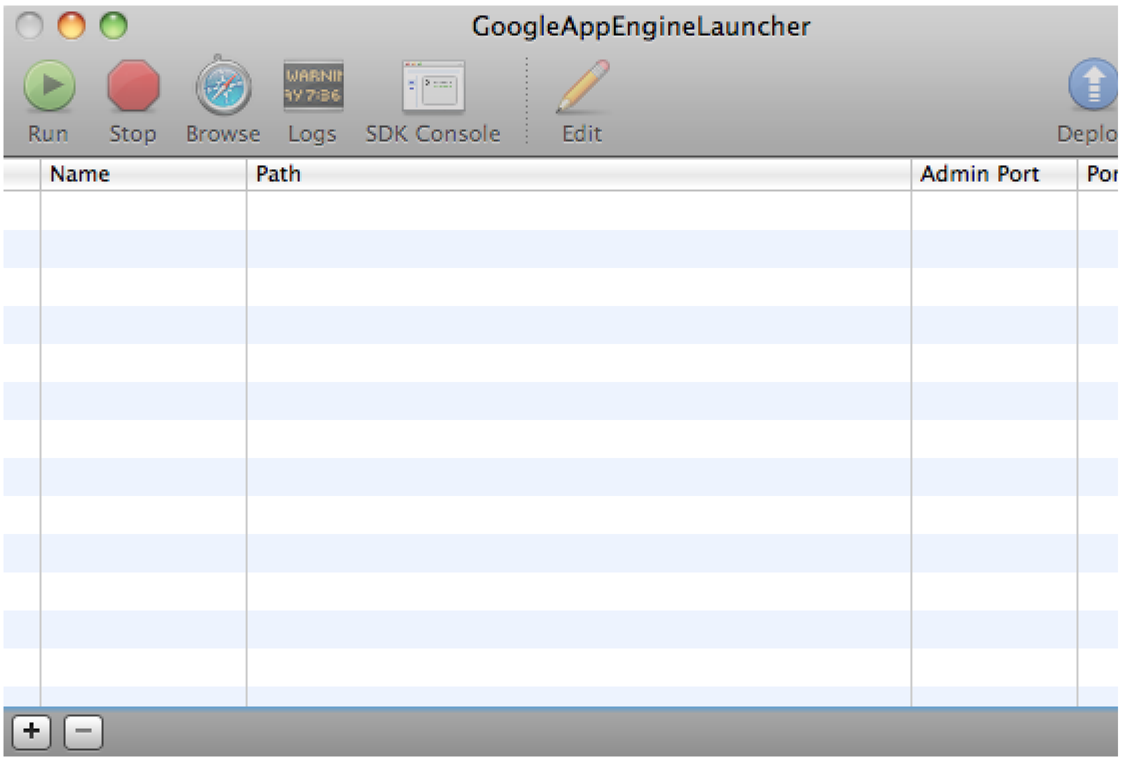
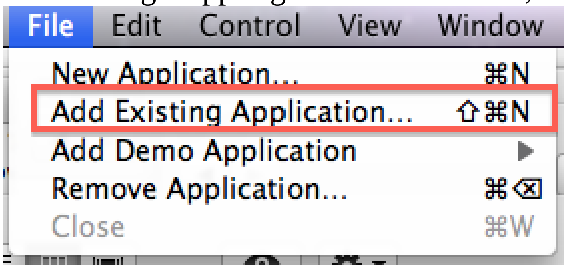
Creating a Custom TinyWebDB service

1. TinyWebDB is an App Inventor component that allows you to store data persistently in a database on the web. Because the data is stored on the web instead of a particular device, TinyWebDB can be used to facilitate communication between devices and apps (recommendation systems, multi-player games, etc.).
2. By default, the TinyWebDB component stores data on a test service provided by App Inventor, http://appinvtinywebdb.appspot.com/. This service is helpful for testing, but it is shared by all App Inventor users, and it has a limit of 1000 entries. If you use it, your data will be overwritten eventually.
3. For most apps you write, you'll want to create a custom web service that isn't shared with other App Inventor apps and programmers. You don't need to be a programmer to do so --- just follow the instructions below and you'll be able to make your own web service.
4. To create your own web service, follow these instructions:
5. Download and install Windows x86-64 MSI installer https://www.python.org/downloads/release/python-2714/
6. Download the installer SDK for App Engine for Python (standard enviroment) from https://cloud.google.com/appengine/downloads. When you visit this page, you will see several buttons; click on the one that says Google App Engine SDK for Python.
7. Download and install App Engine SDK for Python: https://storage.googleapis.com/appengine-sdks/featured/GoogleAppEngine-1.9.60.msi
8. Run Google APP Engine app and set Edit/Preference menu all path
9. Review the page describing two TinyWeb DB services and download the .zip file from that page corresponding to the particular service that you want. This .zip file contains source code for your custom TinyWebDB web service. You do not have to understand this code in order to use it. Below, for concreteness, let's assume that you have downloaded alltags-deletable-tinywebdb-fall14.zip.
10. Unzip the downloaded alltags-deletable-tinywebdb-fall14.zip file (typically by double-clicking on its icon). This will create a folder named alltags-deletable-tinywebdb-fall14. You can rename it if you want.
11. Run the GoogleAppEngineLauncher by double-clicking its icon. If you are prompted to Make command symlinks, you can say OK, but it doesn't really matter if this step succeeds or not. This should result in creating an empty GoogleAppEnginerLauncher window:

