Lab Assignment #1 – Interacting with Amazon web services using aws cli, the python sdk (boto3) and chalice frame work.

Due Date: Friday, Week 6

Purpose:

The purpose of this Lab assignment is to:

1. To interact with S3 Amazon web services using the software development toolkit.
2. Perform information extraction using Amazon comprehend and Amazon comprehend medical.
3. Update the architecture and implementation of an AI embedded application to include a new service.
4. Preform text to speech translation using “Amazon polly”.

General Instructions:

Be sure to read the following general instructions carefully:

1. This assignment must be completed individually by all the students.
2. Only provide the requested screenshots and make sure to have a complete screenshot, partial screenshots will not earn any marks.
3. You will have to add all the analysis and screenshots in the analysis report.
4. You will have to provide a **demonstration video for your solution** and upload the video together with the solution on **eCentennial** through the assignment link. See the **video recording instructions** at the end of this document.
5. In your 8-minute demonstration video you should explain your solution clearly, going over the main code blocks and the purpose of each method also demoing the execution of the code. Youtube links and links to google drive or any other media <u>are not acceptable</u>, the actual recording file must be submitted. The demonstration video is required for <u>exercises #1 and 3 only.</u>
6. Any submission without an accompanying video will lose 25% of the grade.
7. Any submission without an accompanying Analysis report will lose 20% of the grade.

Assignment Pre-requisites:

1. Aws cli
2. boto3 client
3. Chalice
4. pipenv virtual environment
5. Files as attached to this assignment

Exercise #1: Files upload (25%)

Uploading a group of files to s3 bucket using boto3
Exercise requirements:
Download the three txt files attached to this assignment, rename them to firstname1.txt, firstname2.txt, firstname3.txt, where firstname is your firstname.
Using the boto3 SDK write a python script to upload the three files to your S3 bucket. Take into consideration the following:

1. Create an especial environment, using pipenv to run your code.
2. Catch any client errors, checkout this link https://boto3.amazonaws.com/v1/documentation/api/latest/guide/error-handling.html to learn how to handle client errors. (hint: import logging)
3. Use a loop and printout the name of each file once the loading starts.
4. Record and printout the start time and the end time of the loading process for all three files.
5. Test the code.
6. Check the management console and take a full screenshot and add it to your analysis report.
7. Attach the python script to your submission name it firstname_filesuplolad.py where first name is your first name.
8. In the video you only need to explain your code.

(Once you are done, remove the text files from your S3 bucket to avoid charges)

Exercise # 2: Entity extraction (25%)

Introduction:
“Amazon Comprehend” is a Natural Language Processing (NLP) platform available as a service on AWS. It’s a collection of pre-trained NLP models that can be used through the management interface. Also, the services can be called either using the aws CLI or using an SDK to call the respective APIs’.
“Amazon medical comprehend” is variant of the comprehend service that specializes in extracting information from various medical documents.
Checkout the documentation at the following links:
https://docs.aws.amazon.com/comprehend/latest/dg/how-entities.html
https://aws.amazon.com/comprehend/medical/

Exercise requirements:

1) To check that your user is setup correctly issue the following command on your terminal:

```
$ aws iam get-user
```

   Copy the full response you received into your analysis report. Also, add a screenshot.
2) Prepare a text string that contains: your name, e-mail address, phone number, address, name of an organization you either study or work at.
   Example:
   Mayy Habayeb mhabayeb@my.centennialcollege.ca (416) 222 3441 233 Hoey crescent M3E1R7 Toronto Ontario Centennial college
3) Using aws CLI pass the string to “amazon comprehend” api, to detect all the entities in the string. Save the results of the api call to a json file name it firstname_comprehend_output.json. (Hint: use Linux re-direct)

4) Using aws cli, pass the same string to “amazon medical comprehend”. Use the feature extraction of Protected Health Information (PHI). Again, save the results of the api call to a json file, name it firstname_comprehend_medical_output.json.
5) In your analysis report:
    a. Note the syntax of the two api calls you issued on your Linux terminal.
    b. Create a table and compare the results of both api calls, check how good they did on identifying the type of entity and level of confidence. In addition, note the differences in how the service split the text string. Finally write some conclusions.
    c. Check the pricing of the comprehend service at this link https://aws.amazon.com/comprehend/pricing/ and estimate how much did it cost to execute each call and note that in your analysis report.
6) Attach both json files you created to the submission.

    Note: No need to cover in demonstration video.

Exercise # 3: Text to Speech (50%)

Recall the pictorial translator we built during weeks 4&5, using the same environment and code base, add a feature to read out the translated text to the end user.

