHP script (i.e., congress.php) should use the **corresponding** "bioguide_id" value to make another request to the restful web service URL to query detailed information about the selected legislator:

```
https://congress.api.sunlightfoundation.com/legislators?chamber=house&state=WA&bioguide_id=N000189&apikey=YOUR_API_KEY_HERE
```

The PHP script should parse the returned JSON-formatted object and extract some fields. After extracting the data, the PHP script should display the data in a tabular format below the search form. A sample output is shown in Figure 7.

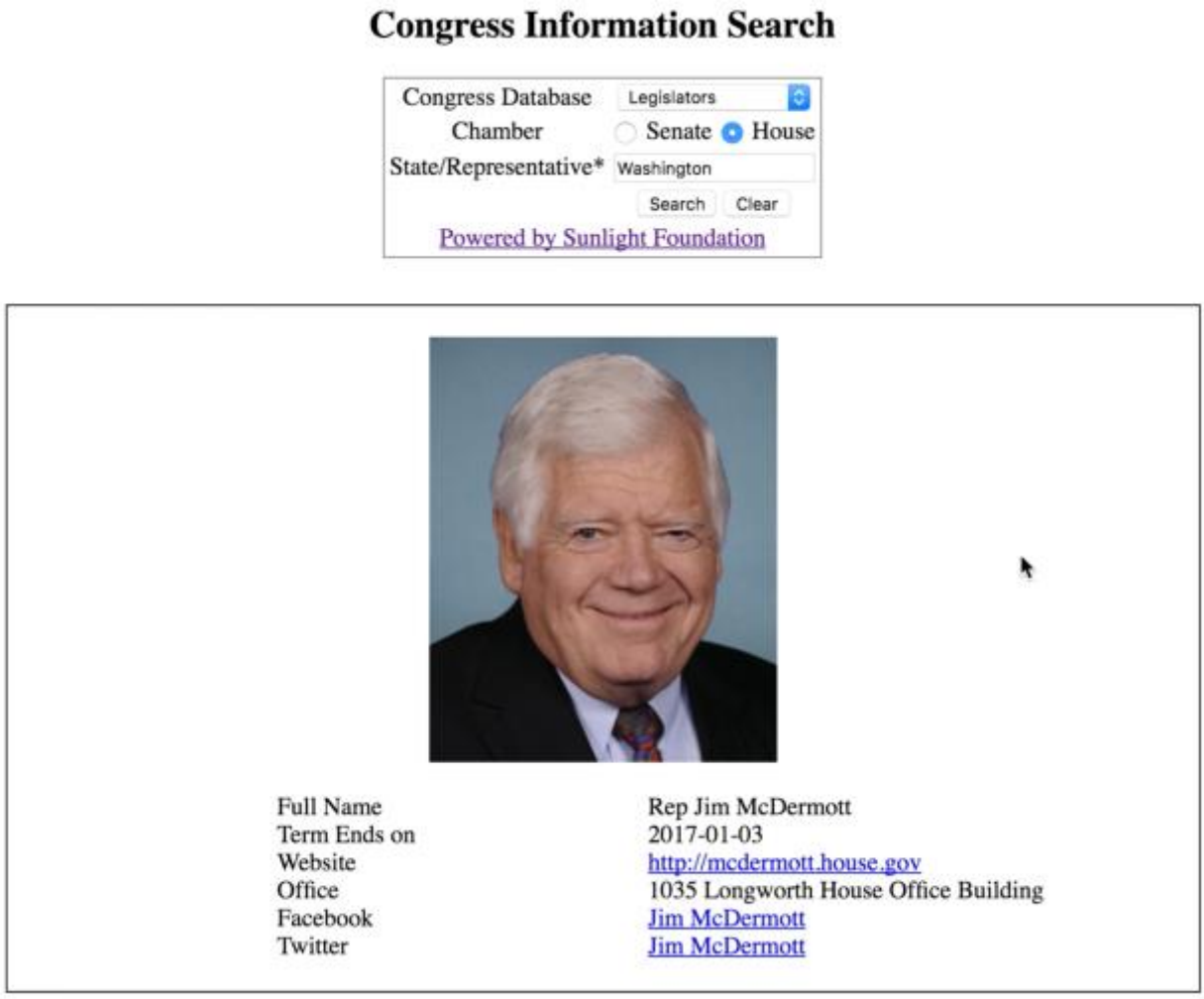


Figure 7: Search Result When clicking the View Details link

The PHP script should map the data retrieved from the API service response to render the legislator information table using the following mapping:

Table 2: Mapping the result from Legislator API into HTML Table

Table Column	**API Service Response**
The legislator image	You should combine the url https://theunitedstates.io/images/congress/225x275/ with the value of "bioguide_id" attribute and the string ".jpg" to get a full url of the image. Set this link as the url of image tag.
Full Name	The value of "title" "first_name" "last_name" attributes
Terms Ends on	The value of "term_end"

Website	The value of "website"
Office	The value of "office"
Facebook	The value of "facebook_id". You should concatenate https://www.facebook.com/ with this value to get a link to this legislator's Facebook page. The displaed text of this link is the legislator's full name.
Twitter	The value of "twitter_id". You should concatenate https://twitter.com/ with this value to get a link to this legislator's Twitter page. The dsiplayed text of this link is the legislator's full name.

If the API service returns an empty result set, the page should display "***The API returned zero results for the request.***" instead of a result table.

In summary, the search mechanism to be implemented behaves as follows:

- Based on the input data in the search form, construct a web service URL to retrieve the output from the API service.
- Parse the returned JSON and extract the values.
- Display the legislator information in tabular format.

Important Note:
Regarding the "State/Representative" field, if the user enters a US state name (instead of a Representative name), then your service URL should be like this:

```
https://congress.api.sunlightfoundation.com/legislators?chamber=house&state=STATE_NAME_HERE&apikey=YOUR_API_KEY_HERE
```

This means that the value provided in the "state/representative" field is considered as a state if it is the full name of a US state. The script should convert the name of the state into a two-letter code. An example of searching legislators by state name is shown as Figure 5.

If the value entered in the "State/Representative" field is not a US state name, it is considered to be a "representative name" and the service URL should be like this:

```
https://congress.api.sunlightfoundation.com/legislators?chamber=house&query=REPRESENTATIVE_NAME_HERE&apikey=YOUR_API_KEY_HERE
```

This means that the value provided in the "state/representative" field is considered as partial the value of a representative name (e.g., first name, last name, or middle name).

An example of searching legislators by representative name is:

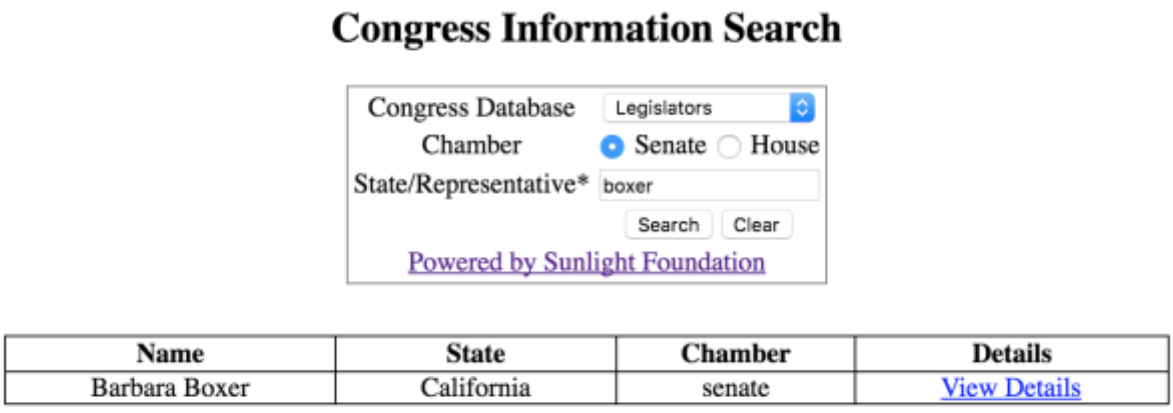


Figure 8: Search Legislators by representative name

2.3. Displaying Search Results of Committees

Figure 9 shows the search form when Committees is selected as the Congress Database.

