C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku login

Enter your Heroku credentials:

Email: chrisguthrie1600@comcast.net

Password: \*\*\*\*\*\*\*\*\*\*\*\*

Logged in as chrisguthrie1600@comcast.net

C:\cd Users\Chrisg\BootCampClasswork\Project-GW>

C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku create

Creating app... done, nameless-sea-62811

https://nameless-sea-62811.herokuapp.com/ | https://git.heroku.com/nameless-sea-

62811.git

C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku rename Project-GW

Renaming nameless-sea-62811 to Project-GW... !

! Name must start with a letter and can only contain lowercase letters,

! numbers, and dashes.

C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku rename project-gw

Renaming nameless-sea-62811 to project-gw... done

https://project-gw.herokuapp.com/ | https://git.heroku.com/project-gw.git

Git remote heroku updated

! Don't forget to update git remotes for all other local checkouts of the

! app.

C:\Users\Chrisg\BootCampClasswork\Project-GW>git remote -v

heroku https://git.heroku.com/project-gw.git (fetch)

heroku https://git.heroku.com/project-gw.git (push)

origin https://github.com/chrisg4github/Project-GW.git (fetch)

origin https://github.com/chrisg4github/Project-GW.git (push)

# Create the postgres database hobby-dev is a free database level

C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku addons:create heroku-postgresql:hobby-dev

Creating heroku-postgresql:hobby-dev on project-gw... free

Database has been created and is available

! This database is empty. If upgrading, you can transfer

! data from another database with pg:copy

Created postgresql-animated-62565 as DATABASE\_URL

Use heroku addons:docs heroku-postgresql to view documentation

# List any heroku addons you have set up

C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku addons

Add-on Plan Price State

--------------------------------------------- --------- ----- -------

heroku-postgresql (postgresql-animated-62565) hobby-dev free created

+- as DATABASE

The table above shows add-ons and the attachments to the current app (project-gw

) or other apps.

# List information about the database set up

C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku pg:info

=== DATABASE\_URL

Plan: Hobby-dev

Status: Available

Connections: 0/20

PG Version: 9.6.5

Created: 2017-11-14 18:52 UTC

Data Size: 7.1 MB

Tables: 0

Rows: 0/10000 (In compliance)

Fork/Follow: Unsupported

Rollback: Unsupported

Continuous Protection: Off

Add-on: postgresql-animated-62565

# Check the contents of the DATABASE\_URL

C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku config

=== project-gw Config Vars

DATABASE\_URL: postgres://zveqkcpjroxbce:44a0357267af92609c0117ccd9c262d8746e69c5

980140d843cdfef853069cb8@ec2-50-17-253-3.compute-1.amazonaws.com:5432/d7b3o1itpode38

Exit heroku.

Log into the local database:

C:\Users\Chrisg\BootCampClasswork\Project-GW>psql -d gw\_db -U postgres

Password for user postgres:

psql (9.6.4)

Type "help" for help.

gw\_db=# \l

List of databases

Name | Owner | Encoding | Collate |

Ctype | Access privileges

---------------------------+----------+----------+----------------------------+----------------------------+-----------------------

class\_db | chrisg | UTF8 | English\_United States.1252 |

English\_United States.1252 |

companies | postgres | UTF8 | English\_United States.1252 |

English\_United States.1252 |

gw\_db | postgres | UTF8 | English\_United States.1252 |

English\_United States.1252 |

joins\_db | postgres | UTF8 | English\_United States.1252 |

English\_United States.1252 |

pagila | chrisg | UTF8 | English\_United States.1252 |

English\_United States.1252 |

postgres | postgres | UTF8 | English\_United States.1252 |

English\_United States.1252 |

template0 | postgres | UTF8 | English\_United States.1252 |

English\_United States.1252 | =c/postgres +

| | | |

| postgres=CTc/postgres

template1 | postgres | UTF8 | English\_United States.1252 |

English\_United States.1252 | =c/postgres +

| | | |

| postgres=CTc/postgres

tough\_interview\_questions | postgres | UTF8 | English\_United States.1252 |

English\_United States.1252 |

(9 rows)

gw\_db=# \c gw\_db

You are now connected to database "gw\_db" as user "postgres".

gw\_db=# \dt

No relations found.

gw\_db-# \q

**# Log into heroku to push the local database to heroku database**

C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku login

Enter your Heroku credentials:

Email: chrisguthrie1600@comcast.net

Password: \*\*\*\*\*\*\*\*\*\*\*\*

Logged in as chrisguthrie1600@comcast.net

C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku pg:push gw\_db DATABASE\_URL --app project-gw

heroku-cli: Pushing gw\_db ---> postgresql-animated-62565

'env' is not recognized as an internal or external command,

operable program or batch file.

