

APLIKASI PENDETEKSI DUPLIKAT DATA

Khoirul Anam¹, Rendi Alexander Hutagalung², Andre Hadiman Rotua Parhusip³, Chintya Bella⁴, Lisa Diani Amelia⁵

Jurusan Sains Data, Fakultas Sains, Institut Teknologi Sumatera, Lampung Selatan, Indonesia

Email: khoirul.122450039@student.itera.ac.id rendi.122450057@student.itera.ac.id
lisa.122450021@student.itera.ac.id cinty

Pendahuluan

Database mempunyai sistem penyimpanan dengan jumlah yang banyak, dan dalam sebuah data biasanya terdapat data dengan atribut atau kolom yang sama. Karena data ini pengguna biasanya tidak sadar saat memanggil data yang salah akibat adanya kesamaan dalam datanya. Hal ini juga akan berdampak saat data ini akan dipanggil dan diolah serta akan mengakibatkannya banyaknya ruang penyimpanan yang digunakan.

Untuk dapat melihat data yang sama dalam sebuah database secara manual atau memeriksanya satu-satu, hal ini akan sangat merepotkan dan membutuhkan banyak waktu. Maka, sangat dibutuhkan solusi untuk menangani permasalahan ini untuk dapat mendeteksi data yang sama dan menghapusnya dengan cepat. Sehingga pengguna dapat memanggil dan menggunakan data dengan baik serta dapat menghemat ruang penyimpanan.

Dalam pemrograman berbasis fungsi terdapat fungsi `map ()` menerapkan fungsi ke setiap item yang dapat diulang dan menghasilkan hasil serta fungsi `reduce ()`. `Reduce` adalah sebuah fungsi yang digunakan untuk mengeksekusi nilai pada setiap element dengan tipe array dan menampilkan dalam sebuah nilai saja. Untuk menghapus data yang duplikat maka membutuhkan algoritma dengan menggunakan kedua fungsi diatas. Dengan menerapkan algoritma memeriksa di setiap element dan mengulangi ke elemen lainnya. Maka hasil pencarian data yang sama dan menghapusnya akan mempercepat pengguna tanpa harus membuka dan memeriksanya satu dengan lainnya.

Metode

Pada aplikasi pendeteksi duplikat data kami menggunakan built-in function dalam python yaitu;

1. Lambda

fungsi Lambda - juga dikenal sebagai “fungsi anonim” - yang memungkinkan kita untuk membuat dan menggunakan fungsi dalam satu baris. fungsi ini berguna ketika kita membutuhkan fungsi pendek yang hanya akan digunakan sekali.

2. Map

Fungsi `map()` adalah fungsi bawaan yang memungkinkan kita memproses dan mengubah semua item dalam iterable tanpa menggunakan perulangan yang eksplisit, sebuah teknik yang umumnya dikenal sebagai pemetaan. Digunakan ketika kita perlu menerapkan fungsi transformasi ke setiap item dalam sebuah iterable dan mengubahnya menjadi iterable baru.

3. Reduce

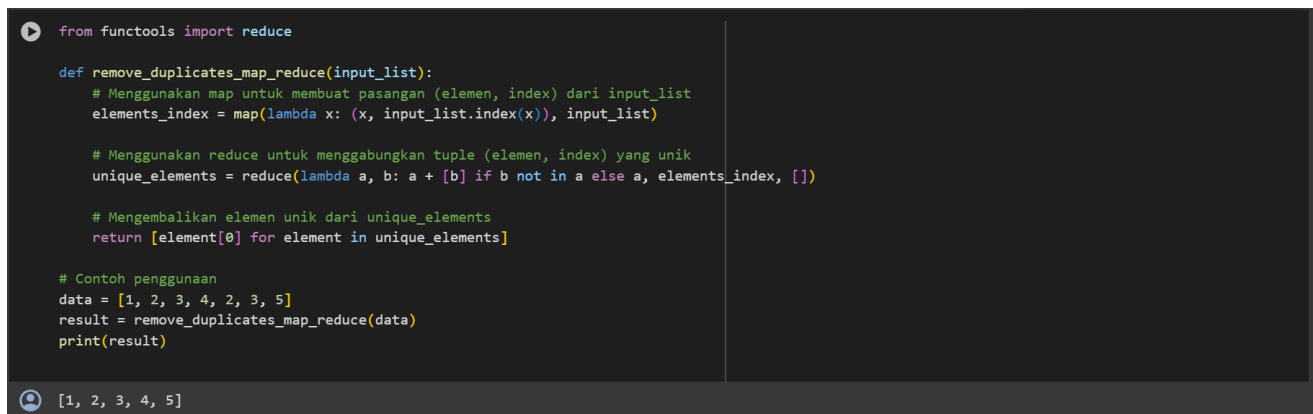
Fungsi `reduce()` Python adalah fungsi yang digunakan untuk menerapkan fungsi ke urutan nilai. Fungsi pengurangan() dirancang untuk melakukan perhitungan kumulatif pada urutan angka. Ini mengurangi urutan nilai menjadi satu nilai dengan melakukan operasi tertentu pada setiap nilai dalam urutan tersebut.

