**MAKALAH PEMROGRAM DASAR**

**TOKO BUKU SEDERHANA**

*Jl. Kaliabang No.8, Perwira, Kec. Bekasi Utara, Kota Bks, Jawa Barat 17122*



Tugas ini dibuat untuk memenuhi Nilai Tugas Projek Matakuliah Dasar

Pemrograman

Tim Penyusun :

EFRIANSYAH (19241625)

TIRTA ADI MULYA (19241058)

M.ADRIANSYAH (19241585)

WAHYU NOVIANTO (19241060)

ADAM YASSIR (19241639)

**PROGRAM STUDI SISTEM INFORMASI KAMPUS KALIABANG**

**UNIVERSITAS BINA SARANA INFORMATIKA**

**2024**

# KATA PENGANTAR

Syukur Allhamdulillah senantiasa kami panjakan kehadiran Allah SWT yang telah melimpahkan rahmat dan karunia-Nya sehingga kami dapt menyelesaikan MAKALAH ini guna memenuhi tugas kelompok untuk mata kuliah Dasar Pmrograman dengan judul “**TOKO BUKU SEDERHANA**”.

Kami menyadari bahwa dalam penulisan makalah ini tidak terlepas dari bantuan banyak pihak yang dengan tulus memberikan doa, saran dan kritik sehingga makalah ini dapat terselesaikan.

Kami menyadari sepenuh nya bahwa makalah ini masih jauh dari sempurna dikarenakan terbatasnya pengalaman dan pengetahuan yang kami miliki.Oleh karena itu kami mengharapkan segala bentuk saran serta masukan bahkan kritik yang membangun dari berbagai pihak. Akhirnya kami berharap semoga makalah ini dapat memberikan manfaat bagi perkembangan dunia pendidikan.

**DAFTAR ISI**

[KATA PENGANTAR 2](#_Toc186396142)

[BAB I 4](#_Toc186396143)

[PENDAHULUAN 4](#_Toc186396144)

[1.1. Latar Belakang 4](#_Toc186396145)

[1.1. Ruang Lingkup 5](#_Toc186396146)

[BAB II 7](#_Toc186396147)

[LANDASAN TEORI 7](#_Toc186396148)

[2.1. Pengertian Bahasa Pemrograman Python 7](#_Toc186396149)

[2.2. Kelebihan dan Kekurangan Bahasa Python 7](#_Toc186396150)

[BAB III 8](#_Toc186396151)

[PERANCANGAN DAN IMPLEMENTASI 8](#_Toc186396152)

[3.1. Rancangan Flowchart Aplikasi 8](#_Toc186396153)

[3.3. Source Code 13](#_Toc186396154)

[BAB IV 24](#_Toc186396155)

[PENUTUP 24](#_Toc186396156)

[4.1. Kesimpulan 24](#_Toc186396157)

[4.2. Saran 24](#_Toc186396158)

[DAFTAR PUSTAKA 25](#_Toc186396159)

# BAB I

# PENDAHULUAN

## Latar Belakang

Di era digital saat ini, kemajuan teknologi informasi telah memberikan dampak besar dalam dalam berbagai aspek kehidupan, termasuk dalam pengelolaan bisnis. Salah satu sektor yang turut merasakan manfaat dari penerapan teknologi adalah dunia retail, khususnya toko buku. Proses pengelolaan data penjualan, stok barang, hingga transaksi sering kali menjadi tantangan utama yang memerlukan efisiensi tinggi.

Penggunaan aplikasi berbasis terminal untuk mengelola toko buku sederhana menjadi solusi praktis untuk menghadapai tantangan tersebut. Aplikasi ini dirancang untuk membantu pengguna dalam melakukan manajemen toko buku secara sistematis, mulai dari pengelolaan data buku (CRUD), penambahan stok, hingga transaksi, aplikasi ini diharapkan mampu meningkatkan produktivitas serta meminimalisir kesalahan manusia dalam pengelolaan data.

