DSSE – SRE - Position

**Craft demo**

The Accounting team would like to quickly look up country codes to speed up processing of expense reports, they currently have a website bookmarked and the process is manual and tedious. Help them speed things up by creating a simple cli.

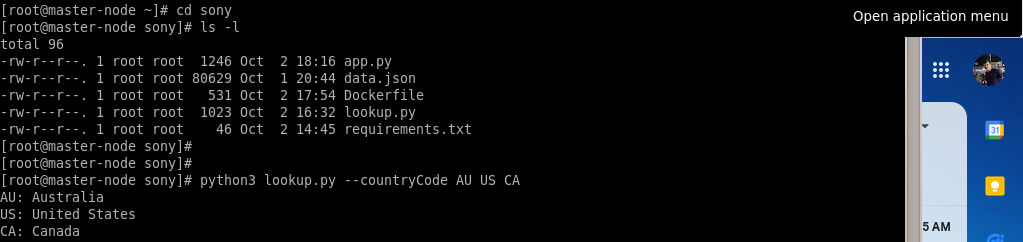
* Write a country lookup service using your favourite programming language.
* Explore the following API <https://www.travel-advisory.info/api>
* Given country code -> return country name example:

lookup --countryCode=AU

Australia

* Save the data from <https://www.travel-advisory.info/api> to a file called data.json and add functionality to your program to work with a file instead of real api endpoint
* Make sure your program supports multiple country codes as input
* Be ready to demo execution of your pro
* gram in the terminal

The code which I have written lookup.py

**Output:**

**Bonus**

Demand for your tool is increasing and there is an ask from multiple teams to be able to use county code lookup automation. Decision has been made to extend your tool to expose its functionality via REST.

* Convert code written in Craft demo to a REST service with two routes
* /health returns health of your service
* /diag check returns status of the api <https://www.travel-advisory.info/api> return {"api\_status":{... "code":200,"status":"ok"} that you obtained from hitting an API
* /convert – converts country name to country code
* Create local k8s cluster on your workstation
* Deploy your service to local k8s cluster

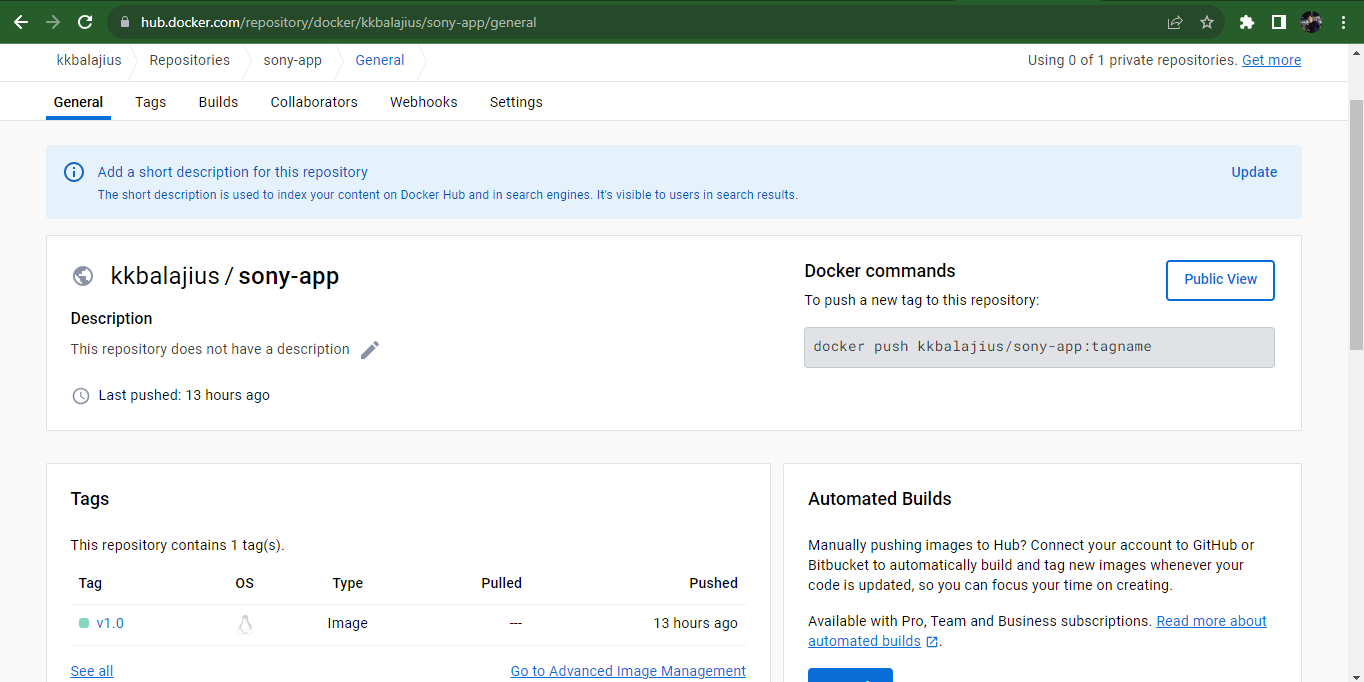
The Operations team likes your service, but to make their job easier and keep things operating at an optimal level

