



ALGORITMA DAN PEMROGRAMAN

TUGAS BESAR

BY KELOMPOK 20



GROUP MEMBER



- 1 Naufal Rasydan Ramdhani - 102022300116
- 2 Ardi Khalifah - 102022300162
- 3 M Ilham Fathurahman - 102022330055
- 4 Muhammad Adifa Faiz - 102022300146
- 5 Muhammad Faris Alfaqih - 102022300372

TOPICS

LATAR BELAKANG

PROSES CRUD (CREATE, READ, UPDATE DAN DELETE)

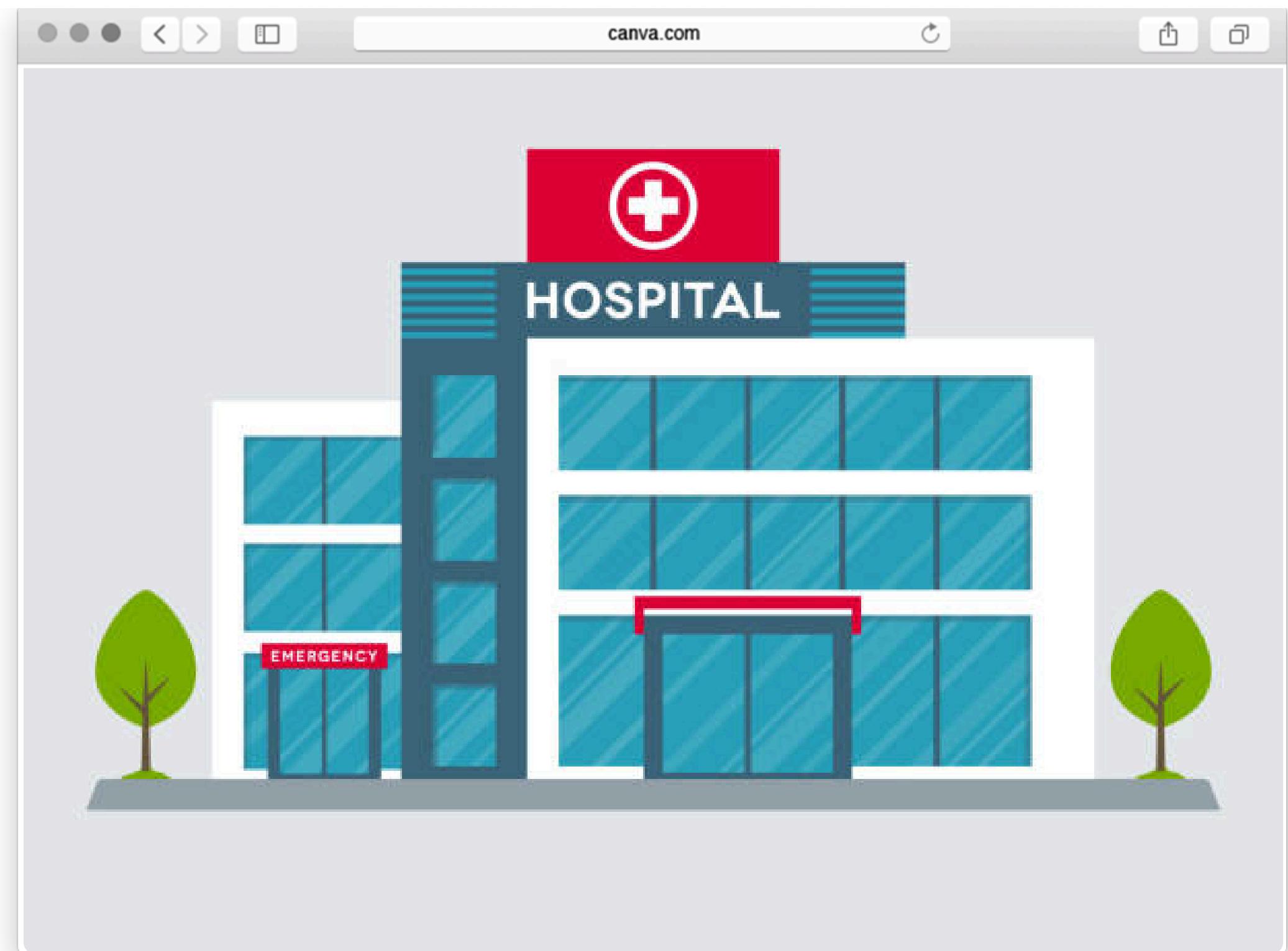
PROSES SEARCHING

PROSES FILTERING

DATA VISUALISASI

KONEKSI KE DATABASE

MENGGUNAKAN LIBRARY





Latar belakang

Program ini dibuat dengan tujuan untuk membangun sistem pendataan rumah sakit yang terkomputerisasi. Sistem ini dirancang untuk membantu rumah sakit dalam mengelola data pasien, dokter, dan penyakit secara lebih efisien dan efektif.

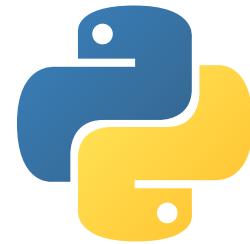


```
1 import streamlit as st
2 from st_on_hover_tabs import on_hover_tabs
3 import base64
4 import sqlite3 as sq
5 import pandas as pd
6 import matplotlib.pyplot as plt
7 import os
8 from streamlit import components
9 import json
10 from pathlib import Path
```

Library

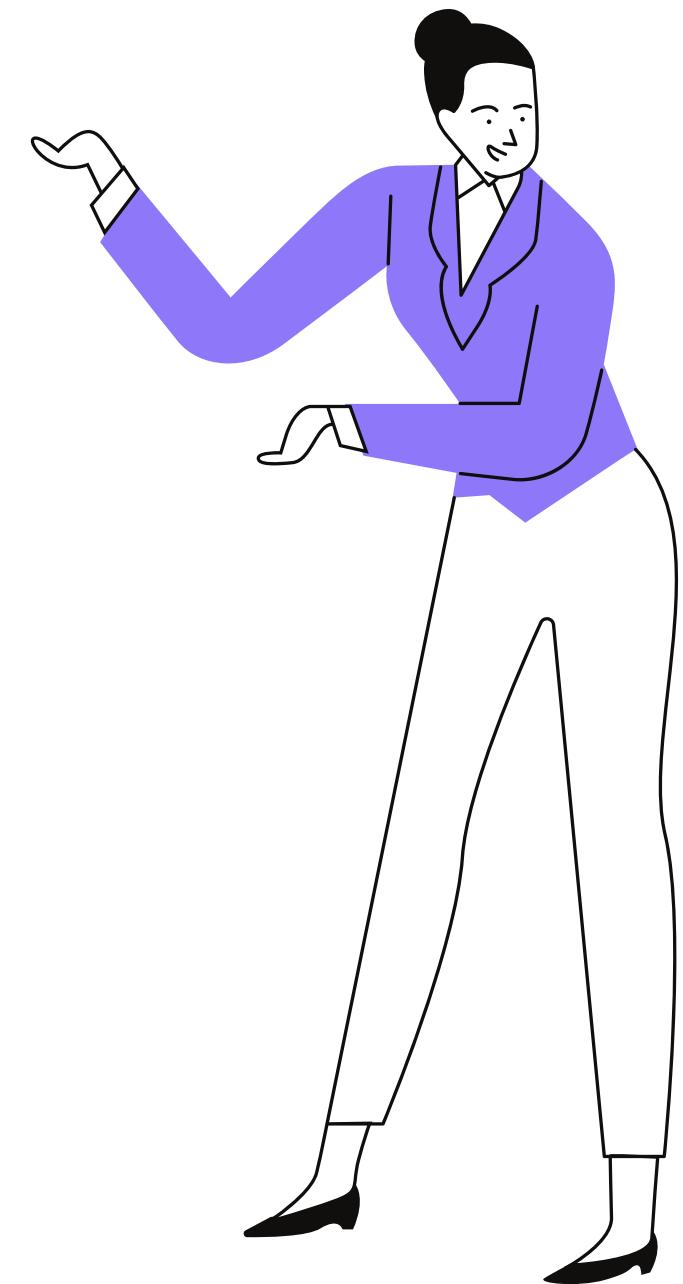
Import Library yang Diperlukan :

