SDEV 400 – Homework 2
AWS CLI and DynamoDB

Overview:

In this homework, you will demonstrate the successful creation and use of tables using AWS CLI and AWS DynamoDB services.

Assignment: Total 100 points, 17.5% of the overall class grade

Prior to attempting this assignment, be sure you have completed the readings and activities in week 3 and 4. These activities are foundational and required to be successful in completing this homework.

Also, please get started early as this assignment will take you longer than you think.

1. (35 points) Using your AWS Educate Cloud9 **AWS CLI** environment create a Table named Sensors with a Hash Key named Sensor. Sensor is an attribute that uniquely identifies an item in the database. Sensor will be of type String. Additional attributes that may be included in each database item include:
   - SensorDescription (String) – A string that describes the Sensor
   - ImageFile (String) – A filename that provides an image of the sensor. The filename is fully qualified to include to path location. (e.g. /Sensors/images/acoustic8-elementarray.jpg)
   - SampleRate (Number) – The sample rate of the sensor (e.g. 2048)
   - Locations (Set) – A set of locations where this Sensor has been used to collect data. (e.g. {Aberdeen MD, Warren MI, Orlando FL})

   After successfully creating the table, continue to use the AWS-CLI to load at least 20 different sensor items. The sensor items should be stored in a JSON file. When creating the JSON file, be sure to not include all attributes for each item. For example, all items must have the Sensor attribute, but some items might be missing the ImageFile, Sample, Locations or SensorDescription attributes. In addition, add additional attributes (of your choosing), on the fly, to at least 5 of the items.

   Finally, while still using the AWS-CLI, write a statement that prints all items in the table.

   Provide and describe your AWS-CLI commands used for this exercise. Provide screen captures supporting the successful execution of each command.

2. (40 points) Using the AWS SDK and Python within your AWS Educate Cloud9 environment, create a DynamoDB table and functionality as described below:

- Table named Courses to hold attribute including Subject (e.g. SDEV), CatalogNbr (e.g. 400), Title (e.g. Secure Programming in the Cloud), NumCredits (e.g. 3), and a CourseID (e.g. 001). The CourseID should represent the Hash Key for the Courses table.
- Input 10 Course items of your choice. Note that all attributes are required for each item entered in the Courses table.
- Provide a command line simple interface that allows a user to search for a title given the Subject and CatalogNbr. The search program should continue to loop until the user requests to exit. Also, if both Subject and CatalogNbr are not entered, the program should request the data be re-entered. For example the following interface would fulfill this requirements:

```
Enter the Subject:
SDEV
Enter the CatalogNbr:
300
The title of SDEV 300 is Building Secure Python Applications.
Would you like to search for another title? (Y or N)
Y
Enter the Subject:
SDEV
 Enter the CatalogNbr:
A CatalogNbr is required. Enter the CatalogNbr:
400
The title of SDEV 400 is Secure Programming in the Cloud.
Would you like to search for another title? (Y or N)
N
Thanks for using the Catalog Search program.
```

Provide your fully test Python code to accomplish these tasks. Provide screen captures supporting the successful execution of the program and running of the command line search query code.

3. (15 points) Use the AWS CLI to delete all DynamoDB tables that were created for this exercise. Provide AWS CLI responses and the commands used to successfully complete this task in your report.

4. (10 points) Submit your report in Word or PDF format will all required components no later than the due date.

This includes the following Submission details:

1. Page numbers should be included for all pages (except the title page) and be at the top right of the page.
2. Paragraphs should be double-spaced with 1" margins on all sides.
3. 12 pt. Times New Roman font or similar should be used.
4. Figures should have titles and numbers.
5. The document should contain minimal spelling and grammar errors.
6. References are included (you should reference the code used) and provided in APA format.

Any submissions that do not represent work originating from the student will be submitted to the Dean's office and evaluated for possible academic integrity violations and sanctions.