Melalui aplikasi ini, toko buku dapat mengotomatiskan beberapa proses yang sebelumnya dilakukan secara manual, seperti pencatatan stok dan transaksi. Selain itu, aplikasi ini dirancang agar mudah digunakan, bahkan untuk pengguna yang memiliki pengetahuan teknis yang terbatas. Hal ini menjadikan aplikasi “ Toko Buku Sederhana” sebagai salah satu langkah awal menuju digitalisasi bisnis retail secara sederhana dan efektif.

* Maksud

Pengembangan aplikasi “Toko Buku Sederhana” bertujua untuk menyediakan alat bantu berbasis teknologi yang dapat digunakan untuk mempermudah pengelolaan operasional toko buku, baik dari sisi admin maupun pembeli. Aplikasi ini dirancang agar mudah diakses melalui terminal dan mampu memberikan solusi praktis untuk manajemen stok, transaksi, dan penyimpanan data.

* Tujuan

Tujuan dari pembuatan program Toko Buku Sederhana adalah :

1. Meningkatkan Efisiensi:

Mengotomatisasi proses pengelolaan data buku, seperti penambahan, pengeditan, dan penghapusan data buku, sehingga lebih efisien dibandingkan pencatatan manual.

1. Mempermudah Transaksi:

Mempermudah pembeli dalam melihat daftar buku, menambahkan buku ke keranjang, dan melakukan pembayaran dengan fitur yang sistematis.

1. Meminimalisir Kesalahan:

Mengurangi risiko kesalahan dalam pencatatan stok, harga, dan transaksi penjualan melalui sistem yang terstruktur dan berbasis data.

1. Meningkatkan Akurasi Data:

Menyediakan fitur penyimpanan dan pemuatan data buku secara otomatis dari file eksternal, sehingga data lebih terorganisir dan dapat diakses kapan saja.

1. Menyediakan Dasar Digitalisasi:

Menjadi langkah awal untuk digitalisasi pengelolaan toko buku, yang dapat dikembangkan lebih lanjut menjadi aplikasi berbasis GUI atau web di masa depan.

## Ruang Lingkup

Aplikasi "Toko Buku Sederhana" memiliki ruang lingkup sebagai berikut:

1. **Pengelolaan Data Buku (Admin):**
   * Menambah, mengedit, melihat, dan menghapus data buku yang meliputi kode buku, judul, harga, dan stok.
   * Penyimpanan dan pemuatan data buku dilakukan melalui file eksternal (dataBuku.txt).
2. **Fungsi untuk Pembeli:**
   * Melihat daftar buku yang tersedia.
   * Menambahkan buku ke keranjang belanja.
   * Melihat isi keranjang belanja dan melakukan checkout dengan perhitungan total biaya dan kembalian.
3. **Sistem Keamanan:**
   * Akses ke menu admin dilindungi dengan username dan password untuk mencegah perubahan data oleh pengguna yang tidak berwenang.
4. **Platform:**
   * Aplikasi berjalan di terminal berbasis teks dan menggunakan bahasa pemrograman Python.
5. **Batasan Sistem:**
   * Aplikasi tidak mendukung antarmuka grafis (GUI).
   * Transaksi hanya dapat dilakukan satu pembeli dalam satu waktu.
   * Data buku tidak memiliki kategori atau genre sebagai pengelompokan tambahan.

# BAB II

# LANDASAN TEORI

## 2.1. Pengertian Bahasa Pemrograman Python

Python adalah bahasa pemrograman tingkat tinggi yang bersifat dinamis, interpretatif, dan mudah dibaca. Python dikembangkan oleh Guido van Rossum dan pertama kali dirilis pada tahun 1991. Python dirancang untuk mengutamakan keterbacaan kode, menggunakan indentasi untuk memisahkan blok kode, sehingga menjadikannya pilihan populer bagi pengembang perangkat lunak.