- **Stremalit** : Library Python untuk membuat aplikasi web interaktif dengan cepat dan mudah.
- **Base64** : Digunakan untuk encoding dan decoding data menjadi format Base64.
- **Sqlite3** : Python library untuk berinteraksi dengan database SQLite
- **Pandas** : Python library untuk manipulasi dan analisis data
- **Matplotlib.pyplot** : Submodul dari Matplotlib yang menyediakan fungsi untuk membuat berbagai jenis plot dan visualisasi data secara mudah.
- **Os** : Library Python yang menyediakan berbagai fungsi untuk berinteraksi dengan sistem operasi, seperti mengelola file dan direktori.
- **Json** : Library standar untuk membaca dan menulis data dalam format JSON
- **St_on_hover_tabs** : Custom component untuk Streamlit yang memungkinkan pembuatan tab interaktif yang merespon hover
- **Pathlib** : Library standar untuk mengelola path file dan direktori dengan cara yang lebih mudah dan intuitif



```
1 with open('data rumah sakit_0920.json') as f:  
2     data = json.load(f)
```

- Membuka file dengan nama (**data rumah sakit_0920**) dan mewakilkannya sebagai **f**
- Membuat variabel “**data**” yang berisi fungsi untuk membaca dataset “**f**”.





```
1 def database():
2     conn = sq.connect('tesss.db')
3     return conn
```

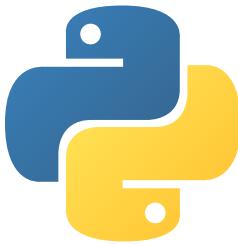


```
1 def membuat_table():
2     conn = database()
3     c = conn.cursor()
4     c.execute('''CREATE TABLE IF NOT EXISTS dokter (
5         ID_Dokter INTEGER PRIMARY KEY AUTOINCREMENT,
6         nama TEXT NOT NULL,
7         spesialisasi TEXT NOT NULL)''')
8     c.execute('''CREATE TABLE IF NOT EXISTS pasien (
9         ID_Pasien INTEGER PRIMARY KEY AUTOINCREMENT,
10        nama TEXT NOT NULL,
11        umur INTEGER NOT NULL,
12        tanggal_masuk DATE,
13        alamat TEXT NOT NULL,
14        dokter_id INTEGER,
15        FOREIGN KEY (dokter_id) REFERENCES dokter (ID_Dokter)'''
16    )
17    c.execute('''CREATE TABLE IF NOT EXISTS penyakit (
18        ID_Penyakit INTEGER PRIMARY KEY AUTOINCREMENT,
19        nama TEXT NOT NULL,
20        pasien_id INTEGER,
21        tanggal_masuk DATE,
22        FOREIGN KEY (pasien_id) REFERENCES pasien (ID_Pasien)'''
```

Fungsi “database” berfungsi untuk mendefinisikan sebuah fungsi untuk membuat koneksi ke database “tesss.db”.

Fungsi "membuat_table" berfungsi untuk membuat tabel-tabel yang belum dibuat sebelumnya ke dalam sebuah database.

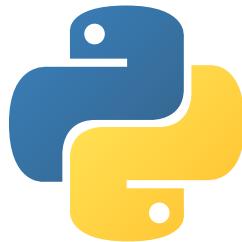




```
 1 def muat_data():
 2     conn = database()
 3     c = conn.cursor()
 4     c.execute('SELECT COUNT(*) FROM dokter')
 5     dokter_count = c.fetchone()[0]
 6     c.execute('SELECT COUNT(*) FROM pasien')
 7     pasien_count = c.fetchone()[0]
 8     if dokter_count == 0 and pasien_count == 0:
 9         for dokter in data['dokter']:
10             c.execute('INSERT INTO dokter (nama, spesialisasi) VALUES (?,?)', (dokter['nama'], dokter['spesialisasi']))
11         for pasien in data['pasien']:
12             c.execute('INSERT INTO pasien (nama, umur, alamat, tanggal_masuk) VALUES (?,?,?,?,?)',
13                       (pasien['nama'], pasien['umur'], pasien['alamat'], pasien['tanggal_masuk']))
14             pasien_id = c.lastrowid
15             c.execute('INSERT INTO penyakit (nama, pasien_id, tanggal_masuk) VALUES (?,?,?)',
16                       (pasien['penyakit'], pasien_id, pasien['tanggal_masuk']))
17     conn.commit()
18     conn.close()
```

Fungsi ini memuat data dokter dan pasien dari variabel data ke dalam database. Jika database kosong, semua data dokter dan pasien akan dimuat. Jika database tidak kosong, data tidak akan dimuat ulang.





Membuat variabel "ADMIN_0920" yang berisi username dari admin dan password. Ini akan berguna ketika kita ingin masuk sebagai admin pada program.



```
1 ADMIN_0920 = {  
2     "admin": "Ut0mo.27"  
3 }
```



```
1 def login_admin(username, password):  
2     return ADMIN_0920.get(username) == password
```

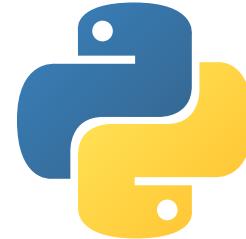
Membuat fungsi "login_admin" dengan range username dan password.



```
1 def login_pasien(nama):  
2     conn = database()  
3     c = conn.cursor()  
4     c.execute('SELECT * FROM pasien WHERE nama = ?', (nama,))  
5     result = c.fetchone()  
6     conn.close()  
7     return result
```

fungsi "login_pasien" bertujuan untuk melakukan proses login bagi pasien berdasarkan nama mereka di database.





```
1 def get_role(username):
2     if username in ADMIN_0920:
3         return "admin"
4     else:
5         return "pasien"
6
```

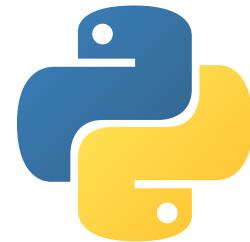
Fungsi "get_role" bertujuan untuk menentukan peran (role) dari seorang pengguna berdasarkan username mereka.



```
1 def masukan_dokter(nama_dokter_0920, spesialisasi_dokter_0920):
2     conn = database()
3     c = conn.cursor()
4     c.execute('INSERT INTO dokter (nama, spesialisasi) VALUES (?, ?)', (nama_dokter_0920, spesialisasi_dokter_0920))
5     conn.commit()
6     conn.close()
```

Fungsi "masukan_dokter" bertujuan untuk memasukkan data dokter dan spesialisasi baru ke dalam tabel dokter di database.





