

**KERANGKA PORTOFOLIO KOMPETENSI PADA
PROGRAM KLINIK GIGI**

**Disusun Oleh
ABDULLAH AZZAM RABBANI
10240038**



**PROGRAM STUDI REKAYASA PERANGKAT LUNAK
UNIVERSITAS BINA SARANA INFORMATIKA
MARGONDA
2024/2025**

KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas rahmat dan karunia-Nya, sehingga laporan portofolio ini dapat diselesaikan tepat waktu. Laporan ini disusun sebagai bagian integral dari pemenuhan tugas Portofolio Sertifikasi Kompetensi untuk Unit Kompetensi Mengimplementasikan Pemrograman Terstruktur (Klinik Gigi).

Laporan ini berfokus pada analisis mendalam terhadap sebuah proyek yang telah dikembangkan sebelumnya, yaitu Program Sistem Informasi Klinik Gigi yang dibangun menggunakan bahasa pemrograman Python. Pemilihan studi kasus ini bertujuan untuk mendemonstrasikan penguasaan konsep pemrograman terstruktur, mulai dari penentuan tipe data, penggunaan operator, hingga implementasi struktur kontrol percabangan (selection), perulangan (iteration), serta prinsip modularitas melalui fungsi.

Penulis menyadari bahwa laporan ini masih memiliki keterbatasan. Untuk itu, kritik dan saran yang konstruktif sangat diharapkan demi perbaikan di masa mendatang. Semoga laporan ini dapat memberikan kontribusi nyata dalam pengembangan pemahaman mengenai implementasi pemrograman terstruktur.

Jakarta, 28 September 2025

penyusun

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Tujuan Analisis	1
BAB II ANALISIS STUDI KASUS.....	2
2.1. Deskripsi Proyek	2
2.2. Input dan Output	2
BAB III IMPLEMENTASI KONSEP PEMROGRAMAN TERSTRUKTUR	2
3.1 Aturan Penamaan Variabel dan Tipe Data.....	2
3.2. Operator yang Digunakan	4
3.3. Struktur Kontrol Percabangan (Selection)	5
3.4. Struktur Kontrol Perulangan (Iteration/Looping)	6
3.5 Penggunaan Fungsi (Function / Prosedur)	6
3.6 Array (Struktur Data Koleksi).....	7
3.7. Library.....	7
BAB IV DOKUMENTASI SCRIPT.....	8
4.1. Dokumentasi Struktur Data dan Inisialisasi.....	8
4.2 Dokumentasi Fungsi Pembantu dan Perhitungan	9
4.3 Dokumentasi Fungsi Alur Kontrol dan Perulangan	9
BAB V PENUTUP.....	11
5.1. Kesimpulan	11

BAB I

PENDAHULUAN

1.1. Latar Belakang

- Jelaskan pentingnya Pemrograman Terstruktur (Prosedural) dalam pengembangan software (kemudahan maintenance, keterbacaan kode, dan efisiensi).
- Sebutkan tujuan dibuatnya laporan ini, yaitu mendemonstrasikan Unit Kompetensi "Mengimplementasikan Pemrograman Terstruktur" melalui analisis kode proyek.
- Perkenalkan studi kasus: Program Sistem Informasi Klinik Gigi sebagai objek analisis.

1.2. Tujuan Analisis

- Menganalisis penggunaan elemen-elemen Pemrograman Terstruktur (variabel, operator, percabangan, perulangan, function, array) dalam kode program Klinik Gigi.
- Menyusun dokumentasi script yang terperinci.

BAB II

ANALISIS STUDI KASUS

2.1. Deskripsi Proyek

- **Nama Proyek:** Program Sistem Informasi Klinik Gigi.
- **Fungsi Utama:** Melakukan pencatatan layanan gigi, menghitung biaya, menerapkan diskon, dan mencetak struk.
- **Bahasa Pemrograman:** Python.