**# May need this command to empty target database**

>heroku pg:reset DATABASE\_URL

PGUSER=postgres PGPASSWORD='password' heroku pg:push gw\_db DATABASE\_URL --app project-gw

**# View heroku postgres log**

heroku logs -p postgres -t

heroku logs -p postgres

**# Screen output of pg:push attempts:**

$ heroku pg:push gw\_db DATABASE\_URL --app project-gw

heroku-cli: Pushing gw\_db ---> postgresql-animated-62565

pg\_restore: [archiver] input file is too short (read 0, expected 5)

$ PGUSER=postgres PGPASSWORD='password' heroku pg:push gw\_db DATABASE\_URL --app project-gw

heroku-cli: Pushing gw\_db ---> postgresql-animated-62565

pg\_dump: [archiver (db)] connection to database "gw\_db" failed: FATAL: password authentication failed for user "postgres"

pg\_restore: [archiver] input file is too short (read 0, expected 5)

Chrisg@Chrisg-PC MINGW64 ~/bootcampclasswork/Project-GW (master)

$ PGUSER=postgres PGPASSWORD='password' heroku pg:push gw\_db DATABASE\_URL --app project-gw

heroku-cli: Pushing gw\_db ---> postgresql-animated-62565

pg\_restore: [archiver] did not find magic string in file header

pg\_dump: last built-in OID is 16383

pg\_dump: reading extensions

pg\_dump: identifying extension members

pg\_dump: reading schemas

pg\_dump: reading user-defined tables

pg\_dump: reading user-defined functions

pg\_dump: reading user-defined types

pg\_dump: reading procedural languages

pg\_dump: reading user-defined aggregate functions

pg\_dump: reading user-defined operators

pg\_dump: reading user-defined access methods

pg\_dump: reading user-defined operator classes

pg\_dump: reading user-defined operator families

pg\_dump: reading user-defined text search parsers

pg\_dump: reading user-defined text search templates

pg\_dump: reading user-defined text search dictionaries

pg\_dump: reading user-defined text search configurations

pg\_dump: reading user-defined foreign-data wrappers

pg\_dump: reading user-defined foreign servers

pg\_dump: reading default privileges

pg\_dump: reading user-defined collations

pg\_dump: reading user-defined conversions

pg\_dump: reading type casts

pg\_dump: reading transforms

pg\_dump: reading table inheritance information

pg\_dump: reading event triggers

pg\_dump: finding extension tables

pg\_dump: finding inheritance relationships

pg\_dump: reading column info for interesting tables

pg\_dump: finding the columns and types of table "public.station"

pg\_dump: finding default expressions of table "public.station"

pg\_dump: finding the columns and types of table "public.weather"

pg\_dump: finding default expressions of table "public.weather"

pg\_dump: flagging inherited columns in subtables

pg\_dump: reading indexes

pg\_dump: reading indexes for table "public.station"

pg\_dump: reading indexes for table "public.weather"

pg\_dump: reading constraints

pg\_dump: reading triggers

pg\_dump: reading rewrite rules

pg\_dump: reading policies

pg\_dump: reading row security enabled for table "public.station\_id\_seq"

pg\_dump: reading policies for table "public.station\_id\_seq"

pg\_dump: reading row security enabled for table "public.weather\_id\_seq"

pg\_dump: reading policies for table "public.weather\_id\_seq"

pg\_dump: reading row security enabled for table "public.station"

pg\_dump: reading policies for table "public.station"

pg\_dump: reading row security enabled for table "public.weather"

pg\_dump: reading policies for table "public.weather"

pg\_dump: reading large objects

pg\_dump: reading dependency data

pg\_dump: saving encoding = UTF8

pg\_dump: saving standard\_conforming\_strings = on

pg\_dump: saving database definition

pg\_dump: [custom archiver] WARNING: ftell mismatch with expected position -- ftell used

pg\_dump: [custom archiver] WARNING: ftell mismatch with expected position -- ftell used

pg\_dump: dumping contents of table "public.station"

pg\_dump: [custom archiver] WARNING: ftell mismatch with expected position -- ftell used

pg\_dump: dumping contents of table "public.weather"

pg\_dump: [custom archiver] could not write to output file: Invalid argument

**Examples of pg\_dump commands:**

PGPASSWORD=mypassword pg\_dump -Fc --no-acl --no-owner -U postgres gw\_db > gw\_db11142017.dump

