

Program Studi Teknik Informatika Institut Teknologi Sumatera

2024

**Pemrograman Berorientasi Objek**

**Laporan Praktikum**

|  |  |
| --- | --- |
| **Modul :** | **Class & Object** |
| **Nama :** | **Rayhan Fadel Irwanto** |
| **NIM :** | **122140236** |
| **Kelas (Kelas Asal) :** | **RA** |

Instruksi sederhana :

* Disarankan kepada **Praktikan Pemrograman Berorientasi Objek** untuk mengeditnya menggunakan Google Docs agar tidak berantakan dan rapi,
* Silahkan mengganti **Nama Modul** baik yang ada pada **Cover** dan **Header** sesuai dengan materi praktikum,
* Gunakan text styling seperti **Heading 1**, **Normal Text** yang telah terformat / Text Style lainnya yang digunakan untuk menjaga estetika laporan,
* Gunakan [**Syntax Highlighter**](https://highlight.hohli.com/index.php)untuk merapikan kode yang sudah Praktikan buat ke dalam Laporan Praktikum.

# Materi Praktikum

Dalam Python, sebuah class berperan sebagai kerangka kerja untuk membuat objek yang memiliki atribut (variabel) dan metode (fungsi) yang didefinisikan di dalamnya. Objek sendiri adalah contoh konkret dari suatu class, yang memiliki sifat dan perilaku yang sesuai dengan definisi class tersebut. Misalnya, kita bisa memiliki class "Mahasiswa" dengan atribut seperti nama dan usia, serta metode seperti "belajar" dan "istirahat". Dengan menciptakan objek-objek berdasarkan class "Mahasiswa", kita bisa memiliki mahasiswa-mahasiswa dengan nama dan usia yang berbeda, yang memiliki kemampuan untuk belajar dan beristirahat sesuai dengan definisi class tersebut. Dengan menggunakan konsep class dan objek, kita dapat mengorganisir dan mengelola data dengan lebih terstruktur dalam program Python.

# Link Source Code

<https://onlinegdb.com/qVNlJZtyN>

# Source Code

Gunakan [**Syntax Highlighter**](https://highlight.hohli.com/index.php)untuk merapikan Source Code yang dipindahkan dari text editor anda ke dokumen ini.

*# Class for books*

**class** Book:

**def** \_\_init\_\_(self, title, genre, author, release\_year):

        self.title = title

        self.genre = genre

        self.author = author

        self.release\_year = release\_year

**class** BookList(Book):

**def** \_\_init\_\_(self, title, genre, author, release\_year):

        super().\_\_init\_\_(title, genre, author, release\_year)

        self.book\_list = []

**def** add\_book(self):

        title = input("Judul buku: ")

        genre = input("Masukkan genre buku: ")

        author = input("Masukkan nama pengarang: ")

        release\_year = int(input("Masukkan tahun penerbitan: "))

        new\_book = Book(title, genre, author, release\_year)

        self.book\_list.append(new\_book)

**def** display\_book\_list(self):

        print("Daftar Buku:")

        for book in self.book\_list:

            print(**f**"Judul: {book.title}, Genre: {book.genre}, Pengarang: {book.author}, Tahun Terbit: {book.release\_year}")

**def** edit\_book(self):

        search\_book = input("Judul buku yang ingin dimodifikasi: ")

        for book in self.book\_list:

            if book.title == search\_book:

                book.title = input("Judul baru: ")

                book.genre = input("Genre baru: ")

                book.author = input("Nama pengarang baru: ")

                book.release\_year = int(input("Tahun penerbitan baru: "))

                print("Buku berhasil dimodifikasi.")

                return

        print(**f**"Buku dengan judul {search\_book} tidak ditemukan...")

**def** delete\_book(self):

        search\_book = input("Masukkan judul buku yang ingin dihapus: ")

        for book in self.book\_list:

            if book.title == search\_book:

                self.book\_list.remove(book)

                print("Buku berhasil dihapus.")

                return

        print(**f**"Buku dengan judul {search\_book} tidak ditemukan...")

**def** exit\_program(self):

        print("Keluar dari program.")

*# Main Program*

initial\_books = int(input("Masukkan jumlah buku mula-mula: "))

book\_data = BookList("", "", "", "")

i = 0

while True:

    print(**f**"Masukkan data buku ke-{i + 1}:")

    book\_data.add\_book()

    i += 1

*# Setelah blok kode dijalankan sekali, cek apakah pengguna ingin melanjutkan*

    choice = input("Tambah buku lagi? (y/n): ")

    if choice.lower() != 'y':

        break

while True:

    print("\nToko Buku Raihan Fadel:")

    print("1. Tambah Data Buku")

    print("2. Lihat List Buku")

    print("3. Modifikasi Buku Lama")

    print("4. Hapus Buku Lama")

    print("5. Keluar Program")

    choice = input("Masukkan nomor inputan (1/2/3/4/5): ")

    if choice == '1':

        book\_data.add\_book()

    elif choice == '2':

        book\_data.display\_book\_list()

    elif choice == '3':

        book\_data.edit\_book()

    elif choice == '4':

        book\_data.delete\_book()

    elif choice == '5':

        book\_data.exit\_program()

        break

    else:

        print("Pilihan tidak valid. Silakan pilih kembali.")

# Dokumentasi Hasil Running

|  |
| --- |
|  |
|  |
| **Gambar 1. Output Class & Object Kasus Toko Buku** |

Code di atas merupakan implementasi program manajemen buku sederhana menggunakan Python. Program ini terdiri dari dua kelas utama, yaitu kelas "Book" untuk mendefinisikan atribut dari buku seperti judul, genre, pengarang, dan tahun terbit, serta kelas "BookList" yang bertanggung jawab untuk menambah, menampilkan, mengedit, dan menghapus buku dalam daftar.

Pertama, pengguna diminta untuk memasukkan jumlah buku awal yang ingin dimasukkan ke dalam daftar. Selanjutnya, pengguna dapat memilih menu untuk melakukan operasi tertentu, seperti menambah buku baru, melihat daftar buku, mengedit buku yang sudah ada, atau menghapus buku dari daftar. Setiap kali pengguna memilih untuk menambah buku, dia diminta untuk memasukkan informasi tentang buku tersebut, termasuk judul, genre, pengarang, dan tahun terbit. Setelah itu, buku baru akan ditambahkan ke dalam daftar.

Kelas "BookList" memiliki metode-metode untuk menjalankan operasi-operasi tersebut. Metode "add\_book" digunakan untuk menambahkan buku baru ke dalam daftar, "display\_book\_list" untuk menampilkan daftar buku yang sudah ada, "edit\_book" untuk mengedit informasi buku yang sudah ada dalam daftar, dan "delete\_book" untuk menghapus buku dari daftar. Setelah setiap operasi, pengguna diberikan opsi untuk melanjutkan atau keluar dari program. Program akan terus berjalan hingga pengguna memilih untuk keluar.