

# Modul : Pengantar Python dengan Google Colab



# Daftar Isi

---

1. Latar Belakang.....	1
2. Menggunakan Google Colab .....	3
3. Menjalankan Kode Python di Google Colab .....	6
4. Menggunakan GPU di Google Colab .....	8
5. Instalasi dan Import Library di Google Colab.....	9
6. Upload File dan Akses Google Drive .....	10
7. Contoh Pemograman .....	11
Referensi.....	13

# 1. Latar Belakang

---

Python adalah bahasa pemrograman *interpreter* tingkat tinggi, berorientasi objek, dengan memiliki semantik yang dinamis. Bahasa tingkat tinggi yang dibangun dalam struktur data, dikombinasikan dengan penyetikan dinamis dan pengikatan dinamis, membuatnya sangat menarik untuk Pengembangan Aplikasi Cepat, serta untuk digunakan sebagai bahasa *scripting*. Sintaksis Python yang sederhana dan mudah dipelajari menekankan keterbacaan dan karenanya mengurangi biaya pemeliharaan program. Python mendukung modul dan paket, yang mendorong modularitas program dan penggunaan kode kembali. Interpreter Python dan pustaka standar yang luas tersedia dari berbagai sumber dan dapat didistribusikan secara bebas.

Python mendukung pemrograman dengan paradigma multiguna, utamanya; namun tidak dibatasi; pada pemrograman berorientasi objek, pemrograman imperatif, dan pemrograman fungsional. Salah satu fitur yang tersedia pada python adalah sebagai bahasa pemrograman dinamis yang dilengkapi dengan manajemen memori otomatis. Sama halnya dengan bahasa pemrograman dinamis lainnya, python umumnya digunakan sebagai bahasa *scripting* namun penggunaan bahasa ini lebih luas mencakup konteks pemanfaatan. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi. Kode python dapat dijalankan di berbagai platform sistem operasi, antara lain:

1. Linux/Unix
2. Windows
3. Mac OS X
4. Java Virtual Machine
5. OS/2
6. Amiga
7. dan sebagainya.

Beberapa fitur yang dimiliki Python adalah:

1. memiliki library atau pusaka yang sangat lengkap yang siap dipakai untuk berbagai memiliki tata bahasa yang jernih dan mudah dipelajari.
2. Tata Bahasa pemrogramannya mudah dipahami
3. memiliki aturan layout kode sumber yang memudahkan pengecekan, pembacaan kembali dan penulisan ulang kode sumber.