pg\_dump -Fc --no-owner -U postgres gw\_db > gw\_db\_11142017.dump

pg\_dump -Fc gw\_db > gw\_db\_11142017.dump

**Examples of pg\_restore:**

pg\_restore --verbose --clean --no-acl --no-owner -U postgres -d gw\_db gw\_db\_1142017.dump

heroku pg:backups:restore 'https://s3.amazonaws.com/me/items/3H0q/mydb.dump' DATABASE\_URL

pg\_restore -U <username> -d <dbname> -1 <filename>.dump

PGPASSWORD=<PASS> pg\_restore --verbose --clean --no-acl --no-owner -h <HOSTNAME> -U <USER> -d <DATABASE> -p <PORT> <FILEPATH>

**# Try this command 1st**

heroku pg:pg\_restore --verbose --clean --no-acl --no-owner -h ec2-50-17-253-3.compute-1.amazonaws.com -U zveqkcpjroxbce -d gw\_db -p 5432 https://github.com/chrisg4github/Project-GW/blob/master/gw\_db\_11142017.dump password=44a0357267af92609c0117ccd9c262d8746e69c5980140d843cdfef853069cb8

heroku pg\_restore --verbose --clean --no-acl --no-owner -h ec2-50-17-253-3.compute-1.amazonaws.com -U zveqkcpjroxbce -d gw\_db -p 5432 https://github.com/chrisg4github/Project-GW/blob/master/gw\_db\_11142017.dump password=44a0357267af92609c0117ccd9c262d8746e69c5980140d843cdfef853069cb8

**# Try this command 2nd**

PGPASSWORD='44a0357267af92609c0117ccd9c262d8746e69c5980140d843cdfef853069cb8' pg\_restore -C --verbose --clean --no-acl --no-owner -h ec2-50-17-253-3.compute-1.amazonaws.com -U zveqkcpjroxbce -d gw\_db -p 5432 https://github.com/chrisg4github/Project-GW/blob/master/gw\_db\_11142017.dump

**# Try this command 3rd**

PGPASSWORD='44a0357267af92609c0117ccd9c262d8746e69c5980140d843cdfef853069cb8' pg\_restore -C --verbose --clean --no-acl --no-owner -h ec2-50-17-253-3.compute-1.amazonaws.com -U zveqkcpjroxbce -d gw\_db -p 5432 C:/Users/Chrisg/BootCampClasswork/Project-GW/gw\_db\_11142017.dump

**# Try this command 4th**

PGPASSWORD='44a0357267af92609c0117ccd9c262d8746e69c5980140d843cdfef853069cb8' pg\_restore -C --verbose --no-acl --no-owner -h ec2-50-17-253-3.compute-1.amazonaws.com -U zveqkcpjroxbce -p 5432 C:/Users/Chrisg/BootCampClasswork/Project-GW/gw\_db\_11142017.dump

Latest log file for pg\_restore is pg\_restore\_1a.log: The following didn't make it into the log file.

Chrisg@Chrisg-PC MINGW64 ~/bootcampclasswork/Project-GW (master)

$ PGPASSWORD='44a0357267af92609c0117ccd9c262d8746e69c5980140d843cdfef853069cb8' pg\_restore -C --verbose --no-acl --no-owner -h ec2-50-17-253-3.compute-1.amazon aws.com -U zveqkcpjroxbce -p 5432 C:/Users/Chrisg/BootCampClasswork/Project-GW/ gw\_db\_11142017.dump >pg\_restore\_1a.log

pg\_restore: creating DATABASE "gw\_db"

pg\_restore: connecting to new database "gw\_db"

pg\_restore: creating COMMENT "gw\_db"

pg\_restore: creating SCHEMA "public"

pg\_restore: creating COMMENT "SCHEMA public"

pg\_restore: creating EXTENSION "plpgsql"

pg\_restore: creating COMMENT "EXTENSION plpgsql"

pg\_restore: creating SEQUENCE "public.station\_id\_seq"

pg\_restore: creating TABLE "public.station"

pg\_restore: creating SEQUENCE "public.weather\_id\_seq"

pg\_restore: creating TABLE "public.weather"

pg\_restore: processing data for table "public.station"

pg\_restore: executing SEQUENCE SET station\_id\_seq

pg\_restore: processing data for table "public.weather"

pg\_restore: executing SEQUENCE SET weather\_id\_seq

pg\_restore: creating CONSTRAINT "public.station station\_pkey"

pg\_restore: creating CONSTRAINT "public.weather weather\_pkey"

Ticket #530343 - Not able to move my local postgres database to Heroku.

