globe

distance

Description

Allow to return the distances between multiple geographical coordinates and another geographical point.

Usage

```
distance(lat1, long1, lat_l, long_l, alt_l=None, alt1=None)
```

`lat1` is the latiitude of the established point

`long1` is the longitude of the established point

`alt1` is the altitude of the established point, if not given the distances calculated won't take in count this parameter

`lat_l` is a list containing the latitudes of the geographical points to be compared to the established point

`long_l` is a list containing the longitudes of the geographical points to be compared to the established point

`alt_l` is a list containing the altitudes of the geographical points to be compared to the established point, if not given the distances calculated won't take in count this parameter

unnester

Class composed of functions for list manipulation.

nestfind

Description

Allow to access to the list or the element of the n list in depth from the main list.

Usage

```
nestfind(input_l, dim_search)
```

`input_l` is the input list

`dim_search` is the dimension of the list or the element to find

Example

```
>l = [[1, 3, 2, [5, 6], 5], 4, [7, "ee"]]

>dim_search = [0, 3, 1]

>nestfind(input_l=ldim_search=dim_search)

6

>dim_search = [0, 3]

[5, 6]
```

ns

Description

Function whose goal is to manipulate nested list.

Usage

```
>unnester.ns(input_l, dim_end=1, strt_l=[], rtn_l=[], id_rec_main=0,
wrk_l=None, flag_l=[])
```

The two parameters you need to know are input_l and dim_end. The fact that it is a recursive function requires the presence of the others paramters that are used each iteration of the recursive function.

`input_l` The nested list you want to unnest to a certain point. `dim_end` The dimension from which you want to keep.

Example

```
>unnester.ns(input_l=[1, [5, [[2], 4, [23, 3, 3]]], 2, 3334,
[4, [55, 56], 7, [77, [66, 67], 78], 2, [33, 5]], 3, [5, 6], 4],
dim_end=3, strt_l=[], rtn_l=[], flag_l=[])


[1, 5, [2], 4, [23, 3, 3], 2, 3334, 4, 55, 56, 7, 77, [66, 67],
78, 2, 33, 5, 3, 5, 6, 4]
```

```
>unnester.ns([1, [2], 3], dim_end=1, strt_l=[], rtn_l=[], flag_l=[])

[1, [2], 3]

>unnester.ns([1, [2], 3], dim_end=2, strt_l=[], rtn_l=[], flag_l=[])

[1, 2, 3]
```

Here, we are forced to declare the list parameters in the function call because if not declared, it will take their last value. This is the case for python 3.11.6.