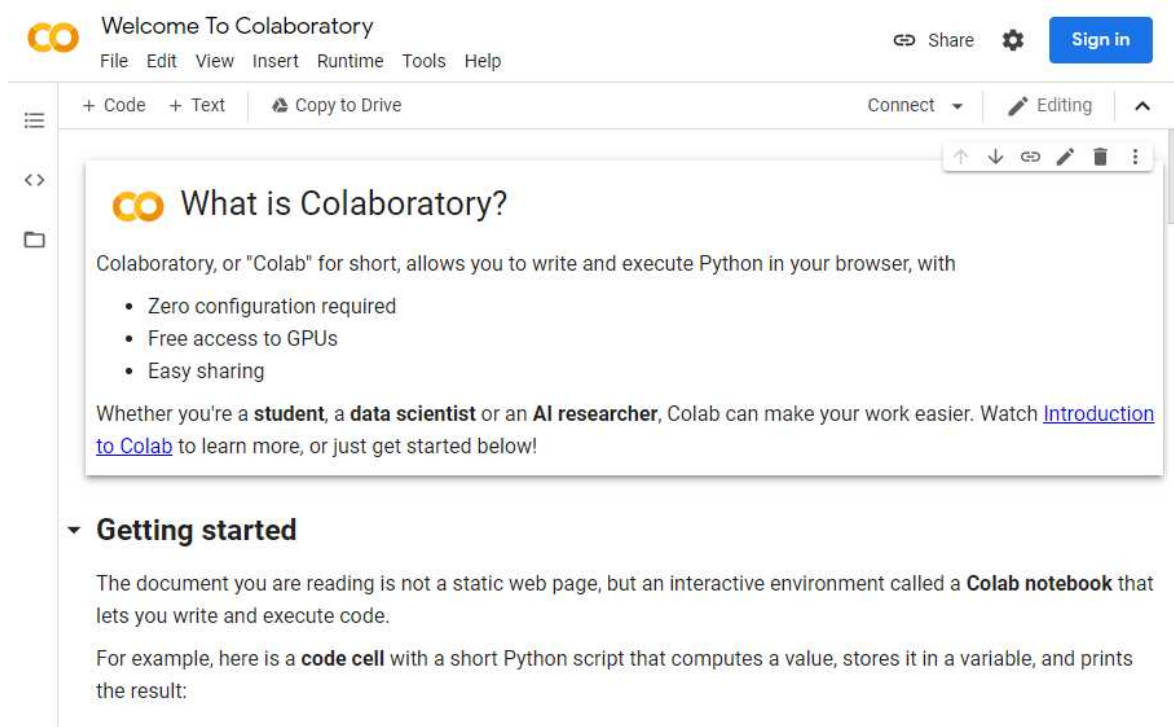
4. berorientasi objek.
5. memiliki sistem pengelolaan memori otomatis (garbage collection, seperti java)
6. modular, mudah dikembangkan dengan menciptakan modul-modul baru; modul-modul tersebut dapat dibangun dengan bahasa Python maupun C/C++.
7. Garbage collection otomatis
8. Fasilitas pengaturan penggunaan ingatan komputer sehingga para pemrogram tidak perlu melakukan pengaturan ingatan komputer secara langsung.
9. memiliki banyak fasilitas pendukung sehingga mudah dalam pengoperasiannya.

Python dapat digunakan dengan berbagai macam antarmuka aplikasi seperti Anaconda, IntelliJ IDEA, Visual Studio Code, Sublime Text, Spyder, Google Colab, dan sebagainya. Google Colab (Google Collaboratory) merupakan tools yang berbasis cloud dan free untuk tujuan penelitian. Google colab dibuat dengan environment jupyter notebook dan mendukung hampir semua library yang dibutuhkan dalam berbagai lingkungan pengembangan, seperti Machine Learning, Artificial Intelligence (AI), Kriptografi, pengolahan data, dan sebagainya. Berikut adalah beberapa kelebihan dalam menggunakan google colab, antara lain:

1. Penggunaan google colab ditujukan bagi para peneliti yang sedang mengembangkan penelitian dan membutuhkan spesifikasi komputer yang tinggi. Hanya perlu diingat bahwa google colab membutuhkan koneksi internet.
2. Google memberikan akses cloud komputer dengan spesifikasi:
  - a. Intel(R) Xeon(R) CPU @ 2.30GHz
  - b. 12,6 GB VRAM
  - c. Tesla P100-PCIE-16GB, Cuda Cores: 2496
  - d. 33 GB
3. Dalam menggunakan google colab kita tidak memerlukan konfigurasi apapun, namun dapat menginstall Pustaka di Google Colab.
4. Dapat diintegrasikan dengan Google Drive dan berbagi dengan pengguna lainnya.
5. Dapat digunakan dimana saja karena bersifat cloud