Beberapa fitur utama dari Python adalah:

1. Sintaks yang mudah dipahami
2. Pemrograman yang Berioentasi Objek
3. Portabilitas
4. Ekosistem yang luas

## 2.2. Kelebihan dan Kekurangan Bahasa Python

1. Kelebihan Python

* **Sederhana:** Sintaksis mudah dipahami, cocok untuk pemula.
* **Fleksibel:** Bisa digunakan untuk berbagai macam proyek.
* **Komunitas besar:** Banyak sumber belajar dan dukungan.
* **Pustaka kaya:** Tersedia banyak library untuk berbagai keperluan.
* **Platform independen:** Bisa dijalankan di berbagai sistem operasi.

1. Kekurangan Python:

* **Kecepatan:** Lebih lambat dari bahasa yang dikompilasi.
* **Konsumsi memori:** Agak boros dibandingkan bahasa lain.
* **GIL:** Membatasi kinerja multithreading.
* **Dynamic typing:** Bisa menimbulkan kesalahan saat runtime.
* **Kurang populer untuk mobile:** Pilihan lain lebih umum untuk pengembangan aplikasi mobile.

1. Pengertian Flowchart

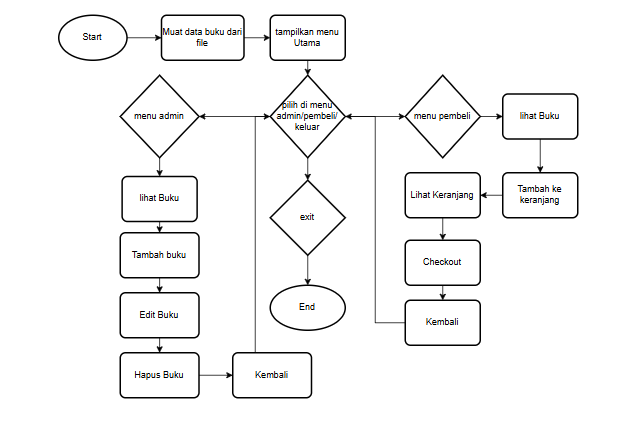
**Flowchart** adalah diagram alir yang menggunakan simbol-simbol grafis untuk menggambarkan langkah-langkah atau alur logika dari suatu proses.

# BAB III

# PERANCANGAN DAN IMPLEMENTASI

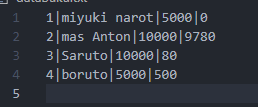
## 3.1. Rancangan Flowchart Aplikasi

Cth:



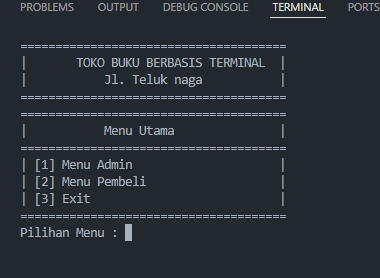
**3.2. Tampilan Aplikasi**

**File data\_buku.txt**

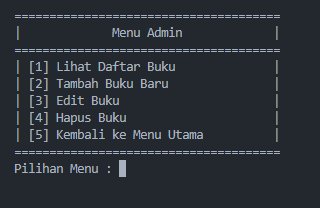
****

**File main.py**

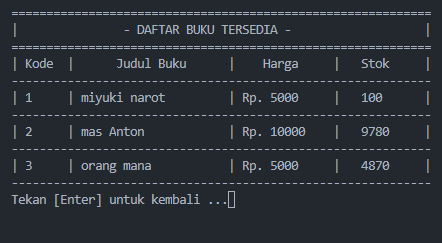
* Menu Utama



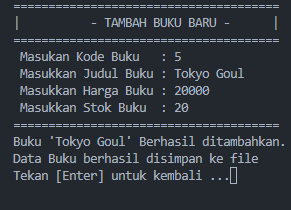
* Menu Admin



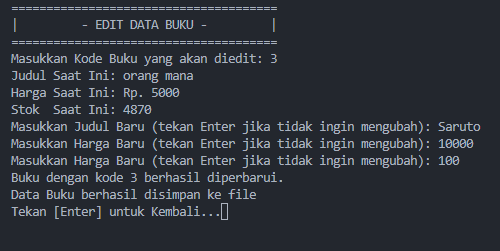
* Lihat Daftar Buku



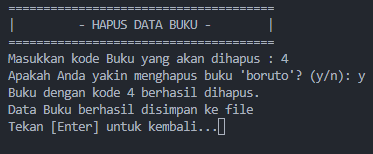
* Tambah Buku



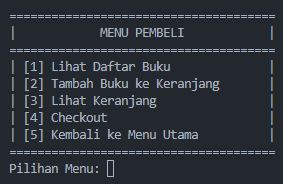
* Edit Buku



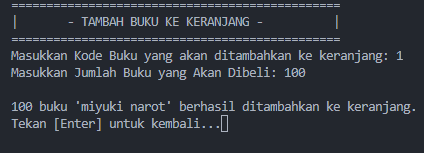
* Hapus Buku



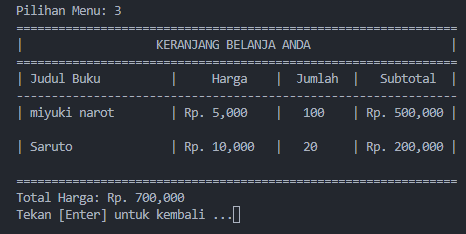
* Menu Pembeli



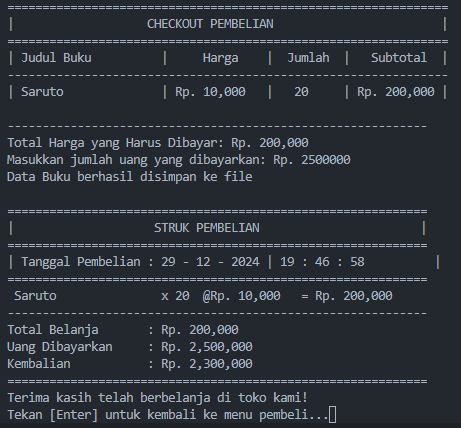
* Tambah Buku ke Keranjang



* Lihat Keranjang



* Checkout & Sturk



## Source Code

* File data\_buku.py

1. import os
2. # Dictionary untuk menyimpan data buku
3. DataBuku = {}
4. # Fungsi untuk membaca data buku
5. def baca\_data\_buku():
6. global DataBuku
7. try:
8. with open('dataBuku.txt', 'r') as file:
9. for line in file:
10. kode, judul, harga, qty = line.strip().split('|')
11. DataBuku[kode] = {"judul": judul, "harga": int(harga), "qty": int(qty)}
12. print("Data Buku berhasil dimuat dari file.")
13. except FileNotFoundError:
14. print("File dataBuku.txt tidak ditemukan. Memulai dengan data kosong.")
15. # Fungsi untuk menyimpan data buku
16. def simpan\_data\_buku():
17. with open('dataBuku.txt', 'w') as file:
18. for kode, buku in DataBuku.items():
19. file.write(f"{kode}|{buku['judul']}|{buku['harga']}|{buku['qty']}\n")
20. print("Data Buku berhasil disimpan ke file")
21. def clear\_os():
22. os.system("CLS")

