1. Pengurutan (Sorting) dalam Python

Pengurutan dengan `sorted()` dan `sort()`

- `sorted()`: Fungsi ini mengembalikan list baru yang sudah diurutkan, sementara list asli tidak berubah.
- `list.sort()`: Metode ini mengurutkan list di tempat (in-place) dan mengubah list asli.

Contoh penggunaan `sorted()`:

```python
data = [3, 1, 4, 1, 5, 9, 2, 6, 5, 3, 5]
sorted_data = sorted(data)
print(sorted_data)
```

Contoh penggunaan `list.sort()`:

```python
data = [3, 1, 4, 1, 5, 9, 2, 6, 5, 3, 5]
data.sort()
print(data)
```

Pengurutan dengan Kriteria Khusus

Anda bisa menggunakan parameter `key` untuk mengurutkan dengan kriteria khusus.

Contoh pengurutan berdasarkan panjang string:

```python
```

```
words = ["apple", "banana", "cherry", "date"]

sorted_words = sorted(words, key=len)

print(sorted_words)
```

2. Fungsi-fungsi Lain untuk Mengolah Array

`map()`

`map()` menerapkan fungsi tertentu ke setiap item dalam iterable (misalnya list) dan mengembalikan map object.

Contoh penggunaan `map()`:

```python
numbers = [1, 2, 3, 4, 5]

squared = map(lambda x: x**2, numbers)

print(list(squared))
```

`filter()`

`filter()` menyaring item dalam iterable berdasarkan fungsi yang mengembalikan nilai `True` atau `False`.

Contoh penggunaan `filter()`:

```python
numbers = [1, 2, 3, 4, 5]

even_numbers = filter(lambda x: x % 2 == 0, numbers)

print(list(even_numbers))
```

```

#### `reduce()`

`reduce()` menerapkan fungsi secara berurutan ke item dalam iterable, menggabungkan hasil sebelumnya dengan item berikutnya.

Contoh penggunaan `reduce()`:

```python
from functools import reduce

numbers = [1, 2, 3, 4, 5]
product = reduce(lambda x, y: x * y, numbers)
print(product)
```

3. Menghitung Jarak Menggunakan Rumus Haversine

Rumus Haversine digunakan untuk menghitung jarak antara dua titik pada permukaan bola, seperti bumi. Rumus ini memperhitungkan kelengkungan bumi.

Berikut adalah implementasi rumus Haversine dalam Python:

```python
import math

def haversine(coord1, coord2):
```
```

```
    # Koordinat (latitude, longitude)
    lat1, lon1 = coord1
    lat2, lon2 = coord2

    # Konversi derajat ke radian
    lat1, lon1, lat2, lon2 = map(math.radians, [lat1, lon1, lat2, lon2])

    # Haversine formula
    dlat = lat2 - lat1
    dlon = lon2 - lon1
    a = math.sin(dlat/2)**2 + math.cos(lat1) * math.cos(lat2) * math.sin(dlon/2)**2
    c = 2 * math.asin(math.sqrt(a))

    # Radius bumi dalam kilometer
    r = 6371
    return c * r

# Contoh penggunaan
coord1 = (-6.200000, 106.816666)  # Jakarta
coord2 = (-7.250445, 112.768845)  # Surabaya
jarak = haversine(coord1, coord2)
print(f"Jarak antara Jakarta dan Surabaya adalah {jarak:.2f} km")
```