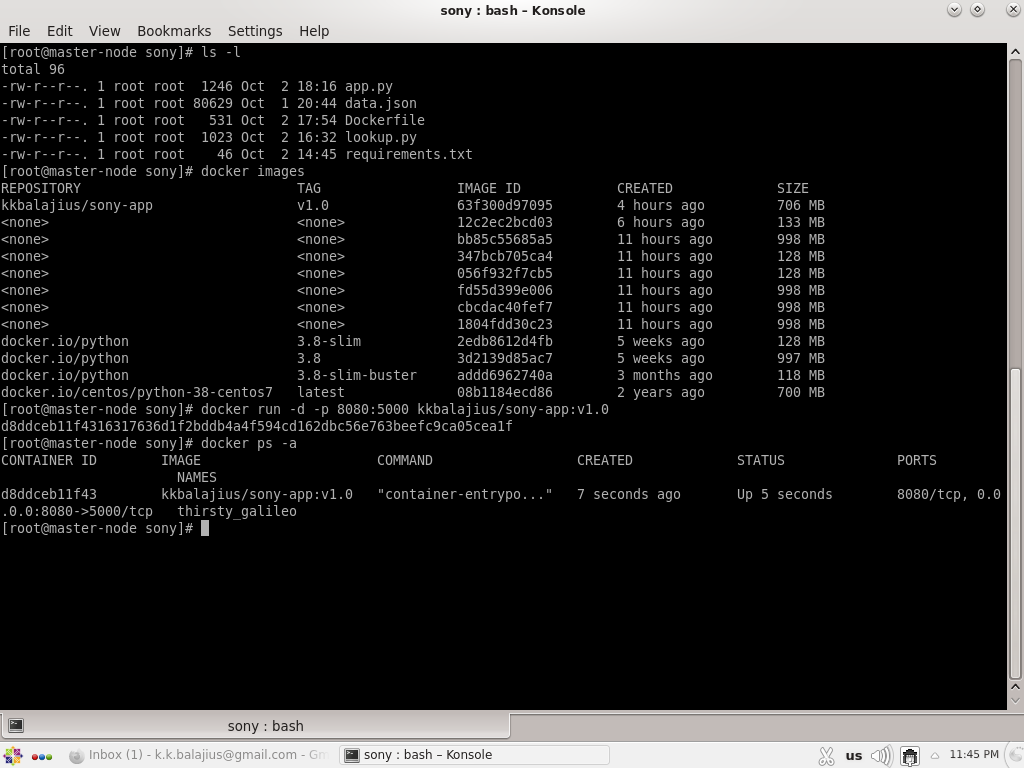
* Setup basic monitoring

Prerequisites:

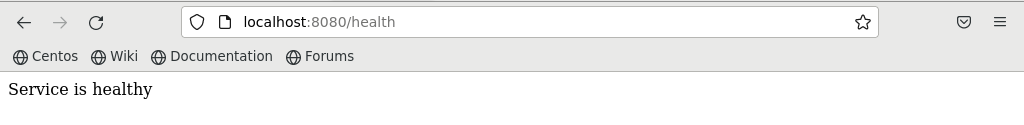
* Virtualbox
* K8S-Master – IP: 10.0.2.15
* K8S-Slave – IP: 10.0.2.16
* Python 2.7.5 – Preinstalled
* Docker Installed on both server

**Dokcer Hub Converted image uploaded in Repos:**

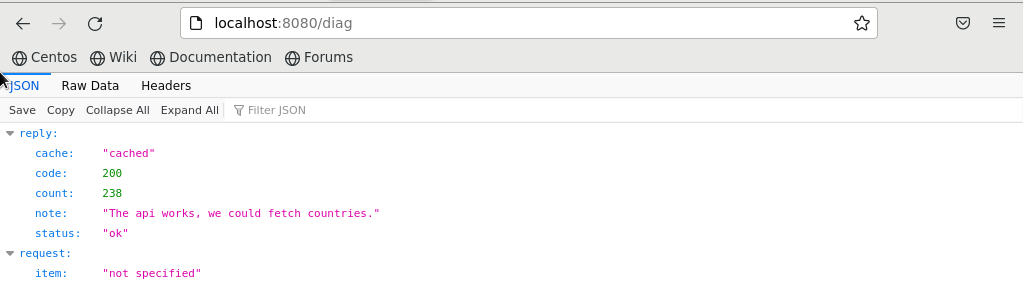




* Convert code written in Craft demo to a REST service with two routes
* **/health** returns health of your service



* **/diag** check returns status of the api <https://www.travel-advisory.info/api> return {"api\_status":{... "code":200,"status":"ok"} that you obtained from hitting an API



* /convert – converts country name to country code

