PySpark Exercises

1. **Answer the following questions by using pyspark.RDD**
   a. Using **Capitals.txt** dataset;
      i. Find two closest capital cities and the distance between them.
      ii. Find two capital cities furthest away from each other and the distance between them.
   b. Using **EarthquakeDataset-Latest.txt** dataset;
      i. Please find the list of foreshocks and aftershocks (within 20 km and in 24 hours) for top ten earthquakes between years 1900 and 2022.
   c. Using **Lottery.txt** dataset,
      i. Please find top most commonly drawn **triple numbers**.
   d. Using **DollarDataset.txt** dataset,
      i. Please find top 5 greatest daily increase (by percentage)

2. **Answer the following questions by using pyspark.sql.DataFrame**
   Extract following informations from movielens dataset. There are several versions of movielens dataset you can use the smallest one (ml-latest-small.zip) https://grouplens.org/datasets/movielens/latest/
   a. For each tag (not genre), find average ratings of movies and sort them according to average rating. (Sample results below are not correct !)

Tag	Average Rating
Funny	3.8
Pixar	3.5

   b. Find top 10 (sorted by their similarity) most similar users for each user. Similarity: You can use any similarity measure like Cosine, Manhattan, Euclidean ...etc. (or you can implement your own similarity measure)

3. Find the best classification method for leaf dataset.
   Original link for the dataset:
   https://archive.ics.uci.edu/ml/machine-learning-databases/00288/
   a. For each classification method find the best parameters by using cross validation (or train validation split.)
   b. After finding best method and parameter values, please create a table that shows all of your results as shown below (given values are not correct!)

Method	Parameters	Accuracy

Random Forest Classifier	Param1=0.1
	Param2=0.5
	…
		0.0001

Gradient-Boosted Classifier	Param1=0.3
	Param2=6
	…
		0.000000000000002

…	…	…

4. Use “auto-mpg.data.txt” dataset to answer the following question:
   a. Find the cars with the worst fuel efficiency (lowest mpg) for each **origin**. (1→USA, 2→Europe, 3→Asia)
   b. Add a new column named “Car-Type” that has following values according to acceleration. (User Defined Function)
      (0 - 7 secs → Fast Car, 7 - 12 secs Average Car, 12+ secs Slow Car)

c. We want to predict mpg for given automobile info. Choose one of the ML algorithms from Spark ML library and prepare data for training.
    i. **origin** column should be one hot encoded
    ii. **mpg** column is the label value.
    iii. try to use **PCA** to decrease the number of features by 1.

d. Create a model and print your test accuracy. (Evaluation)