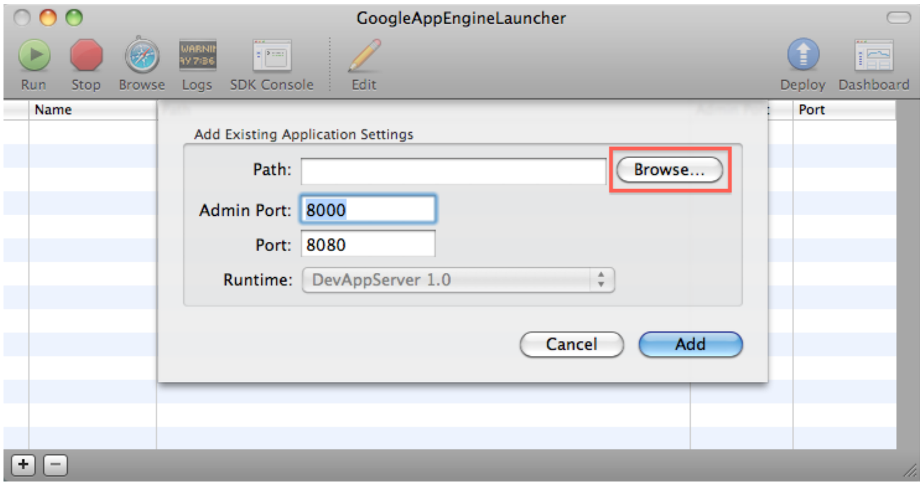
12. 

13. In the GoogleAppEngineLauncher menu, choose File>Add Existing Application.

14. 

15.

16. Click Browse

17. 

Add Existing Application Settings

Path:

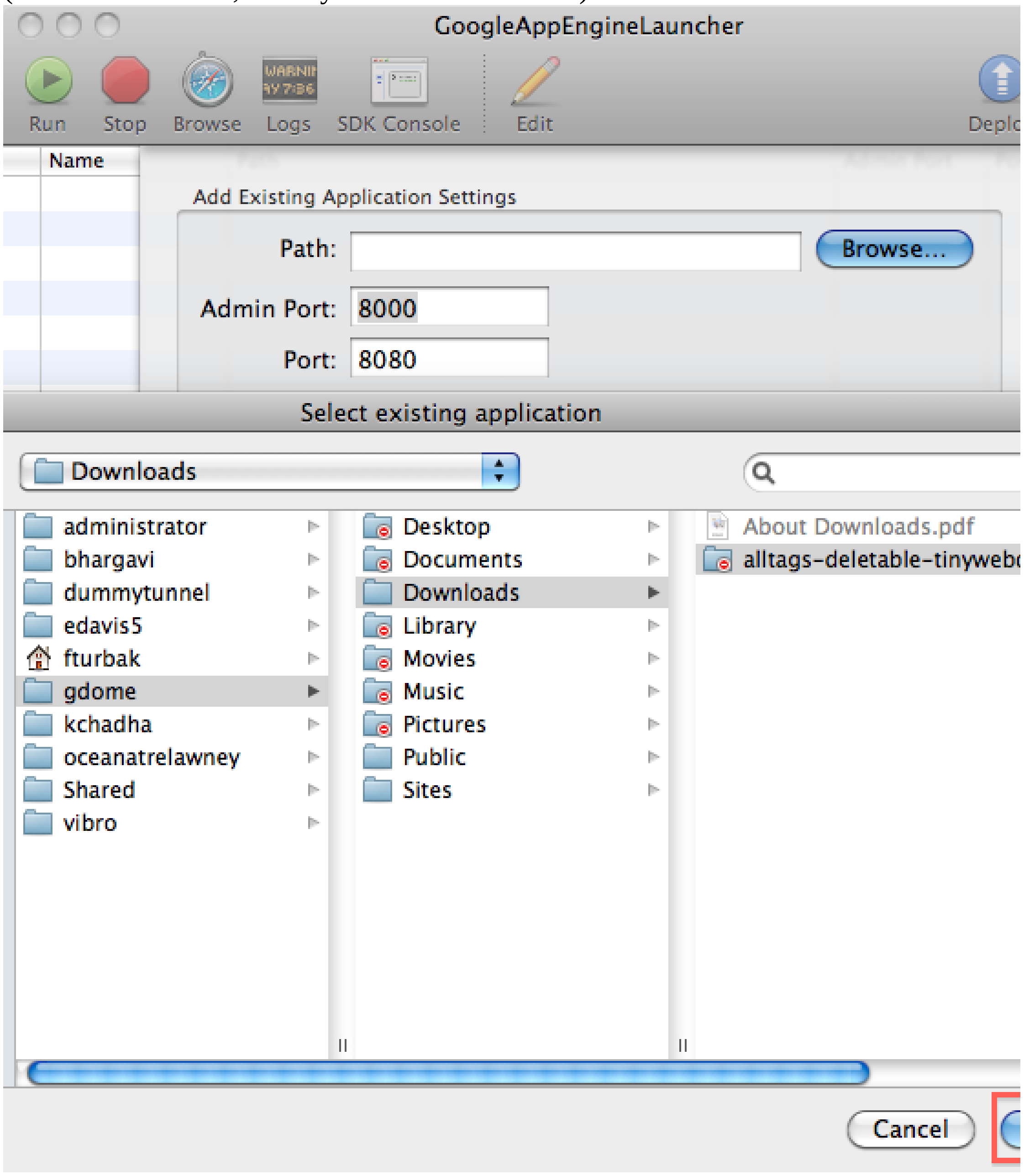
Admin Port: 8000

Port: 8080

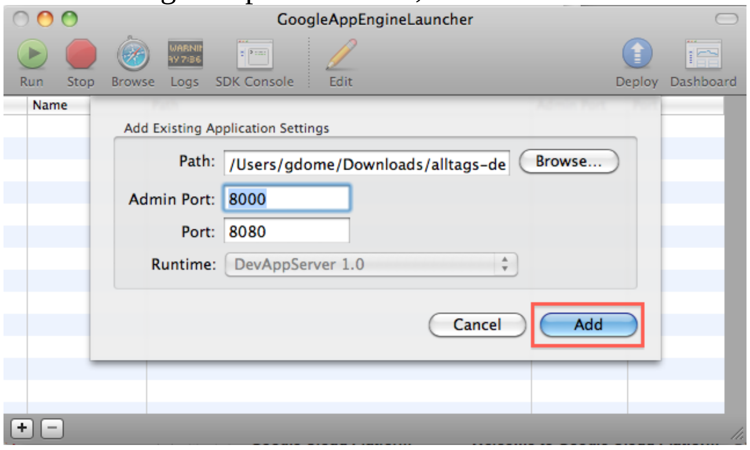
Runtime: DevAppServer 1.0

18. Navigate to the alltags-deletable-tinywebdb-fall14 folder you unzipped above, and click Choose

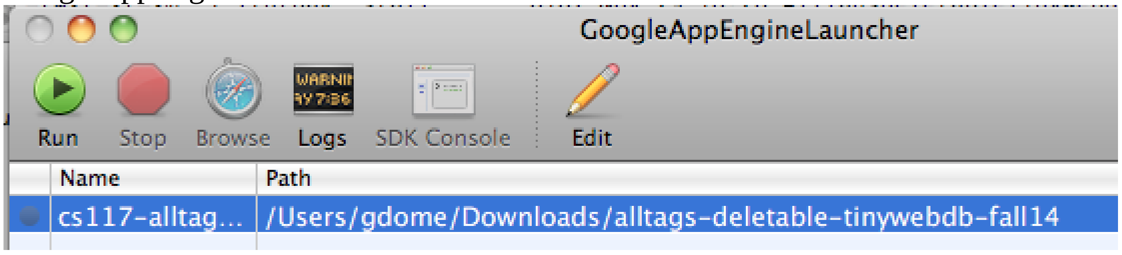
(for the folder itself, not any of its contained files).



19. Do not change the port numbers, and click Add.

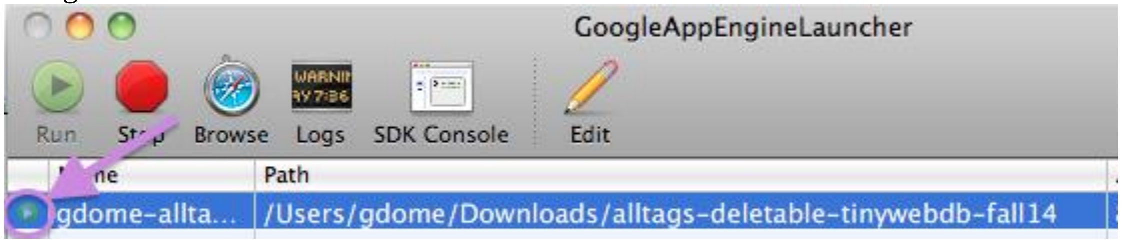


20. This alltags-deletable-tinywebdb-fall14 service has now been added to your GoogleAppEngineLauncher:



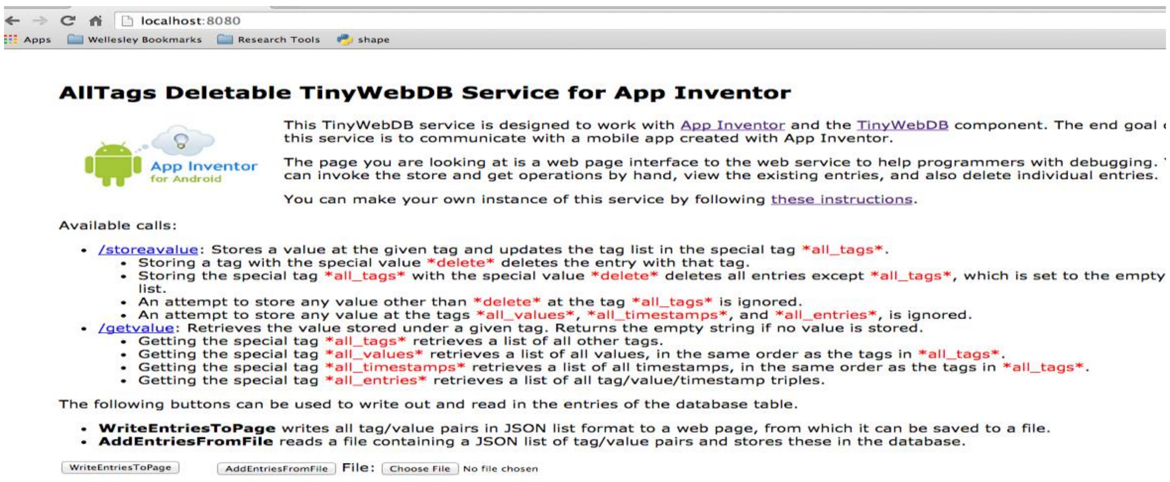
Name	Path
cs117-alltag...	/Users/gdome/Downloads/alltags-deletable-tinywebdb-fall14

21. In the GoogleAppEngineLauncher, click the Run button. This will launch a test web service that runs locally on your computer (not remotely on the web). The app has correctly launched if you see a small green circle icon containing a white right triangle at the far left of the selected line.



Name	Path
gdome-allta...	/Users/gdome/Downloads/alltags-deletable-tinywebdb-fall14

22.
23. If you instead see a warning icon --- a yellow triangle containing an exclamation point --- most likely GoogleAppEngineLauncher is confused about which version of Python to run. Fix this in GoogleAppEnginerLauncher>Preferences by setting the Python path to be the executable for Python 2.7
24. You can test the local test web service by opening a web browser and entering localhost:8080 as the URL. You'll see the web page interface to your web service. The end-goal of this service is to communicate with a mobile app created with App Inventor. But the service provides a web page interface to the service to help programmers with debugging. You can experiment with your web service through this local web page. You can invoke the store and get operations by hand (by clicking on /storeavalue and /getvalue), view the existing entries, and also delete individual entries.



