Hands-on Steps for AWS Rekognition:

1. Make sure you have completed the module steps available here:
   https://github.com/norawebbwilliams/cambridge_elements/blob/master/notes/02-use-aws
   -rekognition.md
2. Find your 'auto_tagger_example' directory (ours is on the Desktop) and take a quick look
   at the images in the 'data' directory to get a sense of what's there.
3. To run the Rekognition code on those images, you need to make **3 changes** to the
   python code. For each change, you need to update a **path**, meaning you need to point
   the program to a specific folder or file.

   Here's an example of a Windows path:
   'C:/Users/Nora/Desktop/auto_tagger_example/notebooks/' [Note the direction of the
   slashes and the slash at the end!]

   A Mac path might look more like:
   '/Users/Andreu/Desktop/auto_tagger_example/notebooks/' [Note the direction of the
   slashes and the slash at the end!]

4. You can update the code and run it in one of two ways:
   a. Update and execute using a jupyter notebook
      i. Open a bash terminal.
      ii. Navigate to your 'auto_tagger_example' directory by using the command
          'ls' (which **lists** the available files in your current location) and 'cd' which
          moves you to a new directory (**c**hange **d**irectory). In my bash terminal,
          this looks like the following, where the $ indicates that I've entered a
          command:

          ```
          $ cd Desktop/auto_tagger_example/
          $ ls
          code/  data/  keys/  notebooks/  results/
          ```

          If you need to go up a directory, type 'cd ..'

      iii. Start a jupyter notebook by typing 'jupyter notebook' and hitting
           enter in your bash terminal from the 'auto_tagger_example' directory. It
           may take a few minutes to start.
      iv. When the notebook opens in your web browser, navigate to 'notebooks'
          and then open '01_aws_rekog_labels.ipynb'
      v. In the **Paths** section, update the code with the paths to your results and
         data directories.
      vi. In the **Connect to Rekognition API** section, update the code with the
          path to your AWS access keys.

vii. Run each section of the code. If all goes well, you should see labels pop up, and there will be a new 'awsrekognition_detect_labels.csv' file in your 'results' directory.

viii. You can save the jupyter notebook to keep the paths you've entered for next time.

ix. Common errors: Make sure you are using a python 3.5+ kernel; make sure that your paths are correct; make sure that your have boto3 installed.

b. Update with a text editor and execute in bash

i. Use a text editor (Sublime, etc.) to open the '01_aws_rekog_labels_example_py3.5.py' script, located in the 'code' directory.

ii. In the **Paths** section, update the code with the paths to your results and data directories.

iii. In the **Connect to AWS Rekognition API** section, update the code with the path to your AWS access keys.

iv. Save the python script.

v. Open a bash terminal.

vi. Navigate to your 'auto_tagger_example/code' directory by using the command 'ls' (which **lists** the available files in your current location) and 'cd' which moves you to a new directory (**c**hange **d**irectory). Then navigate to the code directory In my bash terminal, this looks like the following, where the $ indicates that I've entered a command:

```
$ cd Desktop/auto_tagger_example/
$ ls
code/  data/  keys/  notebooks/  results/
$ cd code/
$ ls
01_aws_rekog_labels_example_py3.5.py
02_aws_rekog_celeb_example_py3.5.py
```

vii. Run the python script by typing

```
$ python 01_aws_rekog_labels_example_py3.5.py
```

then hit enter.

viii. If all goes well, you should see labels pop up in your terminal (it may take a few minutes), and there will be a new 'awsrekognition_detect_labels.csv' file in your 'results' directory.

ix. Common errors: Make sure you are in a python 3.5+ environment; make sure that your paths are correct; make sure that your have boto3 installed.

5. Follow the same steps to run the AWS celebrity recognition tagger (notebook and script 02).