Fungsi “melihat_data_dokter” digunakan untuk mengambil data dari tabel dokter dalam database.

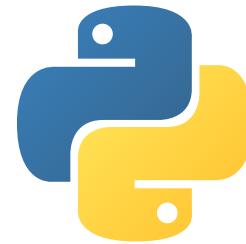
```
● ● ●  
1 def melihat_data_dokter():  
2     conn = database()  
3     c = conn.cursor()  
4     c.execute('SELECT * FROM dokter')  
5     dokter_0920 = pd.DataFrame(c.fetchall(), columns=[desc[0] for desc in c.description])  
6     conn.close()  
7     return dokter_0920
```

```
● ● ●  
1 def melihat_data_pasien():  
2     conn = database()  
3     query = ''  
4     SELECT pa.ID_Pasien, pa.nama AS Nama_Pasien, pa.umur, pa.alamat, py.nama AS Nama_Penyakit, py.tanggal_masuk  
5     FROM pasien pa  
6     LEFT JOIN penyakit py ON pa.ID_Pasien = py.pasien_id  
7     ''  
8     c = conn.cursor()  
9     c.execute(query)  
10    pasien_0920 = pd.DataFrame(c.fetchall(), columns=[desc[0] for desc in c.description])  
11    conn.close()  
12    return pasien_0920
```

Fungsi “melihat_data_pasien” bertujuan untuk mengambil data pasien beserta informasi penyakit yang dialami berdasarkan data dari tabel pasien dan penyakit.



Visual Studio Code



```
1 def cari_dokter(nama_dokter_0920):
2     conn = database()
3     c = conn.cursor()
4     c.execute('SELECT * FROM dokter WHERE nama LIKE ?', (nama_dokter_0920,))
5     dokter_0920 = pd.DataFrame(c.fetchall(), columns=[desc[0] for desc in c.description])
6     conn.close()
7     return dokter_0920
8
```

Fungsi “cari_dokter” digunakan untuk mencari dokter berdasarkan nama dokter yang diberikan.

Fungsi “cari_pasien” digunakan untuk mencari pasien berdasarkan nama pasien yang diberikan.

```
1 def cari_pasien(nama_pasien_0920):
2     conn = database()
3     c = conn.cursor()
4     c.execute('SELECT * FROM pasien WHERE nama LIKE ?', (nama_pasien_0920,))
5     pasien_0920 = pd.DataFrame(c.fetchall(), columns=[desc[0] for desc in c.description])
6     conn.close()
7     return pasien_0920
```





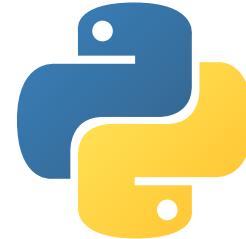
```
1 def melihat_data_diri(username_0920):
2     conn = database()
3     c = conn.cursor()
4     c.execute('SELECT * FROM pasien WHERE nama = ?', (username_0920,))
5     data_diri_0920 = pd.DataFrame(c.fetchall(), columns=[desc[0] for desc in c.description])
6     conn.close()
7     return data_diri_0920
```

Fungsi “melihat_data_diri” digunakan untuk mencocokan nama pasien dalam database dengan nama pasien yang diberikan.

```
1 def melihat_penyakit_diri(username_0920):
2     conn = database()
3     query = '''
4     SELECT p.nama as Penyakit, p.tanggal_masuk as Tanggal_Masuk
5     FROM penyakit p
6     JOIN pasien pa ON p.pasien_id = pa.ID_Pasien
7     WHERE pa.nama = ?
8     ...
9
10    c = conn.cursor()
11    c.execute(query, (username_0920,))
12    penyakit_diri_0920 = pd.DataFrame(c.fetchall(), columns=[desc[0] for desc in c.description])
13    conn.close()
14    return penyakit_diri_0920
```

Fungsi “melihat_penyakit_diri” digunakan untuk mengambil data tentang penyakit yang diderita oleh seorang pasien berdasarkan nama pasien yang diberikan.





```
● ● ●  
1 def memilih_dokter(username_0920, dokter_id):  
2     conn = database()  
3     c = conn.cursor()  
4     c.execute('UPDATE pasien SET dokter_id = ? WHERE nama = ?', (dokter_id, username_0920))  
5     conn.commit()  
6     conn.close()
```

Fungsi “memilih_dokter” digunakan untuk mengupdate data pada tabel “pasien”, berdasarkan nama pasien dan ID dokter yang diberikan.

```
● ● ●  
1 def masukan_pasien_dan_penyakit(nama_pasien_0920, umur_pasien_0920, alamat_pasien_0920, nama_penyakit_0920, tanggal_masuk_0920):  
2     conn = database()  
3     c = conn.cursor()  
4     c.execute('INSERT INTO pasien (nama, umur, alamat) VALUES (?, ?, ?)', (nama_pasien_0920, umur_pasien_0920, alamat_pasien_0920))  
5     pasien_id = c.lastrowid  
6     c.execute('INSERT INTO penyakit (nama, pasien_id, tanggal_masuk) VALUES (?, ?, ?)', (nama_penyakit_0920, pasien_id, tanggal_masuk_0920))  
7     conn.commit()  
8     conn.close()
```

Fungsi “masukan_pasien_dan_penyakit” digunakan untuk memasukkan data seorang pasien beserta informasi penyakit yang dideritanya ke dalam database.





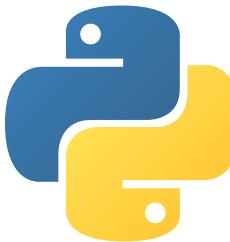
```
● ● ●  
1 def visualisasi_jumlah_penyakit(nama_penyakit):  
2     conn = database()  
3     query = ''  
4     SELECT tanggal_masuk, COUNT(ID_Penyakit) as jumlah_kasus  
5     FROM penyakit  
6     WHERE nama = ?  
7     GROUP BY tanggal_masuk  
8     ORDER BY tanggal_masuk  
9     ''  
10    c = conn.cursor()  
11    c.execute(query, (nama_penyakit,))  
12    data_0920 = pd.DataFrame(c.fetchall(), columns=[desc[0] for desc in c.description])  
13    conn.close()  
14    return data_0920
```

Fungsi “visualisasi_jumlah_penyakit” bertujuan untuk menyusun query SQL untuk mengambil data jumlah kasus penyakit berdasarkan tanggal masuk.

```
● ● ●  
1 def membandingkan_data_penyakit():  
2     conn = database()  
3     query = ''  
4     SELECT p.nama as Penyakit, COUNT(p.ID_Penyakit) as Jumlah_Kasus  
5     FROM penyakit p  
6     GROUP BY p.nama  
7     ''  
8     c = conn.cursor()  
9     c.execute(query)  
10    perbandingan_0920 = pd.DataFrame(c.fetchall(), columns=[desc[0] for desc in c.description])  
11    conn.close()  
12    return perbandingan_0920
```

Fungsi “membandingkan_data_penyakit” bertujuan untuk menyusun query SQL untuk mengambil data jumlah kasus untuk setiap jenis penyakit.





