\*\*Documentation for the Python program "Arrays "\*\*

The following Python program generates an array of values based on the specified parameters.

## Function `arraygen(min, max, steps, scale)`

### Parameters:

- `min`: The minimum value of the array.

- `max`: The maximum value of the array.

- `steps`: The number of steps between `min` and `max`.

- `scale`: The scale on which the array should be generated (either 'Linear' or 'Exponential').

### Return Value:

The function returns an array `Pn\_array` along with the number of steps `steps`.

### Array Generation:

1. \*\*Linear Scale\*\* (`scale == 'Linear'`):

- The array is generated incrementally from `min` to `max`.

- Each step corresponds to a uniform interval between the values.

- The formula is: `Pn\_array[i] = min + i \* (max - min) / steps`.

2. \*\*Exponential Scale\*\* (`scale == 'Exponential'`):

- The array is generated incrementally from `min` to `max`.

- The values are scaled exponentially.

- The formula is: `Pn\_array[i] = exp(log(min) + i \* (log(max) - log(min)) / steps)`.

## Example Calls:

```python

import numpy as np

# Linear Scale

linear\_array, num\_steps = arraygen(0, 10, 5, 'Linear')

print(f"Linear array: {linear\_array} (Number of steps: {num\_steps})")

# Exponential Scale

exp\_array, num\_steps = arraygen(1, 100, 5, 'Exponential')

print(f"Exponential array: {exp\_array} (Number of steps: {num\_steps})")

```

Feel free to adjust the values of `min`, `max`, `steps`, and `scale` to explore different results.

4. \*\*Notes\*\*:

- Ensure that the required libraries (`numpy` and `matplotlib`) are installed.

- Customize the data and axis labels according to your application.