

**TERRITORY MANAGEMENT PROJECT**

ENGINE GEOCODING DOCUMENTATION

OPERATIONAL PROCHEDURE AND LOGIC FLOW

Dipersiapkan oleh



**Dattabot - PT Mediatrac Sistem Komunikasi**  
Grha Tirtadi 2nd Floor   |   Jl. Senopati 71-73   |   Jakarta 12190   |   Indonesia   |   **Map** 6°13'57.37"S 106°48'42.29"E 

**P** +62 21 520 2568   |   **F** +62 21 520 4180   |   [**www.ci-mediatrac.com**](http://www.ci-mediatrac.com/)

**Daftar Isi**

[Daftar Isi 1](#_Toc485809532)

[1. Pendahuluan 1](#_Toc485809533)

[1.1. Tujuan dokumen 1](#_Toc485809534)

[1.2. Definisi dan Singkatan 1](#_Toc485809535)

[1.3 Daftar Revisi 1](#_Toc485809536)

[2. Menyiapkan Environment 1](#_Toc485809537)

[2.1. Python 1](#_Toc485809538)

[2.2. Cloning Project dari Gitlab 1](#_Toc485809539)

[2.3. Python Library 1](#_Toc485809540)

[3. Menjalankan Engine 1](#_Toc485809541)

[3.1. Mempersiapkan Input Data 1](#_Toc485809542)

[3.2. Menjalankan Engine via Command Line 1](#_Toc485809543)

[3.3. Memahami Output dari Engine Geocoding 1](#_Toc485809544)

[4. Engine Architecture 1](#_Toc485809545)

[4.1. Logic Flow 1](#_Toc485809546)

[4.2. Struktur Code 1](#_Toc485809547)

# 1. Pendahuluan

Dokumen ini berisi informasi mengenai Prosedur Penggunaan Engine Geocoding serta architecture dari engine tersebut sehingga memudahkan proses pemeliharaan atau versioning code. Dokumen ini disiapkan untuk Data dan GIS engineer untuk keperluan pekerjaan Geocoding. User yang dapat menggunakan engine ini ditujukan pada user yang memiliki pengetahuan mengenai basic command line dan pemahaman akan database.

## 1.1. Tujuan dokumen

Tujuan dari pembuatan dokumen ini adalah:

1. Dokumentasi proses atau prosedur tetap dalam melakukan proses geocoding.
2. Dokumentasi engine architecture untuk memudahkan proses pemeliharaan atau versioning code.

## 1.2. Definisi dan Singkatan

|  |  |
| --- | --- |
| **Singkatan** | **Definisi** |
| MSK | PT. Mediatrac Sistem Komunikasi |

## 1.3 Daftar Revisi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Editor** | **Versi** | **Tanggal** | **Deskripsi** |
| 1 | Mia Renauly | 1.0 | 21-06-2017 | Initiator |
| 2 |  |  |  |  |

# 2. Menyiapkan Environment

## Python

Engine Geocode menggunakan Python versi 3.6. Disarankan untuk menggunakan versi tersebut atau versi Python yang lebih baru. Untuk file instalasi Python dapat diunduh langsung di website resmi Python: [https://www.python.org/downloads/](https://www.python.org/downloads/%20).

Jika membutuhkan lebih dari satu environment Python, sebaiknya lakukan instalasi Python menggunakan Anaconda dari Continuum Analytics, kunjungi link berikut: <https://www.continuum.io/downloads>. Pilih Python 3.6 Version versi 64 bit.

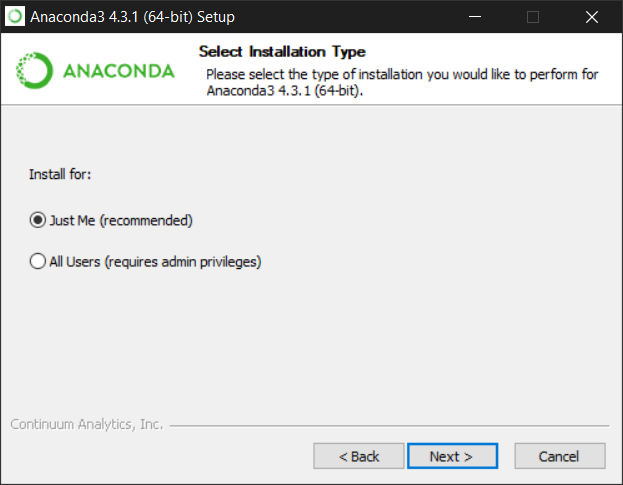
Berikut ini merupakan langkah instalasi dan cara membuat environment pada Anaconda:

1. Buka file Installer Anaconda. Akan muncul Jendela instalasi:



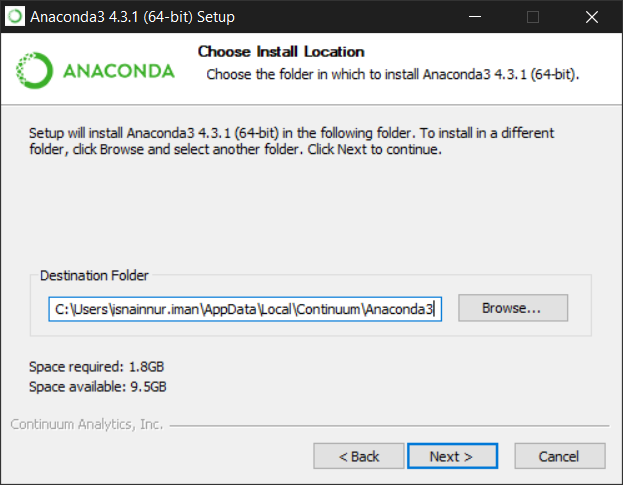
Gambar . Jendela Instalasi Anaconda 64-bit

1. Pada License Agreement klik I Agree. Akan muncul jendela Installation Type. Memilih All Users mewajibkan anda untuk menjalankan Anaconda Navigator sebagai Administrator:



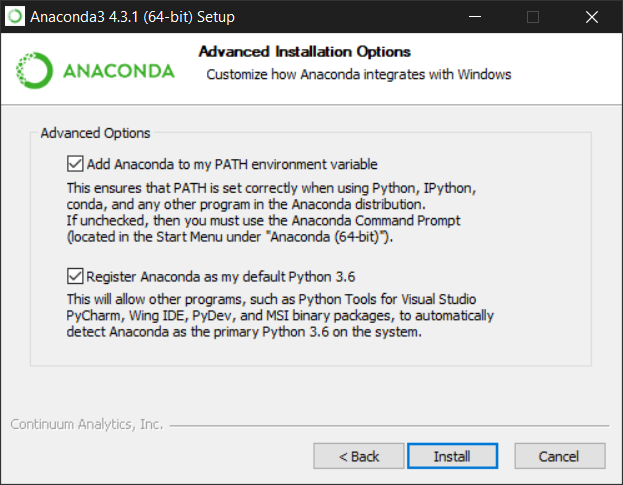
Gambar . Jendela Installation Type

1. Klik Next, lalu pilih direktori yang anda inginkan untuk instalasi:



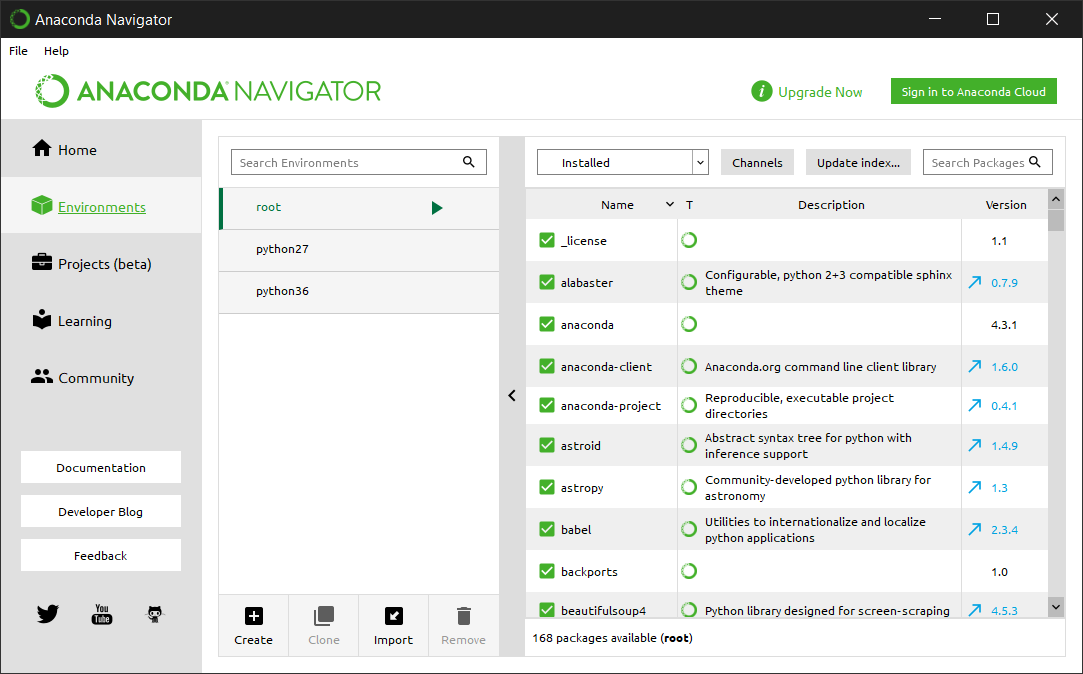
Gambar . Jendela Install Location

1. Centang pada kedua Check Box, lalu klik Install:



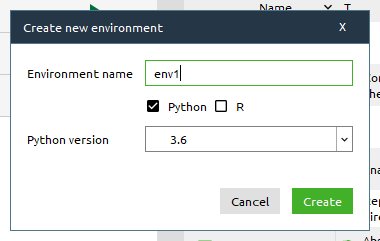
Gambar . Jendela Installation Options

1. Setelah instalasi selesai, buka Anaconda Navigator -> Environments. Jangan lupa untuk Run as Administrator jika anda sebelumnya instalasi dilakukan untuk All User.



Gambar . Jendela Environment Anaconda Navigator

1. Klik Create pada Jendela Environment, akan muncul jendela Create New Environment:



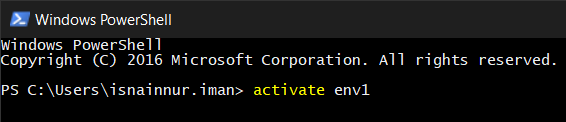
Gambar . Jendela Create New Environment

Pada bagian environment name, masukkan nama yang diinginkan. Centang pada Python, lalu pilih versi 3.6. Klik Create.

1. Setelah Environment dibuat. Kita harus mengaktifkan environment yang dibuat tadi, jalankan command berikut pada Command Prompt atau PowerShell:

activate [Environment Name]

Contohnya seperti berikut:

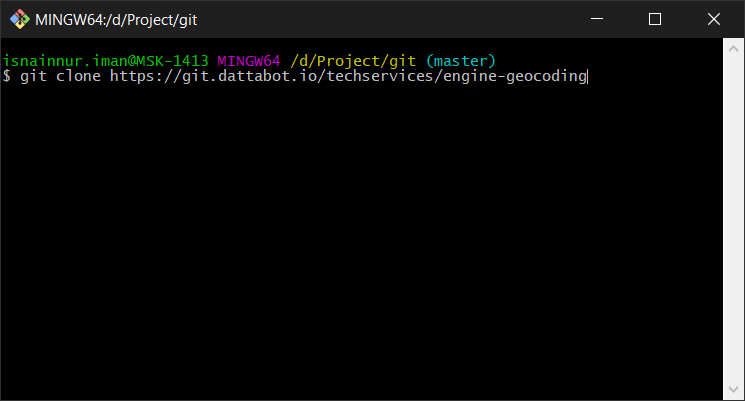


Gambar . Mengaktifkan Environment Anaconda

## Cloning Project dari Gitlab

Untuk Cloning Project dapat dilakukan dengan menjalankan command berikut pada jendela git bash:

**git** clone **[**url**]**



Gambar . Cloning Project menggunakan git Bash

Jika terkendala masalah akses atau error, alternative lainnya adalah dengan mengunduh Engine Geocode secara langsung di halaman git Engine Geocode: <https://git.dattabot.io/techservices/engine-geocoding>.

