

Modul Praktikum PBO



DISUSUN OLEH :

1. FALASHIFFA GADING
2. JOANA DEIKA PRATAMA

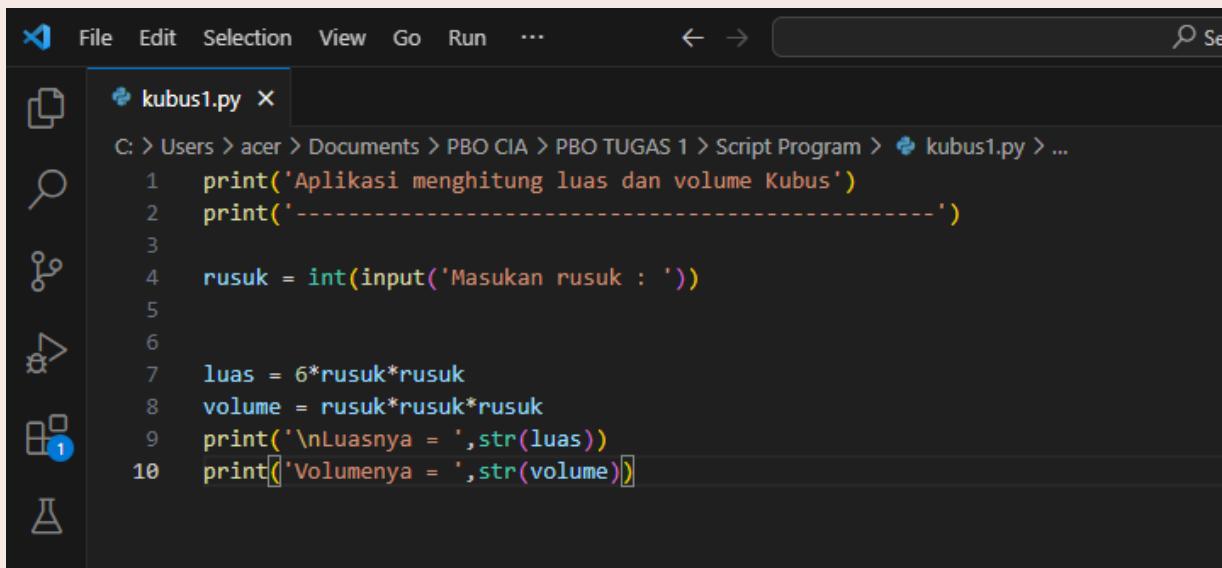
FAKULTAS TEKNIK
PRODI TEKNIK INFORMATIKA KELAS TI22B

UNIVERSITAS MUHAMMADIYAH CIREBON
2022/2023

Sebelum memulai module tiap pertemuan,kami disini akan membahas apa itu type data dan macam-macam type data

- Type data ialah cara untuk memudahkan program dalam mendeteksi berbagai jenis informasi dan memastikan sistem operasional dapat berjalan dan meminimalkan kesalahan (error)
- Macam-macam type data:
 - a.Character (Char): Huruf, angka, atau simbol
 - b.Floating Point: Bilangan desimal
 - c.Integer: Bilangan bulat
 - d.Boolean (bool): Data yang memiliki nilai true atau false
 - e.String: Serangkaian karakter
 - f.Array: Serangkaian elemen yang sama

Pertemuan 1 (Membuat program perhitungan luas dan volume Bangun Ruang)



```
File Edit Selection View Go Run ... ← → ⌂ Search
kubus1.py ×
C: > Users > acer > Documents > PBO CIA > PBO TUGAS 1 > Script Program > kubus1.py > ...
1 print('Aplikasi menghitung luas dan volume Kubus')
2 print('-----')
3
4 rusuk = int(input('Masukan rusuk : '))
5
6
7 luas = 6*rusuk*rusuk
8 volume = rusuk*rusuk*rusuk
9 print('\nLuasnya = ', str(luas))
10 print('Volumenya = ', str(volume))
```

Diatas ini ada script menghitung luas dan volume dalam bangun ruang kubus dengan menggunakan bahasa pemrograman Python.

- Perintah “Print” gunanya adalah untuk menampilkan output dalam python.
- Dan dalam rumus diatas didalamnya ada type data int dengan artian type data dalam bentuk bilangan bulat.
- Dan ada karakter “\n” itu gunanya untuk membuat baris baru saat digunakan.
- Ada pula type data “str” itu gunanya ialah digunakan untuk mengkonversi segala jenis data ke bentuk string.

Pertemuan 2 (Mengerjakan Soal Yang Bermateri Tentang IF,Loop & Array di Kertas Polio)

Tugas 1 PBO - Baca-saja

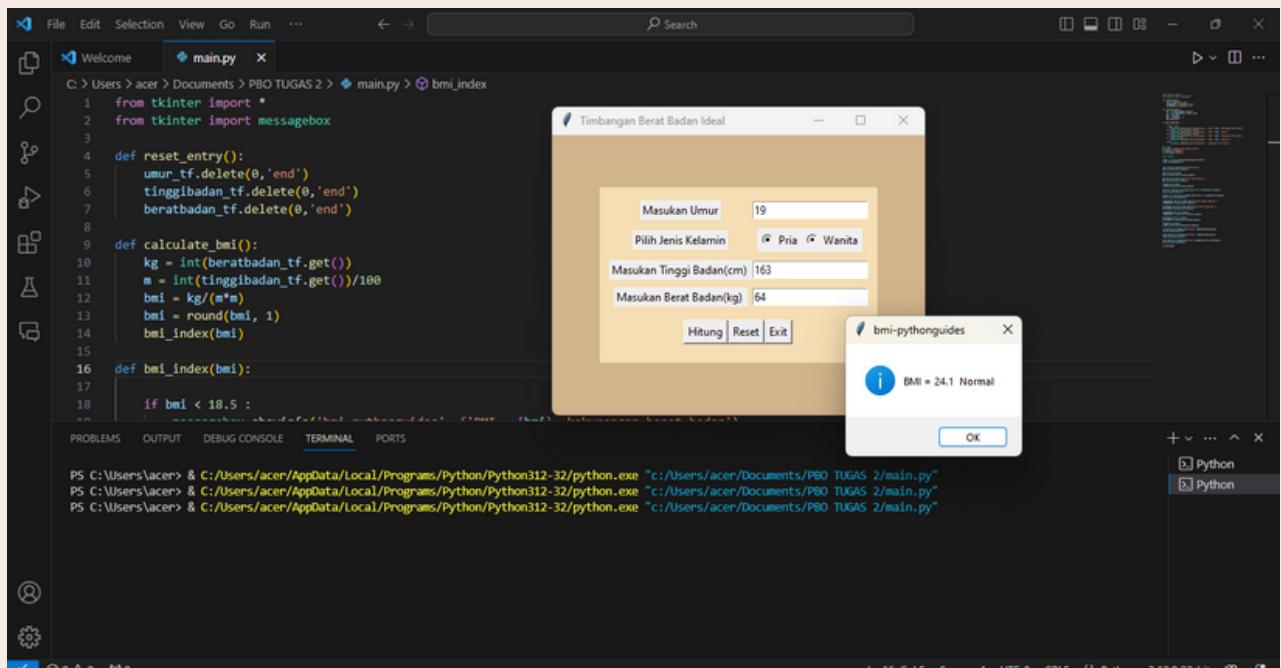
Jika beberapa teks terlihat kacau, Anda da... ▾