2.2. Input dan Output

Input Program:

- Input pilihan menu (1, 2, 3).
- Input jumlah layanan yang akan di beli.
- Input nomer urut layanan.

Output Program:

- Daftar layanan dan harga.
- Pesan Peringatan (jika input salah).
- **Struk Transaksi** yang memuat detail pembelian, diskon, dan total bayar.

BAB III

IMPLEMENTASI KONSEP PEMROGRAMAN TERSTRUKTUR

Implementasi pemrograman terstruktur diwujudkan dengan membagi program menjadi fungsi-fungsi modular (prosedur) dan mengatur alur eksekusi menggunakan struktur kontrol dasar: sekuensial (urutan), percabangan (seleksi), dan perulangan (iterasi).

3.1 Aturan Penamaan Variabel dan Tipe Data

Ketentuan penamaan variabel dalam studi kasus ini umumnya mengikuti gaya penulisan **snake_case** untuk variabel lokal (seperti `harga_final`, `total_pembelian`), namun juga menggunakan **camelCase** untuk beberapa atribut kelas (seperti `banyakJenisLayanan`, `statusBeli`).

A. Ketentuan Nama Variabel

- 1) Atribut Kelas (Internal Data): Menggunakan awalan self. untuk membedakannya sebagai properti milik objek, seperti self.layanan dan self.diskon.
- 2) Variabel Lokal: Menggunakan nama yang deskriptif dan tidak boleh menggunakan kata kunci Python (contoh: if, while, for).

B. Implementasi Tipe Data

Tabel berikut menguraikan tipe data yang digunakan dalam program, menunjukkan bagaimana struktur data primitif dan terstruktur dimanfaatkan untuk mengelola informasi layanan dan transaksi.

Tipe Data (Python)	Variabel/Atribut	Contoh Nilai	Fungsi dalam Program
Dictionary (dict)	self.layanan	{"Pembersihan Gigi": 50000, ...}	Struktur data terstruktur untuk menyimpan daftar layanan dan harganya . Memungkinkan akses data cepat menggunakan nama layanan sebagai kunci.
Dictionary (dict)	self.diskon	{"Pembersihan Gigi": 0.1, ...}	Struktur data terstruktur untuk menyimpan persentase diskon yang berlaku untuk layanan tertentu.
List of Dictionaries	self.pembelian	[{"layanan": "...", "biaya": 150000, ...}, ...]	Struktur data terstruktur multidimensi yang bertindak sebagai array transaksi. Menyimpan detail lengkap setiap layanan yang dipilih oleh pengguna.
List(list)	total_pembelian	[50000, 120000, 75000]	Struktur data koleksi untuk menampung harga final setiap item yang dibeli, digunakan untuk perhitungan total akhir.
Integer(int)	self.banyakJenisLayanan	2	Struktur data koleksi untuk menampung harga final setiap item

			yang dibeli, digunakan untuk perhitungan total akhir.
Interger(int)	pilihan	3	Menyimpan input numerik bilangan bulat untuk navigasi menu atau pemilihan layanan .
Float (float)	self.diskon["Pembersihan Gigi"]	0.1	Nilai numerik desimal, digunakan untuk menyimpan persentase diskon sebelum dikalikan dengan harga.
Boolean (bool)	self.statusBeli	True atau False	Tipe data logis, digunakan sebagai <i>flag</i> untuk menandai apakah pengguna sudah melakukan transaksi sebelum mencetak struk. Ekspor ke Spreadsheet

3.2. Operator yang Digunakan

Program menggunakan berbagai jenis operator untuk pemrosesan data, perhitungan, dan pengambilan keputusan logis.