## Python Library

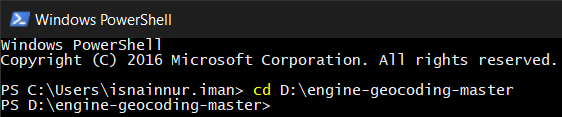
Sebelum melakukan instalasi Library Python, lakukan instalasi pip terlebih dahulu. Pip berfungsi untuk mempermudah instalasi library-library untuk Python. Instalasi pip dapat dilakukan dengan menjalankan command berikut pada Command Prompt / PowerShell:

easy\_install pip

Untuk instalasi library, terlebih dahulu masuk ke direktori utama dari Engine Geocode dengan menjalankan command berikut:

**cd** [Drive:\[directory]]

Direktori pada command diatas merupakan direktori Engine Geocode yang berisi file main.py. Contoh command yang dijalankan adalah sebagai berikut:



Gambar 9. Cara Pindah Direktori menggunakan Command Prompt/PowerShell

Jalankan command berikut untuk Instalasi library menggunakan pip secara langsung dari requirements.txt:

install python3 library (pip install -r requirements.txt)

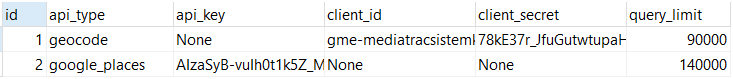
# 3. Menjalankan Engine

## 3.1. Mempersiapkan Input Data

Sebelum menjalankan Engine Geocode ada beberapa hal yang perlu dipersiapkan terlebih dahulu:

1. Tabel mt\_api\_list

Tabel ini berisi informasi detail untuk mengakses API Google Maps Geocoding dan Google Places. Detail yang ditambahkan pada tabel tersebut dapat dilihat pada gambar di bawah ini:



Gambar 10. Tabel List API

* Kolom api\_type

Berisikan jenis api yang ingin ditambahkan:

* + - geocode – Untuk Google Maps Geocoding API
    - google\_places – Untuk Google Places API
* Kolom api\_key, client\_id, client\_secret
  + - Jika crecedentials dari API berupa API Key maka isi kolom api\_key dengan API Key yang dimiliki, dan dua kolom lain dengan ‘None’
      * Jika berupa Client ID dan Client Secret merupakan crecedential dari API anda, maka api\_key diisi dengan ‘None’
* Kolom query\_limit

Isi dengan limit query yang diinginkan. Limit disini hanya menghitung jumlah query pada satu database saja. Jika API digunakan pada database yang berbeda gunakan fungsi limit API di API Manager.

1. Tabel mt\_batas\_zona\_tambahan

Ada 3 tabel batas negara yang saat kini tersedia, dibedakan atas jarak dari garis pangkal:

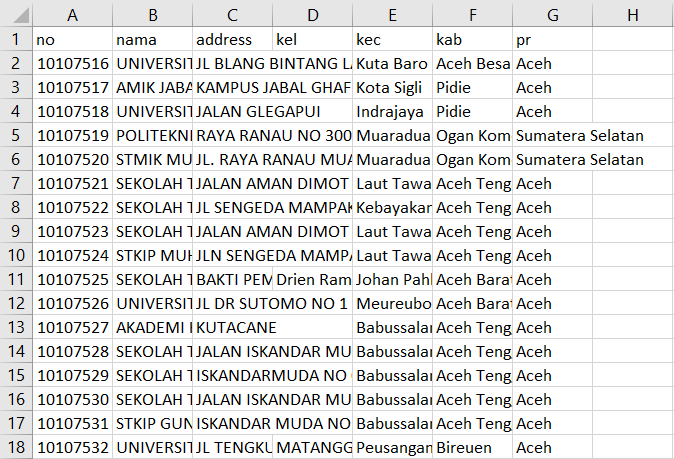
* + - * mt\_batas\_laut\_territorial = 12 mil dari garis pangkal
      * mt\_batas\_laut\_tambahan = 24 mil dari garis pangkal
      * mt\_batal laut\_territorial = 200 mil dari garis pangkal

Tabel batas digunakan untuk menentukan lokasi titik POI apakah masih berada di wilayah Indonesia atau tidak. Proses ini dilakukan apabila titik tidak bertampalan dengan polygon master shapefile. Jika masih berada di Wilayah Indonesia, maka Engine Geocode akan mencari wilayah administrasi terdekat dari titik yang tidak bertampalan tadi.

Batas Zona Tambahan digunakan karena jaraknya tidak terlalu dekat dengan garis pangkal, sehingga cukup banyak titik yang masih dapat ditemukan. Tidak terlalu jauh – jaraknya cukup masuk akal jika dibandingkan Zona Ekonomi Ekslusif untuk menentukan wilayah administrasi terdekat.

1. Tabel input csv

Tabel ini berisi informasi mengenai data yang akan di proses menggunakan Engine Geocoding.