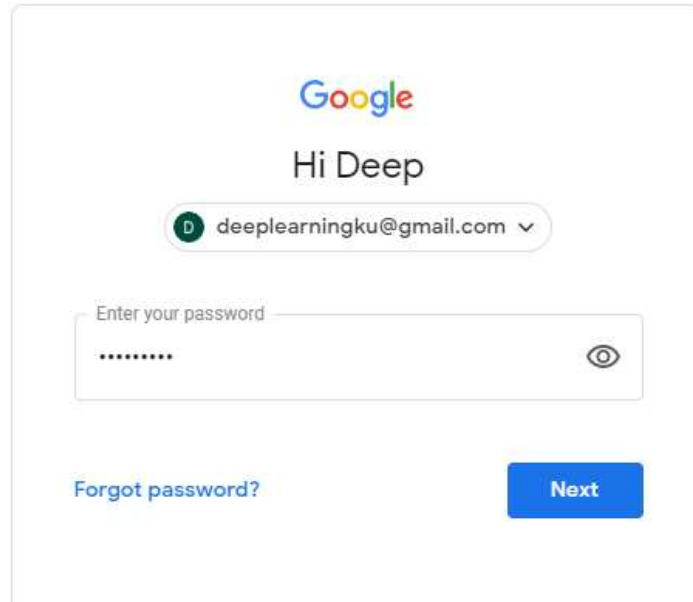
## 2. Menggunakan Google Colab

1. Akses Halaman Google Colab melalui laman ini : <https://colab.research.google.com/> , maka akan muncul halaman pembuka Google Colab



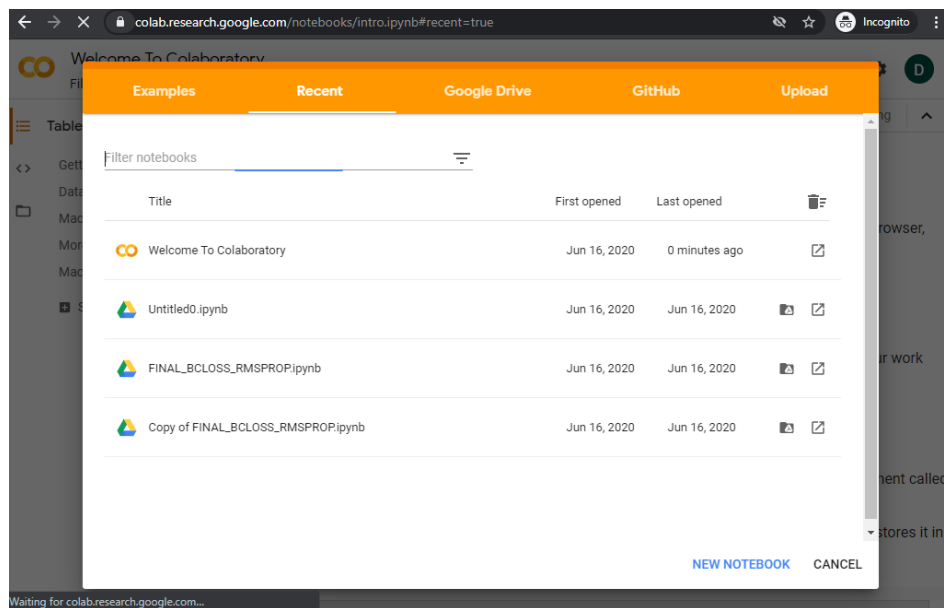
Gambar 1 Tampilan Awal Google Colab

2. Untuk menggunakan layanan Google Colab, kita diharuskan memiliki akun Google. Jika belum memiliki akun, silahkan daftar melalui link ini <https://accounts.google.com/>
3. Jika sudah memiliki akun Google, silahkan Sign In menggunakan akun Google anda. Berikut contoh Sign In ke Google Colab