<https://help.heroku.com/tickets/530343?n=1>

CREATE DATABASE gw\_db; # I get permission denied when using this command.

C:\Users\Chrisg\BootCampClasswork\Project-GW

**Get the heroku postgres credentials using: heroku pg:credentials:url**

C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku pg:credentials:url

Connection information for default credential.

**Connection info string:**

"dbname=d7b3o1itpode38 host=ec2-50-17-253-3.compute-1.amazonaws.com port=5432

user=zveqkcpjroxbce password=44a0357267af92609c0117ccd9c262d8746e69c5980140d843

cdfef853069cb8 sslmode=require"

**Connection URL *aka* DATABASE\_URL:**

postgres://zveqkcpjroxbce:44a0357267af92609c0117ccd9c262d8746e69c5980140d843c

dfef853069cb8@ec2-50-17-253-3.compute-1.amazonaws.com:5432/d7b3o1itpode38

# Local verification of the database gw\_db (hawaii tables)

Chrisg@Chrisg-PC MINGW64 ~/bootcampclasswork/Project-GW (master)

$ psql -d gw\_db -U postgres

Password for user postgres:

psql (9.6.4)

WARNING: Console code page (437) differs from Windows code page (1252)

8-bit characters might not work correctly. See psql reference

page "Notes for Windows users" for details.

Type "help" for help.

gw\_db=# \dt

List of relations

Schema | Name | Type | Owner

--------+---------+-------+----------

public | station | table | postgres

public | weather | table | postgres

(2 rows)

gw\_db=# \q (to quit psql)

**# Pushing code from local repo to Heroku repo:**

To deploy your app to Heroku, you typically use the git add, commit, push commands

to push the code from your local repository’s master branch to your heroku remote.

$ git add .

$ git commit -m "Move new files to heroku"

$ git push heroku master

# Why do we have a local repo copy in our GitHub repo? We can't see the Heroku directory structure.

# Are we supposed to move local repo to GitHub to Heroku? Move from local to GitHub and then Heroku.

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Chrisg@Chrisg-PC MINGW64 ~/bootcampclasswork/Project-GW (master)

$ git push heroku master

Counting objects: 92, done.

Delta compression using up to 4 threads.

Compressing objects: 100% (86/86), done.

Writing objects: 100% (92/92), 131.12 MiB | 176.00 KiB/s, done.

Total 92 (delta 28), reused 0 (delta 0)

remote: Compressing source files... done.

remote: Building source:

remote:

remote: -----> Python app detected

remote: -----> Installing python-3.6.2

remote: -----> Installing pip

remote: -----> Installing requirements with pip

remote: Collecting gunicorn==19.7.1 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 1))

remote: Downloading gunicorn-19.7.1-py2.py3-none-any.whl (111kB)

remote: Collecting certifi==2017.7.27.1 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 2))

remote: Downloading certifi-2017.7.27.1-py2.py3-none-any.whl (349kB)

remote: Collecting chardet==3.0.4 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 3))

remote: Downloading chardet-3.0.4-py2.py3-none-any.whl (133kB)

remote: Collecting click==6.7 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 4))

remote: Downloading click-6.7-py2.py3-none-any.whl (71kB)

remote: Collecting Flask==0.12.2 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 5))

remote: Downloading Flask-0.12.2-py2.py3-none-any.whl (83kB)

remote: Collecting idna==2.6 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 6))

remote: Downloading idna-2.6-py2.py3-none-any.whl (56kB)

remote: Collecting itsdangerous==0.24 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 7))

remote: Downloading itsdangerous-0.24.tar.gz (46kB)

remote: Collecting Jinja2==2.9.6 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 8))

remote: Downloading Jinja2-2.9.6-py2.py3-none-any.whl (340kB)

remote: Collecting kdtree==0.15 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 9))

remote: Downloading kdtree-0.15-py2.py3-none-any.whl

remote: Collecting MarkupSafe==1.0 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 10))

remote: Downloading MarkupSafe-1.0.tar.gz

remote: Collecting numpy==1.13.1 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 11))

remote: Downloading numpy-1.13.1-cp36-cp36m-manylinux1\_x86\_64.whl (17.0MB)

remote: Collecting pandas==0.20.3 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 12))

remote: Downloading pandas-0.20.3-cp36-cp36m-manylinux1\_x86\_64.whl (24.5MB)

remote: Collecting python-dateutil==2.6.1 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 13))

remote: Downloading python\_dateutil-2.6.1-py2.py3-none-any.whl (194kB)

remote: Collecting pytz==2017.2 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 14))