Gambar 11. Format Tabel Input CSV

Berikut kolom-kolom yang harus terdapat pada file csv:

* no – berisi informasi id (unique identifier), dapat berupa angka, huruf, maupun kombinasi keduanya
* nama – berisi informasi mengenai nama tempat
* address – berisi informasi alamat tempat
* kel – berisi informasi nama kelurahan
* kec – berisi informasi nama kecamatan
* kab – berisi informasi nama kabupaten
* pr – berisi informasi nama propinsi

1. Tabel shp\_admin\_dattabot

Tabel ini berisi informasi polygon wilayah administrasi. Tabel ini digunakan pada proses Intersects untuk mendapatkan informasi wilayah administrasi dari koordinat hasil geocode pada Engine Geocode

1. Tabel Singkatan

Tabel singkatan terdiri atas 3 tabel:

* Tabel mt\_singkatan – berisi informasi singkatan jalan, nama tempat, dan singkatan lain yang bersifat umum.
* Tabel mt\_singkatanpoi – berisi informasi singkatan yang berkaitan dengan nama POI
* Tabel mt\_singkatanadm – berisi informasi singkatan untuk nama wilayah administrasi

1. Tabel mt\_tb\_kodepos

Tabel ini berisi informasi mengenai data kodepos untuk masing-masing wilayah administrasi.

1. Tabel mt\_tb\_sinonim\_vi, mt\_tb\_sinonim\_su, mt\_tb\_sinonim\_di, mt\_tb\_sinonim\_pr

* mt\_tb\_sinonim\_vi – berisi singkatan dan nama lain kelurahan
* mt\_tb\_sinonim\_su – berisi singkatan dan nama lain kecamatan
* mt\_tb\_sinonim\_di – berisi singkatan dan nama lain kabupaten
* mt\_tb\_sinonim\_pr – berisi singkatan dan nama lain propinsi

## 3.2. Menjalankan Engine via Command Line

Untuk menjalankan Engine Geocode, masuk terlebih dahulu ke direktori utama Engine Geocode di Command Prompt / Powershel. Kemudian jalankan command dengan format berikut:

python main**.**py **-**i "[Drive:\[directory]]" **-**o "[Output Table]"

Direktori pada command diatas merupakan direktori folder tempat menaruh file csv. Sedangkan Output Table merupakan nama tabel keluaran yang diinginkan di database. Contoh command yang dijalankan adalah sebagai berikut:



Gambar 12. Menjalankan Engine Geocode

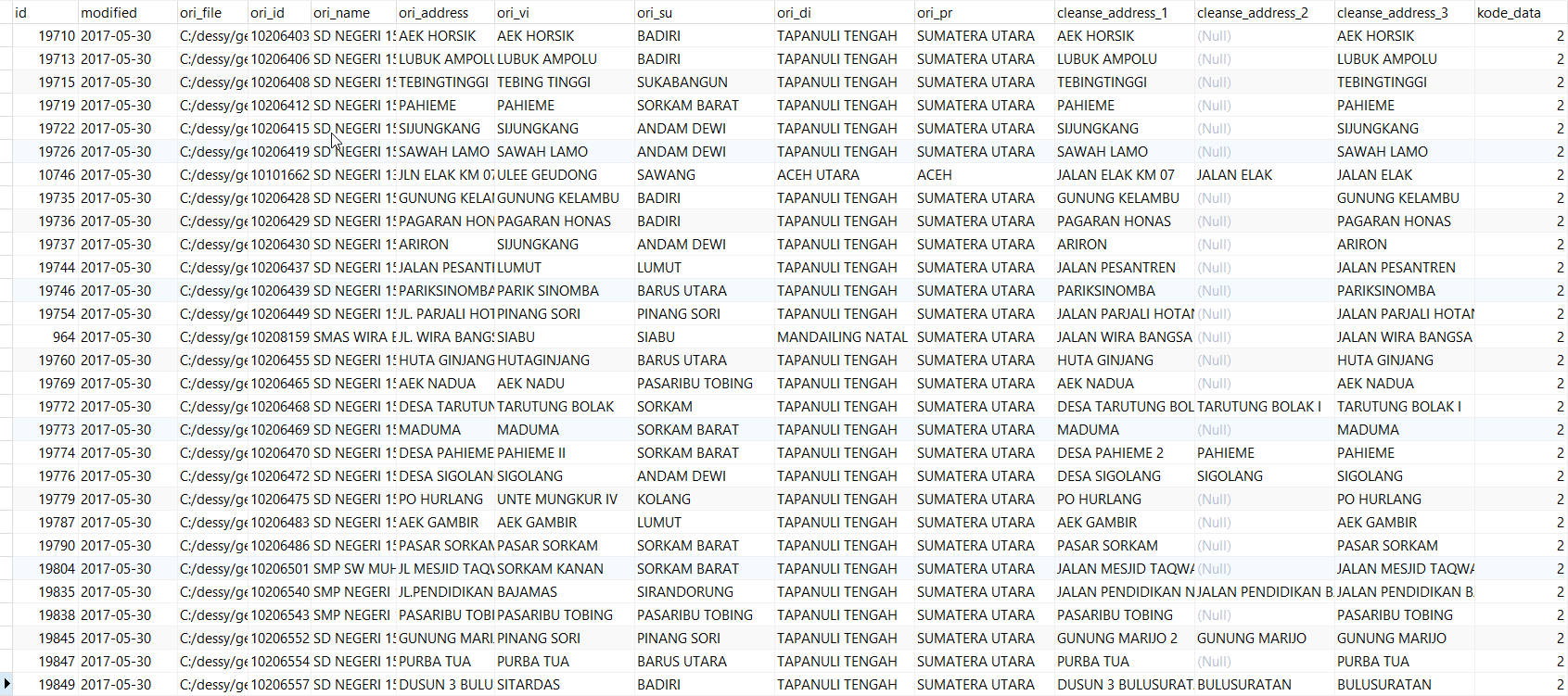
## 3.3. Memahami Output dari EngineGeocoding