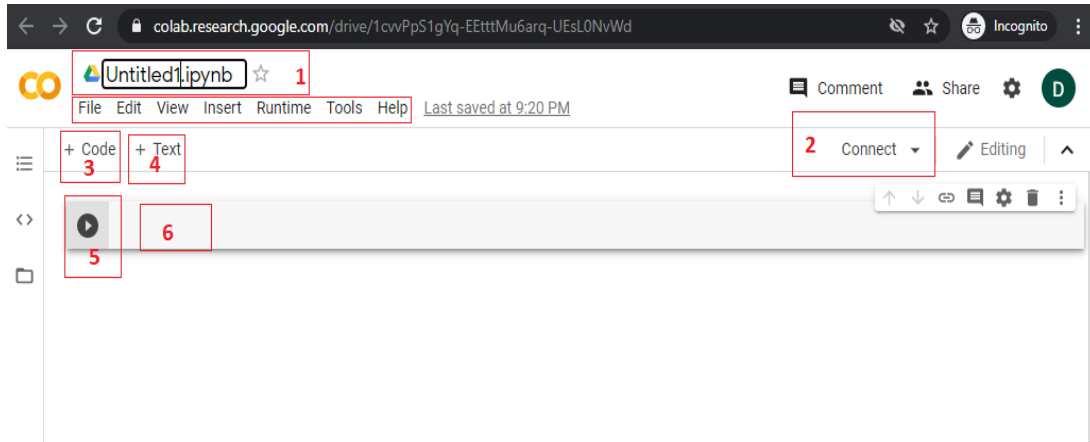
Gambar 2 Halaman Sign In Google

- Setelah login, maka akan muncul sebuah jendela yang memberitahukan apakah kita mau membuat File Notebook yang baru atau menggunakan File yang sudah ada sebelumnya



Gambar 3 Jendela Awal Setelah Login Google Colab

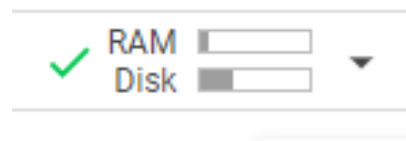
5. Klik NEW NOTEBOOK untuk membuat sebuah file notebook baru untuk menjalankan kode python di Google Colab, maka akan dibuat sebuah file Notebook yang baru seperti gambar dibawah ini.



Gambar 4 File Notebook pada Google Colab


Keterangan Gambar 4:

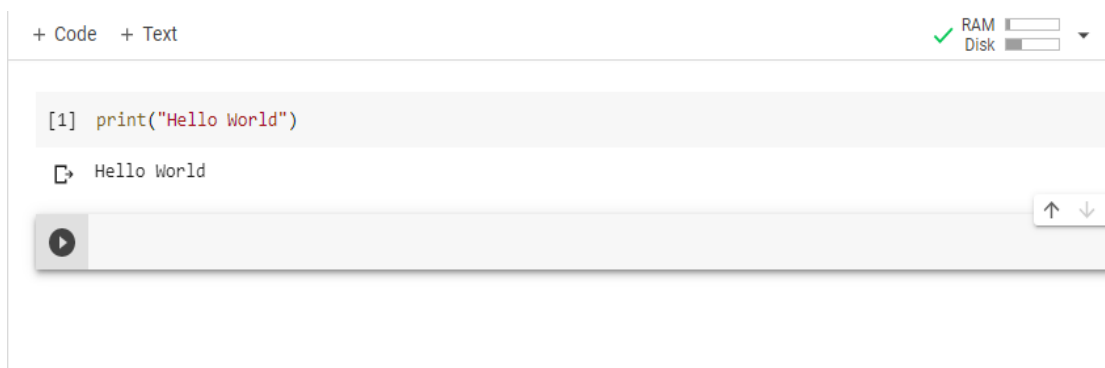
1. Nama File
  2. Untuk mengaktifkan (menghubungkan) Notebook kita di Google Colab
  3. Menambah baris kode Notebook
  4. Menambahkan keterangan Teks
  5. Eksekusi kode python yang dituliskan kode baris
  6. Cell Tempat baris kode dituliskan
6. Untuk mengaktifkan File Notebook yang telah dibuat, klik Connect (Seperti yang ditunjukkan pada Gambar 4, No 2)
  7. Setelah Connect, maka File Notebook sudah aktif dan bisa menjalankan kode python



Gambar 5 Notebook Sudah Aktif

### 3. Menjalankan Kode Python di Google Colab

1. Ketik baris kode python di Cell kode python, kemudian klik tombol  atau tekan tombol **Shift + Enter** bersamaan.
2. Maka kode baris yang kita ketik, akan dieksekusi, dan memberikan Output dibawah baris kode yang kita buat.