First, you will need to investigate the use of polly amazon service at: https://docs.aws.amazon.com/polly/latest/dg/how-text-to-speech-works.html

Exercise requirements:

1) In your written analysis report you should provide the following:
    a. An updated architecture document for the AI framework mentioned in Module #4 WLO#2.
    b. An updated communications diagram for the project, illustrating the interactions with the various components including any new components, you can do that using Microsoft Visio.
    c. List and explain the design decisions you have taken and justify why you have taken them, there is at least one design decision to take for this problem.
2) Update the project code to include the new feature of reading out the translated text.
3) Attach to your submission, all the python code scripts, test cases and voice output files in a zipped folder name it firstrname_speaking_pictorial. Please document every change in the code.
4) Test case two images with foreign language text, make sure you use reasonably sized images not to exceed 300KB. Also check the supported languages.
5) For Html changes you are free to use any code, refer to the following as one option for audio https://www.w3schools.com/html/html5_audio.asp.
6) In the demo video:
    a. Explain your code changes.
    b. Show case one test case.

(Once you are done, remove the images from your S3 bucket to avoid charges)

Naming and Submission Rules:

1. You must name your submission as specified in each exercise.
2. Upload the submission file on e-Centennial using the Assignment link(s).
3. In total you should submit the following:
    a. One demonstration video, name it firstname_comp264_assignment#1_demo.
    b. One analysis report for all three exercises, the format is either as a word document or pdf, name it firstname_comp264_assignment#1_analysis_report.
    c. 1 python script for exercise #1.
    d. 2 json files for exercise #2.
    e. 1 zipped file containing the project code for exercise #3.

Rubric *(applies to each exercise, for exercise #2 the demonstration weight will be part of the written analysis content)*

Evaluation criteria	**Not acceptable**	**Below Average**	**Average**	**Competent**	**Excellent**
	0% - 24%	**25%-49%**	**50-69%**	**70%-83%**	**84%-100%**
Requirements in exercises 50%	Missing all requirements required	Some requirements are implemented.	Majority of requirements are implemented but some are malfunctioning.	Majority of requirements implemented.	All requirements are implemented Correctly.
Instruction/ Code Documentation on python script 5%	No comments explaining code. Missing all or majority of attachments.	Minor comments are implemented. Missing some required attachments.	Some code is correctly commented. Missing some required attachments.	Majority of code is correctly commented. Requested screen shots and files are attached.	All code is correctly commented. Requested screen shots and files are attached.
Written analysis report content 15%	Missed all the key ideas; very shallow.	Shows some thinking and reasoning but most ideas are underdeveloped.	Indicates thinking and reasoning applied with original thought on a few ideas.	Indicates original thinking and develops ideas with sufficient and firm evidence.	Indicates synthesis of ideas, in-depth analysis and evidences original thought and support for the topic.
Written analysis report format and organization 5%	Writing lacks logical organization. It shows no coherence and ideas lack unity. Serious errors. No transitions. Format is very messy.	Writing lacks logical organization. It shows some coherence but ideas lack unity. Serious errors. Format needs attention, some major errors.	Writing is coherent and logically organized. Some points remain misplaced. Format is neat but has some assembly errors.	Writing is coherent and logically organized with transitions used between ideas and paragraphs to create coherence. Overall unity of ideas is present. Format is neat and correctly assembled.	Writing shows high degree of attention to logic and reasoning of all points. Unity clearly leads the reader to the conclusion. Format is neat and correctly assembled with professional look.
Demonstration Video 25%	Very weak no mention of the code changes. Execution of code not demonstrated.	Some parts of the code changes presented. Execution of code partially demonstrated.	All code changes presented but without explanation why. Code demonstrated.	All code changes presented with explanation, exceeding time limit. Code demonstrated.	A comprehensive view of all code changes presented with explanation, within time limit. Code demonstrated.

Demonstration Video Recording

Please record a short video (max 8 minutes) to explain/demonstrate your assignment solution. You may use the Windows 10 Game bar to do the recording:

1. Press the Windows key + G at the same time to open the Game Bar dialog.
2. Check the "Yes, this is a game" checkbox to load the Game Bar.
3. Click on the Start Recording button (or Win + Alt + R) to begin capturing the video.
4. Stop the recording by clicking on the red recording bar that will be on the top right of the program window.

(If it disappears on you, press Win + G again to bring the Game Bar back.)

You'll find your recorded video (MP4 file), under the Videos folder in a subfolder called Captures.

Or

You can use any other video recording package freely available.