4. List Comprehension

List comprehension terdiri dari tanda kurung berisi ekspresi, yang dieksekusi untuk setiap elemen bersama dengan perulangan `for` untuk mengulangi setiap elemen dalam daftar Python. Pemahaman daftar mungkin mencakup tes `if` opsional

Pembahasan

Pada jurnal kali ini terdapat program pendeteksi duplikat data sebagai berikut:



```
from functools import reduce

def remove_duplicates_map_reduce(input_list):
    # Menggunakan map untuk membuat pasangan (elemen, index) dari input_list
    elements_index = map(lambda x: (x, input_list.index(x)), input_list)

    # Menggunakan reduce untuk menggabungkan tuple (elemen, index) yang unik
    unique_elements = reduce(lambda a, b: a + [b] if b not in a else a, elements_index, [])

    # Mengembalikan elemen unik dari unique_elements
    return [element[0] for element in unique_elements]

# Contoh penggunaan
data = [1, 2, 3, 4, 2, 3, 5]
result = remove_duplicates_map_reduce(data)
print(result)
```

[1, 2, 3, 4, 5]

Gambar 1. Program

Pada baris pertama kita menggunakan *define of module* dari library `functools` guna untuk menggunakan beberapa tools yang ada pada library tersebut seperti `reduce`. Pada baris ketiga kita mendefinisikan sebuah objek berdasarkan *user defined* yaitu `remove_duplicates_map_reduce` dengan atribut (`input_list`). Kemudian pada pendefinisian objek kita isi dengan suatu objek baru berdasarkan *user defined* yaitu `elements_index` dengan menggunakan `map` function di mana menerapkan fungsi ke setiap item yang dapat diulang dan

menghasilkan hasil. Lalu, pada baris berikutnya, kita isi dengan objek baru yang berdasarkan *user defined* yaitu `unique_elements` dengan menggunakan `reduce` function di mana daftar yang diberikan menjadi satu nilai. Lalu, pada baris berikutnya, kita mengembalikan elemen unit dari `unique_elements`. Selanjutnya, kita membuat daftar numerik dari angka satu sampai dengan 5 dengan nama objek data. Kemudian, menerapkan fungsi sebagai objek yakni fungsi `result` sama dengan objek `remove_duplicates_map` dengan atribut data (daftar numerik). Tahap akhirnya kita memanggil fungsi `result` itu sendiri sehingga menghasilkan output daftar `[1, 2, 3, 4, 5]`.

Kesimpulan

Aplikasi pendeteksi duplikat data ini menggunakan metode pemrograman fungsional dalam Python untuk mengidentifikasi dan menghapus data duplikat dengan efisien. Permasalahan duplikat data dalam database dapat menyebabkan kesalahan pemanggilan data, memperlambat pengolahan data, dan menggunakan ruang penyimpanan yang tidak efisien. Untuk mengatasi permasalahan tersebut, diperlukan solusi untuk mendeteksi dan menghapus data duplikat dengan cepat. Metode yang digunakan dalam aplikasi pendeteksi duplikat data mencakup penggunaan fungsi built-in Python seperti `lambda`, `map`, `reduce`, dan `list comprehension`. Program pendeteksi duplikat data menggunakan modul dari library `functools`, dan mengimplementasikan fungsi `map` dan `reduce` untuk menghapus duplikat dari sebuah daftar numerik.