AllTags Deletable TinyWebDB Service for App Inventor

This TinyWebDB service is designed to work with App Inventor and the TinyWebDB component. The end goal
this service is to communicate with a mobile app created with App Inventor.

The page you are looking at is a web page interface to the web service to help programmers with debugging.
can invoke the store and get operations by hand, view the existing entries, and also delete individual entries.

You can make your own instance of this service by following these instructions.

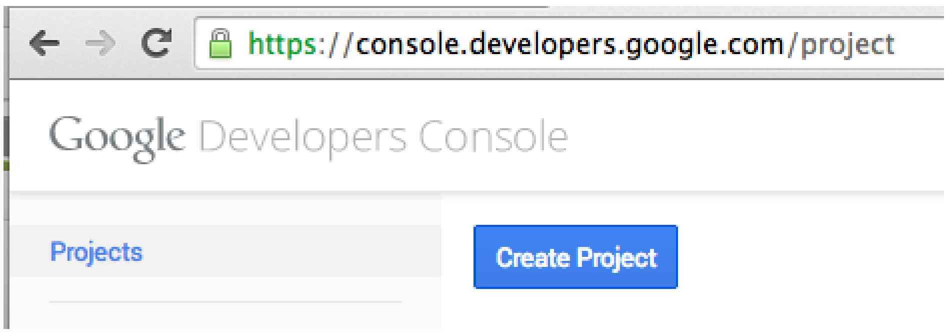
Available calls:

- /storeavalue: Stores a value at the given tag and updates the tag list in the special tag *all_tags*.
  - Storing a tag with the special value *delete* deletes the entry with that tag.
  - Storing the special tag *all_tags* with the special value *delete* deletes all entries except *all_tags*, which is set to the empty list.
  - An attempt to store any value other than *delete* at the tag *all_tags* is ignored.
  - An attempt to store any value at the tags *all_values*, *all_timestamps*, and *all_entries*, is ignored.
- /getvalue: Retrieves the value stored under a given tag. Returns the empty string if no value is stored.
  - Getting the special tag *all_tags* retrieves a list of all other tags.
  - Getting the special tag *all_values* retrieves a list of all values, in the same order as the tags in *all_tags*.
  - Getting the special tag *all_timestamps* retrieves a list of all timestamps, in the same order as the tags in *all_tags*.
  - Getting the special tag *all_entries* retrieves a list of all tag/value/timestamp triples.

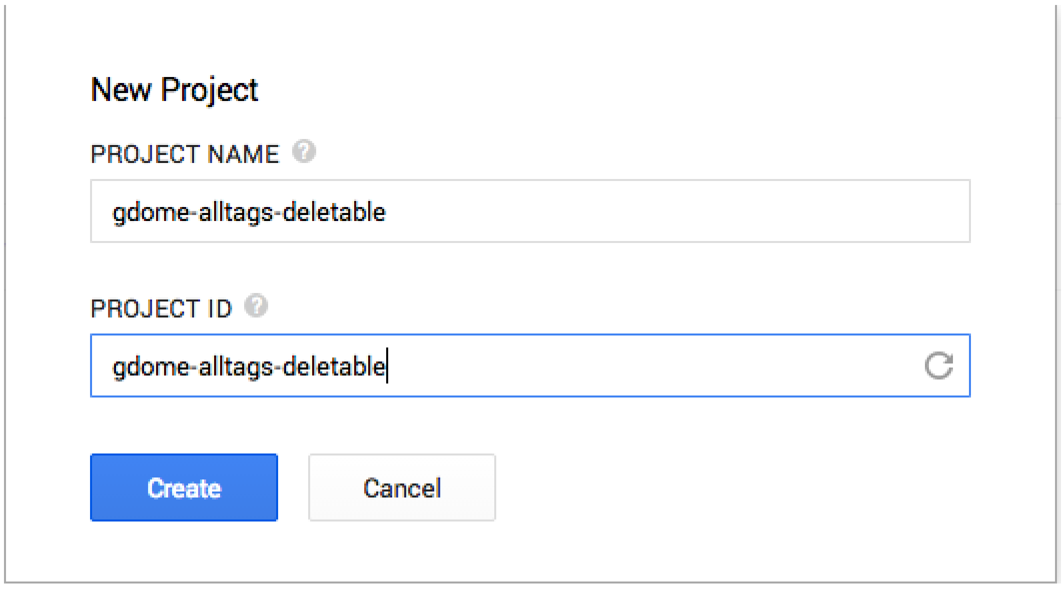
The following buttons can be used to write out and read in the entries of the database table.

- **WriteEntriesToPage** writes all tag/value pairs in JSON list format to a web page, from which it can be saved to a file.
- **AddEntriesFromFile** reads a file containing a JSON list of tag/value pairs and stores these in the database.