1. **Tabel Output Geocode**

Engine Geocode akan menghasilkan sebuah tabel dengan nama sesuai nama yang telah ditetapkan saat menjalankan command. Berikut ini merupakan penjelasan dari setiap nama kolom pada Output Table:

|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| id | Unique identifier yang di generate saat input csv ke database |
| modified | Tanggal saat data file csv di import ke database. |
| ori\_file | Direktori lokasi file csv |
| ori\_id | Unique Identifier pada data input file csv |
| ori\_name | Nama Tempat pada data input file csv |
| ori\_address | Lokasi Tempat pada data input file csv |
| ori\_vi | Nama Kelurahan pada data input file csv |
| ori\_su | Nama Kecamatan pada data input file csv |
| ori\_di | Nama Kabupaten/Kota pada data input file csv |
| ori\_pr | Nama Propinsi pada data input file csv |
| cleanse\_address\_1 | ori\_address yang telah di cleansing dengan pattern 1 |
| cleanse\_address\_2 | ori\_address yang telah di cleansing dengan pattern 2 |
| cleanse\_address\_3 | ori\_address yang telah di cleansing dengan pattern 3 |
| kode\_data | Kode data untuk metode geocoding yang digunakan |
| flag\_geocode | Status Data:  Q: Data menunggu antrian untuk diproses P: Data sedang diproses D: Data selsesai diproses dan lokasi data ditemukan E: Data selsesai diproses dan lokasi data tidak ditemukan |
| google\_address | Nama lokasi hasil geocoding |
| place\_name | Nama tempat hasil geocoding |
| google\_vi | Nama Kelurahan hasil geocoding |
| google\_su | Nama Kecamatan hasil geocoding |
| google\_di | Nama Kota/Kabupaten hasil geocoding |
| google\_pr | Nama Provinsi hasil geocoding |
| adm\_vi | Nama Kelurahan hasil proses Intersect koordinat hasil geocoding dan polygon wilayah administrasi Indonesia. |
| adm\_su adm\_vi | Nama Kecamatan hasil proses Intersect koordinat hasil geocoding dan polygon wilayah administrasi Indonesia. |
| adm\_di | Nama Kabupaten/Kota hasil proses Intersect koordinat hasil geocoding dan polygon wilayah administrasi Indonesia. |
| adm\_pr | Nama Provinsi hasil proses Intersect koordinat hasil geocoding dan polygon wilayah administrasi Indonesia. |
| adm\_vi\_id | ID Kelurahan hasil proses Intersect koordinat hasil geocoding dan polygon wilayah administrasi Indonesia. |
| adm\_su\_id | ID Kecamatan hasil proses Intersect koordinat hasil geocoding dan polygon wilayah administrasi Indonesia. |
| adm\_di\_id | ID Kabupaten hasil proses Intersect koordinat hasil geocoding dan polygon wilayah administrasi Indonesia. |
| adm\_pr\_id | ID Kabupaten/Propinsi hasil proses Intersect koordinat hasil geocoding dan polygon wilayah administrasi Indonesia. |
| match\_pr | Status matching administrasi Provinsi |
| match\_di | Status matching administrasi Kabupaten/Kota |
| match\_su | Status matching administrasi Kecamatan |
| match\_vi | Status matching administrasi Kelurahan |
| match\_jln | Status Matching nama jalan hasil geocode dan nama jalan awal. |
| Keterangan | Metode geocode yang digunakan dalam proses geocoding jika data ditemukan. Jika geocode tidak ditemukan, maka akan muncul keterangan ‘ERROR’. |
| Summary | Kualitas data hasil geocoding, terdiri atas High, Medium dan Low. |
| Summary\_detail | Parameter untuk menentukan kualitas data untuk kolom summary. |
| Parsing | Kolom untuk proses matching data. |

