**Assignment - AC50002 - Programming Languages for Data Engineering**

***Python Assignment 2***

Student ID: 2499976

**Report**

1. First of all, I read the assignment file to understand the problem.
2. Then I identified that the following actions must be done in order to solve the problem and then created functions for the relevant action required to perform each task:
   1. Map image (map7.png) must be used to mark the map locations
   2. BBox (Bounding box) must be set create the desired map
3. I used PyCharm as the development IDE.
4. The code below is uploaded in Github as a Python file:

Github URL: <https://github.com/MoonlightO2/Python/blob/main/Peiris%20-%20Assignment02.py>

**Coding**

| # import folium  # from folium.plugins import minimap  import numpy as np  import pandas as pd  import webbrowser  import matplotlib.pyplot as plt  from matplotlib.animation import FuncAnimation  import time  import gc  start = time.process\_time()  # Load data from CSV to Dataset "data"  data = pd.read\_csv('GrowLocations.csv')  print("\nGrow Locations Dataset - Shape (Before cleaning)")  print(data.shape)  # Displaying the list of column names  print('\nList of column names : ', list(data.columns))  # Removing erroneous data  print("\nShow all erroneous data....")  print(data.isna().sum())  print("\nRemoving erroneous data....")  data = data.dropna(subset=['Serial'])  print("\nErroneous data removed....")  print("\nGrow Locations Dataset - First 5 rows")  print(data.head())  print("\nGrow Locations Dataset - Shape (After cleaning)")  print(data.shape)  print("\nGrow Locations Dataset - Data Types")  print(data.dtypes)  # Print dataset into CSV - export to csv to see whether the dataset has all clean data  print("\nExport dataset into CSV - File created - GrowLocations-Cleaned.csv...")  data.to\_csv('GrowLocations-Cleaned.csv', index=False)  print("\nShow all unique types")  data = pd.DataFrame(data)  data.Type = data.Type.replace('Thingful.Connectors.GROWSensors.AirTemperature', 'Air Temperature')  data.Type = data.Type.replace('Thingful.Connectors.GROWSensors.BatteryLevel', 'Battery Level')  data.Type = data.Type.replace('Thingful.Connectors.GROWSensors.FertilizerLevel', 'Fertilizer Level')  data.Type = data.Type.replace('Thingful.Connectors.GROWSensors.Light', 'Light')  data.Type = data.Type.replace('Thingful.Connectors.GROWSensors.SoilMoisture', 'Soil Moisture')  data.Type = data.Type.replace('Thingful.Connectors.GROWSensors.WaterTankLevel', 'Water Tank Level')  # print(data.Type.unique())  print("\nTypes list")  print(data['Type'].value\_counts())  # find max/min, plug into a website, snip area as png and insert as plotmap  BBox = ((-10.592, 1.6848, 50.681, 57.985))  # read the image in, plot points over image  grow\_map = "map7.png"  truth\_plot = plt.imread(grow\_map)  fig, ax = plt.subplots(figsize=(8, 8), linewidth=0.1)  plot\_title = "Grow Locations UK"  ax.set\_title(plot\_title)  ax.set\_xlabel("Longitude")  ax.set\_ylabel("Latitude")  ax.set\_xlim(BBox[0], BBox[1])  ax.set\_ylim(BBox[2], BBox[3])  scat = ax.scatter(data.Latitude, data.Longitude, zorder=1, alpha=0.5, c='b', s=10)  color\_data = np.random.random((500, len(data.Latitude)))  def update(frame):  scat.set\_array(color\_data[frame])  return scat,  anime = FuncAnimation(fig, update, frames=range(500), blit=True)  ax.imshow(truth\_plot, zorder=0, extent=BBox, aspect='equal')  print("\nLoading map image....")  print("\nTime taken to load map image....", time.process\_time() - start)  plt.show()  print("\nMap image closed....")  print("\n\nTotal time taken: ", time.process\_time() - start)  print("\nDeleting data to clear memory....")  del data  gc.collect()  try:  print(data)  except:  print("\nMemory cleared successfully....") |
| --- |

**Assumptions**

1. None

**References**

1. How to create a map using Python - <https://www.youtube.com/watch?v=SasGzg3jm4o>
2. Read image - <https://www.youtube.com/watch?v=mnWbSWU9w4U>
3. How to Display, Modify and Save Images in Matplotlib - <https://www.youtube.com/watch?v=3oXZTy7hrAQ>
4. Creating a Simple Map with Folium and Python - <https://towardsdatascience.com/creating-a-simple-map-with-folium-and-python-4c083abfff94>
5. Bounding box - <https://pythonhosted.org/planar/bbox.html>
6. Tile layers - <https://nagasudhir.blogspot.com/2021/07/introduction-to-folium-for-interactive.html>
7. Drawing a track on top of a map picture with matplotlib - <https://blog.finxter.com/display-modify-and-save-images-with-matplotlib/>, <https://stackoverflow.com/questions/53651899/drawing-a-track-on-top-of-a-map-picture-with-matplotlib>
8. How to drop rows in Python Pandas - <https://www.youtube.com/watch?v=X_OaCq-E3Sw>
9. Location map - <https://github.com/manuelcaeiro/locaton-map>
10. Marker cluster - <https://www.youtube.com/watch?v=n7HUBNmXB5Y>
11. Python maps with folium - <https://www.youtube.com/watch?v=t9Ed5QyO7qY>
12. Count occurrences in Column - <https://www.marsja.se/pandas-count-occurrences-in-column-unique-values/>
13. CV2 image reading - <https://www.youtube.com/watch?v=ADV-AjAXHdc>, <https://learnopencv.com/how-to-select-a-bounding-box-roi-in-opencv-cpp-python/>
14. Time taken - <https://stackoverflow.com/questions/14452145/how-to-measure-time-taken-between-lines-of-code-in-python>
15. Write / export into CSV file - <https://www.easytweaks.com/write-python-list-txt-file/>
16. Clear memory - <https://www.pythonpool.com/python-clear-memory/>
17. Minimap - <https://www.youtube.com/watch?v=t9Ed5QyO7qY>
18. Animate a plot over an image in Python - <https://stackoverflow.com/questions/61942985/animate-a-plot-over-an-image-in-python>