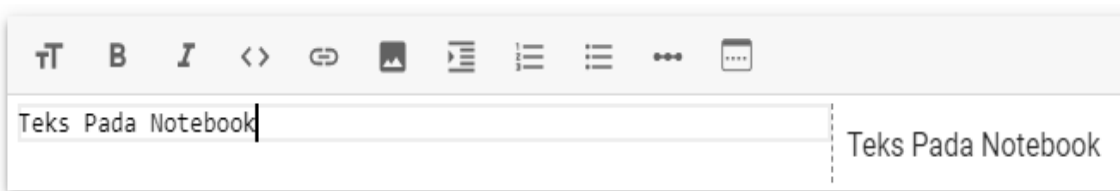
Gambar 6 Hasil Eksekusi Kode Python

3. Pada Cell kode baris tersebut, ada beberapa tools yang terdapat pada Cell, kita dapat memindahkan sebuah cell ke atas atau bawah, membuat *hyperlink*, menghapus cell, maupun memberikan komentar pada cell tersebut.



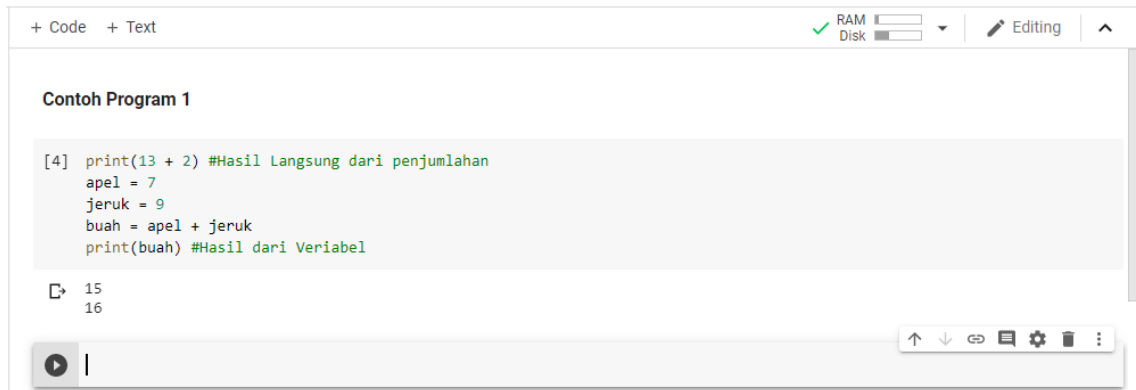
Gambar 7 Pengaturan pada Cell Notebook Google Colab

4. Kita dapat menambahkan teks pada Notebook, dengan meng-klik tombol +Text. Ketikkan teks yang mau dibuat pada cell text editor, kemudian eksekusi dengan tombol  atau tekan tombol **Shift+Enter** bersamaan.



Gambar 8 Text Editor untuk menambahkan text pada Notebook

## 5. Contoh penggunaan Cell +Code dan +Text



The image shows a Jupyter Notebook interface. At the top, there are tabs for '+ Code' and '+ Text', and a status bar with 'RAM', 'Disk', and 'Editing' indicators. The main area is titled 'Contoh Program 1' and contains a code cell. The code cell has a prompt '[4]' and contains the following Python code: `print(13 + 2) #Hasil Langsung dari penjumlahan`, `apel = 7`, `jeruk = 9`, `buah = apel + jeruk`, and `print(buah) #Hasil dari Variabel`. Below the code, the output is displayed: `15` and `16`. At the bottom of the cell, there is a toolbar with icons for up, down, refresh, help, settings, and delete.

```
[4] print(13 + 2) #Hasil Langsung dari penjumlahan
    apel = 7
    jeruk = 9
    buah = apel + jeruk
    print(buah) #Hasil dari Variabel
```

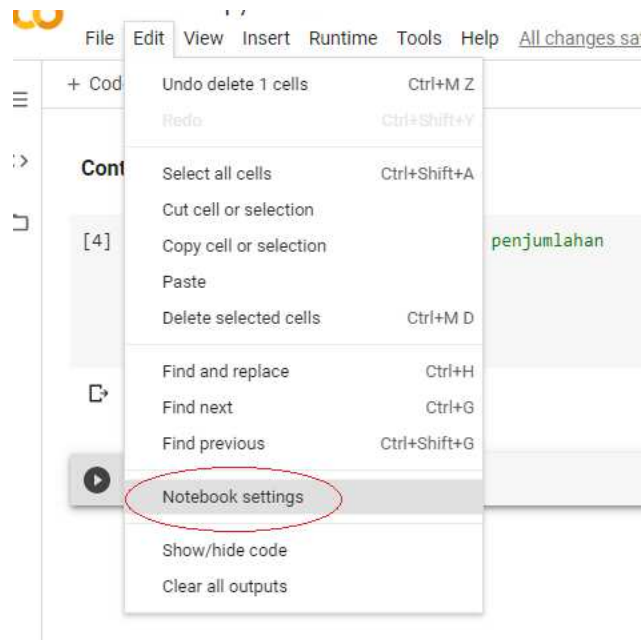
```
15
16
```

Gambar 9 Contoh Kode Python dan Text

## 4. Menggunakan GPU di Google Colab

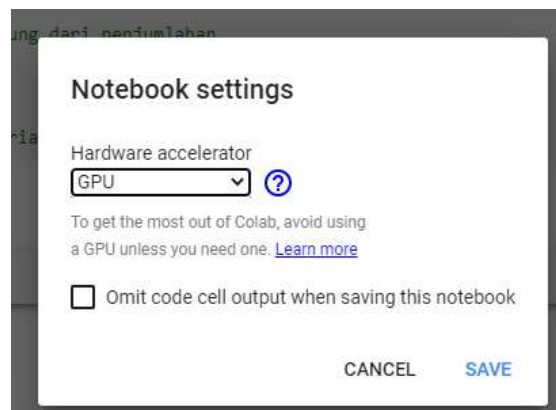
Google Colab menyediakan akses proses GPU secara gratis untuk penggunaannya. Kita dapat melakukan pengaturan pada Notebook kita untuk dapat menggunakan akses GPU yang telah disediakan oleh Google Colab.

1. Klik menu **Edit** → **Notebook settings**



Gambar 10 Setting Notebook

2. kemudian ubah pada bagian **Hardware accelerator** menjadi **GPU** kemudian Klik **Save**, maka kita dapat mengubah runtime versi Python pada notebook sedang aktif.

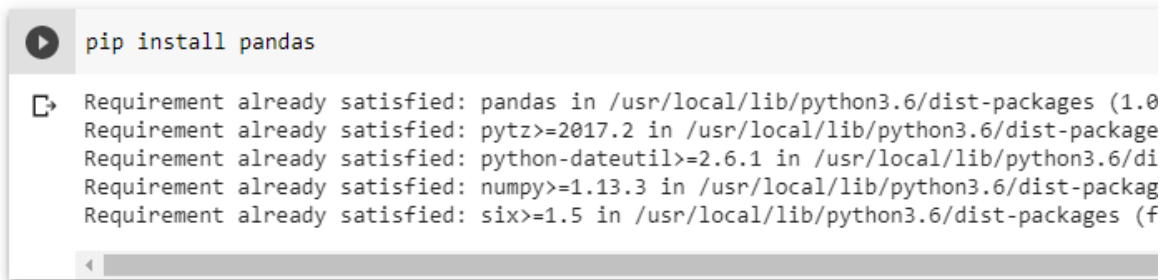


Gambar 11 Setting GPU pada Notebook aktif

## 5. Instalasi dan Import Library di Google Colab

1. Untuk menambahkan library baru pada google colab, gunakan perintah “pip install (nama\_package)”, kemudian eksekusi baris perintah tersebut

