MPLNet Data Tool

Last updated

3 December 2023

Contents

1   Description

(PLACEHOLDER) The objective of this project is to develop a functional graphical user interface for seamless communication and data retrieval from NASA's Micro Pulse Lidar Network (MPLNet).

The application is designed to streamline the downloading process, providing users with a user-friendly interface while eliminating the need for direct access to NASA's data site.

It enables comprehensive data selection from AppState's MPLNet site, facilitating the aggregation and transformation of data from netCDF4 format to CSV format. This enhancement aims to expedite further research endeavors.

2 buildDirList

Description
Returns a list of directories to be used for downloading the files

Usage
buildDirList(years, months, days, data_path='..data')

Arguments
years (list): The list of years
months (list): The list of months
days (list): The list of days
data_path (str): The path to the data folder (default: '../data/')
This is the path relative to the current working directory

Returns
dirs (list): The list of directories to be downloaded

3 buildFileList

Description
Returns a list of files to be downloaded based upon the user input

Usage
buildFileList(years, months, days, fileTypes)

Arguments
years (list): The list of years
months (list): The list of months
days (list): The list of days
fileTypes (list): The list of file types

Returns
files (list): The list of files to be downloaded

4 buildURLList

Description
Returns a list of urls to be downloaded based upon the user input

Usage
buildURLList(years, months, days, fileTypes)

Arguments
years (list): The list of years
months (list): The list of months
days (list): The list of days
fileTypes (list): The list of file types

Returns
urls (lis1): The list of urls to be downloaded

5 buildSelectionList

Description
Returns a list of months, days, and files for selection

Usage
buildSelectionList(years)

Arguments
years (list): The years available for a given site

Returns
months (list): The months available for a given site and year
days (list): The days available for a given site, year, and month
fileTypes (list): The files available for a given site, year, month, and day

6 create_directory

Description
Returns the url for the mplnet file path

Usage
create_directory(path)

Arguments
path (str): The path to be folder to be created

Returns
None

7 create_export_name

Description

Usage
create_export_name(selectedVars, variable)
Arguments
selectedVars (object): object containing selected variables
variable (str): name of variable to export

Returns
filename (str): name of export file

8 directory_select

Description
Returns user selected directory from a list
Depreciated as GUI will guide user selection

Usage
directory_select(name, options)

Arguments
name (str): The name of the directory
options (list): The options for the directory

Returns
str: The user selected directory

9 export

Description
Export data to csv file
Uses current data directory to save csv file

Usage
export(filename, files, variable)

Arguments
filename (str): name of output csv file
files (list): list of full file paths
variable (str): name of variable to export

Returns
None

10 file_select

Description
Returns user selected file from a list
Depreciated as GUI will guide user selection

Usage
file_select(file_list)

Arguments
file_list (list): The list of files

Returns
str: The user selected file

11   get_mplnet_data

Description
Writes the mplnet data from file at url to file_path_name

Usage
get_mplnet_data(url, file_path_name)

Arguments
url (str): The url for the mplnet data file
file_path_name (str): The full path and name of the file to be saved

Returns
Boolean value indicating if the file was downloaded and saved successfully

12   get_mplnet_html

Description
Returns the html for the mplnet address given
Used to scrape for file variables to assist in downloading

Usage
get_mplnet_html(url)

Arguments
url (str): The url for the mplnet html

Returns
requests.Response (object): The html for the mplnet address given

13   get_user_path

Description
Returns the user specified path for the mplnet data

Usage
get_user_path()

Arguments
None

Returns
str: The user specified local path for the mplnet data

14 leap_year

Description
Returns true if year is a leap year

Usage
leap_year(year)

Arguments
year (int): The year to be checked

Returns
bool: True if year is a leap year

15 parse_day

Description
Returns a list of the days available for a given site, year, and month

Usage
parse_day(html)

Arguments
html (str): The html for the mplnet address given

Returns
list: The days available for a given site, year, and month

16 parse_file

Description
Returns a list of the files available for a given site, year, month, and day

Usage
parse_file(html)

Arguments
html (str): The html for the mplnet address given

Returns
list: The files available for a given site, year, month, and day

17 parse_month

Description
Returns a list of the months available for a given site and year

Usage
parse_month(html)

Arguments
html (str): The html for the mplnet address given

Returns
list: The months available for a given site and year

18 parse_year

Description
Returns a list of the years available for a given site
Usage
parse_year(html)

Arguments
html (str): The html for the mplnet address given

Returns
list: The years available for a given site

19 FileVariables (class)

Description
Class to build and store file variables

Usage
fileVar = FileVariables()

