Guidelines for Utilizing the Source Code in ERA5 Metocean Data Acquisition and Analysis

1) Metocean Data Acquisition from ERA5 Dataset

Before running this cell, users should:

- **Register for a CDS Account**:
    - Create an account with the Copernicus Climate Data Store at https://cds.climate.copernicus.eu if you don't already have one. This account is necessary to access ERA5 data.
- **Configure API Key and Client**:
    - Obtain the API key from the CDS website (under the user profile) and set up the **cdsapi** on the local machine. This involves creating a **.cdsapirc** file in the home directory with the API credentials. Detailed setup instructions can be found in the CDS API documentation.
- **Set Up API Client**:
    - Ensure that the **cdsapi** library is installed and set up properly. If not installed, use **pip install cdsapi** to install it.
- **Specify Output Directory**:
    - Path: Update the **output_directory** variable with the desired file path where the .nc files will be saved.
- **Customize Filename**:
    - Adapt the **output_filename** to suit the naming convention. For example, change **f'era5_data{year}.nc'** to include specific identifiers relevant to the project or location.
- **Modify Data Retrieval Parameters**:
    - **Time Range**: Adjust the year range in **for year in range(1972, 2023)** if a different period is in interest.
    - **Variables**: Customize the 'variable' list in the **c.retrieve** function to include only the required data parameters.
    - **Geographical Area**: Fill in the **area** parameter with the specific latitude and longitude values for the desired data collection location.
- **Review Data Format**: Confirm that the **format** parameter in the **c.retrieve** function is set to 'netcdf' or change it to another format if required.

2) Processing and Combining ERA5 Metocean .nc File Data into Excel File

Before running this cell, users should:

- **Specify the Directory for NetCDF Files**:

    - Update **nc_files_directory** with the path where the collected .nc (NetCDF) files are stored.

- **Customize File Naming**:
    - Modify the filename in **file_path** to match the naming convention of the.nc files. For instance, replace **"era5_data_{year}.nc"** with the specific file naming pattern.

- **Adjust Date Range for Data Collection**:
    - Change the year range in **for year in range(1972, 2023)** to match the period for which you have data.

- **Verify and Adjust NetCDF Data Variables**:
    - Check the variable names in the NetCDF file (like **'u10'**, **'v10'**, **'shww'**) and ensure they align with what you intend to analyze.

- **Wind Speed Calculation**:
    - The script calculates wind speed at 150m from 10m level data. If a different height is relevant to the analysis, adjust the calculation accordingly.

- **Data Frame Creation**:
    - The DataFrame columns are named **'Wind Speed at 10m (m/s)'**, **'Wind Speed at 150m (m/s)'**, and **'Significant Wave Height (m)'**. Rename these columns if different data is being extracted or if a different naming convention is preferred.

- **Saving the Combined Data**:
    - Customize the **output_excel_file** variable to name the Excel file according to the project's naming standards.

- **Confirmation Message Customization**:
    - Modify the print statement at the end of the cell to reflect the name of the output Excel file or to provide a different confirmation message.

3) Import Libraries and Load Data

Before running this cell, users should:

- **Install Required Libraries** (if not already installed):
    - Ensure libraries like **numpy**, **pandas**, **matplotlib**, **xarray**, **scipy**, and any others used in the cell are installed in the Python environment. Install these using pip (e.g., **pip install numpy pandas matplotlib xarray scipy**).

- **Update File Path for Data Loading**:
    - Modify **file_path** to the location where the Excel file is stored. Make sure the path is correctly formatted for the operating system. Change **path_to_your_excel_file** to something like **'C:\\Users\\era5_files\\combined_era5_data.xlsx'**

- **Data Verification**:

- After loading the data, the script prints the first few rows with **df.head()**. This is useful for a quick verification to ensure the data is loaded correctly.

4) Statistical Analysis of Wind Speed Data Using Weibull Distribution and Significant Wave Height Data Using Rayleigh Distribution

Before running this cell, users should:

- **Ensure Relevant Data in DataFrame**:
    - Ensure that the DataFrame df contains columns for 'Time (h)', 'Wind Speed at 150m (m/s)', and 'Significant Wave Height (m)'.
- **Libraries**:
    - Confirm that libraries such as **numpy**, **matplotlib.pyplot**, and **scipy.stats** are installed and available in the Python environment.
- **Plot Execution**:
    - Run the cell to generate the Weibull PDF and CDF plots. Ensure the Python environment supports graphical output from **matplotlib**.

6) Detailed Monthly Analysis of Weather Conditions in 8-Hour Intervals Over Years

Before running this cell, users should:

- **Check DataFrame Format**:
    - Ensure that the DataFrame **df** contains columns for **'Time (h)'**, **'Wind Speed at 150m (m/s)'**, and **'Significant Wave Height (m)'**.
    - You can update the lambda functions defining the weather conditions (WC1 to WC5) based on wind speed and wave height.
- **Analyze Data in 8-Hour Chunks**:
    - The script processes the data in 8-hour intervals for each month and year, applying the weather conditions to classify each interval for later visualization in Cell 7.
- **Modify Analysis Range (if necessary)**:
    - You can adjust the **years** range and **months** list if the analysis period differs from 1972 to 2023, but it’s a must.
- **Customize File Path for Output**:
    - Change **output_path** to the desired location for saving the Excel file with the analysis results.
- **Run and Review the Output**:
    - Execute the cell and verify the output in the generated excel file, especially if modifications were made to the conditions or data pre-processing steps.

7) Visualization of Monthly Weather Condition Availabilities for Selected Location

Before running this cell, users should:

- **Ensure Data Availability**:
    - Verify that the file **era5_monthly_weather_conditions.xlsx** generated by Cell 6(or the specific file) is available and contains the required data.
- **Update File Path**:
    - Modify **file_path** to the correct path where the Excel file is stored.
- **Review Data Structure**:
    - Ensure that the DataFrame **df** after loading should have columns for 'Year', 'Month', and weather conditions ('WC_1' to 'WC_5').
- **Customize Plot Settings**:
    - Adjust the figure size, aspect ratios, and color scheme in the plot as per your requirements.
    - Modify the **month_labels** and **year_labels** if the dataset has a different time range or specific labeling needs.
- **Execution and Verification**:
    - Run the cell to generate the plot. Verify that the visualization accurately represents the data and adjust settings if necessary.

8) 8-hour Interval-Based Weather Condition Analysis for Offshore Wind Installation Operations

Before running this cell, users should:

- **Review and Understand Weather Conditions**:
    - You can update the lambda functions defining the weather conditions. These functions categorize the data based on wind speed and wave height thresholds.
- **Verify Data Granularity**:
    - Ensure that the data in **df** is structured to allow for analysis in 8-hour intervals.
    - Rerun Cell 3 If Needed: If you face issues like KeyError while running Cell 8, rerun Cell 3 to reload and reprocess the DataFrame df, as it's sometimes required to refresh data in the Jupyter environment.
- **Modify Analysis Range**:
    - Adjust the **years** and **months** variables if the analysis period is different from 1972 to 2023.

- **Set Output File Path**: Update the **output_path** to the preferred location for saving the analysis results in an Excel file.
- **Execute and Review Output**: After running Cell 8, carefully review the generated Excel file to confirm its alignment with the requirements for the simulation tool. This file is critical for simulating installations of floating offshore wind plants tool and needs be generated separately for each interest of location and provided to the simulation tool's for cell.