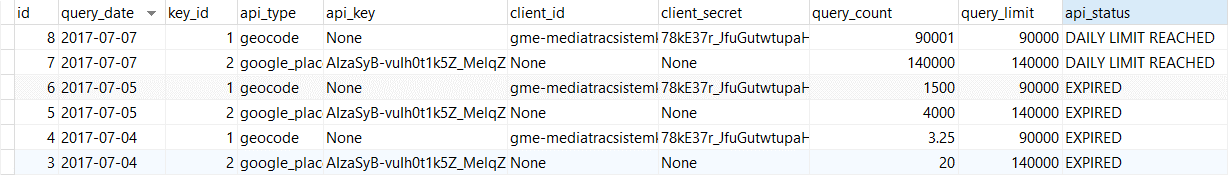
Contoh tampilan Output Table:



Gambar 13. Tabel Output Geocoding

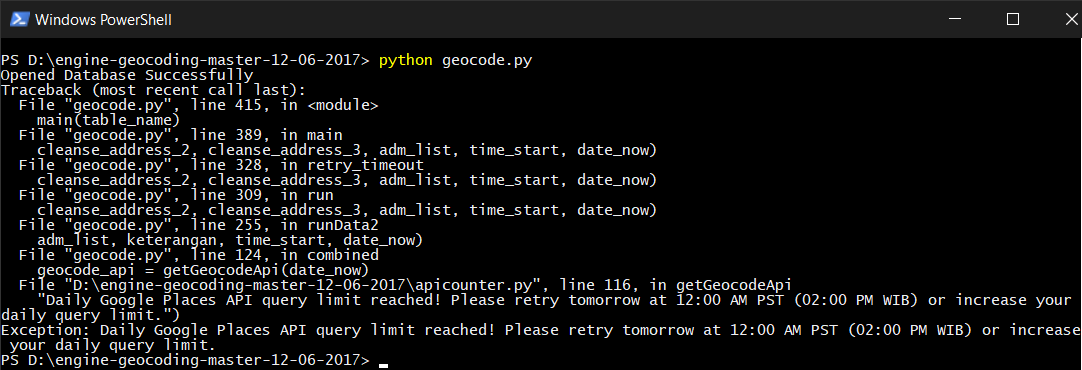
1. **Tabel API Query Count**

Berikut ini merupakan tampilan tabel mt\_api\_query\_count:



Gambar 14. Tabel Counter API

Tabel mt\_api\_query\_count berisi informasi mengenai jumlah query yang telah digunakan dalam 1 hari pada database yang digunakan untuk menjalankan Engine Geocode. Apabila kolom query\_count nilainya lebih besar atau sama dengan query\_limit kolom api\_status akan menghasilkan pesan ‘DAILY QUERY LIMIT REACHED’. Proses pada Engine Geocode akan berhenti, dan pada jendela command prompt akan menampilkan pesan berikut:



Gambar . Error Jika Query Limit Terlewati

Anda dapat mencoba untuk menjalankan Engine Geocode pada 12:00 AM [Waktu Pasifik](https://id.wikipedia.org/wiki/Waktu_Standar_Pasifik) esok harinya. Atau dapat menambah jumlah query\_limit pada tabel tersebut. Namun, yang perlu diperhatikan adalah jumlah query\_count pada tabel ini merupakan hasil perhitungan setiap proses pada Engine Geocode. Untuk perhitungan yang lebih akurat dapat dilihat pada Console API Manager Google.

1. **Tabel Hasil Process Matching**

# 4. Engine Architecture

## 4.1. Logic Flow

## 4.2. Struktur Code