The API to query all of the legislators is:

```
https://congress.api.sunlightfoundation.com/legislators?apikey=YOUR_API_KEY_HERE&per_page=all
```

The response is as follows:

```
{
  - results: [
    - {
        bioguide_id: "D000626",
        birthday: "1970-03-01",
        chamber: "house",
        contact_form: null,
        crp_id: "N00038767",
        district: 8,
        fax: null,
      - fec_ids: [
          "H6OH08315"
        ],
        first_name: "Warren",
        gender: "M",
        govtrack_id: "412675",
        in_office: true,
        last_name: "Davidson",
        leadership_role: null,
        middle_name: null,
        name_suffix: null,
        nickname: null,
        oc_email: null,
        ocd_id: "ocd-division/country:us/state:oh/cd:8",
        office: "1011 Longworth House Office Building",
        party: "R",
        phone: "202-225-6205",
        state: "OH",
        state_name: "Ohio",
        term_end: "2017-01-03",
        term_start: "2016-06-09",
        thomas_id: "02296",
        title: "Rep",
        votesmart_id: 166760,
        website: null
      },
    - {
        bioguide_id: "L000585",
        birthday: "1968-07-05",
```

Figure 22

4.2 Bills API

The API to query the first page of 50 Bills is:

```
https://congress.api.sunlightfoundation.com/bills?apikey=YOUR_API_KEY_HERE&per_page=50
```

The response is as follows:

```
{
 - results: [
    - {
         bill_id: "hr5929-114",
         bill_type: "hr",
         chamber: "house",
       - committee_ids: [
            "HSED"
         ],
         congress: 114,
         cosponsors_count: 0,
         enacted_as: null,
       - history: {
            active: false,
            awaiting_signature: false,
            enacted: false,
            vetoed: false
         },
         introduced_on: "2016-07-25",
         last_action_at: "2016-07-25",
       - last_version: {
            version_code: "ih",
            issued_on: "2016-07-25",
            version_name: "Introduced in House",
            bill_version_id: "hr5929-114-ih",
          - urls: {
               html: "https://www.gpo.gov/fdsys/pkg/BILLS-114
               pdf: "https://www.gpo.gov/fdsys/pkg/BILLS-114h
               xml: "https://www.gpo.gov/fdsys/pkg/BILLS-114h
            },
            pages: 2
         },
         last_version_on: "2016-07-25",
         last_vote_at: null,
         number: 5929,
         official_title: "To amend the Occupational Safety and
         for employees.",
         popular_title: null,
       - related_bill_ids: [
            "hr5915-114"
         ],
         short_title: "Menstrual Products for Employees Act",
       - sponsor: {
            first_name: "Grace",
            last_name: "Meng",
            middle_name: null,
            name_suffix: null,
            nickname: null,
```

Figure 23

4.3 Committees API

The API to query the first page of all committees is:

```
https://congress.api.sunlightfoundation.com/committees?apikey=YOUR_API_KEY_HERE&per_page=all
```

The response is as follows:

```
{
  - results: [
      - {
            chamber: "senate",
            committee_id: "SSGA18",
            name: "Federal Spending Oversight and Emergency Management",
            parent_committee_id: "SSGA",
            subcommittee: true
        },
      - {
            chamber: "senate",
            committee_id: "SSGA15",
            name: "Financial and Contracting Oversight",
            parent_committee_id: "SSGA",
            subcommittee: true
        },
      - {
            chamber: "senate",
            committee_id: "SSGA17",
            name: "Emergency Management, Intergovernmental Relations, and the D:
            parent_committee_id: "SSGA",
            subcommittee: true
        },
      - {
            chamber: "senate",
            committee_id: "SSGA01",
            name: "Permanent Subcommittee on Investigations",
            parent_committee_id: "SSGA",
            subcommittee: true
        },
      - {
            chamber: "senate",
            committee_id: "SSFR14",
            name: "State Department and USAID Management, International Operatio
            parent_committee_id: "SSFR",
            subcommittee: true
        },
```

Figure 24

5. Hints

5.1 Images

The images for this homework have been provide in http://cs-server.usc.edu:45678/hw/hw8/images/

5.2 Get started with the Bootstrap Library

To know how to get started with Bootstrap, please refer to the page at http://getbootstrap.com/getting-started/. You need to import the necessary CSS file and JS file provided by Bootstrap.

5.3 Bootstrap UI Components

Bootstrap provides a complete mechanism to make Web pages responsive to different mobile devices. In this exercise you will get hands-on experience with responsive design using the Bootstrap Grid System.

At a minimum, you will need to use Bootstrap Form, Tab, Wells, Carousel and Glyphicons to implement the required functionality. Information about these components can be found here:

Bootstrap Form   http://getbootstrap.com/css/ - forms

Bootstrap Tabs	http://getbootstrap.com/javascript/ - tabs
Bootstrap Wells	http://getbootstrap.com/components/ - wells
Bootstrap Carousel	http://getbootstrap.com/javascript/ - carousel
Bootstrap Glyphicons	http://getbootstrap.com/components/ - glyphicons

5.4 Google App Engine/Amazon Web Services

You should use the domain name of the Google App Engine/Amazon Web Services you created in HW#7 to make the request. For example, if you're GAE/AWS server domain is called example.appspot.com or example.elasticbeanstalk.com, and the user performs a GET request with operation name="legislator", then a query of the following type will be generated:

(GAE) - http://example.appspot.com/?operation=legislators

(AWS) - http://example.elasticbeanstalk.com/?operation=legislators

5.5 AJAX call

You can send the request to the PHP script by passing the URL to `$.ajax()`. You must use a **GET** method to request the resource since you are required to provide this link to your homework list to let graders check whether the PHP code is running on Google GAE or AWS (please refer to the grading guideline for details).

The is the JQuery AJAX call:

```
$.ajax({
  url: 'URL in GAE',
  //parameter list
  data: {symbol:"AAPL"},
  type: 'GET',
  success: function(response,status,xhr){
      //parse the output
  },
  error: function(xhr, status, error){
      //parse the error
  }
});
```

Figure 25 – JQuery AJAX call

The is the AngularJS AJAX call:

```
// Simple GET request example:
$http({
  method: 'GET',
  url: '/someUrl'
}).then(function successCallback(response) {
    // this callback will be called asynchronously
    // when the response is available
  }, function errorCallback(response) {
    // called asynchronously if an error occurs
    // or server returns response with an error status.
  });
```

Figure 26 – AngularJS AJAX call

5.6. HTML5 Local Storage

Local storage is more secure, and large amounts of data can be stored locally, without affecting website performance. Unlike cookies, the storage limit is far larger (at least 5MB) and information is never transferred to the server. There are two method, getItem() and setItem(), that can be used for this purpose. The local storage could only store strings. So you need to convert the data to strings before storing it in the local storage. See:

https://developer.mozilla.org/en-US/docs/Web/API/Window/localStorage

http://www.w3schools.com/html/html5_webstorage.asp

5.7 Get Started with AngularJS

AngularJS extends HTML with new attributes.

AngularJS is perfect for Single Page Applications (SPAs).

AngularJS is easy to learn.

To get started with AngularJS. The W3SCHOOL site has easy tutorials:

http://www.w3schools.com/angular/

Or, you can follow the tutorials at the official website:

https://docs.angularjs.org/tutorial

5.8 AngularJS Pagination and Filters

The Pagination Directive of AngularJS is truly plug-n-play - no need to do any set-up or logic in your controller. Just add an attribute, drop in your navigation wherever you like, and boom - instant, full-featured pagination.

The Github page is at:

https://github.com/michaelbromley/angularUtils/tree/master/src/directives/pagination

The Demo page is at:

http://plnkr.co/edit/Wtkv71LIqUR4OhzhgpqL?p=preview

5.9 AngularJS Progress Bar

The AngularJS Progress Bar Directive allows you to auto compute the progress with a simple set up.

The documentation is located at:

https://angular-ui.github.io/bootstrap/ - /progressbar

5.10 External Libraries

This is the list of external libraries that you may also want to use:

*   JQuery – https://code.jquery.com/
*   Moment JS - http://momentjs.com/ for time conversion

6. FAQ's

Q1. Which server should I use, where would server be hosted?

You can use either host it in Google App Engine or Amazon Web Services. While grading we just need to see the data

Q2. Can I use any other way to store data?

You are allowed only to use browsers local storage. No other way of storing data like cookies, session storage is allowed.

Q3. How long should the local storage be persistent?

The local storage data should be persistent until the browser is closed. Even if the page is refreshed, the local storage data should not be deleted.

Q4. Is it required to use Bootstrap Carousel?

Yes. It is mandatory to use bootstrap carousel.

Q5. Facing issues with Cross Origin Request?

There are several ways to avoid cross origin issues. You can use CORS. If you like to use $.ajax() you can use datatype: jsonp with callback function appended with the URL.