* File main.py

1. import time
2. from admin import admin
3. from pembeli import pembeli
4. from data\_buku import baca\_data\_buku, simpan\_data\_buku,clear\_os
5. # Fungsi untuk menampilkan menu utama
6. def menu\_utama():
7. clear\_os()
8. print("======================================")
9. print("|       TOKO BUKU BERBASIS TERMINAL  |")
10. print("|           Jl. Teluk naga           |")
11. print("======================================")
13. print('======================================')
14. print('|           Menu Utama               |')
15. print('======================================')
16. print('| [1] Menu Admin                     |')
17. print('| [2] Menu Pembeli                   |')
18. print('| [3] Exit                           |')
19. print('======================================')
20. pilihan = int(input("Pilihan Menu : "))
22. if pilihan == 1:
23. # Meminta input username dan password
24. username = input("Username: ")
25. password = input("Password: ")
26. # Mengecek apakah username dan password benar
27. if (username == "admin" or username == "ADMIN") and (password == "admin123" or password == "ADMIN123"):
28. admin()  # Jika login berhasil, panggil fungsi admin()
29. else:
30. print("Username atau password salah.\n")
31. input("Tekan [Enter] untuk kembali ke menu utama...")
32. menu\_utama()
33. elif pilihan == 2:
34. pembeli()
35. elif pilihan == 3:
36. print('Terimakasih Telah mampir😊')
37. exit()
38. else:
39. print('Pilihan tidak valid. Silahkan coba lagi.')
40. time.sleep(1)
41. menu\_utama()
42. if \_\_name\_\_ == "\_\_main\_\_":
43. baca\_data\_buku()
44. menu\_utama()

* File admin.py

1. import time
2. from data\_buku import DataBuku, simpan\_data\_buku,clear\_os
3. # Fungsi admin
4. def admin():
5. clear\_os()
6. print('======================================')
7. print('|             Menu Admin             |')
8. print('======================================')
9. print('| [1] Lihat Daftar Buku              |')
10. print('| [2] Tambah Buku Baru               |')
11. print('| [3] Edit Buku                      |')
12. print('| [4] Hapus Buku                     |')
13. print('| [5] Kembali ke Menu Utama          |')
14. print('======================================')
15. pilihan = int(input('Pilihan Menu : '))
17. if pilihan == 1:
18. lihat\_buku()
19. elif pilihan == 2:
20. tambah\_buku()
21. elif pilihan == 3:
22. edit\_buku()
23. elif pilihan == 4:
24. hapus\_buku()
25. elif pilihan == 5:
26. import main
27. main.menu\_utama()
28. else:
29. print('Pilihan tidak valid. Silahkan coba lagi.')
30. time.sleep(1)
31. admin()
32. # fungsi untuk menampilkan daftar buku (admin)
33. def lihat\_buku():
34. print('============================================================')
35. print('|               - DAFTAR BUKU TERSEDIA -                   |')
36. print('============================================================')
37. print('| Kode  |      Judul Buku      |    Harga     |   Stok     |')
38. print('------------------------------------------------------------')
39. if not DataBuku:
40. print("|          Tidak Ada Buku Tersedia                          |")
41. print("=============================================================")
42. else:
43. for kode, buku in DataBuku.items():
44. print(f"| {kode:<5} | {buku['judul']:<20} | Rp. {buku['harga']:<8} |   {buku['qty']:<8} |")
45. print('------------------------------------------------------------')
46. input('Tekan [Enter] untuk kembali ...')
47. admin()
48. # fungsi untuk menambah buku baru
49. def tambah\_buku():
50. print('======================================')
51. print('|          - TAMBAH BUKU BARU -      |')
52. print('======================================')
54. try:
55. kode\_buku = input(' Masukan Kode Buku   : ').upper()
56. if kode\_buku in DataBuku:
57. print(f"Error: Buku dengan kode {kode\_buku} Sudah Ada.")
58. else:
59. judul\_buku = input(' Masukkan Judul Buku : ')
60. harga\_buku = int(input(' Masukkan Harga Buku : '))
61. stok\_buku  = int(input(' Masukkan Stok Buku  : '))
62. DataBuku[kode\_buku] = {'judul': judul\_buku, 'harga': harga\_buku, 'qty': stok\_buku}
63. print('======================================')
64. print(f"Buku '{judul\_buku}' Berhasil ditambahkan.")
65. simpan\_data\_buku()
67. except ValueError:
68. print("Error: Harga dan stok buku harus berupa angka. Silahkan Coba Lagi.")
69. input('Tekan [Enter] untuk kembali ...')
70. admin()
72. # fungsi untuk mengedit buku
73. def edit\_buku():
74. print('======================================')
75. print('|         - EDIT DATA BUKU -         |')
76. print('======================================')
77. kode\_buku = input('Masukkan Kode Buku yang akan diedit: ').upper()
78. if kode\_buku in DataBuku:
79. print(f"Judul Saat Ini: {DataBuku[kode\_buku]['judul']}")
80. print(f"Harga Saat Ini: Rp. {DataBuku[kode\_buku]['harga']}")
81. print(f"Stok  Saat Ini: {DataBuku[kode\_buku]['qty']}")
82. judul\_baru = input('Masukkan Judul Baru (tekan Enter jika tidak ingin mengubah): ')
83. harga\_baru = input('Masukkan Harga Baru (tekan Enter jika tidak ingin mengubah): ')
84. stok\_baru =  input('Masukkan Harga Baru (tekan Enter jika tidak ingin mengubah): ')
86. if judul\_baru.strip():
87. DataBuku[kode\_buku]['judul'] = judul\_baru
89. # update harga jika diisi
90. if harga\_baru.strip():
91. try:
92. DataBuku[kode\_buku]['harga'] = int(harga\_baru)
93. except ValueError:
94. print("Error: Harga baru harus berupa angka.")
96. # update stok jika diisi
97. if stok\_baru.strip():
98. try:
99. DataBuku[kode\_buku]['qty'] = int(stok\_baru)
100. except:
101. print("Error: stok baru harus berupa angka.")
102. print(f"Buku dengan kode {kode\_buku} berhasil diperbarui.")
103. simpan\_data\_buku()
104. else:
105. print(f"Error: Buku dengan kode {kode\_buku} tidak ditemukan.")
107. input('Tekan [Enter] untuk Kembali...')
108. admin()