25. Your web service is not yet on the web, and thus not yet accessible to an App Inventor app. To move it to the web, you need to upload it to Google's App Engine servers using a Google account. Unfortunately, Wellesley's gmail accounts are not able to use App Engine, so you will need to use a Google/gmail account that is different from your Wellesley account. Create a new browser window that is logged into a non-Wellesley gmail account and visit https://console.developers.google.com/project. This will bring you to your Google Developer Console page:


https://console.developers.google.com/project
Google Developers Console
Projects
Create Project

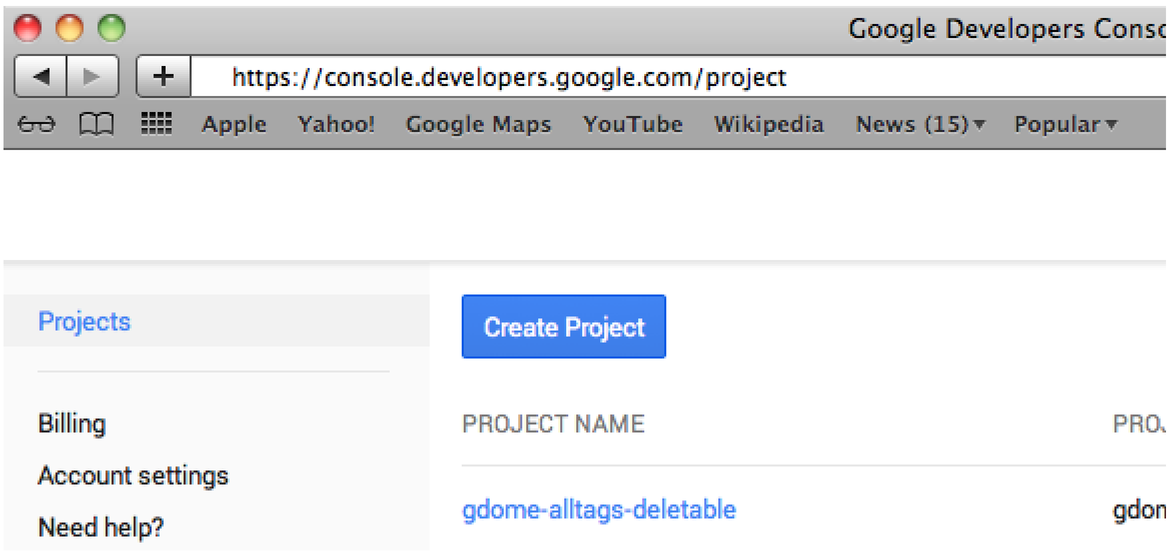
26. At some points during the following process, you may be encouraged to sign up for a free trial. DO NOT DO THIS.
27. Click Create Project. You'll need to specify a Project Name and a globally unique Project ID. For simplicity, choose the same name for each of these, something of the form yourGmailName-alltags-deletable. Remember the Project ID because you'll need it later. Click Create to create your project. If your Project ID was unique, you now have a new, empty project on Google's servers.