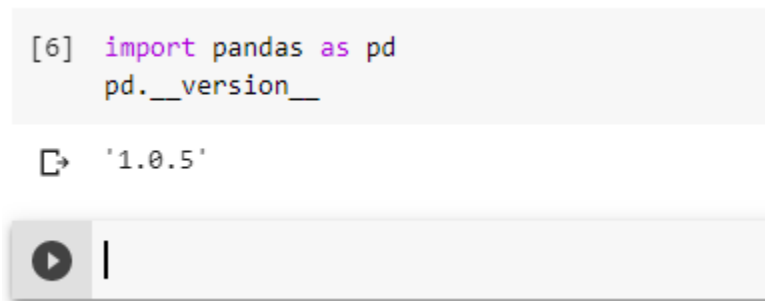
Contoh install library



```
▶ pip install pandas  
↳ Requirement already satisfied: pandas in /usr/local/lib/python3.6/dist-packages (1.0  
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.6/dist-package  
Requirement already satisfied: python-dateutil>=2.6.1 in /usr/local/lib/python3.6/di  
Requirement already satisfied: numpy>=1.13.3 in /usr/local/lib/python3.6/dist-packag  
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.6/dist-packages (f
```

Gambar 12 Contoh install library

2. Untuk import library yang sudah diinstall, gunakan perintah “import (nama\_pustaka)” atau “import (nama\_pustaka) as inisial\_pustaka” jika ingin menggunakan inisial

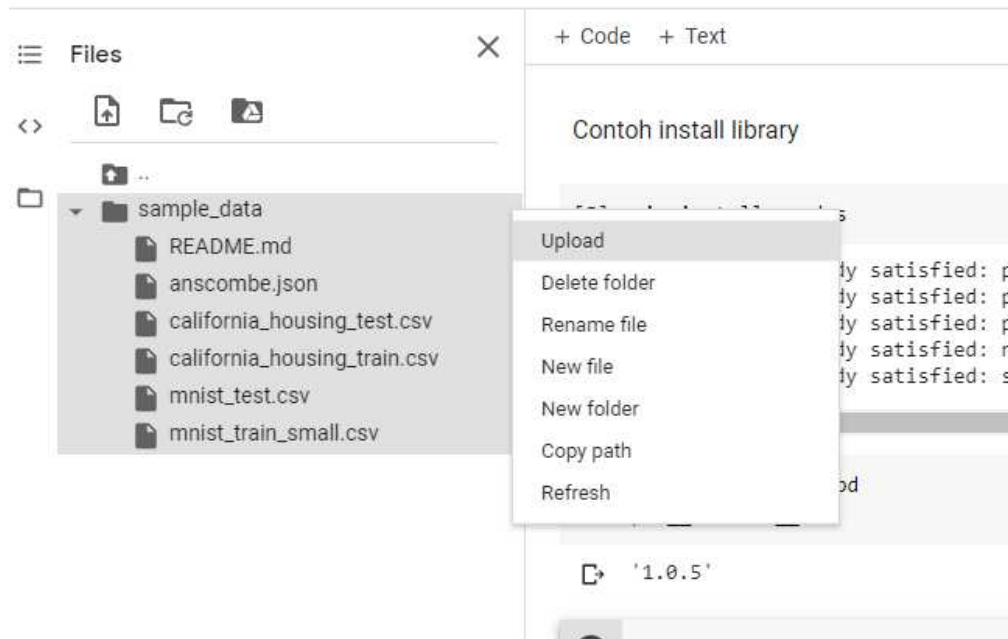


```
[6] import pandas as pd  
    pd.__version__  
↳ '1.0.5'  
▶ |
```


Gambar 13 Import Pustaka/ Package di Notebook Google Colab

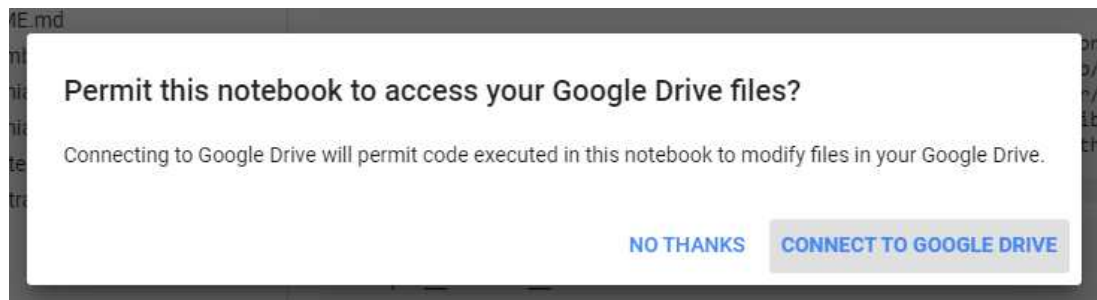
## 6. Upload File dan Akses Google Drive