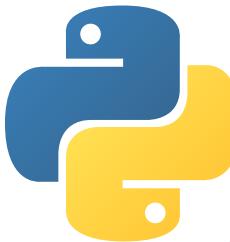
Fungsi “memfilter_jenis_penyakit” digunakan untuk memfilter data penyakit berdasarkan jenis penyakit yang dipilih.

```
1 def memfilter_jenis_penyakit(pilih_penyakit_0920):
2     conn = database()
3     query = ''
4     SELECT p.nama as Penyakit, COUNT(p.ID_Penyakit) as Jumlah_Kasus
5     FROM penyakit p
6     WHERE p.nama = ?
7     GROUP BY p.nama
8     ...
9     c = conn.cursor()
10    c.execute(query, (pilih_penyakit_0920,))
11    hasil_filter_0920 = pd.DataFrame(c.fetchall(), columns=[desc[0] for desc in c.description])
12    conn.close()
13    return hasil_filter_0920
```

```
1 def hapus_dokter(dokter_id_0920):
2     conn = database()
3     c = conn.cursor()
4     c.execute('DELETE FROM dokter WHERE ID_Dokter = ?', (dokter_id_0920,))
5     conn.commit()
6     conn.close()
```

Fungsi “hapus_dokter” digunakan untuk menghapus data seorang dokter dari tabel dokter berdasarkan ID dokter yang diberikan.





```
● ● ●  
1 def hapus_pasien(pasien_id_0920):  
2     conn = database()  
3     c = conn.cursor()  
4     c.execute('DELETE FROM pasien WHERE ID_Pasien = ?', (pasien_id_0920,))  
5     conn.commit()  
6     conn.close()
```

**Fungsi hapus_pasien untuk menghapus database
(pasien_id_0920)**

```
● ● ●  
1 def get_image_path(image_name):  
2     script_dir = os.path.dirname(__file__)  
3     return os.path.join(script_dir, image_name)
```

**Fungsi berikut memiliki tujuan untuk memanipulasi
jalur file dan direktori.**





```
● ● ●
1 @st.cache_data
2 def get_img_as_base64(file):
3     with open(file, "rb") as f:
4         return base64.b64encode(f.read()).decode()
5
6 img = get_img_as_base64(get_image_path("hd_hospital.jpg"))
7
8 page_bg_img = f"""
9 <style>
10 [data-testid="stAppViewContainer"] {{
11     background-image: url("data:image/png;base64,{img}");
12     background-size: cover;
13 }}
14 </style>
15 """
16
17 st.markdown(page_bg_img, unsafe_allow_html=True)
18
19 st.markdown(
20     """
21     <style>
22     .stButton>button {
23         width: 200px; /* Lebar tombol */
24         height: 50px; /* Tinggi tombol */
25         font-size: 18px; /* Ukuran font */
26     }
27     </style>
28     """,
29     unsafe_allow_html=True,
30 )
```

```
● ● ●
1 st.markdown(
2     """
3     <style>
4     [data-testid="stHeader"] {
5         background: transparent;
6     }
7     .main > div:first-child {
8         background: transparent;
9     }
10    </style>
11    """,
12    unsafe_allow_html=True,
13 )
14
15 st.markdown(
16     """
17     <style>
18     [data-testid="stAppViewBlockContainer"] {
19         width: 35%;
20         padding: 15rem 1rem 10rem;
21         max-width: 46rem;
22         box-sizing: border-box;
23     }
24     </style>
25     """,
26     unsafe_allow_html=True,
27 )
28
29 st.markdown(
30     """
31     <style>
32     [data-testid="stMarkdownContainer"] h1 {
33         color: white;
34         font-family: sans-serif;
35         font-size: 20px;
36     }
37     </style>
38     """,
39     unsafe_allow_html=True,
40 )
```

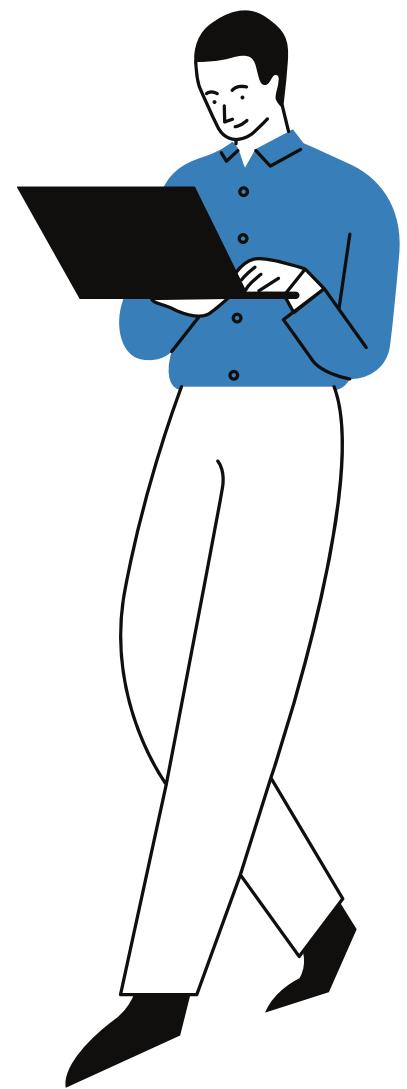
Pada bagian yang menggunakan `st.markdown` merupakan salah satu cara untuk mengubah User Interface (UI) menggunakan CSS dan menerapkannya secara langsung tanpa menggunakan atau membuat file CSS terpisah. Untuk menggunakan fitur tersebut terdapat beberapa hal yang perlu diperhatikan yaitu dengan mencari nama CLASS yang sesuai pada LIBRARY Streamlit.

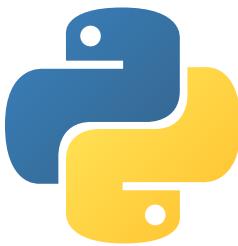




```
● ● ●  
1 if not st.session_state['logged_in_0920']:  
2     st.header("Login")  
3     login_type_0920 = st.selectbox("Login sebagai", ["Admin", "Pasien"])  
4  
5     if login_type_0920 == "Admin":  
6         username_0920 = st.text_input("Username")  
7         password_0920 = st.text_input("Password", type="password")  
8         login_button_0920 = st.button("Login sebagai Admin")  
9  
10    if login_button_0920:  
11        if login_admin(username_0920, password_0920):  
12            st.session_state['logged_in_0920'] = True  
13            st.session_state['role_0920'] = "admin"  
14            st.session_state['username_0920'] = username_0920  
15            st.success("Login berhasil!")  
16        else:  
17            st.error("Username atau password salah")  
18    else:  
19        username_0920 = st.text_input("Nama Pasien")  
20        login_button_0920 = st.button("Login sebagai Pasien")  
21  
22    if login_button_0920:  
23        if login_pasien(username_0920):  
24            st.session_state['logged_in_0920'] = True  
25            st.session_state['role_0920'] = "pasien"  
26            st.session_state['username_0920'] = username_0920  
27            st.success("Login berhasil!")  
28        else:  
29            st.error("Nama pasien tidak ditemukan")
```

