Ajani Mnyandu – Python script

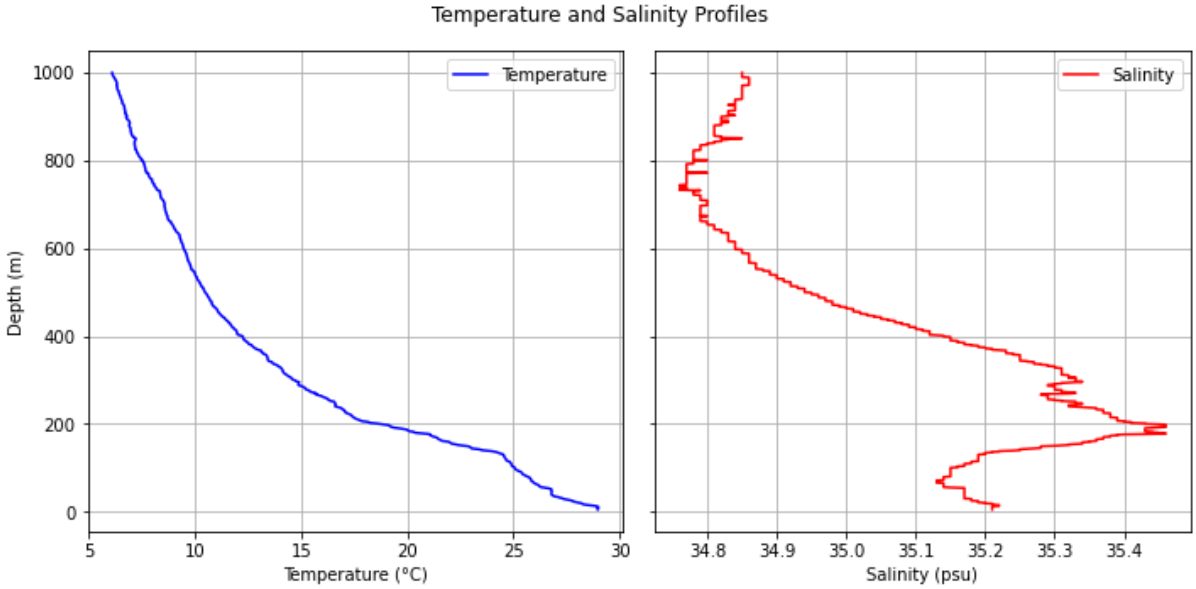
Part 1 – Temperature and salinity profiles with figures

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

salinity_data = 'C:\\Users\\LENOVO\\Documents\\Python Scripts'
salinity_data =  pd.read_csv('C:\\Users\\LENOVO\\Documents\\Python Scripts', index_col= 0)
print(data.head(salinity_data))

def ddmm2dd(ddmm):
    thedeg = np.floor(ddmm/100.)
    themin = (ddmm-thedeg*100.)/60.
    return thedeg+themin
ddmm2dd()

#define variables
date = salinity_data.iloc[:,0]
depth = salinity_data.iloc[:,2]
tempC = salinity_data.iloc[:,3]
salinity = salinity_data.iloc[:,4]

fig, axs = plt.subplots(1, 2, figsize=(10, 5), sharey=True)



axs[0].plot(tempC, depth, color='blue', label='Temperature')
axs[0].set_xlabel('Temperature (°C)')
axs[0].set_ylabel('Depth (m)')
axs[0].invert_yaxis()  # Invert y-axis to have depth increasing downwards
axs[0].grid(True)
```

```
axs[0].legend()
```

```
axs[1].plot(salinity, depth, color='red', label='Salinity')
axs[1].set_xlabel('Salinity (psu)')
axs[1].invert_yaxis()  # Invert y-axis to have depth increasing downwards
axs[1].grid(True)
axs[1].legend()
```

```
plt.suptitle('Temperature and Salinity Profiles')
plt.tight_layout()
plt.show()
```



Part 2

```
import pandas as pd
import matplotlib.pyplot as plt
```

```
data = pd.read_csv('C:/Users/LENOVO/Documents/Python Scripts/SAA2_WC_2017_metocean_10min_avg.csv')
```

```
# Check for missing values
print("Missing values before handling:\n", data.isnull().sum())



# Select data from departure to July 4th
departure_date = '2017/06/28 17:10 '  # Replace with actual
departure date
july_4th_date = '2017/07/04 11:20'   # Replace with July 4th date
selected_data = data.loc[departure_date:july_4th_date]



# Plot time series of temperature
plt.figure()
selected_data['AIR_TEMPERATURE'].plot(label='Temperature')
plt.xlabel('TIME_SERVER')
plt.ylabel('Temperature (°C)')
plt.title('Time Series of Temperature')
plt.style.use('grayscale')
plt.legend()
plt.savefig('temperature_time_series.png', dpi=200)

# Plot histogram of salinity
plt.figure()
selected_data['TSG_SALINITY'].plot.hist(bins=range(30, 36),
edgecolor='black')
plt.xlabel('Salinity (psu)')
plt.ylabel('Frequency')
plt.title('Histogram of Salinity')
plt.savefig('salinity_histogram.png', dpi=200)

# Create scatter plot of wind speed vs air temperature with latitude
encoded in color
plt.figure()
```

```
plt.scatter(selected_data['WIND_SPEED_TRUE'],
selected_data['AIR_TEMPERATURE'], c=selected_data['LATITUDE'],
cmap='viridis')

plt.colorbar(label='Latitude')

plt.xlabel('Wind Speed (m/s)')

plt.ylabel('Air Temperature (°C)')

plt.title('Scatter Plot of Wind Speed vs Air Temperature')

plt.savefig('wind_speed_vs_air_temperature_scatter.png', dpi=200)


plt.show()
```