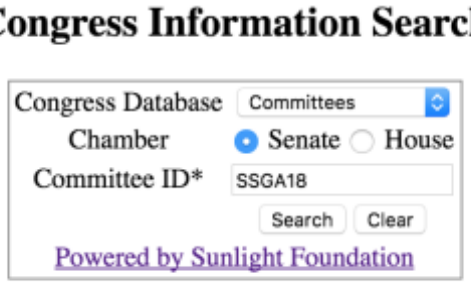


Figure 9: Search form of Committees

You should use the /committees service provided by Sunlight Congress API and construct the service URL as follow based on user input:

```
https://congress.api.sunlightfoundation.com/committees?committee_id=COMMITTEE_ID_HERE&
chamber=CHAMBER_TYPE_HERE&apikey=YOUR_API_KEY_HERE
```

A sample JSON response is:

```
{
    "results": [
        {
            "chamber": "senate",
            "committee_id": "SSGA18",
            "name": "Federal Spending Oversight and Emergency Management",
            "parent_committee_id": "SSGA",
            "subcommittee": true
        }
    ],
    "count": 1,
    "page": {
        "count": 1,
        "per_page": 20,
        "page": 1
    }
}
```


Figure 10: Response of /committees service

The result table should be like this:

Committee ID	Committee Name	Chamber
SSGA18	Federal Spending Oversight and Emergency Management	senate

Figure 11: Result of Committee search

The PHP script should map the data retrieved from the API service response to render the committee information table using the following mapping:

Table 3: Mapping the result from Committee API into HTML Table

Table Column	API Service Response
Committee ID	The value of "committee_id" attribute

Committee Name	The value of "name" attribute
Chamber	The value of "chamber" attribute

If the API service returns an empty result set, the page should display "***The API returned zero results for the request.***" instead of a result table.

2.4. Displaying Search Results of Bills

Figure 12 shows the search form when Bills is selected as the Congress Database.

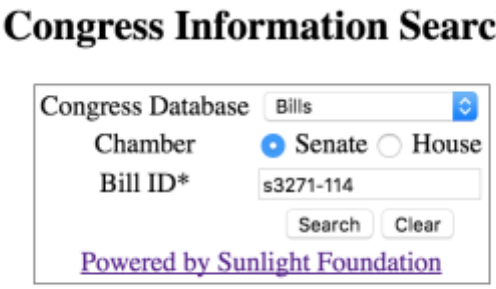


Figure 12: Search form of Bills

You should use the /bills service provided by Sunlight Congress API and construct the service URL as follow based on user input:

```
https://congress.api.sunlightfoundation.com/bills?bill_id=BILL_ID_HERE&chamber=CHAMBER_TYPE_HERE&apikey=YOUR_API_KEY_HERE
```

A sample JSON response is:

```
{ ⊟
   "results":[ ⊟
      { ⊟
         "bill_id":"s3271-114",
         "bill_type":"s",
         "chamber":"senate",
         "committee_ids":[ ⊟
            "SSHR"
         ],
         "congress":114,
         "cosponsors_count":1,
         "enacted_as":null,
         "history":{ ⊞ },
         "introduced_on":"2016-07-14",
         "last_action_at":"2016-07-14",
         "last_version":{ ⊞ },
         "last_version_on":"2016-07-14",
         "last_vote_at":null,
         "number":3271,
         "official_title":"A bill to amend the Carl D. Perkins Career and Techni
         "popular_title":null,
         "related_bill_ids":[ ⊞ ],
         "short_title":"Workforce Advance Act",
         "sponsor":{ ⊞ },
         "sponsor_id":"B001267",
         "urls":{ ⊟
            "congress":"http://beta.congress.gov/bill/114th/senate-bill/3271",
            "govtrack":"https://www.govtrack.us/congress/bills/114/s3271",
            "opencongress":"https://www.opencongress.org/bill/s3271-114"
         },
         "withdrawn_cosponsors_count":0
      }
   ],
   "count":1,
   "page":{ ⊟
      "count":1,
      "per_page":20,
      "page":1
   }
}
```


Figure 13: Response of /bills service

The result table should be like this:

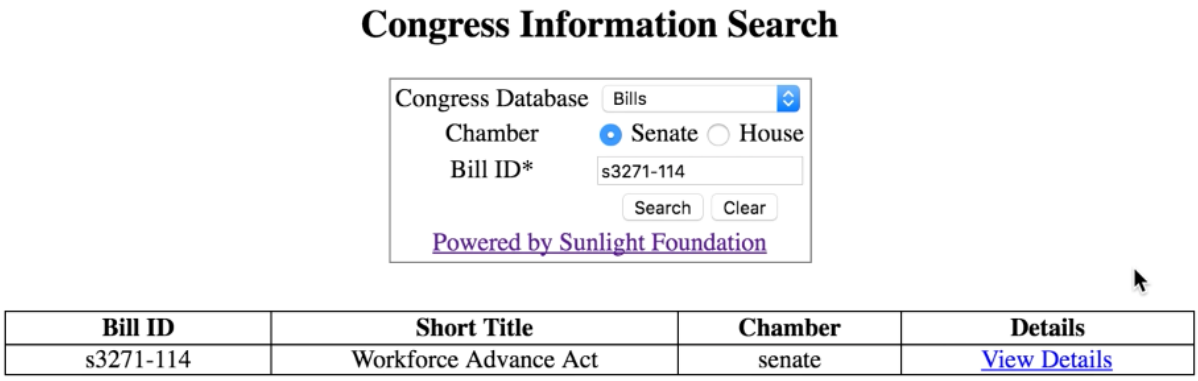


Figure 14: Result of Bill search

If the API service returns an empty result set, the page should display "***The API returned zero results for the request.***" instead of a result table.

The PHP script should map the data retrieved from the API service response to render the bill information table using the following mapping:

Table 4: Mapping the result from Bill API into HTML Table

Table Column	**API Service Response**
Bill ID	The value of "bill_id" attribute
Short Title	The value of "short_title" attribute
Chamber	The value of "chamber" attribute

The Details column contains a "View Details" link. When users click on the "View Details" link of this bill. You should display the detail information of this Bill.

Figure 15: Detail information of Bill search

The PHP script should map the data retrieved from the API service response to render the bill detail information table using the following mapping:

Table 5: Mapping the result from Bill API into HTML Table

Table Column	**API Service Response**
Bill ID	The value of "bill_id" attribute
Bill Title	The value of "short_title" attribute
Sponsor	The value of "title" "first_name" "last_name" attributes under "sponsor" attribute
Introduced On	The value of "introduced_on" attribute
Last action with date	The value of "version_name" attribute under "last_version", the value of "last_action_at" attribute
Bill URL	The value of "pdf" attribute under "urls" attribute under "last_version" attribute. The text of the link should be the value of "short_title" attribute.

2.5. Displaying Search Results of Amendments

Figure 16 shows the search form when Committees is selected as the Congress Database.

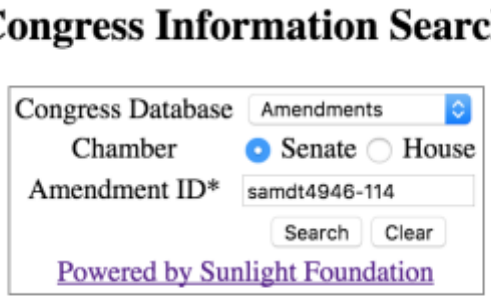


Figure 16: Search form of Committees

You should use the /amendments service provided by Sunlight Congress API and construct the service URL as follow based on user input:

```
https://congress.api.sunlightfoundation.com/amendments?amendment_id=AMENDMENT_ID_HERE&
chamber=CHAMBER_TYPE_HERE&apikey=YOUR_API_KEY_HERE
```

A sample JSON response is:


```
{
  "results":[
    {
      "amendment_id":"samdt4946-114",
      "amendment_type":"samdt",
      "amends_bill_id":"s764-114",
      "chamber":"senate",
      "congress":114,
      "description":null,
      "introduced_on":"2016-06-29",
      "last_action_at":"2016-06-29",
      "number":4946,
      "purpose":null,
      "sponsor_id":"M000355",
      "sponsor_type":"person"
    }
  ],
  "count":1,
  "page":{
    "count":1,
    "per_page":20,
    "page":1
  }
}
```


Figure 17: Response of /committees service

The result table should be like this:

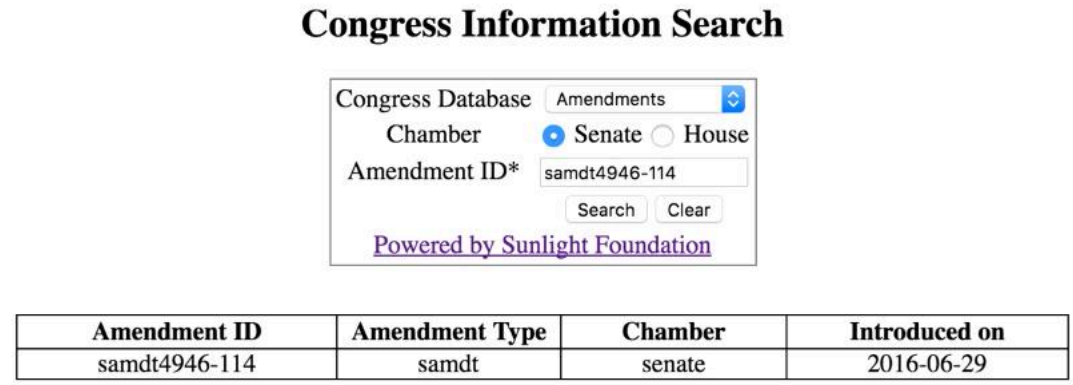


Figure 18: Result of Committee search

The PHP script should map the data retrieved from the API service response to render the committee information table using the following mapping:

Table 6: Mapping the result from Amendments API into HTML Table

Table Column	API Service Response
Amendment ID	The value of "amendment_id" attribute
Amendment Type	The value of "amendment_type" attribute
Chamber	The value of "chamber" attribute
Introduce on	The value of "introduced_on" attribute

If the API service returns an empty result set, the page should display "***The API returned zero results for the request.***" instead of a result table.

2.6. Saving Previous Inputs

In addition to displaying the results, the PHP page should maintain the provided values to display the current result.

For example, if a user searches the Legislator database for senate members in "Washington", the user should see what was provided in the search form when displaying the results. Specifically, when clicking on the "Search" button, the page should display the result retrieved from the API service and keep the value provided in the search form. In addition, when clicking on the "View Details" link, the page should display the result retrieved from the API service and keep the value provided in the search form. It follows that you need to keep the whole search form/input fields and buttons even while displaying results/errors.

3. Hints

3.1. Sunlight Congress API

In order to make requests to Sunlight Congress API, you need to get an API key. Go to the website at http://sunlightfoundation.com/api/, and click "Get a key!", fill the sign up form, and you will get a confirm email. Click the link in the email. Then login at the Sunlight Foundation website and go to your "Profile Settings". You will see your API key string right above you name.

For information about the Sunlight Congress API, please go to:

https://sunlightlabs.github.io/congress/
and
http://tryit.sunlightfoundation.com/congress

3.2. Parsing JSON-formatted data in PHP

In PHP 5, you can parse JSON-formatted data using the "*json_decode*" function. For more information, please go to http://php.net/manual/en/function.json-decode.php.

To read the contents of a JSON-formatted object, you can use the "*file_get_contents*" function.

3.3 List of US States and Their Two-Letter Abbreviations

Table 7: Mapping the State Full Name into Two-Letter Code

Two-Letter Abbreviation	State	Two-Letter Abbreviation	State
AL	Alabama	MT	Montana
AK	Alaska	NE	Nebraska
AZ	Arizona	NV	Nevada
AR	Arkansas	NH	New Hampshire
CA	California	NJ	New Jersey
CO	Colorado	NM	New Mexico
CT	Connecticut	NY	New York
DE	Delaware	NC	North Carolina
DC	District Of Columbia	ND	North Dakota

FL	Florida	OH	Ohio
GA	Georgia	OK	Oklahoma
HI	Hawaii	OR	Oregon
ID	Idaho	PA	Pennsylvania
IL	Illinois	RI	Rhode Island
IN	Indiana	SC	South Carolina
IA	Iowa	SD	South Dakota
KS	Kansas	TN	Tennessee
KY	Kentucky	TX	Texas
LA	Louisiana	UT	Utah
ME	Maine	VT	Vermont
MD	Maryland	VA	Virginia
MA	Massachusetts	WA	Washington
MI	Michigan	WV	West Virginia
MN	Minnesota	WI	Wisconsin
MS	Mississippi	WY	Wyoming
MO	Missouri		

4. Files to Submit

In your course homework page, you should update the **HW6 link** to refer to your new initial web page for this exercise. Also, submit your file (A single .php file named congress.php) electronically to the csci571 account so that they can be graded and compared to all other students' code via the MOSS code comparison tool.

****IMPORTANT**:**
All discussions and explanations in Piazza related to this homework are part of the homework description. So please review all Piazza threads before finishing the assignment.