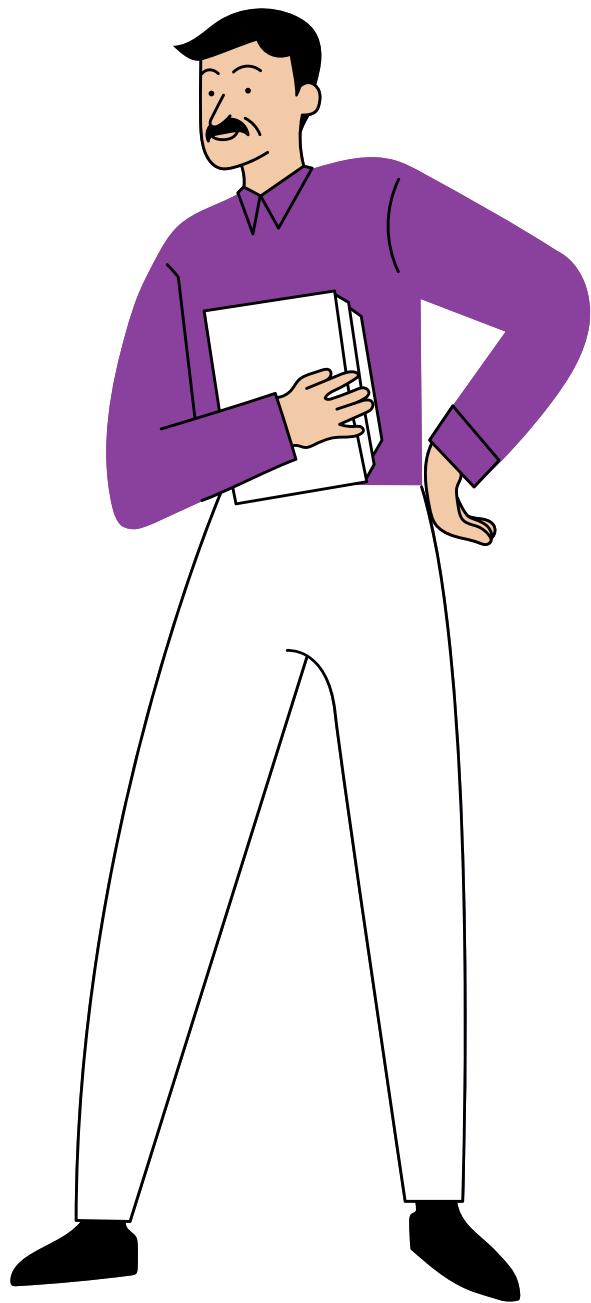
Kode ini berfungsi untuk menampilkan formulir login dan menangani proses login pengguna sebagai admin atau pasien dalam aplikasi Streamlit.





Kode ini mengatur tampilan aplikasi Streamlit dengan memuat file CSS untuk styling dan menampilkan menu navigasi di sidebar. Menu navigasi berbeda tergantung pada peran pengguna (admin atau pasien).

```
1  else:
2      BASE_DIR = os.path.dirname(os.path.abspath(__file__))
3
4      with open(os.path.join(BASE_DIR, 'style.css')) as f:
5          st.markdown(f<style>{f.read()}</style>, unsafe_allow_html=True)
6
7      with st.sidebar:
8          st.sidebar.title("MENU")
9          if st.session_state['role_0920'] == "admin":
10              tabs = on_hover_tabs(tabName=['Dashboard', 'Lihat Dokter', 'Lihat Pasien', 'Cari Dokter', 'Cari Pasien', 'Tambahkan Dokter', 'Tambahkan Data Pasien', 'Hapus Dokter', 'Hapus Pasien', 'Filter Penyakit', 'Bandingkan Penyakit', 'Visualisasi Jumlah Penyakit', 'Logout'],
11                               iconName=['dashboard', 'search', 'search', 'search', 'search', 'add', 'delete', 'filter', 'filter', 'filter', 'logout'], default_choice=0)
12          else:
13              tabs = on_hover_tabs(tabName=['Dashboard', 'Lihat Data Diri', 'Lihat Penyakit Diri', 'Lihat Dokter', 'Pilih Dokter', 'Logout'],
14                               iconName=['dashboard', 'search', 'search', 'search', 'logout'], default_choice=0)
```

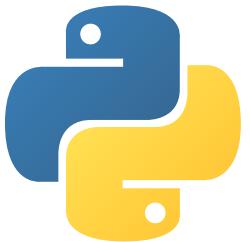




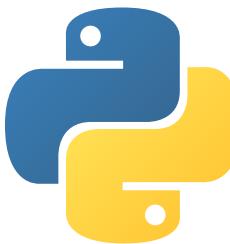
```
● ● ●  
1 if tabs == "Logout":  
2     st.session_state['logged_in_0920'] = False  
3     st.session_state['role_0920'] = None  
4     st.session_state['username_0920'] = None  
5     st.success("Anda telah logout")
```

