

1.Question 1

What should I run to revoke access to all users logged in on my Google Cloud SDK?

1 / 1 point

- ☐ gcloud auth revoke account
- ☒ gcloud auth revoke --all
- ☐ gcloud account revoke

Correct

Correct! gcloud auth revoke will revoke access to a specific account while adding the --all flag will revoke access immediately to all accounts currently logged in

2.Question 2

Firebase Authentication only allows 2 methods of authentication: Google and Email

1 / 1 point

- ☐ True
- ☒ False

Correct

Correct! You can select from multiple methods in the Firebase console, e.g. Facebook, Twitter, Email, Google etc.

3.Question 3

What is a typical Python AppEngine app structure?

1 / 1 point

- ☒ A static folder with static resources, a templates folder with frontend templates, a YAML configuration file, one or more Python scripts, a requirements file and, if datastore is in use, a relevant service key to be provided.
- ☐ A static folder with static resources, a templates folder with frontend templates, a YAML configuration file, a requirements file
- ☐ A static folder with static resources, a config file, a template resource, a main script, a datastore key, a secondary set up file, a configuration file

Correct

Correct! This is the typical structure of a web service as seen and set in Task 4

4.Question 4

"gcloud app deploy" command will deploy the app on the currently set project via the currently active account

1 / 1 point

- ☒ True
- ☐ False

Correct

Correct! In Task 11 we have seen how to deploy the app.

5.Question 5

"gcloud app getsite" will send me to the online location where my deployed app lives

1 / 1 point

- ☒ False
- ☐ True

Correct

Correct, the right command is "browse" as seen in Task 11, deploy the app.

6.Question 6

How do you bind a URL to a function creating a route (endpoint) in Flask with POST and GET methods, so to direct a URL to a certain template and performing certain actions?

1 / 1 point

- ☒ @app.route('/', methods = ['POST', 'GET']), def function(): and return render_template
- ☐ @app.route render template
- ☐ @app.route POST/GET, render_template print(routename, mapURL)

Correct

Correct! Review Task 6 to explore Flask and Python app set up

7.Question 7

How to run your app locally?

1 / 1 point

- ☒ Make sure to add if __name__ == '__main__': app.run(host='127.0.0.1', port=8080, debug=True) to your Python script, run python "SCRIPTNAME.py" in your Powershell terminal, and direct the browser to localhost:8080
- ☐ Make sure to add if __name__ == '__main__': app.run and open directly the browser to localhost/
- ☐ Run Windows IIS, copy the app in the root folder and open localhost on port 8085

Correct

Correct! In Task 7 we have run and tested the Python app

8.Question 8

python -r requirements.txt will install automatically all listed dependencies for your app

1 / 1 point

- ☒ True
- ☐ False

Correct

Correct! Remember to add all of your dependencies (needed packages) to requirements.txt !

9.Question 9

You need to add Firebase project-specific configuration information to your Jinja2 templates

1 / 1 point

- ☒ True
- ☐ False

Correct

Correct! Check Task 8 to see how we edited the static resources.

10.Question 10

app.yaml file is a configuration file that contains information on how the app should be handled once deployed e.g. URL handlers or what runtime environment is used

1 / 1 point

- ☒ True
- ☐ False

Correct

Correct!

11.Question 11

What is a kind in datastore?

1 / 1 point

- ☒ An entity category for query purposes
- ☐ An entity property that can be changed by any user
- ☐ A yaml file setting for datastore usage

Correct

Correct! Check Task 10 to review datastore!

12.Question 12

What is GOOGLE_APPLICATION_CREDENTIALS?

1 / 1 point

- ☐ A variable set in Python with the project name
- ☒ An environmental variable containing the key to supply credentials to the datastore service for authentication and usage
- ☐ A javascript array that contains all accounts information to be used in the templates

Correct

Incorrect! Check task 9 for more information on how to set up datastore