New Project

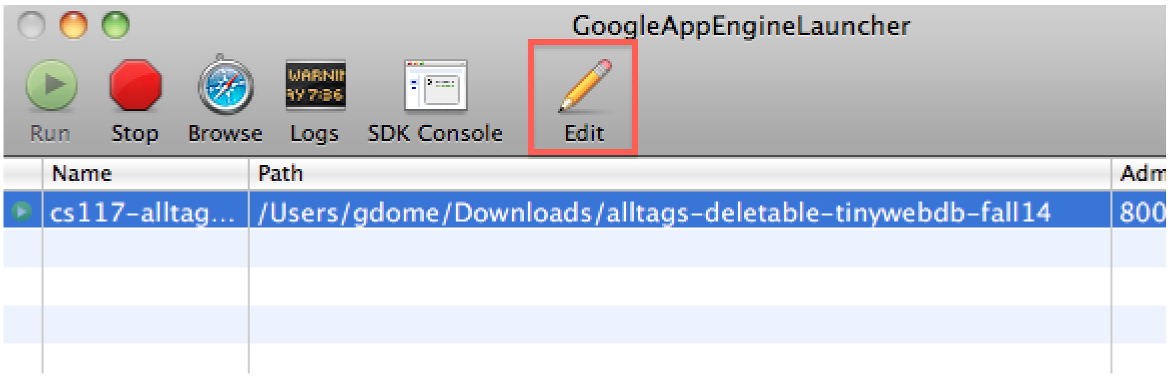
PROJECT NAME
gdome-alltags-deletable

PROJECT ID
gdome-alltags-deletable

28. You will see your new project under the Create Project button:

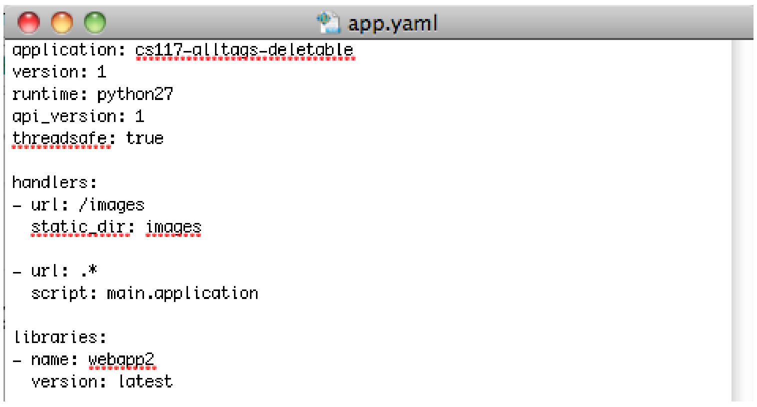


29. Now go back to the GoogleAppEngineLauncher window and click the Edit button.



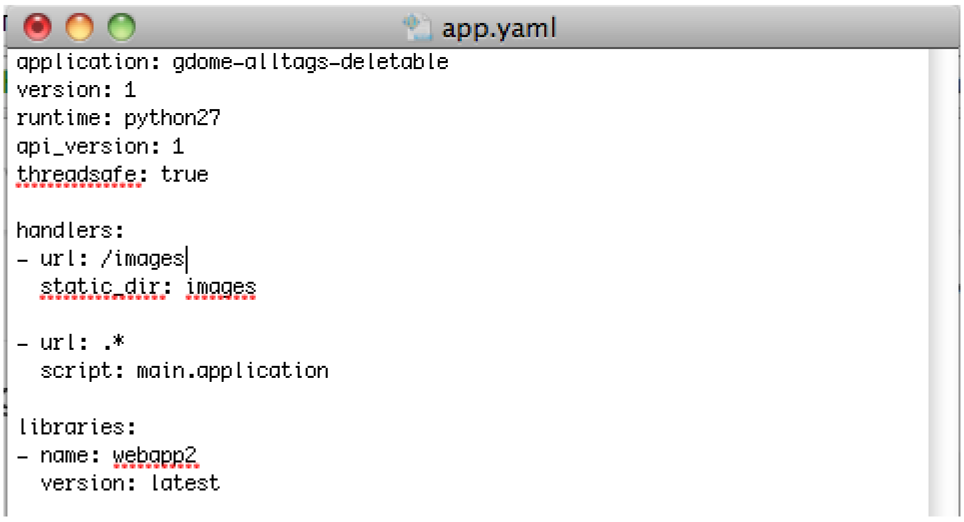
Name	Path	Adm
cs117-alltag...	/Users/gdome/Downloads/alltags-deletable-tinywebdb-fall14	800

30. This will open a text editor for the app.yaml file within the alltags-deletable-tinywebdb folder:



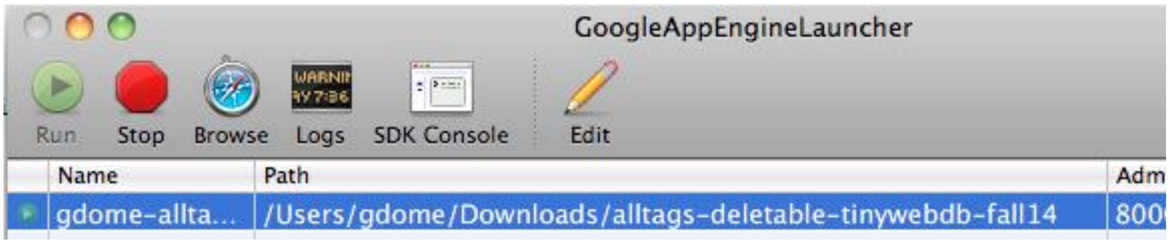
```
application: cs117-alltags-deletable
version: 1
runtime: python27
api_version: 1
threadsafe: true

handlers:
- url: /images
  static_dir: images

- url: .*
  script: main.application

libraries:
- name: webapp2
  version: latest
```

31. In the text editor, modify the first line so that application: is followed by the Project ID you choose when you created a project in step 19.



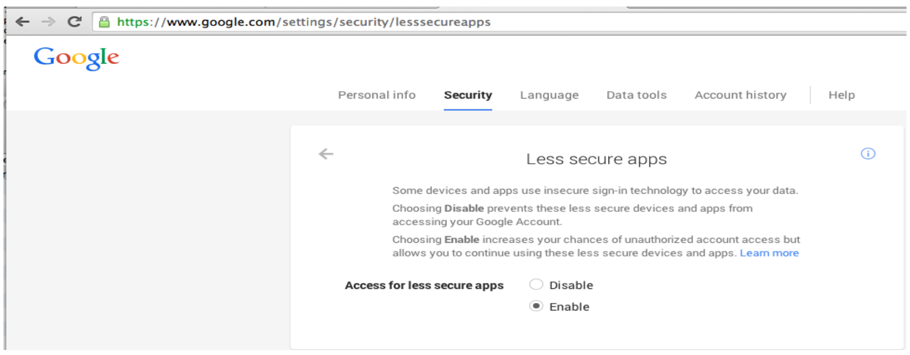
```
application: gdome-alltags-deletable
version: 1
runtime: python27
api_version: 1
threadsafe: true

handlers:
- url: /images
  static_dir: images

- url: .*
  script: main.application

libraries:
- name: webapp2
  version: latest
```

32. Save the changes to this file and close the editor window. The name of the project within the GoogleAppEngineLauncher window will change:



Name	Path	Adm
gdome-allta...	/Users/gdome/Downloads/alltags-deletable-tinywebdb-fall14	800

33. In a browser where you're logged in to your non-Wellesley gmail account, visit https://www.google.com/settings/security/lesssecureapps, and select Enable. It turns out this is necessary to deploy you service to App Engine. It's odd that Google should require you choose this less secure option for a service on its own App Engine servers, but you must do this before the next step.

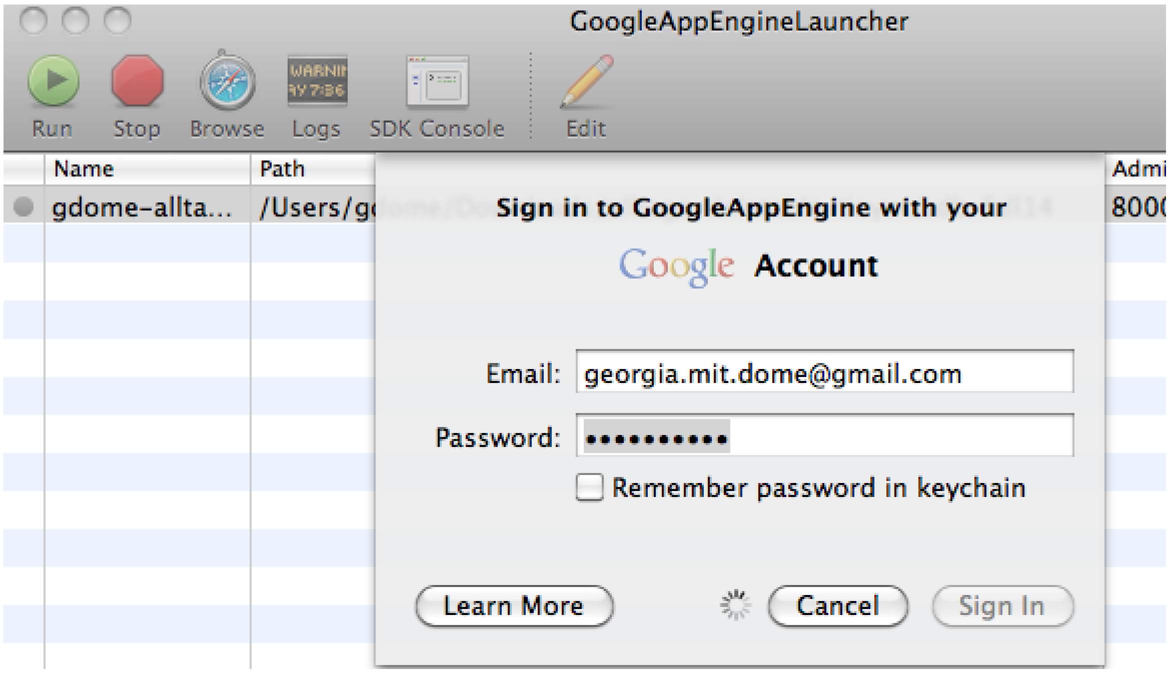


Less secure apps

Some devices and apps use insecure sign-in technology to access your data. Choosing **Disable** prevents these less secure devices and apps from accessing your Google Account.
Choosing **Enable** increases your chances of unauthorized account access but allows you to continue using these less secure devices and apps. Learn more

Access for less secure apps Disable [✓ Enable]

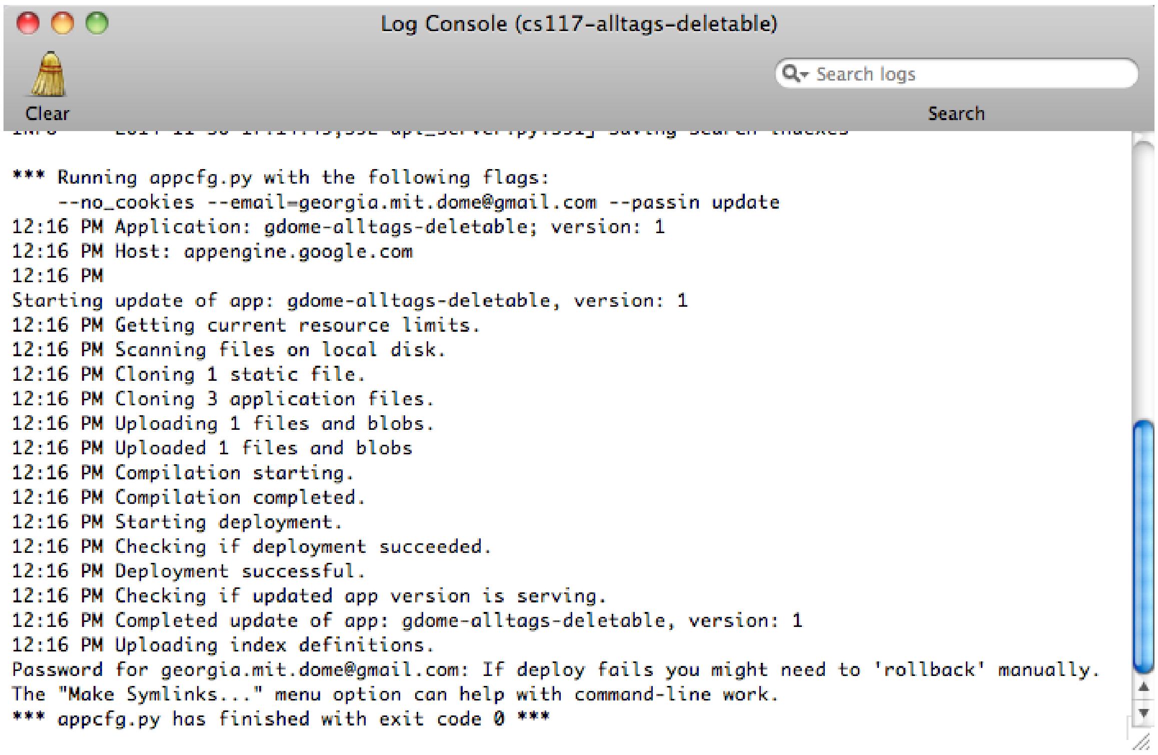
34. In GoogleAppEngineLauncher, click on the Deploy button and log into the same non-Wellesley gmail account you used to create your project in step 19.



Email: georgia.mit.dome@gmail.com
Password: ••••••••••

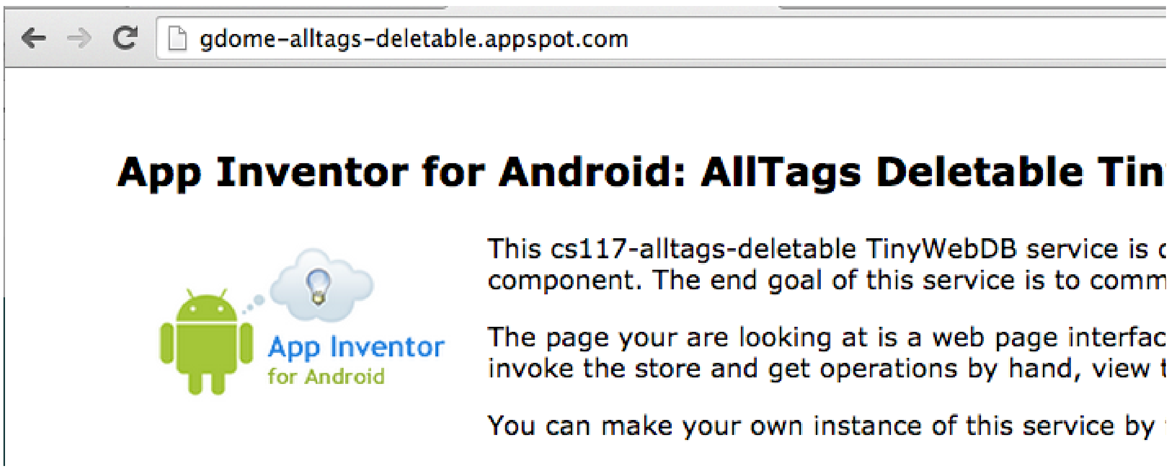
35.