Kode ini mengatur proses logout pengguna dengan menghapus informasi login dan menampilkan pesan konfirmasi.





```
1 elif tabs == "Dashboard":
2     def apply_style(style):
3         st.markdown(f"<style>{style}</style>", unsafe_allow_html=True)
4         apply_style(
5             """
6                 body, /* or your parent container */
7                 [data-test-id="stappStaffBlockContainer"] {
8                     display: flex;
9                     flex-direction: column;
10                    min-height: 100vh;
11                }
12                [data-test-id="stappVivaldBlockContainer"] {
13                    flex: 1;
14                    width: 100%;
15                    padding: 0 10px 10px 10px;
16                    max-width: 64rem;
17                    box-sizing: border-box;
18                    background: white !important;
19                }
20            """
21        )
22
23     import streamlit.components.v1 as components
24     components.html(
25         """
26             <!DOCTYPE html>
27             <html>
28                 <head>
29                     <meta name="viewport" content="width=device-width, initial-scale=1">
30
31                     <!-- Box-Sizing: border-box -->
32                     <body font-family: Verdana, sans-serif;>
33                         <mySlides> (display: none);
34                         
35
36                     <div class="slideshow-container">
37                         <div class="mySlides fade">
38                             
39                         <div class="mySlides fade">
40                             
41                         </div>
42                         <br>
43                         <div style="text-align:center">
44                             <span class="dot"></span>
45                             <span class="dot"></span>
46                             <span class="dot"></span>
47                         </div>
48                         <script>
49                             slideIndex = 0;
50                             showSlides();
51
52                             function showSlides() {
53                                 let i;
54                                 let slides = document.getElementsByClassName("mySlides");
55                                 let dots = document.getElementsByTagName("span");
56                                 for (i = 0; i < slides.length; i++) {
57                                     dots[i].style.display = "none";
58                                     slides[i].style.display = "none";
59                                 }
60                                 if (slideIndex > slides.length) {slideIndex = 1}
61                                 if (slideIndex < 1) {slideIndex = slides.length}
62                                 dots[slideIndex-1].style.display = "block";
63                                 dots[slideIndex-1].classList.add("active");
64                                 setTimeout(showSlides, 4000); // Change image every 2 seconds
65                             }
66                         </script>
67                     </div>
68                 </body>
69             </html>
70             <!--
71                 height:600;
72             </-->
73         """
74     )
75
76     apply_style(
77         """
78             <div class="main">
79                 <stappStaffBlockContainer>
80                     <myHomePage/>
81                 </stappStaffBlockContainer>
82             </div>
83
84             <div class="slideshow-container">
85                 <div class="mySlides fade">
86                     
87                 </div>
88                 <div class="mySlides fade">
89                     
90                 </div>
91                 <div class="mySlides fade">
92                     
93                 </div>
94                 <br>
95                 <div style="text-align:center">
96                     <span class="dot"></span>
97                     <span class="dot"></span>
98                     <span class="dot"></span>
99                 </div>
100                 <script>
101                     slideIndex = 0;
102                     showSlides();
103
104                     function showSlides() {
105                         let i;
106                         let slides = document.getElementsByClassName("mySlides");
107                         let dots = document.getElementsByTagName("span");
108                         for (i = 0; i < slides.length; i++) {
109                             dots[i].style.display = "none";
110                             slides[i].style.display = "none";
111                         }
112                         if (slideIndex > slides.length) {slideIndex = 1}
113                         if (slideIndex < 1) {slideIndex = slides.length}
114                         dots[slideIndex-1].style.display = "block";
115                         dots[slideIndex-1].classList.add("active");
116                         setTimeout(showSlides, 4000); // Change image every 2 seconds
117                     }
118                 </script>
119             </div>
120         """
121     )
122
123     apply_style(
124         """
125             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
126                 <h1>HomePage</h1>
127             </div>
128         """
129     )
130
131     apply_style(
132         """
133             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
134                 <h1>HomePage</h1>
135             </div>
136         """
137     )
138
139     apply_style(
140         """
141             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
142                 <h1>HomePage</h1>
143             </div>
144         """
145     )
146
147     apply_style(
148         """
149             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
150                 <h1>HomePage</h1>
151             </div>
152         """
153     )
154
155     apply_style(
156         """
157             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
158                 <h1>HomePage</h1>
159             </div>
160         """
161     )
162
163     apply_style(
164         """
165             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
166                 <h1>HomePage</h1>
167             </div>
168         """
169     )
170
171     apply_style(
172         """
173             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
174                 <h1>HomePage</h1>
175             </div>
176         """
177     )
178
179     apply_style(
180         """
181             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
182                 <h1>HomePage</h1>
183             </div>
184         """
185     )
186
187     apply_style(
188         """
189             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
190                 <h1>HomePage</h1>
191             </div>
192         """
193     )
194
195     apply_style(
196         """
197             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
198                 <h1>HomePage</h1>
199             </div>
200         """
201     )
202
203     apply_style(
204         """
205             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
206                 <h1>HomePage</h1>
207             </div>
208         """
209     )
210
211     apply_style(
212         """
213             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
214                 <h1>HomePage</h1>
215             </div>
216         """
217     )
218
219     apply_style(
220         """
221             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
222                 <h1>HomePage</h1>
223             </div>
224         """
225     )
226
227     apply_style(
228         """
229             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
230                 <h1>HomePage</h1>
231             </div>
232         """
233     )
234
235     apply_style(
236         """
237             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
238                 <h1>HomePage</h1>
239             </div>
240         """
241     )
242
243     apply_style(
244         """
245             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
246                 <h1>HomePage</h1>
247             </div>
248         """
249     )
250
251     apply_style(
252         """
253             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
254                 <h1>HomePage</h1>
255             </div>
256         """
257     )
258
259     apply_style(
260         """
261             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
262                 <h1>HomePage</h1>
263             </div>
264         """
265     )
266
267     apply_style(
268         """
269             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
270                 <h1>HomePage</h1>
271             </div>
272         """
273     )
274
275     apply_style(
276         """
277             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
278                 <h1>HomePage</h1>
279             </div>
280         """
281     )
282
283     apply_style(
284         """
285             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
286                 <h1>HomePage</h1>
287             </div>
288         """
289     )
290
291     apply_style(
292         """
293             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
294                 <h1>HomePage</h1>
295             </div>
296         """
297     )
298
299     apply_style(
299         """
300             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
301                 <h1>HomePage</h1>
302             </div>
303         """
304     )
305
306     apply_style(
306         """
307             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
308                 <h1>HomePage</h1>
309             </div>
310         """
311     )
312
313     apply_style(
313         """
314             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
315                 <h1>HomePage</h1>
316             </div>
317         """
318     )
319
320     apply_style(
320         """
321             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
322                 <h1>HomePage</h1>
323             </div>
324         """
325     )
326
327     apply_style(
327         """
328             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
329                 <h1>HomePage</h1>
330             </div>
331         """
332     )
333
334     apply_style(
334         """
335             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
336                 <h1>HomePage</h1>
337             </div>
338         """
339     )
340
341     apply_style(
341         """
342             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
343                 <h1>HomePage</h1>
344             </div>
345         """
346     )
347
348     apply_style(
348         """
349             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
350                 <h1>HomePage</h1>
351             </div>
352         """
353     )
354
355     apply_style(
355         """
356             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
357                 <h1>HomePage</h1>
358             </div>
359         """
360     )
361
362     apply_style(
362         """
363             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
364                 <h1>HomePage</h1>
365             </div>
366         """
367     )
368
369     apply_style(
369         """
370             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
371                 <h1>HomePage</h1>
372             </div>
373         """
374     )
375
376     apply_style(
376         """
377             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
378                 <h1>HomePage</h1>
379             </div>
380         """
381     )
382
383     apply_style(
383         """
384             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
385                 <h1>HomePage</h1>
386             </div>
387         """
388     )
389
390     apply_style(
390         """
391             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
392                 <h1>HomePage</h1>
393             </div>
394         """
395     )
396
397     apply_style(
397         """
398             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
399                 <h1>HomePage</h1>
400             </div>
401         """
402     )
403
404     apply_style(
404         """
405             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
406                 <h1>HomePage</h1>
407             </div>
408         """
409     )
410
411     apply_style(
411         """
412             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
413                 <h1>HomePage</h1>
414             </div>
415         """
416     )
417
418     apply_style(
418         """
419             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
420                 <h1>HomePage</h1>
421             </div>
422         """
423     )
424
425     apply_style(
425         """
426             <div style="background-color: white; color: black; padding: 10px; border-radius: 10px; border: 1px solid #ccc; width: fit-content; margin: auto; text-align: center; font-size: 1.2em; font-weight: bold; letter-spacing: 0.1em; font-family: sans-serif;">
427                 <h1&gt
```



```
1 elif tabs == "Lihat Dokter":  
2     dokter_0920 = melihat_data_dokter()  
3     st.dataframe(dokter_0920)
```



```
1 elif tabs == "Lihat Pasien":  
2     if st.session_state['role_0920'] == "admin":  
3         pasien_0920 = melihat_data_pasien()  
4         st.dataframe(pasien_0920)  
5     else:  
6         st.error("Anda tidak memiliki akses untuk fitur ini.")
```

Kode ini berfungsi untuk menampilkan data dokter dalam aplikasi Streamlit. Ketika pengguna memilih tab "Lihat Dokter", fungsi `melihat_data_dokter()` dipanggil untuk mengambil data dokter dari sumber data (misalnya, database atau file). Data dokter yang diperoleh kemudian ditampilkan dalam bentuk tabel menggunakan fungsi `st.dataframe()`.

Kode ini menampilkan data pasien dalam bentuk tabel jika pengguna adalah admin. Jika pengguna bukan admin, akan muncul pesan error yang menyatakan bahwa pengguna tidak memiliki akses.

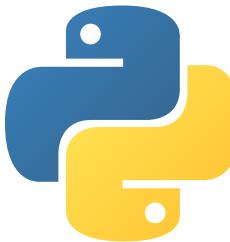


```
● ● ●  
1 elif tabs == "Cari Dokter":  
2     nama_dokter_0920 = st.text_input("Masukkan nama dokter yang ingin dicari:")  
3     if nama_dokter_0920:  
4         hasil_cari_dokter_0920 = cari_dokter(nama_dokter_0920)  
5         st.dataframe(hasil_cari_dokter_0920)
```

Kode ini memungkinkan pengguna mencari data dokter berdasarkan nama, kode akan memanggil fungsi `cari_dokter` dan akan menampilkan hasilnya dalam bentuk tabel.