remote: Downloading pytz-2017.2-py2.py3-none-any.whl (484kB)

remote: Collecting requests==2.18.4 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 15))

remote: Downloading requests-2.18.4-py2.py3-none-any.whl (88kB)

remote: Collecting scikit-learn==0.19.1 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 16))

remote: Downloading scikit\_learn-0.19.1-cp36-cp36m-manylinux1\_x86\_64.whl (12.4MB)

remote: Collecting scipy==1.0.0 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 17))

remote: Downloading scipy-1.0.0-cp36-cp36m-manylinux1\_x86\_64.whl (50.0MB)

remote: Collecting selenium==3.5.0 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 18))

remote: Downloading selenium-3.5.0-py2.py3-none-any.whl (921kB)

remote: Collecting six==1.11.0 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 19))

remote: Downloading six-1.11.0-py2.py3-none-any.whl

remote: Collecting sklearn==0.0 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 20))

remote: Downloading sklearn-0.0.tar.gz

remote: Collecting splinter==0.7.6 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 21))

remote: Downloading splinter-0.7.6.tar.gz

remote: Collecting SQLAlchemy==1.1.14 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 22))

remote: Downloading SQLAlchemy-1.1.14.tar.gz (5.2MB)

remote: Collecting urllib3==1.22 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 23))

remote: Downloading urllib3-1.22-py2.py3-none-any.whl (132kB)

remote: Collecting Werkzeug==0.12.2 (from -r /tmp/build\_f6d14843e3dd9052c457ccf6cb737c01/requirements.txt (line 24))

remote: Downloading Werkzeug-0.12.2-py2.py3-none-any.whl (312kB)

remote: Installing collected packages: gunicorn, certifi, chardet, click,itsdangerous, MarkupSafe, Jinja2, Werkzeug, Flask, idna, kdtree, numpy, six, py thon-dateutil, pytz, pandas, urllib3, requests, scikit-learn, scipy, selenium, sklearn, splinter, SQLAlchemy

remote: Running setup.py install for itsdangerous: started

remote: Running setup.py install for itsdangerous: finished with status 'done'

remote: Running setup.py install for MarkupSafe: started

remote: Running setup.py install for MarkupSafe: finished with status 'done'

remote: Running setup.py install for sklearn: started

remote: Running setup.py install for sklearn: finished with status 'done'

remote: Running setup.py install for splinter: started

remote: Running setup.py install for splinter: finished with status 'done'

remote: Running setup.py install for SQLAlchemy: started

remote: Running setup.py install for SQLAlchemy: finished with status 'done'

remote: Successfully installed Flask-0.12.2 Jinja2-2.9.6 MarkupSafe-1.0 SQLAlchemy-1.1.14 Werkzeug-0.12.2 certifi-2017.7.27.1 chardet-3.0.4 click-6.7 gun icorn-19.7.1 idna-2.6 itsdangerous-0.24 kdtree-0.15 numpy-1.13.1 pandas-0.20.3 python-dateutil-2.6.1 pytz-2017.2 requests-2.18.4 scikit-learn-0.19.1 scipy-1.0.0 selenium-3.5.0 six-1.11.0 sklearn-0.0 splinter-0.7.6 urllib3-1.22

remote:

remote: -----> Discovering process types

remote: Procfile declares types -> web

remote:

remote: -----> Compressing...

remote: Done: 236M

remote: -----> Launching...

remote: Released v4

remote: https://project-gw.herokuapp.com/ deployed to Heroku

remote:

remote: Verifying deploy... done.

To https://git.heroku.com/project-gw.git

\* [new branch] master -> master

Chrisg@Chrisg-PC MINGW64 ~/bootcampclasswork/Project-GW (master)

**# to view the live app**

# open in a browser to https://project-gw.herokuapp.com

**Test URLs for the Hawaii API test application:**

https://project-gw.herokuapp.com/api/v1.0/stations

https://project-gw.herokuapp.com/api/v1.0/precipitation

**# Error message clean up by uninstalling and then reinstalling Heroku CLI**

C:\Users\Chrisg\BootCampClasswork\Project-GW>heroku login

! Error parsing plugin

! C:\Users\Chrisg\AppData\Local\heroku\client\node\_modules\heroku-pg

! Unexpected token ;

! Error parsing plugin

! C:\Users\Chrisg\AppData\Local\heroku\client\node\_modules\heroku-pipelines

! Error parsing

! C:\Users\Chrisg\AppData\Local\heroku\client\node\_modules\heroku-pipelines\

package.json:

! Unexpected token m in JSON at position 1

Enter your Heroku credentials:

Email: