

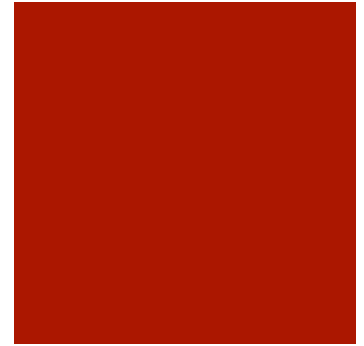


Walkthrough 8

Marta Bralic

Agenda

- Import
- Controls
- Search
- News



Import

- Top of file
 - `#!/usr/bin/php`
 - `require constants.php`
- go to phpMyAdmin and make a table
 - `mysql_connect`
 - `mysql_select_db`



CSV

- `$fp`
 - fopen like in pset5 and pset6
- skip first row of file
 - `fgetcsv($fp)`
- store data via while loop
 - `$data = fgetcsv($fp)`
 - `$data[0]` is zipcode
 - `$data[1]` is whether it is primary
 - `$data[2]` is population
- INSERT into your table
- `fclose($fp)`

Controls

- In load function
 - browse the API to add controls
 - <http://code.google.com/apis/maps/documentation/introduction.html>
 - <http://code.google.com/apis/maps/documentation/controls.html>
 - `map.addControl()`



Search

- form in index.html

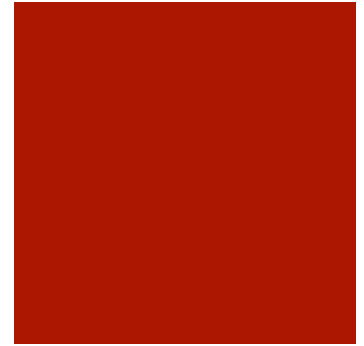
```
<form action="" method="get" onsubmit="go(); return  
false;">
```

```
  <input autocomplete="off" id="q" size="50"  
    type="text" />
```

```
  <input type="submit" value="Go" />
```

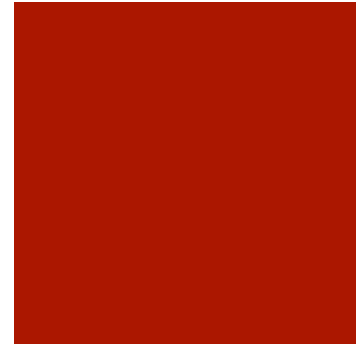
```
</form>
```

go()



- Example geocoding from API
 - Their “address” is our “q”, but ours is an id
 - `document.getElementById("q").value` is our version of “address”
 - lambda function that takes “point”
 - returned by geocoder
 - `if (!point)`
 - `alert!`
 - `else`
 - `map.setCenter(point, 13);`

Send to Proxy



- must have SW and NE coordinates first
 - `var sw = bounds.getSouthWest();`
 - `var ne = bounds.getNorthEast();`
- look at ajax examples from lecture
 - prepare request
 - prepare URL
 - `var url = "proxy.php?sw=" + sw.lat() + "," + sw.lng() + "&ne=" + [...]`
 - send request
 - `request.open("GET", url, true);`

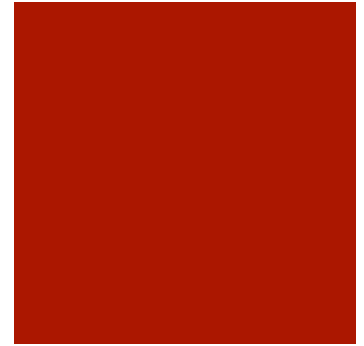
Proxy News

- <http://news.google.com/news?geo=02138&output=rss>
 - 10 articles about 02138
- proxy.php
 - takes coordinates of SW and NE
 - returns JSON (JavaScript Object Notation)
 - cities is an array – 5 biggest cities
 - articles is an array within each city – 10 for each



Get Response

- `var cities = eval("(" + request.responseText + ")");`
 - will evaluate this string as js
- `for (var i = 0; i < cities.length; i++)`
 - `for (var j = 0; j < cities[i].articles.length; j++)`
 - `cities[i].articles[j].link`
 - `cities[i].articles[j].title`



Creating the News Bubble



- addMarker takes a point and xhtml
 - point will be where to put the marker
 - new GLatLng
 - cities[i].Latitude
 - cities[i].Longitude
 - xhtml will be what shows up in the bubble

```
var xhtml = "<b>This is CS 50.</b>";  
xhtml += "<br /><br />";
```

 - also add a link