1. Jelaskan apa itu pemrograman python
2. Jelaskan sejarah singkat python
3. Bagaimana proses instalasi python
4. Bagaimana cara memastikan keberhasilan instalasi python
5. Jelaskan Tipe Data yang ada di Python
6. Jelaskan tentang variabel di python dan berikan contohnya
7. Jelaskan Operator yang ada di python dan berikan contohnya
8. Jelaskan tentang IF (kondisi) dalam pemrograman python serta berikan contohnya
9. Jelaskan tentang Pengulangan (Loop) serta berikan contohnya
10. Jelaskan tentang List, Tuple, Set, dan Dictionary serta berikan contohnya

Jawaban ditulis tangan dengan jelas (dapat dibaca orang lain) di atas kertas folio bergaris, dikumpulkan paling lambat hari sabtu tanggal 28 oktober 2023 jam 12.00 siang.

**Saya sudah mengerjakan
di kertas folio,
dan sudah dikumpulkan**

Pertemuan 3 (Membuat aplikasi Kalkulator BMI berbasis windows)

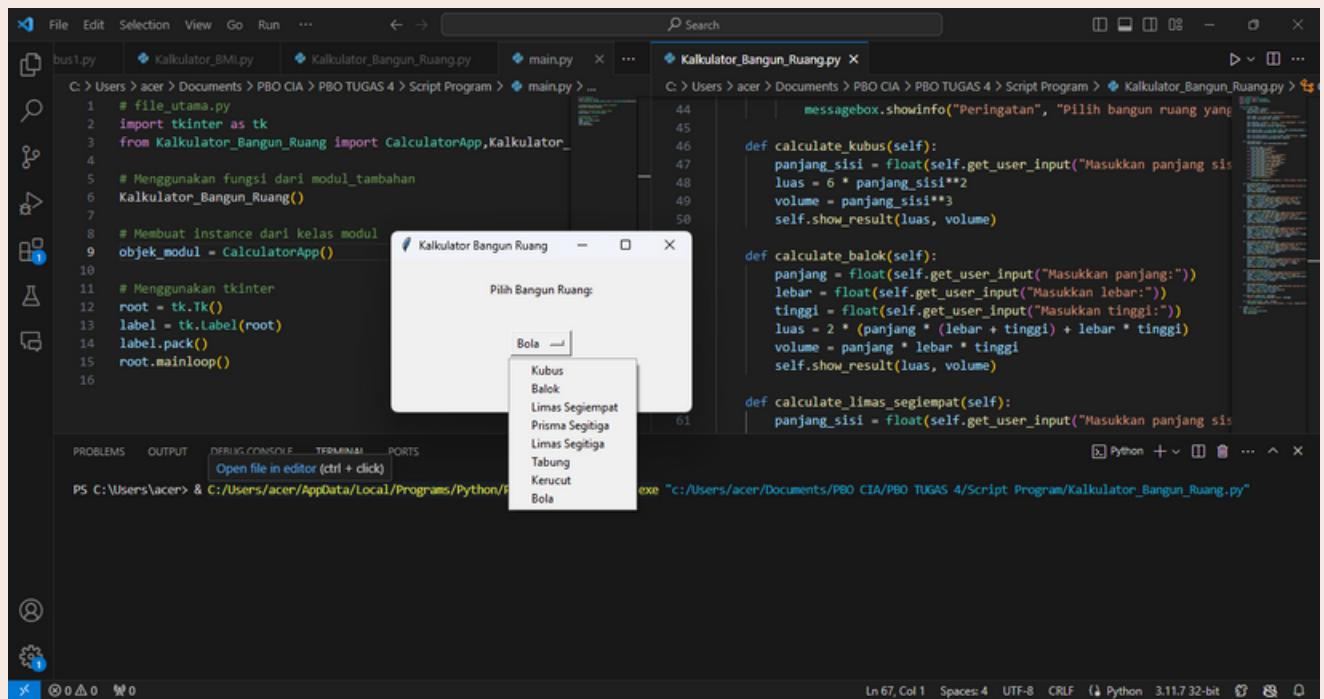


Diatas ini ada script membuat aplikasi Kalkulator BMI berbasis windows dengan menggunakan bahasa pemrograman Python.

Script diatas menggunakan Function pada program kalkulator BMI

Dalam contoh di atas, fungsi `hitung_bmi` menerima dua parameter, yaitu `berat` (dalam kg) dan `tinggi` (dalam cm). Pertama, tinggi dikonversi ke meter. Selanjutnya, BMI dihitung menggunakan rumus `berat/(tinggi**2)`. Hasil BMI kemudian dikembalikan. Fungsi ini dapat digunakan untuk menghitung BMI seseorang berdasarkan berat dan tinggi badan. Selain itu, program juga dapat menggunakan struktur kondisional if-else untuk menentukan kategori BMI berdasarkan hasil perhitungan.

Pertemuan 4 (Membuat aplikasi Bangun Ruang menghitung luas & volume Berbasis Windows)



Script diatas menggunakan Module pada program aplikasi perhitungan luas & volume bangun ruang

Module dalam bahasa pemrograman Python adalah sebuah file yang berisi kode program Python yang dapat digunakan kembali dalam program lain. Modul dapat berisi fungsi, kelas, variabel, dan kode Python lainnya. Dengan menggunakan modul, kita dapat mengorganisir kode program secara logis dan memudahkan penggunaan kembali kode terkait. Modul dapat diimpor ke dalam program Python menggunakan pernyataan `import`. Python juga memiliki modul bawaan yang dapat digunakan tanpa perlu menginstal modul tambahan.

Pertemuan 5 (Mengerjakan quiz menggunakan Aplikasi Quizizz)



Pertemuan 6 (Membuat aplikasi Gambar to teks,Mp3,Mp4,Teks to Sound)

1. Gambar To Teks

```
Gambar ke teks > GambarToTekst.py > ...
1 import tkinter as tk
2 from tkinter import filedialog, image_names
3
4 def open_image():
5     file_path = filedialog.askopenfilename()
6     image = image_names.open(file_path)
7     text = image_to_text(image)
8     text_area.insert(tk.END, text)
9
```

- ‘`filedialog.askopenfilename()`’ adalah sebuah fungsi dalam modul `filedialog` pada library Tkinter di Python. Fungsi ini digunakan untuk menampilkan dialog untuk memilih file dari sistem file. Fungsi ini mengembalikan path file yang dipilih oleh pengguna. Fungsi ini memiliki beberapa parameter opsional seperti `initialdir` untuk menentukan direktori awal yang ditampilkan dalam dialog, dan `filetypes` untuk menentukan jenis file yang dapat dipilih oleh pengguna
- ‘`image_names.open(file_path)`’ adalah sebuah fungsi dalam modul PIL (Python Imaging Library) yang digunakan untuk membuka dan mengidentifikasi gambar yang disebutkan dalam `file_path`. Fungsi ini mengembalikan objek gambar PIL (Python Imaging Library).
- Kode `text = image_to_text(image)` dan `text_area.insert(tk.END, text)` mungkin merupakan bagian dari program yang menggunakan modul Tkinter untuk membuat GUI dalam Python. Fungsi `image_to_text` kemungkinan merupakan fungsi buatan pengembang program tersebut yang digunakan untuk melakukan konversi teks dari gambar menggunakan modul seperti pytesseract.

- `text_area.insert(tk.END, text)` digunakan untuk menampilkan hasil teks ke dalam area teks pada GUI menggunakan modul Tkinter. Pada baris ini, `tk.END` digunakan untuk menentukan posisi akhir dari area teks, sehingga teks yang ditampilkan akan ditambahkan pada akhir teks yang sudah ada. Dengan menggunakan modul Tkinter, kita dapat dengan mudah membuat GUI dan menampilkan hasil teks dari konversi gambar ke teks dalam program Python

2. Mp3

```
MP3 > playmp3.py > ...
1 import os
2 from tkinter import Tk, Button, Label, filedialog
3 import pygame
4
```

Pygame adalah sebuah library cross-platform yang gratis dan terbuka sumber untuk pengembangan aplikasi multimedia, seperti permainan video game, menggunakan Python. Pygame menggunakan library Simple DirectMedia Layer (SDL) dan beberapa library lain yang popular untuk abstraksi pembuatan program menjadi tugas lebih intuitif.

Berikut adalah beberapa fitur utama dari Pygame:

- Multimedia : Pygame memungkinkan pengembangan aplikasi multimedia, seperti permainan video game, audio, dan video.
- Grafik: Pygame memungkinkan pengembangan grafik, seperti pembuatan objek, animasi, dan efek visual.
- Input: Pygame memungkinkan pengembangan input, seperti pengolahan input mouse dan keyboard.
- Networking: Pygame memungkinkan pengembangan networking, seperti pengiriman dan penerimaan data antara aplikasi.

Pygame memiliki beberapa tutorial dan dokumentasi yang dapat membantu pengembang dalam memahami dan menggunakan library ini.

```
# Tombol untuk memilih folder dan memainkan MP3
self.select_button = Button(master, text="Pilih Folder", command=self.select_folder)
self.select_button.pack()
self.label.pack(pady=10, padx=5)

self.play_button = Button(master, text="Mainkan MP3", command=self.play_music)
self.play_button.pack()
self.label.pack(pady=10, padx=5)
```

self.select_button digunakan untuk memilih folder dan memainkan MP3, dan **self.play_button** digunakan untuk memainkan MP3. Dengan menggunakan modul Tkinter, kita dapat dengan mudah membuat GUI dan menampilkan nama file MP3 serta memainkan file MP3 dalam program Python

```
def select_folder(self):
    folder_path = filedialog.askdirectory()
    self.label.config(text=f"Folder MP3: {folder_path}")
    self.folder_path = folder_path

def play_music(self):
    try:
        # Ambil semua file MP3 di folder
        mp3_files = [f for f in os.listdir(self.folder_path) if f.endswith(".mp3")]

        # Pilih file MP3 pertama
        mp3_file = os.path.join(self.folder_path, mp3_files[0])

        # Muat dan mainkan file MP3
        pygame.mixer.music.load(mp3_file)
        pygame.mixer.music.play()
        self.label.config(text=f"Sedang memainkan: {mp3_file}")
    except Exception as e:
        self.label.config(text="Error: " + str(e))
```

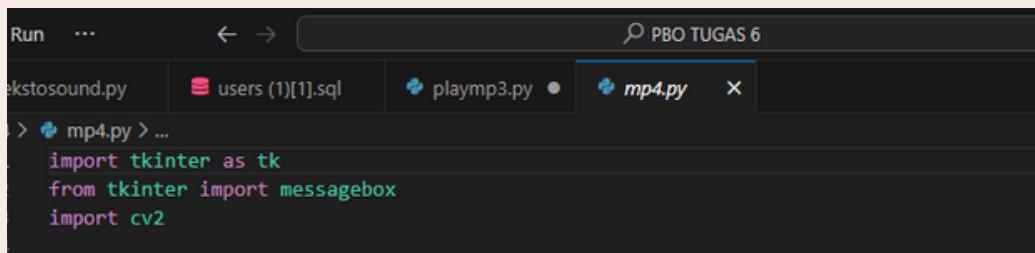
def select_folder(self):

dan

def play_music(self):

Merupakan bagian dari program yang menggunakan modul Tkinter untuk membuat GUI (Graphical User Interface) dalam Python. Fungsi **select_folder** digunakan untuk memilih folder dan menampilkan nama folder yang dipilih, sementara fungsi **play_music** digunakan untuk memainkan file MP3 yang ada dalam folder yang dipilih. Dengan menggunakan modul Tkinter, kita dapat dengan mudah membuat GUI dan memilih folder serta memainkan file MP3 dalam program Python.

3. Mp4



```
Run ... ← → PBO TUGAS 6
tekstosound.py users (1)[1].sql playmp3.py mp4.py
> mp4.py > ...
import tkinter as tk
from tkinter import messagebox
import cv2
```

CV dalam Python adalah singkatan dari Computer Vision, yang merujuk pada kemampuan komputer untuk mengidentifikasi, mendeteksi, dan menganalisis objek dalam gambar atau video. OpenCV adalah sebuah library open-source yang digunakan untuk melakukan computer vision dan image processing dalam Python.

- Pengolahan gambar: OpenCV memungkinkan pengolahan gambar, seperti pembuatan objek, animasi, dan efek visual.
- Pengolahan video: OpenCV memungkinkan pengolahan video, seperti object tracking.
- Pemeriksaan kamera: OpenCV memungkinkan pemeriksaan kamera, seperti camera calibration dan stereo imaging.
- Pemeriksaan objek: OpenCV memungkinkan pemeriksaan objek, seperti feature detection, object detection, dan object tracking.

```
4
5 class VideoPlayerApp:
6     def __init__(self, root, video_path):
7         self.root = root
8         self.root.title("Video Player App")
9
10        self.video_path = video_path
11        self.cap = cv2.VideoCapture(video_path)
12
13        if not self.cap.isOpened():
14            messagebox.showerror("Error", "Could not open video file.")
15            self.root.destroy()
16            return
17
```

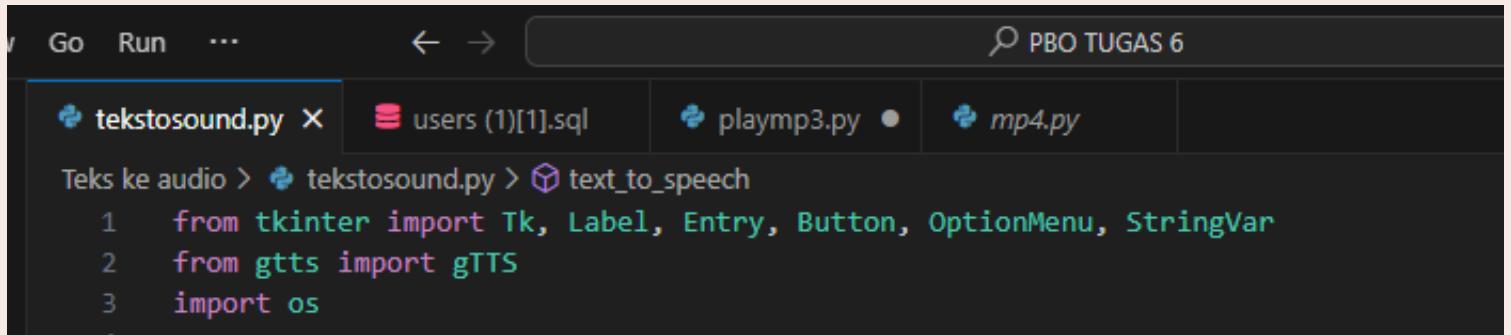
kita jelaskan setiap bagian dari metode `__init__` ini secara detail:

1. `def __init__(self, root, video_path):`: Ini adalah konstruktor dari kelas. Ketika objek dari kelas ini dibuat, metode ini akan dipanggil. Ini mengambil dua parameter: `root`, yang merupakan referensi ke jendela utama (root window) atau jendela aplikasi utama, dan `video_path`, yang merupakan path dari file video yang akan diputar.

2. `self.root = root`: Baris ini menginisialisasi variabel instance `self.root` dengan nilai dari parameter `root`. Ini menunjukkan bahwa kelas ini bergantung pada root window yang diberikan.
3. `self.root.title("Video Player App")`: Ini mengatur judul dari root window menjadi "Video Player App".
4. `self.video_path = video_path`: Baris ini menginisialisasi variabel instance `self.video_path` dengan nilai dari parameter `video_path`. Ini menunjukkan path dari file video yang akan diputar.
5. `self.cap = cv2.VideoCapture(video_path)`: Ini membuat objek capture menggunakan kelas `VideoCapture` dari OpenCV dengan `video_path` yang diberikan. Objek ini digunakan untuk menangkap video dari file, urutan gambar, atau kamera.
6. `if not self.cap.isOpened():`: Ini melakukan pemeriksaan apakah objek capture gagal membuka file video.
7. `messagebox.showerror("Error", "Could not open video file.")`: Jika file video gagal dibuka, pesan kesalahan ditampilkan menggunakan `messagebox` (mungkin dari Tkinter atau pustaka GUI lainnya).
8. `self.root.destroy()`: Jendela root akan ditutup jika terjadi kesalahan saat membuka file video.
9. `return`: Ini mengakhiri metode konstruktor.

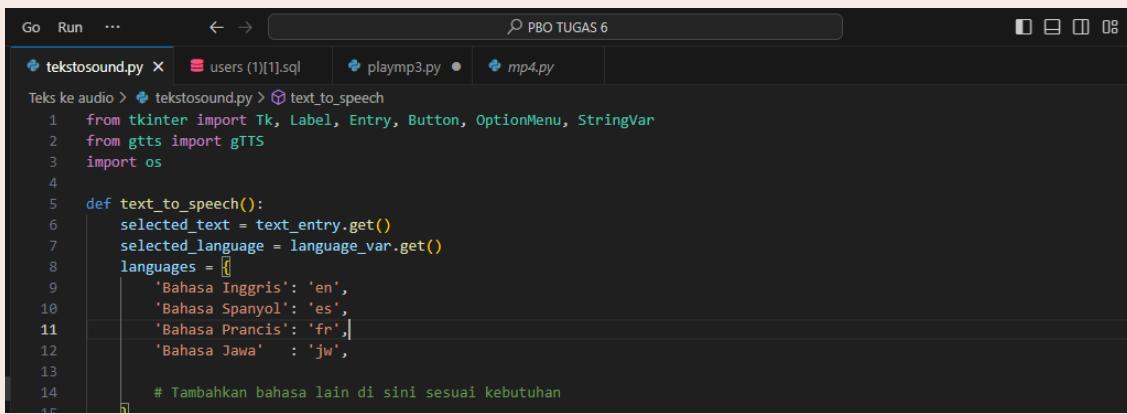
Secara keseluruhan, metode ini digunakan untuk menginisialisasi objek dari kelas yang mungkin merupakan aplikasi pemutar video. Jika file video gagal dibuka, pesan kesalahan akan ditampilkan dan aplikasi akan ditutup.

4. Teks to sound



```
Teks ke audio > tekstosound.py > text_to_speech
1  from tkinter import Tk, Label, Entry, Button, OptionMenu, StringVar
2  from gtts import gTTS
3  import os
```

gTTS adalah sebuah library Python yang digunakan untuk mengonversi teks menjadi suara atau audio. Library ini memanfaatkan API Google Translate text-to-speech untuk menghasilkan suara dari teks yang diberikan. gTTS dapat menghasilkan file audio dalam format mp3 yang dapat disimpan ke dalam file atau dimainkan langsung. Library ini dapat diinstal menggunakan pip dan dapat digunakan sebagai modul Python atau sebagai utilitas baris perintah.

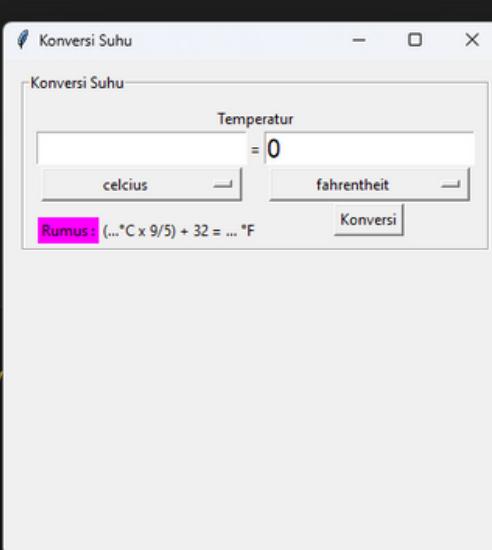


```
Teks ke audio > tekstosound.py > text_to_speech
1  from tkinter import Tk, Label, Entry, Button, OptionMenu, StringVar
2  from gtts import gTTS
3  import os
4
5  def text_to_speech():
6      selected_text = text_entry.get()
7      selected_language = language_var.get()
8      languages = [
9          'Bahasa Inggris': 'en',
10         'Bahasa Spanyol': 'es',
11         'Bahasa Prancis': 'fr',
12         'Bahasa Jawa' : 'jw',
13     ]
14
15     # Tambahkan bahasa lain di sini sesuai kebutuhan
```

Fungsi `text_to_speech()` yang diberikan mengambil teks yang dipilih dan bahasa yang dipilih dari antarmuka pengguna (GUI), lalu mengonversinya menjadi teks-to-suara menggunakan kode tertentu yang tidak terlihat dalam potongan kode yang diberikan

1. `selected_text = text_entry.get()`: Ini mengambil teks yang dimasukkan pengguna melalui antarmuka teks, mungkin dari suatu entri teks atau kotak teks dalam aplikasi GUI. Fungsi `get()` dipanggil pada objek `text_entry`, yang mungkin merupakan objek dari pustaka GUI seperti Tkinter.
2. `selected_language = language_var.get()`: Ini mengambil bahasa yang dipilih oleh pengguna dari antarmuka pengguna. Nilai ini mungkin dipilih dari daftar pilihan atau opsi bahasa yang tersedia dalam aplikasi. Fungsi `get()` dipanggil pada objek `language_var`, yang mungkin merupakan objek variabel dari pustaka GUI.
3. `languages = {...}`: Ini adalah kamus yang memetakan nama bahasa ke kode bahasa tertentu. Setiap entri dalam kamus ini memiliki kunci sebagai nama bahasa dan nilai sebagai kode bahasa yang sesuai. Ini digunakan untuk mencocokkan bahasa yang dipilih oleh pengguna dengan kode bahasa yang diperlukan oleh layanan atau pustaka konversi teks ke suara.

Pertemuan 7 (Aplikasi Konversi Suhu Berbasis Windows)



The screenshot shows a Python application window titled "Konversi Suhu". Inside the window, there is a label "Temperatur" above a text input field containing the value "0". Below the input field are two dropdown menus: "celcius" and "fahrenheit". To the right of these dropdowns is a button labeled "Konversi". A pink box highlights the formula "Rumus: (...°C x 9/5) + 32 = ...°F" located below the input field.

```
tekstosound.py ● kovertersuhu.py ✘ users (1)[1].sql playmp3.py ● mp4.py
C: > Users > acer > Documents > PBO CIA > PBO TUGAS 7 > kovertersuhu.py > ...
1  from tkinter import *
2  import tkinter.font
3
4  root = Tk()
5  root.title("Konversi Suhu")
6  root.geometry("400x400")
7
8  # mengatur ukuran font
9  thefont = tkinter.font.Font(size=15)
10
11 # membuat sebuah frame baru
12 nwframe = LabelFrame(root, text="Konversi Suhu", padx=10, pady=10)
13 nwframe.place(x=15, y=10)
14
15 # judul dan tulisan samadengan
16 judul = Label(nwframe, text="Temperatur").grid(row=0, columnspan=3)
17 samde = Label(nwframe, text="=").grid(row=1, column=1)
18
19
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\acer\Documents\PBO CIA\PBO TUGAS 6> & C:/Users/acer/AppData/Local/Programs/AS 7/kovertersuhu.py
Ln 1, Col 1  Spaces: 4  UTF-8  LF  Python  3.11.7 32-bit
```

```
C:\Users\acer\Documents\PBO CIA\PBO TUGAS 7> kovertersuhu.py ...
53     # menuiskan rumus dan menentukan hitungan
54     if kotaksatu == "celcius" and kotakdua == "fahrentheit":
55         rmsnya = Label(nwframe,text="("+e1.get()+"°C × 9/5) + 32° = ... °F")
56         hitungan = (nomer1 * 9/5) + 32
57     elif kotaksatu == "celcius" and kotakdua == "kelvin":
58         rmsnya = Label(nwframe,text=e1.get()+"°C + 273 = ... K")
59         hitungan = (nomer1 + 273)
60     elif kotaksatu == "celcius" and kotakdua == "reamur":
61         rmsnya = Label(nwframe,text="(4/5 × "+e1.get()+"°C) = ... R")
62         hitungan = (4/5 * nomer1)
63     elif kotaksatu == "fahrentheit" and kotakdua == "celcius":
64         rmsnya = Label(nwframe,text="("+e1.get()+"°F - 32) × 5/9 = ... °C")
65         hitungan = (nomer1 - 32) * 5/9
66     elif kotaksatu == "fahrentheit" and kotakdua == "kelvin":
67         rmsnya = Label(nwframe,text="("+e1.get()+"°F - 32) × 5/9 + 273 = ... K")
68         hitungan = (nomer1 - 32) * 5/9 + 273
69     elif kotaksatu == "fahrentheit" and kotakdua == "reamur":
70         rmsnya = Label(nwframe,text="("+e1.get()+"°F - 32) × 4/9 = ... R")
71         hitungan = (nomer1 - 32) * 4/9
72     elif kotaksatu == "reamur" and kotakdua == "celcius":
73         rmsnya = Label(nwframe,text="5/4 × "+e1.get()+"R) = ... °C")
74         hitungan = (5/4 * nomer1)
75     elif kotaksatu == "reamur" and kotakdua == "fahrentheit":
76         rmsnya = Label(nwframe,text="(9/4 × "+e1.get()+"R) + 32 = ... F")
77         hitungan = (9/4 * nomer1) + 32
78     elif kotaksatu == "reamur" and kotakdua == "kelvin":
79         rmsnya = Label(nwframe,text="(5/4 × "+e1.get()+"R) + 273 = ... K")
80         hitungan = (5/4 * nomer1) + 273
81     elif kotaksatu == "kelvin" and kotakdua == "celcius":
82         rmsnya = Label(nwframe,text=e1.get()+"K - 273 = ... °C")
83         hitungan = (nomer1 - 273)
84     elif kotaksatu == "kelvin" and kotakdua == "fahrentheit":
```

Ln 1, Col 1 Spaces: 4 UTF-8 LF Python 3.1

Aplikasi konversi suhu Python umumnya memiliki fitur-fitur berikut:

- Konversi antar unit suhu: Aplikasi dapat melakukan konversi suhu dari satu unit ke unit lain, misalnya dari Celsius ke Fahrenheit, Kelvin, atau Reamur.
- Antarmuka pengguna (UI): Aplikasi umumnya dilengkapi dengan antarmuka pengguna yang memudahkan pengguna untuk memasukkan nilai suhu dan memilih unit konversi.
- Validasi input: Aplikasi dapat melakukan validasi terhadap input pengguna untuk memastikan bahwa input yang dimasukkan sesuai dengan format yang diharapkan.
- Tampilan hasil konversi: Setelah konversi dilakukan, hasil konversi suhu ditampilkan kepada pengguna dalam unit yang diinginkan.

Pertemuan 8 (UTS)

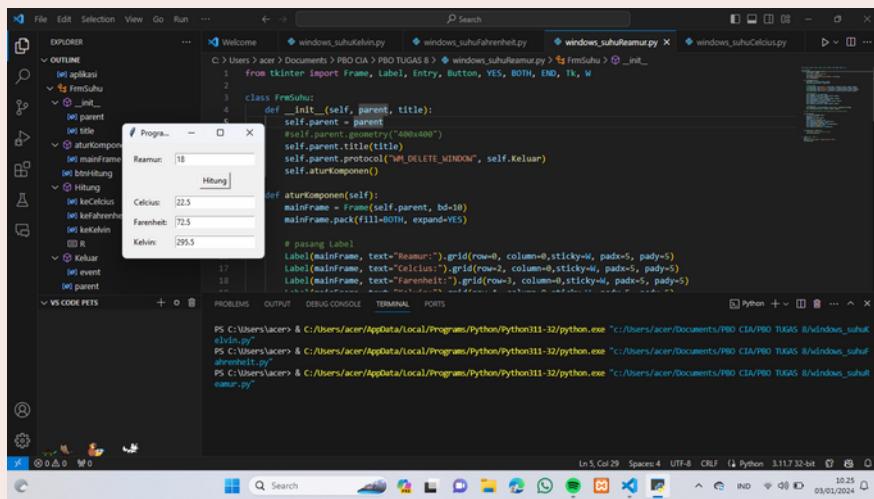
**Melaksanakan UTS 07 Desember
2023**

Pertemuan 9 (Class & Object pada pembuatan program konversi suhu berbasis windows)

```
File Edit Selection View Go Run ... Search ... Welcome windows_suhuKelvin.py windows_suhuFahrenheit.py windows_suhuReamur.py windows_suhuCelcius.py ... EXPLORER OUTLINE aplikasi FmSuhu _init_ parent title aturkomponen mainFrame bntHitung Hitung K Keluar event parent VS CODE PETS + o ... Program... Celsius: 30 Hitung Reamur: 28.8 Fahrenheit: 96.8 Kelvin: 309 # pasang Label self.parent.title(title) self.parent.protocol("WM_DELETE_WINDOW", self.Keluar) self.aturkomponen() turukompon(self): mainframe = Frame(self.parent, bd=10) mainframe.pack(fill=BOTH, expand=YES) Label(mainframe, text="Celsius").grid(row=0, column=0, sticky=W, padx=5, pady=5) Label(mainframe, text="Reamur").grid(row=1, column=0, sticky=W, padx=5, pady=5) Label(mainframe, text="Fahrenheit").grid(row=2, column=0, sticky=W, padx=5, pady=5) Label(mainframe, text="Kelvin").grid(row=3, column=0, sticky=W, padx=5, pady=5) PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + v ... Line 10, Col 1 Spaces:4 UTF-8 CRLF 3.11.7 32-bit ... PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuReamur.py" PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuFahrenheit.py" PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuKelvin.py" PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuCelcius.py" ...
```

```
File Edit Selection View Go Run ... Search ... Welcome windows_suhuKelvin.py windows_suhuFahrenheit.py windows_suhuReamur.py windows_suhuCelcius.py ... EXPLORER OUTLINE aplikasi FmSuhu _init_ parent title aturkomponen mainFrame bntHitung Hitung K Keluar event parent VS CODE PETS + o ... Program... Fahrenheit: 14 Hitung Reamur: -26.66666666666666 Kelvin: -25.333333333333332 Celsius: 247.6666666666666 # pasang Label self.parent.title(title) self.parent.protocol("WM_DELETE_WINDOW", self.Keluar) self.aturkomponen() turukompon(self): mainframe = Frame(self.parent, bd=10) mainframe.pack(fill=BOTH, expand=YES) Label(mainframe, text="Fahrenheit").grid(row=0, column=0, sticky=W, padx=5, pady=5) Label(mainframe, text="Reamur").grid(row=1, column=0, sticky=W, padx=5, pady=5) Label(mainframe, text="Kelvin").grid(row=2, column=0, sticky=W, padx=5, pady=5) Label(mainframe, text="Celsius").grid(row=3, column=0, sticky=W, padx=5, pady=5) PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + v ... Line 8, Col 25 Spaces:4 UTF-8 CRLF 3.11.7 32-bit ... PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuReamur.py" PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuFahrenheit.py" PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuKelvin.py" PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuCelcius.py" ...
```

```
File Edit Selection View Go Run ... Search ... Welcome windows_suhuKelvin.py windows_suhuFahrenheit.py windows_suhuReamur.py windows_suhuCelcius.py ... EXPLORER OUTLINE aplikasi FmSuhu _init_ parent title aturkomponen mainFrame bntHitung Hitung K Keluar event parent VS CODE PETS + o ... Program... Kelvin: 20 Hitung Reamur: -257.0 Celsius: -253 Fahrenheit: -205.0 # pasang Label self.parent.title(title) self.parent.protocol("WM_DELETE_WINDOW", self.Keluar) self.aturkomponen() turukompon(self): mainframe = Frame(self.parent, bd=10) mainframe.pack(fill=BOTH, expand=YES) Label(mainframe, text="Kelvin").grid(row=0, column=0, sticky=W, padx=5, pady=5) Label(mainframe, text="Reamur").grid(row=1, column=0, sticky=W, padx=5, pady=5) Label(mainframe, text="Celsius").grid(row=2, column=0, sticky=W, padx=5, pady=5) Label(mainframe, text="Fahrenheit").grid(row=3, column=0, sticky=W, padx=5, pady=5) PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + v ... Line 8, Col 1 Spaces:4 UTF-8 CRLF 3.11.7 32-bit ... PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuReamur.py" PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuFahrenheit.py" PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuKelvin.py" PS C:\Users\acer> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 8/windows_suhuCelcius.py" ...
```



Class pada Python adalah sebuah blueprint atau cetakan dari objek yang ingin dibuat. Class digunakan untuk mengelompokkan data dan fungsi yang berkaitan dengan objek tertentu. Objek sendiri adalah sebuah instance dari class yang memiliki atribut dan perilaku yang didefinisikan oleh class tersebut. Untuk membuat sebuah class pada Python, digunakan keyword "class" diikuti dengan nama class yang ingin didefinisikan dan diikuti dengan ":". Setelah itu, dapat didefinisikan properties dan methods yang dimiliki oleh class tersebut.

Object pada Python adalah contoh unik dari struktur data yang didefinisikan oleh kelasnya. Object terdiri dari kedua anggota data (variabel kelas dan variabel contoh) dan metode. Variabel kelas adalah data anggota (variabel kelas dan variabel contoh) dan metode, diakses melalui notasi titik.

Pertemuan 10 (Pengenalan Python Database)

The screenshot shows the phpMyAdmin interface for the 'kampusumc' database. The left sidebar lists databases: information_schema, kampusumc, mahasiswa, mysql, performance_schema, phpmyadmin, and test. The 'matakuliah' table under 'kampusumc' is selected. The main area displays the table structure and data. The table has columns: id, kodemk, namamk, and sks. The data shows 8 rows of course information with their respective codes, names, and credit hours.

	id	kodemk	namamk	sks
1	1	KALK22TI	KALKULUS	2
2	2	AIK22TI	AIK 2	2
3	3	KODT22TI	KOMUNIKASI DATA	2
4	4	STPB22TI	STATISTIK DAN PROBABILITAS	2
5	5	AROK22TI	ARSITEKTUR DAN ORGANISASI KOMPUTER	3
6	6	PBO22TI	PEMROGRAMAN II	3
7	7	STDA22TI	STRUKTUR DATA DAN ALGORITMA	3
8	8	APSI22TI	SISTEM INFORMASI	3

Tampilan Database diatas ialah database membuat Mata Kuliah. Dengan Columns 4,yaitu dengan nama ID,Kodemk,NamaMK,beserta SKS.

The screenshot shows the phpMyAdmin interface for the 'kampusumc' database. The left sidebar lists databases: information_schema, kampusumc, mahasiswa, mysql, performance_schema, phpmyadmin, and test. The 'matakuliah' table under 'kampusumc' is selected. The main area displays the table structure. The table has columns: id, kodemk, namamk, and sks. The data shows 8 rows of course information with their respective codes, names, and credit hours.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(100)	utf8mb4_general_ci		No	None		AUTO_INCREMENT	Change Drop More
2	kodemk	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
3	namamk	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
4	sks	enum('1', '2', '3', '4', '6')	utf8mb4_general_ci		No	None			Change Drop More

Dengan 4 table yang dibuat dengan masing-masing ada yang type data nya berbeda. Yaitu :

- 1.Table ‘id’ yaitu dengan type data ‘int’ dengan panjang karakter 100 dan menggunakan Primary Key.
- 2.Table ‘kodemk’ yaitu dengan type data ‘varchar’ dengan panjang karakter 10.
- 3.Table ‘namamk’ yaitu dengan type data ‘varchar’ dengan panjang karakter 255.
- 4.Table ‘sks’ dengan type data ‘enum’

- Setiap table mengapa ada panjang karakternya ? karena agar misalkan kita mau menginputkan kata-kata di dalam tablenya ada jumlah maksimal karakternya.
- Setaip table juga ada type data masing-masing agar pada saat menginputkan angka atau kata.

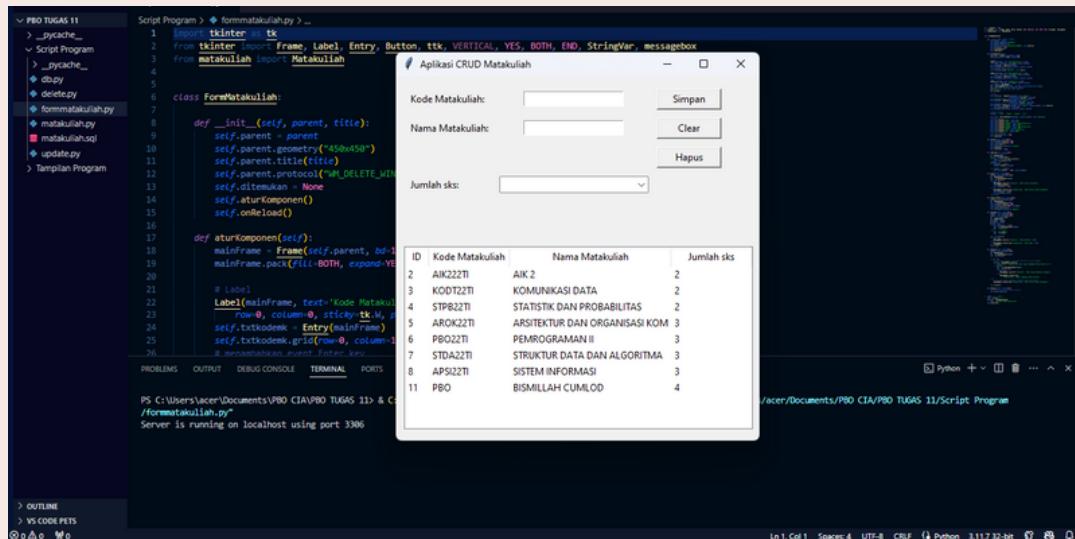
Adapun peraturan dalam penulisan Database :

- Pemilihan Nama Database: Pilihlah nama database yang deskriptif dan mudah dipahami, sesuai dengan tujuan dan konten dari database tersebut.
- Struktur Tabel: Pastikan struktur tabel (kolom dan tipe data) telah direncanakan dengan baik sesuai dengan kebutuhan aplikasi.
- Koneksi Database: Pastikan parameter koneksi ke database seperti *hostname*, *username*, *password*, dan *nama database* telah dikonfigurasi dengan benar dalam skrip aplikasi.
- Keamanan: Selalu perhatikan keamanan database, termasuk hak akses pengguna, enkripsi data sensitif, dan langkah-langkah perlindungan data lainnya.
- Backup dan Restore: Selalu lakukan backup secara berkala dan pastikan prosedur restore database telah disiapkan dengan baik.

Pertemuan 11 (Python Database dengan membuat form matakuliah)

Form matakuliah ini kami buat untuk memudahkan pengguna untuk memasukan matakuliah pada form ini. Dan kami juga mengaplikasikan CRUD pada pembuatan form ini Creat(Untuk menginputkan data matakuliah), Read(untuk melihat output an matakuliah apa asaja yang kami inputkan), Update(untuk kita menambah dan mengubah data matakuliah), dan Delete(untuk menghapus data matakuliah apabila ada kesalahan).

Dibawah ini ada hasil output an data matakuliah yang kami telah buat :

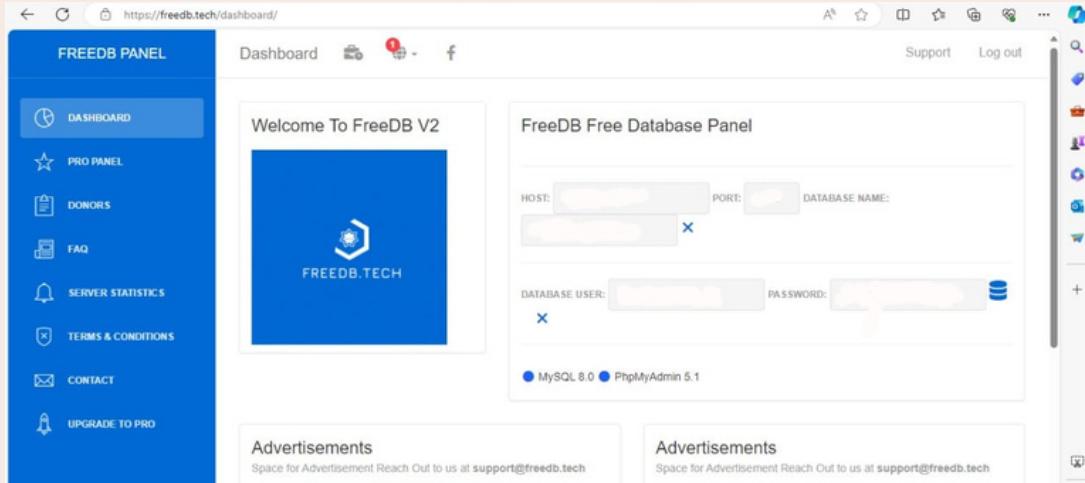


Ini untuk database nya pada file db.py

```
1 import mysql.connector as mc
2
3
4 class DBConnection:
5
6     def __init__(self):
7         self.host = "localhost"
8         self.port = 3306
9         self.name = "pbo"
10        self.user = "root"
11        self.password = ""
12        self.conn = None
13        self.cursor = None
14        self.result = None
15        self.connected = False
16        self.affected = 0
17
18        self.connect()
```

pada "self.name" kita namakan dengan nama tabel sesuai dengan table di database yang kita buat.

Pertemuan 12 (Python Database menggunakan remote mysql server on freedb.tech)



Form matakuliah ini kami buat untuk memudahkan pengguna untuk memasukan matakuliah pada form ini. Dan kami juga mengaplikasikan CRUD pada pembuatan form ini Creat(Untuk menginputkan data matakuliah), Read(untuk melihat output an matakuliah apa asaja yang kami inputkan),

Update(untuk kita menambah dan mengubah data matakuliah), dan Delete(untuk menghapus data matakuliah apabila ada kesalahan). Dibawah ini ada hasil output an data matakuliah yang kami telah buat :

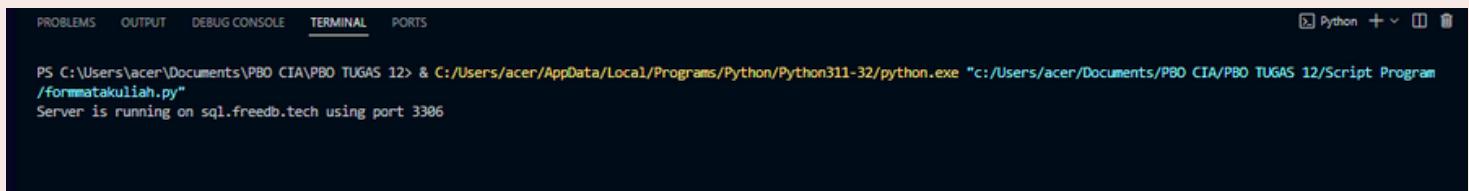
Dibawah ini ada hasil output an data matakuliah yang kami telah buat :

A screenshot of a Python application titled "PBO TUGAS 12". The application window shows a table with columns ID, Kode Matakuliah, Nama Matakuliah, and Jumlah sks. The data is as follows:

ID	Kode Matakuliah	Nama Matakuliah	Jumlah sks
2	AIK2221	AIK 2	2
3	KODT2221	KOMUNIKASI DATA	2
5	AROK2221	ARSTEKTUR DAN ORGANISASI KOMPUTER	3
6	PBO2221	PEMROGRAMAN II	3
7	STD2221	STRUKTUR DATA DAN ALGORITMA	3
8	APSI2221	SISTEM INFORMASI	3

The Python code in the editor shows the creation of a Tkinter-based application for managing course data. It includes functions for saving, clearing, and deleting data, as well as a table view of the course information.

Dan apabila settingan benar maka akan muncul seperti ini :



A screenshot of a terminal window from a code editor. The tab bar at the top shows 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL' (which is underlined), and 'PORTS'. The status bar at the top right shows 'Python + ⌂ ⌂'. The terminal output is as follows:

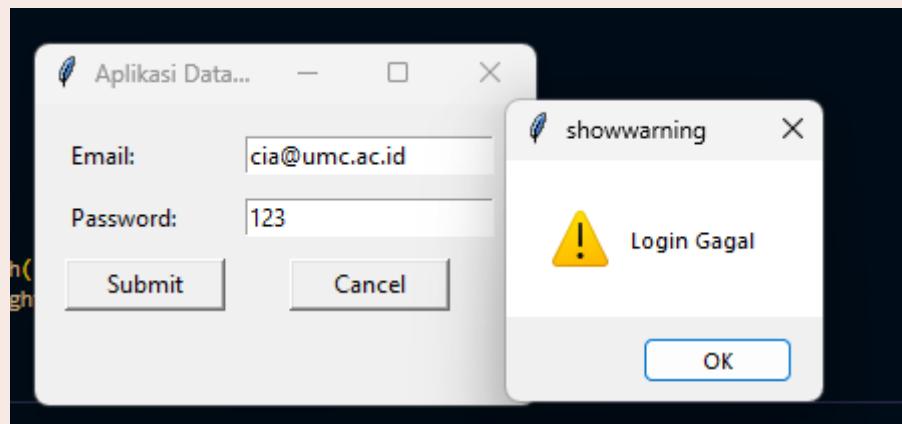
```
PS C:\Users\acer\Documents\PBO CIA\PBO TUGAS 12> & C:/Users/acer/AppData/Local/Programs/Python/Python311-32/python.exe "c:/Users/acer/Documents/PBO CIA/PBO TUGAS 12/Script Program /formmatakuliah.py"
Server is running on sql.freedb.tech using port 3306
```

Pertemuan 13 (Aplikasi login multi user python-mysql)

Form login sering digunakan dalam program Python (dan juga dalam pengembangan perangkat lunak secara umum) untuk beberapa alasan:

1. Keamanan: Dengan menggunakan form login, Kami dapat membatasi akses ke program atau fitur-fiturnya hanya kepada pengguna yang memiliki kredensial yang valid. Ini membantu mencegah akses yang tidak sah atau tidak diinginkan.
 2. Identifikasi Pengguna: Form login memungkinkan pengguna untuk mengidentifikasi diri mereka sendiri sebelum menggunakan program. Ini memungkinkan pengguna untuk memiliki pengaturan pribadi, data yang disimpan, atau hak akses khusus berdasarkan siapa yang masuk.
 3. Pemeliharaan Log: Dengan form login, Kami dapat mencatat siapa yang menggunakan program dan kapan. Ini berguna untuk audit, pemeliharaan, atau bahkan penegakan hukum jika diperlukan.
 4. Personalisasi: Form login memungkinkan kami untuk menyesuaikan pengalaman pengguna berdasarkan pengguna yang masuk. Kami dapat menyajikan pengguna dengan konten yang disesuaikan, opsi yang dipersonalisasi, atau data yang relevan sesuai dengan profil mereka.
 5. Kontrol Akses: Dengan form login, Kami dapat menerapkan kontrol akses yang lebih rumit. Misalnya, Kami dapat memberikan hak akses yang berbeda kepada pengguna berbeda, atau bahkan grup pengguna.
- Meskipun tidak semua program Python memerlukan form login, penggunaannya umum dalam aplikasi web, sistem manajemen basis data, atau program-program yang menangani data sensitif atau penting.

Dibawah ini ada hasil output an form login :



Dan ini outputan pada saat memilih luas mana yang akan dihitung :



Kita disini akan menghitung luas dari persegi dan akan mengoutputkan ini :