Once you've logged in, the your service will be uploaded to an App Engine server in the cloud. The progress of this upload is tracked by the Log window:



```
*** Running appcfg.py with the following flags:
    --no_cookies --email=georgia.mit.dome@gmail.com --passin update
12:16 PM Application: gdome-alltags-deletable; version: 1
12:16 PM Host: appengine.google.com
12:16 PM
Starting update of app: gdome-alltags-deletable, version: 1
12:16 PM Getting current resource limits.
12:16 PM Scanning files on local disk.
12:16 PM Cloning 1 static file.
12:16 PM Cloning 3 application files.
12:16 PM Uploading 1 files and blobs.
12:16 PM Uploaded 1 files and blobs
12:16 PM Compilation starting.
12:16 PM Compilation completed.
12:16 PM Starting deployment.
12:16 PM Checking if deployment succeeded.
12:16 PM Deployment successful.
12:16 PM Checking if updated app version is serving.
12:16 PM Completed update of app: gdome-alltags-deletable, version: 1
12:16 PM Uploading index definitions.
Password for georgia.mit.dome@gmail.com: If deploy fails you might need to 'rollback' manually.
The "Make Symlinks..." menu option can help with command-line work.
*** appcfg.py has finished with exit code 0 ***
```

36. Once the deployment succeeds, you have your own active TinyWebDB service at the URL yourProjectID.appspot.com. For example, the above deployment creates a service at http://gdome-alltags-deletable.appspot.com. The web page at this URL should look the same as when you ran it on the local test server, except that now its in the cloud and accessible by anyone with the address.


gdome-alltags-deletable.appspot.com

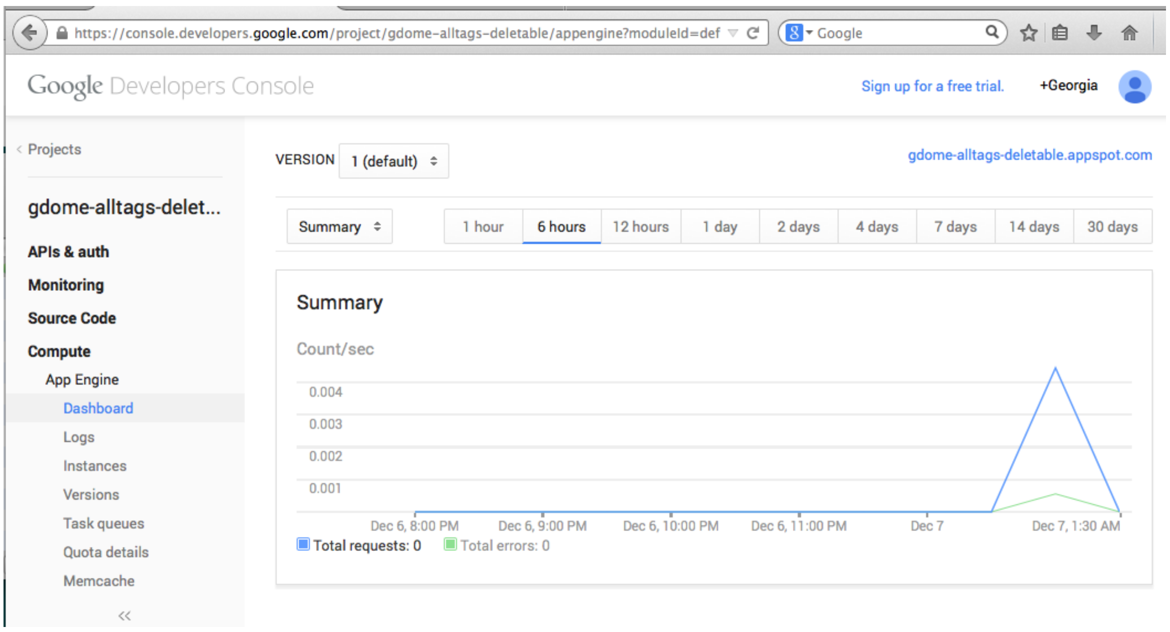
App Inventor for Android: AllTags Deletable Tin

This cs117-alltags-deletable TinyWebDB service is c
component. The end goal of this service is to comm

The page your are looking at is a web page interfac
invoke the store and get operations by hand, view t

You can make your own instance of this service by

37. In the GoogleAppEngineLauncher window, if you press the Dashboard button, it will bring you to a page with details about the monitoring of your new service. Note that your service is completely free. However, it does have some resource restrictions, and if you exceed a quota for the number of accesses to your database during a particular day, the service will shut down until the end of the day, at which point the quota will be reset.


https://console.developers.google.com/project/gdome-alltags-deletable/appengine?moduleId=def

Google Developers Console

Sign up for a free trial. +Georgia

VERSION 1 (default)   gdome-alltags-deletable.appspot.com

Summary | 1 hour | 6 hours | 12 hours | 1 day | 2 days | 4 days | 7 days | 14 days | 30 days

Summary

Total requests: 0   Total errors: 0

38. Your App Inventor apps can store and retrieve data using your new service. Just do the following:
39. Drag in a TinyWebDB component into the App Inventor Designer window.
40. Change the ServiceURL property from the URL for your service. Using the example above, this would be http://gdome-alltags-deletable.appspot.com
41. Now any StoreValue operations (blocks) will store data at your service, and any GetValue operations will retrieve values from your service.