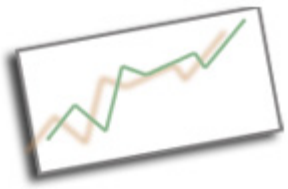

Advanced Online Media
Dr. Cindy Royal
Texas State University - San Marcos
School of Journalism and Mass Communication

Google Maps API

The Google Maps API (application programming interface) gives you a great deal of flexibility in creating your own maps applications. Developers use it for doing mashups with other applications and data. Applications are hosted on your own server.

1. Go to http://code.google.com/apis/maps/signup.html and sign up for a key. You will have to be logged in to a Google acct.
2. The key you generate will be good for a specific domain, so sign up to use it on your own domain. You can sign up for another later, if you move to another domain.
3. Make a note of your key. Put it in a text file from which you can easily copy and paste. That's what you will use to make the Google Maps API work for you.
4. Go to the Developer's Guide and look at the Basics. http://code.google.com/apis/maps/documentation/introduction.html
5. Copy the code under the "Hello World" of Google Maps and paste it into a text editor. You will need to then change the "key" to your own key, removing "abcdefg". For the sensor variable, make it false for now.
6. Save the file as an html document. You can work on it and see the changes locally, but when you launch it to the Web it must be on the Web server associated with your API Key.
7. Open the file in a browser. You should see a Map of Palo Alto, CA.
   Now, find the area in the code where it puts the map center code.
   map.setCenter(new GLatLng(37.4419, -122.1419), 13);
   Change the LatLong numbers to 30.30. -97.75 – this will give you an Austin map
   Change it to 29.88, -97.94 – That's San Marcos
   Play around with the numbers a bit to get comfortable with the longitude, latitude figures.
8. Add the code below to the bottom of the initialize function:
   map.openInfoWindow(map.getCenter(),
   document.createTextNode("Hello, San Marcos"));

   put in any City you want in the example.
9. Now we are going to add markers to the map. Your initialize function should look like this:
   function initialize() {

```
        if (GBrowserIsCompatible()) {
          var map = new
      GMap2(document.getElementById("map_canvas"));
          map.setCenter(new GLatLng(29.88, -97.94), 13);
          map.setUIToDefault();
}
```

Make sure it is closed with curly brace

10. Then add a function to create the markers:

```
    function createMarker(point,html) {
          var marker = new GMarker(point);
          GEvent.addListener(marker, "click", function() {
            marker.openInfoWindowHtml(html);
          });
          return marker;
        }
```

11. For each marker, include this code:

```
     var point = new GLatLng(29.88, -97.94);
        var marker = createMarker(point,'Some stuff to display in the First
    Info Window.')
        map.addOverlay(marker);
```

The script area of your file should look like this. Go to http://cindyroyal.com/advanced/maps/test_map.html to see the working version.

```
     <script type="text/javascript">

       function initialize() {
        if (GBrowserIsCompatible()) {
          var map = new
      GMap2(document.getElementById("map_canvas"));
          map.setCenter(new GLatLng(29.88, -97.94), 13);
          map.setUIToDefault();
      }

      function createMarker(point,html) {
          var marker = new GMarker(point);
          GEvent.addListener(marker, "click", function() {
            marker.openInfoWindowHtml(html);
          });
          return marker;
        }
```

```
// Set up three markers with info windows

        var point = new GLatLng(29.88, -97.94);
      var marker = createMarker(point,'Some stuff to display in the First
Info Window.')
      map.addOverlay(marker);

      var point = new GLatLng(29.8850,-97.9425);
      var marker = createMarker(point,'Some stuff to display in the
Second Info Window')
      map.addOverlay(marker);

      var point = new GLatLng(29.8825,-97.9450);
      var marker = createMarker(point,'Some stuff to display in the
Third Info Window')
      map.addOverlay(marker);



    }


   </script>
```