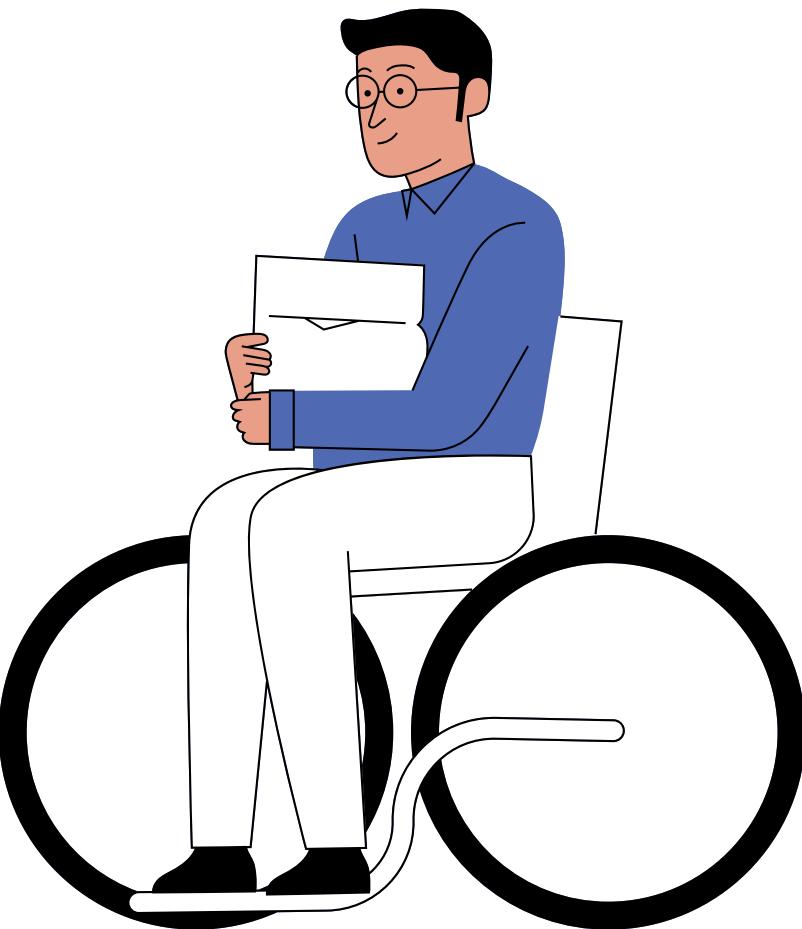
```
● ● ●  
1 elif tabs == "Cari Pasien":  
2     nama_pasien_0920 = st.text_input("Masukkan nama pasien yang ingin dicari:")  
3     if nama_pasien_0920:  
4         hasil_cari_pasien_0920 = cari_pasien(nama_pasien_0920)  
5         st.dataframe(hasil_cari_pasien_0920)
```

Kode ini memungkinkan pencarian data pasien berdasarkan nama , kode akan memanggil fungsi `cari_pasien` dan akan menampilkan hasilnya dalam tabel.



```
● ● ●  
1 elif tabs == "Tambahkan Dokter":  
2     if st.session_state['role_0920'] == "admin":  
3         with st.form("dokter_form_0920", clear_on_submit=True):  
4             nama_dokter_0920 = st.text_input("Nama Dokter")  
5             spesialisasi_dokter_0920 = st.text_input("Spesialisasi Dokter")  
6             tambahkan_0920 = st.form_submit_button("Tambahkan")  
7  
8             if tambahkan_0920:  
9                 masukan_dokter(nama_dokter_0920, spesialisasi_dokter_0920)  
10                st.success("Dokter berhasil ditambahkan!")  
11                st.dataframe(melihat_data_dokter())
```

Kode ini memungkinkan admin untuk menambahkan data dokter baru ke dalam sistem melalui formulir dan menampilkan tabel data dokter yang telah diperbarui.

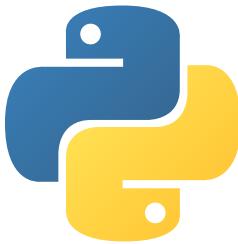




```
● ● ●

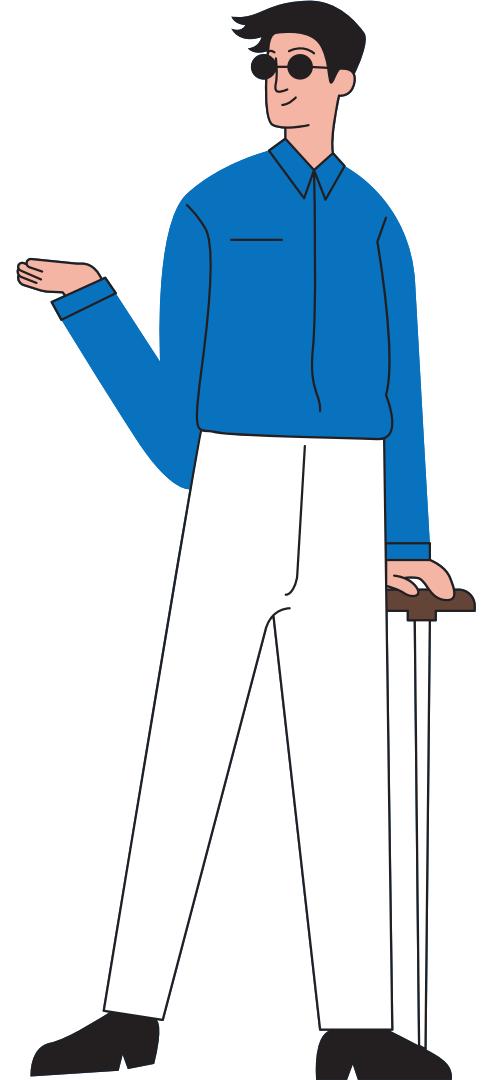
1 elif tabs == "Tambahkan Data Pasien":
2     if st.session_state['role_0920'] == "admin":
3         with st.form("pasien_penyakit_form_0920", clear_on_submit=True):
4             nama_pasien_0920 = st.text_input("Nama Pasien")
5             umur_pasien_0920 = st.number_input("Umur Pasien", min_value=0)
6             alamat_pasien_0920 = st.text_input("Alamat Pasien")
7             nama_penyakit_0920 = st.text_input("Nama Penyakit")
8             deskripsi_penyakit_0920 = st.text_area("Deskripsi Penyakit")
9             tanggal_masuk_0920 = st.date_input("Tanggal Masuk")
10            tambahkan_0920 = st.form_submit_button("Tambahkan")
11
12            if tambahkan_0920:
13                masukan_pasien_dan_penyakit(nama_pasien_0920, umur_pasien_0920, alamat_pasien_0920, nama_penyakit_0920, tanggal_masuk_0920)
14                st.success("Pasien dan Penyakit berhasil ditambahkan!")
15                st.dataframe(melihat_data_pasien())
```

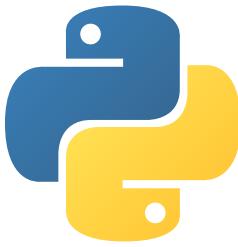
kode ini memungkinkan admin untuk memasukkan data pasien baru beserta informasi penyakitnya melalui formulir, menyimpannya ke dalam sistem, dan menampilkan tabel data pasien yang telah diperbarui.



```
● ● ●  
1 elif tabs == "Hapus Dokter":  
2     if st.session_state['role_0920'] == "admin":  
3         dokter_id_0920 = st.number_input("Masukkan ID Dokter yang Ingin Dihapus", min_value=1)  
4         hapus_button_0920 = st.button("Hapus Dokter")  
5  
6     if hapus_button_0920:  
7         hapus_dokter(dokter_id_0920)  
8         st.success("Dokter berhasil dihapus!")  
9         st.dataframe(melihat_data_dokter())
```

Kode ini memungkinkan admin untuk menghapus data dokter berdasarkan ID yang dimasukkan, kemudian menampilkan tabel data dokter yang sudah diperbarui.

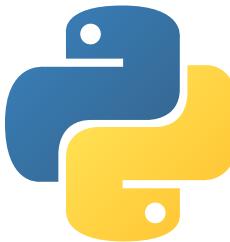




```
● ● ●  
1 elif tabs == "Hapus Pasien":  
2     if st.session_state['role_0920'] == "admin":  
3         pasien_id_0920 = st.number_input("Masukkan ID Pasien yang Ingin Dihapus", min_value=1)  
4         hapus_button_0920 = st.button("Hapus Pasien")  
5  
6     if hapus_button_0920:  
7         hapus_pasien(pasien_id_0920)  
8         st.success("Pasien berhasil dihapus!")  
9         st.dataframe(melihat_data_pasien())
```

Kode ini memungkinkan admin untuk menghapus data pasien berdasarkan ID yang dimasukkan. Setelah admin memasukkan ID dan menekan tombol "Hapus Pasien", sistem akan menghapus data pasien tersebut dan menampilkan pesan sukses serta tabel data pasien yang telah diperbarui.

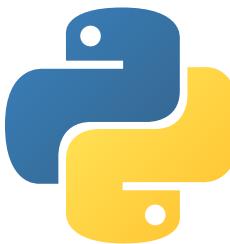




```
● ● ●  
1 elif tabs == "Filter Penyakit":  
2     pilih_penyakit_0920 = st.text_input("Masukkan nama penyakit yang ingin difilter:")  
3     hasil_filter_0920 = memfilter_jenis_penyakit(pilih_penyakit_0920)  
4     st.dataframe(hasil_filter_0920)
```

Kode ini memungkinkan pengguna untuk memfilter data pasien berdasarkan nama penyakit. Setelah pengguna memasukkan nama penyakit, fungsi memfilter_jenis_penyakit akan dipanggil untuk menyaring data pasien yang memiliki penyakit tersebut. Hasil filter kemudian ditampilkan dalam bentuk tabel.





```
1 elif tabs == "Bandingkan Penyakit":  
2     perbandingan_0920 = membandingkan_data_penyakit()  
3     st.dataframe(perbandingan_0920)  
4     plt.figure(figsize=(10, 5))  
5     plt.bar(perbandingan_0920['Penyakit'], perbandingan_0920['Jumlah_Kasus'])  
6     plt.xlabel('Nama Penyakit')  
7     plt.ylabel('Jumlah Kasus')  
8     plt.title('Perbandingan Jumlah Kasus Penyakit')  
9     st.pyplot(plt)
```

Kode ini menampilkan perbandingan jumlah kasus penyakit dan ditampilkan dalam bentuk tabel menggunakan `st.dataframe` dan grafik batang menggunakan library `st.pyplot`.

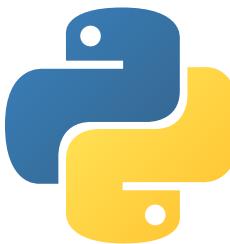
```
1 elif tabs == "Visualisasi Jumlah Penyakit":  
2     penyakit_0920 = st.text_input("Masukkan Nama Penyakit untuk Visualisasi")  
3     if penyakit_0920:  
4         data_0920 = visualisasi_jumlah_penyakit(penyakit_0920)  
5         if not data_0920.empty:  
6             plt.figure(figsize=(10, 5))  
7             plt.plot(data_0920['tanggal_masuk'], data_0920['jumlah_kasus'], marker='o')  
8             plt.xlabel('Tanggal Masuk')  
9             plt.ylabel('Jumlah Kasus')  
10            plt.title(f'Jumlah Kasus {penyakit_0920} per Tanggal')  
11            st.pyplot(plt)  
12        else:  
13            st.warning("Tidak ada data untuk penyakit tersebut.")  
14
```

Kode ini memvisualisasikan jumlah kasus penyakit tertentu dalam bentuk grafik garis berdasarkan tanggal masuk dan jika tidak ada data yang ditemukan, sistem akan menampilkan pesan peringatan.



```
● ● ●  
1 elif tabs == 'Pilih Dokter':  
2     if st.session_state['role_0920'] == 'pasien':  
3         username_0920 = st.session_state['username_0920']  
4         data_diri_0920 = melihat_data_diri(username_0920)  
5         if not data_diri_0920.empty:  
6             pasien_id_0920 = data_diri_0920['ID_Pasien'].values[0]  
7             dokter_0920 = melihat_data_dokter()  
8             dokter_list_0920 = dokter_0920.set_index('ID_Dokter')['nama'].to_dict()  
9             dokter_id_0920 = st.selectbox('Pilih Dokter', options=list(dokter_list_0920.keys()), format_func=lambda x: dokter_list_0920[x])  
10            pilih_dokter_0920 = st.button('Pilih Dokter')  
11  
12            if pilih_dokter_0920:  
13                memilih_dokter(username_0920, dokter_id_0920)  
14                st.success(f'Dokter {dokter_list_0920[dokter_id_0920]} berhasil dipilih!')  
15            else:  
16                st.warning('Data diri pasien tidak ditemukan.')  
17
```

Kode ini menampilkan bagian dari aplikasi berbasis Streamlit yang memungkinkan pasien untuk memilih dokter. pasien dapat memilih dokter dari daftar dokter yang tersedia dan sistem akan menyimpan pilihan mereka.



```
● ● ●  
1 elif tabs == "Lihat Data Diri":  
2     if st.session_state['role_0920'] == "pasien":  
3         username_0920 = st.session_state['username_0920']  
4         data_diri_0920 = melihat_data_diri(username_0920)  
5         st.dataframe(data_diri_0920)
```

Kode ini memungkinkan pasien untuk melihat data diri mereka sendiri, sistem akan mengambil data diri pasien tersebut berdasarkan username yang tersimpan di `st.session_state` dan menampilkannya dalam bentuk tabel.

```
● ● ●  
1 elif tabs == "Lihat Penyakit Diri":  
2     if st.session_state['role_0920'] == "pasien":  
3         username_0920 = st.session_state['username_0920']  
4         penyakit_diri_0920 = melihat_penyakit_diri(username_0920)  
5         st.dataframe(penyakit_diri_0920)
```

Kode ini memungkinkan pasien untuk melihat daftar penyakit yang pernah dideritanya, sistem akan mengambil data penyakit pasien tersebut berdasarkan username yang tersimpan di `st.session_state` dan menampilkannya dalam bentuk tabel.



KAMI MENGUCAPKAN

TERIMA KASIH