Jenis Operator	Simbol	Contoh Kode	Fungsi
penugasan	=	self.statusBeli = True	Memberikan nilai dari sisi kanan ke variabel di sisi kiri.
Aritmatika	+, -, *	i = i + 1, harga - diskon, diskon * 100	Melakukan operasi hitungan untuk menaikkan nilai penghitung (i), menghitung harga setelah diskon, dan

			mengubah desimal menjadi persentase.
Perbandingan	<=, ==, <	if pilihan == 1; if len(y) <= 3:	Membandingkan dua nilai untuk menghasilkan nilai Boolean (True atau False).
Keanggotaan	in	if layanan in self.diskon:	Memeriksa apakah suatu elemen (kunci layanan) ada di dalam suatu koleksi data (self.diskon).

3.3. Struktur Kontrol Percabangan (Selection)

Percabangan digunakan untuk mengarahkan alur program agar dapat mengambil keputusan berdasarkan suatu kondisi.

A. Percabangan if-elif-else untuk Navigasi Menu

Percabangan ini terdapat dalam fungsi main() dan menjadi inti dari interaksi pengguna.

```
if pilihan == 1:
    self.tampilkan_layanan()
elif pilihan == 2:
    self.pilih_layanan()
elif pilihan == 3:
    # Percabangan bersarang untuk validasi status
    if self.statusBeli:
        self.struk()
        break
    else:
        print("Anda Belum Melakukan Pemilihan Menu")
else:
    print("Pilihan tidak valid. Silakan coba lagi.")
```

B. Percabangan untuk Perhitungan Diskon

Dalam fungsi hitung_diskon(), percabangan digunakan untuk menentukan apakah suatu layanan memiliki diskon.

```
def hitung_diskon(self, layanan):
    if layanan in self.diskon: # KONDISI: Cek keberadaan layanan di daftar diskon
        return self.diskon[layanan]
    else:
        return 0 # JIKA FALSE: Kembalikan nilai 0 (tidak ada diskon)
```


3.4. Struktur Kontrol Perulangan (Iteration/Looping)

Perulangan digunakan untuk menjalankan blok kode yang sama secara berulang.

A. Perulangan while untuk Input Berulang

Perulangan while dalam fungsi pilih_layanan() digunakan untuk memastikan pengguna memasukkan layanan sejumlah yang diminta. Perulangan akan terus berjalan selama kondisi `i < self.banyakJenisLayanan` bernilai True.

```
i = 0
# Perulangan dimulai
while i < self.banyakJenisLayanan:
    pilihan = int(input(f"Masukkan nomor layanan ke {i+1}: "))
    # ... kode pemrosesan layanan
    i = i + 1 # Increment counter untuk mengontrol pengulangan
```

B. Perulangan for untuk Iterasi Data

Perulangan for digunakan untuk mengakses setiap elemen dalam koleksi data secara berurutan, seperti saat menampilkan daftar menu dan mencetak struk.

```
# Perulangan untuk menampilkan daftar layanan
for layanan, harga in self.layanan.items():
    print(f"{layanan}: Rp {harga}")

# Perulangan untuk mencetak detail struk
for i, pembelian in enumerate(self.pembelian):
    # ... kode pencetakan detail
```

3.5 Penggunaan Fungsi (Function / Prosedur)

Program Klinik Gigi adalah contoh pemrograman prosedural karena seluruh flow logika dipecah menjadi fungsi-fungsi spesifik yang saling memanggil (modularitas).

Nama Fungsi	Tipe	Tujuan Implementasi
main(self)	Prosedur/Kontrol Utama	Mengatur alur eksekusi, menampilkan menu, dan memanggil fungsi lain.
tampilkan_layanan(self)	Prosedur	Bertanggung jawab tunggal untuk menampilkan semua data layanan.

hitung_biaya(self, layanan)	Fungsi	Mengembalikan nilai harga layanan dari koleksi data, dipanggil oleh fungsi lain.
formatrupiah(self, uang)	Fungsi (Rekursif)	Fungsi Khusus untuk memformat output numerik menjadi format mata uang yang mudah dibaca.
struk(self)	Prosedur	Bertanggung jawab tunggal untuk menghitung total dan menampilkan output laporan (struk).

3.6 Array (Struktur Data Koleksi)

Struktur data koleksi digunakan untuk mengorganisasi data secara efisien. Dalam Python, *array* diimplementasikan dengan list dan dictionary.

Struktur Data	Implementasi	Fungsi dalam Program
Dictionary (Map)	self.layanan	Menyimpan daftar layanan yang bersifat statis dengan akses berdasarkan nama (kunci).
List of Dictionaries	self.pembelian	Menyimpan semua detail transaksi yang dilakukan pengguna, termasuk layanan, biaya awal, diskon, dan harga final.
List	total_pembelian	Digunakan sebagai penampung sementara untuk menyimpan semua Harga Final yang kemudian akan dijumlahkan untuk mendapatkan total bayar.

3.7. Library

Program ini tidak menggunakan *library* atau *module* eksternal. Semua fungsionalitas diimplementasikan menggunakan **fitur bawaan (built-in)** dari bahasa pemrograman Python, seperti fungsi input(), print(), dan struktur data dasar.

BAB IV

DOKUMENTASI SCRIPT

Dokumentasi script adalah langkah penting dalam pemrograman terstruktur untuk meningkatkan keterbacaan kode (readability) dan memudahkan pemeliharaan (maintenance). Dalam program Klinik Gigi ini, dokumentasi dilakukan dengan menambahkan komentar (#) pada setiap fungsi, variabel penting, dan blok struktur kontrol.

4.1. Dokumentasi Struktur Data dan Inisialisasi

Bagian ini menunjukkan inisialisasi kelas dan struktur data utama yang digunakan.

```
class KlinikGigi:
    def __init__(self):
        # Inisialisasi dictionary untuk daftar layanan dan harga
        (Rp)
        self.layanan = {
            "Pembersihan Gigi": 50000,
            "Pengobatan Gigi": 100000,
            "Pengobatan Gigi Berlubang": 150000,
            "Pengobatan Gigi Patah": 200000,
            "Pengobatan Gigi Rusak": 250000
        }

        # Inisialisasi dictionary untuk persentase diskon (contoh:
        0.1 = 10%)
        self.diskon = {
            "Pembersihan Gigi": 0.1,
            "Pengobatan Gigi Berlubang": 0.2,
            "Pengobatan Gigi Rusak": 0.5
        }

        # Inisialisasi list (array) untuk menyimpan detail
        pembelian
        self.pembelian = []

        # Variabel untuk menampung jumlah jenis layanan yang
        dipilih
        self.banyakJenisLayanan = 0

        # Flag Boolean untuk mengecek apakah user sudah melakukan
        pembelian
        self.statusBeli = False
```

4.2 Dokumentasi Fungsi Pembantu dan Perhitungan

Fungsi formatrupiah:

Ini adalah fungsi rekursif untuk memformat angka menjadi format mata uang.

```
def formatrupiah(self, uang):
    # Fungsi rekursif untuk memformat integer menjadi format mata
    uang Rupiah
    y = str(uang)
    if len(y) <= 3:
        # Base Case: jika sisa angka hanya 3 digit atau kurang
        return 'Rp ' + y
    else:
        # Recursive Step: memisahkan 3 digit terakhir dan
        memanggil fungsi kembali
        p = y[-3:] # Ambil 3 digit terakhir
        q = y[:-3] # Ambil sisa digit
        return self.formatrupiah(q) + '.' + p

def hitung_biaya(self, layanan):
    # Fungsi untuk mendapatkan biaya layanan dari dictionary
    self.layanan
    return self.layanan[layanan]

def hitung_diskon(self, layanan):
    # Struktur Percabangan (Selection): Cek apakah layanan ada di
    daftar diskon
    if layanan in self.diskon:
        return self.diskon[layanan] # Kembalikan persentase
    diskon
    else:
        return 0 # Jika tidak ada, kembalikan 0
```

4.3 Dokumentasi Fungsi Alur Kontrol dan Perulangan

Fungsi pilih_layanan

Fungsi ini mengelola proses input layanan menggunakan perulangan **while** dan percabangan.

```
def pilih_layanan(self):
    print("Pilih Layanan:")
    # Perulangan FOR untuk menampilkan opsi layanan
    for i, layanan in enumerate(self.layanan.keys()):
        print(f"{i + 1}. {layanan}")

    # Input jumlah jenis layanan (penentu batas perulangan WHILE)
    self.banyakJenisLayanan = int(input("Masukan banyak layanan :
    "))
    i = 0

    # Percabangan (Selection) untuk validasi input
    if self.banyakJenisLayanan <= 0:
```

```

        print("Data Tidak Ada")
    else:
        # Perulangan WHILE: Berjalan selama counter (i) kurang
        dari total jenis layanan
        while i < self.banyakJenisLayanan:
            pilihan = int(input(f"Masukkan nomor layanan ke {i +
1}: "))

            # Percabangan (Selection) untuk validasi nomor pilihan
            if pilihan <= len(self.layanan.keys()):
                layanan = list(self.layanan.keys())[pilihan - 1]

                # Menambahkan data layanan ke Array/List
                self.pembelian.append({
                    "layanan": layanan,
                    "biaya": self.hitung_biaya(layanan),
                    "diskon": self.hitung_diskon(layanan),
                    # Menggunakan Operator Aritmatika untuk
                    perhitungan harga final
                    "hargaFinal": int(self.hitung_biaya(layanan) -
self.hitung_diskon(layanan) * 100),
                })
                self.statusBeli = True # Set flag status
                pembelian
                i = i + 1 # Increment counter
            else:
                print("Masukan Pilihan Dengan Benar!!,,")
                i = 0 # Reset counter jika input salah (memaksa
input ulang)

```

Fungsi struk

Fungsi ini bertanggung jawab untuk mencetak laporan akhir transaksi.

```

def struk(self):
    print("\nSedang Mencetak Struk ..... \n")
    print("=" * 72)
    print(" " * 14, "Klinik Gigi", " " * 20)
    print("=" * 72)

    total_pembelian = [] # Array/List sementara untuk menampung
    harga final

    # Perulangan FOR: Mencetak detail setiap item dalam Array
    'self.pembelian'
    for i, pembelian in enumerate(self.pembelian):
        nama = pembelian["layanan"]
        harga = pembelian["biaya"]
        diskon = pembelian["diskon"] * 100
        harga_final = pembelian["hargaFinal"]

        # Output ke layar (Struk)
        print(f' {i +
1} \t {nama} \t \t \t {harga} \t \t {diskon} % \t \t \t {harga_final} ')
        total_pembelian.append(harga_final) # Menambahkan harga

```

```

final ke array total

    # Output total bayar, menggunakan fungsi formatrupiah dan
    fungsi sum()
    print("\nTotal Bayar : ",
self.formatrupiah(sum(total_pembelian)))

def main(self):
    # Fungsi utama yang menjalankan program
    while True:
        print("Selamat datang di Klinik Gigi!")
        # ... Tampilkan menu
        try:
            pilihan = int(input("Masukkan pilihan: "))
            # Percabangan (Selection) untuk navigasi menu utama
            if pilihan == 1:
                self.tampilkan_layanan()
            elif pilihan == 2:
                self.pilih_layanan()
            elif pilihan == 3:
                # Percabangan Bersarang: Cek statusBeli
                if self.statusBeli:
                    self.struk()
                    break # Keluar dari perulangan while utama
                else:
                    print("Anda Belum Melakukan Pemilihan Menu")
            else:
                print("Pilihan tidak valid. Silakan coba lagi.")
        except ValueError:
            print(f"Masukan Pilihan dengan benar")

```

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan analisis terhadap kode program Sistem Informasi Klinik Gigi menggunakan bahasa pemrograman Python, dapat disimpulkan bahwa proyek ini telah berhasil mengimplementasikan prinsip-prinsip Pemrograman Terstruktur (Prosedural) yang dipersyaratkan.