1. Untuk upload file kedalam google colab, caranya klik pada **icon folder** pada sidebar sebelah kiri
2. Kemudian klik folder "**content/sample\_data**" kemudian pada sample\_data klik kanan → upload. Jika upload sudah selesai klik refresh.



Gambar 14 Upload file pada Google Colab

3. Cara lain untuk membaca file adalah dengan mengintegrasikan Google Drive kita ke dalam google colab. Untuk langkah awal kita perlu upload file kedalam google drive. Kemudian klik tombol **Mount Drive** 
4. Kemudian akan muncul jendela pemberitahuan untuk mengizinkan Google Colab mengakses Google Drive Anda. Kemudian klik **Connect to Google Drive**



Gambar 15 Jendela permintaan akses Google Drive melalui Google Colab

5. Untuk membaca file yang diupload ke Google Drive, bisa kita akses dengan path **"/content/sample\_data/"**

## 7. Contoh Pemograman

---

### 1. Kondisional If

```
[8] x = 5
    if x % 2 == 0: # jika x dibagi dua bersisa nol
        print ('Genap')
    else:
        print ('Ganjil')
```

↳ Ganjil

Gambar 16 Contoh program kondisional If

### 2. Perulangan For

```
[9] print(range(0, 5))

    for x in range(0, 5):
        print(x)

    print('-----')

    for y in range(9, 0, -3):
        print(y)
```

↳ range(0, 5)  
0  
1  
2  
3  
4  
-----  
9  
6  
3

Gambar 17 Contoh program perulangan For

### 3. Pembuatan Fungsi

```
[10] def apakah_genap(angka):
      if angka % 2 == 0:
          print ('benar, genap')
      else:
          print ('salah, ganjil')

      print ('cek angka 7, apakah genap?')
      apakah_genap(7)
      print ('cek angka 22, apakah genap?')
      x = apakah_genap(angka=100)
      print(x)
```

```
↳ cek angka 7, apakah genap?
    salah, ganjil
    cek angka 22, apakah genap?
    benar, genap
    None
```

Gambar 18 Contoh Program Fungsi

### 4. List

```
[11] l = [1, 2, 3]
      print (l[0])
      l = ['ab', 'bc', 'cd', 'ef']
      print (l[1])
      type(l)
      a = ['ab', 'bc', 1, 2, 1.1, [True, False]]
      print(type(a[4]))
      print (a[5])
      print (a[5][0])
      a.append('xy')
      print(a)
      print(a[6])
      print(a[-1])
```

```
↳ 1
   bc
   <class 'float'>
   [True, False]
   True
   ['ab', 'bc', 1, 2, 1.1, [True, False], 'xy']
   xy
   xy
```

Gambar 19 Contoh List

## Referensi

---

- [1] Colaboratory Frequently Asked Questions. URL: <https://research.google.com/colaboratory/faq.html>
- [2] What is Python? Executive Summary. URL: <https://www.python.org/doc/essays/blurb/>
- [3] Downey, Allen. 2015. Think Python: How to Think Like a Computer Scientist 2nd Edition, Version 2.2.23. Green Tea Press:Needham, Massachusetts
- [4] Pemrograman Python untuk Pemula. URL: [https://colab.research.google.com/drive/1nw0K0noFqtwL\\_S3CPKYjL7KpwQKSgCeF#scrollTo=vIFON3I1qKnf](https://colab.research.google.com/drive/1nw0K0noFqtwL_S3CPKYjL7KpwQKSgCeF#scrollTo=vIFON3I1qKnf)