Attributes
filetype (str): The file type
filevars (list): The file variables
nextIter (int): The next file variable to be filled
selectedFileType (str): The selected file type
selectedFileVars (list): The selected file variables

Methods
setFileTypes(ft): Store the file types from the downloaded files
into the filetype variable to allow user selection
next(): Returns the next file variable to be filled
based upon the user input
peakNext(): Returns the next file variable to be filled
based upon the user input
storeCurrent(value): Returns the current file variable to be filled
based upon the user input
setFileVars(file): Returns the file variables for the selected file type
printSelected(var): Prints the selected file variable

20 FileVariables.next

Description
Returns the next file variable to be filled
based upon the user input

Usage
fileVar.next()

Arguments
None

Returns
The next file variable to be filled

21   FileVariables.peakNext

Description
Returns the next file variable to be filled
based upon the user input

Usage
fileVar.peakNext()

Arguments
None

Returns
Boolean value indicating if there is a next file variable

22   FileVariables.printSelected

Description
Prints the selected file variable

Usage
fileVar.printSelected(var)

Arguments
var (str): The selected file variable

Returns
String with the selected file variable

23   FileVariables.setFileTypes

Description
Store the file types from the downloaded files
into the filetype variable to allow user selection

Usage
fileVar.setFileTypes(ft)

Arguments
ft (list): The list of file types

Returns
None

24   FileVariables.setFileVars

Description
Opens the netcdf file and extracts the variables and populates
the filevars dictionary

Usage
fileVar.setFileVars(file)

Arguments
files (str): A full path to the netcdf file to be opened

Returns
None

25   FileVariables.storeCurrent

Description
Stores the current file variable to be filled
based upon the user input and the nextIter variable

Usage
fileVar.storeCurrent(value)

Arguments
None

Returns
None

26   SelectionVariables (class)

Description
Class to build and store selection variables

Usage
selVar = SelectionVariables()

Attributes
years (list): The list of years
months (list): The list of months
days (list): The list of days
fileTypes (list): The list of file types
selectedYears (list): The list of selected years
selectedMonths (list): The list of selected months
selectedDays (list): The list of selected days
selectedFileTypes (list): The list of selected file types
nextIter (int): The next selection variable to be filled

Methods
getVars(html): Returns the years, months, days, and file types from the html
setSelectVars(years, months, days, fileTypes): Sets the selection variables based upon the user input
next(): Returns the next selection variable to be filled
based upon the user input
peakNext(): Returns the next selection variable to be filled
based upon the user input
storeCurrent(value): Returns the current selection variable to be filled
based upon the user input
prepDownload(): Returns the urls, directories, and files to be downloaded
download(url, dir, file): Downloads the selected files
checkSelection(): Checks to make sure a selection is made in each category
reset(): Resets the nextIter variable to 0
printSelected(): Prints the selected variables

27 SelectionVariables.checkSelection

Description
Checks to make sure a selection is made in each category

Usage
selVar.checkSelection()

Arguments
None

Returns
Boolean value indicating if a selection is made in each category

28 SelectionVariables.download

Description
Downloads the selected files

Usage
selVar.download(url, dir, file)

Arguments
url (str): The url for the mplnet data file
dir (str): The full path of the file to be saved, not including the file name
file (str): The name of the file to be saved

Returns
String with the file name and status of the download

29 SelectionVariables.getVars

Description
Builds the years, months, days, and file types from the html
to be used in the selection of files to be downloaded

Usage
selVar.getVars(html)

Arguments
html (str): The html from the mplnet website

Returns
None

30 SelectionVariables.next

Description
Returns the next selection variable to be filled
based upon the user input

Usage
selVar.next()

Arguments
None

Returns
The next selection variable to be filled or None

31 SelectionVariables.peakNext

Description
Returns the next selection variable to be filled
based upon the user input

Usage
selVar.peakNext()

Arguments
None

Returns
Boolean value indicating if there is a next selection variable

32 SelectionVariables.prepDownload

Description
Prepares a tuple the selected files to be downloaded including the urls, directories, and files
Usage
selVar.prepDownload()

Arguments
None

Returns
Tuple of lists containing the urls, dirs, and, files to be downloaded

33 SelectionVariables.printSelected

Description
Prints the selected variables

Usage
selVar.printSelected()

Arguments
None

Returns
String with the selected variables

34 SelectionVariables.reset

Description
Resets the nextIter variable to 0 and clears the selected values

Usage
selVar.reset()

Arguments
None

Returns
None

35 SelectionVariables.storeCurrent

Description
Returns the current selection variable to be filled
based upon the user input

Usage
selVar.storeCurrent(value)

Arguments
value (list): The current selection values to be stored

Returns
None