111. # fungsi untuk hapus buku
112. def hapus\_buku():
113. print('======================================')
114. print('|         - HAPUS DATA BUKU -        |')
115. print('======================================')
117. kode\_buku = input('Masukkan kode Buku yang akan dihapus : ').upper()
118. if kode\_buku in DataBuku:
119. konfirmasi = input(f"Apakah Anda yakin menghapus buku '{DataBuku[kode\_buku]['judul']}'? (y/n): ").lower()
120. if konfirmasi == 'y':
121. del DataBuku[kode\_buku]
122. print(f"Buku dengan kode {kode\_buku} berhasil dihapus.")
123. simpan\_data\_buku()
124. else:
125. print("Penghapusan dibatalkan.")
126. else:
127. print(f"Error: Buku dengan kode {kode\_buku} tidak ditemukan.")
128. input('Tekan [Enter] untuk kembali...')
129. admin()

* File pembeli.py

1. import time
2. import datetime
3. from data\_buku import DataBuku, simpan\_data\_buku,clear\_os
4. keranjang = []
5. # fungsi pemeli
6. def pembeli():
7. clear\_os()
8. print('======================================')
9. print('|            MENU PEMBELI            |')
10. print('======================================')
11. print('| [1] Lihat Daftar Buku              |')
12. print('| [2] Tambah Buku ke Keranjang       |')
13. print('| [3] Lihat Keranjang                |')
14. print('| [4] Checkout                       |')
15. print('| [5] Kembali ke Menu Utama          |')
16. print('======================================')
17. pilihan = input('Pilihan Menu: ')
18. if pilihan == "1":
19. lihat\_buku\_pembeli()
20. elif pilihan == "2":
21. tambah\_buku\_keranjang()
22. elif pilihan == "3":
23. lihat\_keranjang()
24. elif pilihan == "4":
25. checkout()
26. elif pilihan == "5":
27. import main
28. main.menu\_utama()
29. else:
30. print('Pilihan tidak valid. Silakan coba lagi.')
31. time.sleep(1)  # Memberi jeda sebelum kembali ke menu
32. pembeli()
34. # Menu lihat buku di pembeli
35. def lihat\_buku\_pembeli():
36. print('============================================================')
37. print('|               - DAFTAR BUKU TERSEDIA -                   |')
38. print('============================================================')
39. print('| Kode  |      Judul Buku      |    Harga     |   Stok     |')
40. print('------------------------------------------------------------')
41. if not DataBuku:
42. print("|          Tidak Ada Buku Tersedia                          |")
43. print("=============================================================")
44. else:
45. for kode, buku in DataBuku.items():
46. print(f"| {kode:<5} | {buku['judul']:<20} | Rp. {buku['harga']:<8,} |   {buku['qty']:<8,} |\n")
47. print('============================================================')
48. input('Tekan [Enter] untuk kembali ...')
49. pembeli()
50. # Menu tambah buku keranjang
51. def tambah\_buku\_keranjang():
52. print('===============================================')
53. print('|       - TAMBAH BUKU KE KERANJANG -          |')
54. print('===============================================')
55. kode\_buku = input('Masukkan Kode Buku yang akan ditambahkan ke keranjang: ').upper()
57. if kode\_buku in DataBuku:
58. try:
59. jumlah\_beli = int(input('Masukkan Jumlah Buku yang Akan Dibeli: '))
60. if jumlah\_beli <= DataBuku[kode\_buku]['qty']:
61. # Tambahkan ke keranjang
62. keranjang.append({
63. 'kode': kode\_buku,
64. 'judul': DataBuku[kode\_buku]['judul'],
65. 'harga': DataBuku[kode\_buku]['harga'],
66. 'qty\_beli': jumlah\_beli
67. })
68. print(f"\n{jumlah\_beli} buku '{DataBuku[kode\_buku]['judul']}' berhasil ditambahkan ke keranjang.")
69. else:
70. print("\nError: Stok buku tidak mencukupi.")
71. except ValueError:
72. print("Error: Jumlah buku harus berupa angka.")
73. else:
74. print('\nKode buku tidak ditemukan.')
76. input('Tekan [Enter] untuk kembali...')
77. pembeli()
78. # Menu lihat\_keranjang
79. def lihat\_keranjang():
80. print('===============================================================')
81. print('|                   KERANJANG BELANJA ANDA                    |')
82. print('===============================================================')
83. print('| Judul Buku          |     Harga    |  Jumlah  |   Subtotal  |')
84. print('---------------------------------------------------------------')
85. if not keranjang:
86. print("|                    Keranjang Anda Kosong                    |")
87. print("===============================================================")
88. else:
89. total = 0
90. for buku in keranjang:
91. subtotal = buku['harga'] \* buku['qty\_beli']
92. total += subtotal
93. print(f"| {buku['judul']:<19} | Rp. {buku['harga']:<8,} |   {buku['qty\_beli']:<6} | Rp. {subtotal:<8,}|\n")
94. print('===============================================================')
95. print(f"Total Harga: Rp. {total:,}")
96. input('Tekan [Enter] untuk kembali ...')
97. pembeli()
98. # Menu checkout()
99. def checkout():
100. print('===============================================================')
101. print('|                   CHECKOUT PEMBELIAN                        |')
102. print('===============================================================')
104. if not keranjang:
105. print("|            Keranjang Anda Kosong, Tidak Ada yang Dibeli. |")
106. print("============================================================")
107. input('Tekan [Enter] untuk kembali ke menu pembeli...')
108. pembeli()  # Kembali ke menu pembeli
109. else:
110. total = 0
111. print('| Judul Buku          |     Harga    |  Jumlah  |   Subtotal  |')
112. print('---------------------------------------------------------------')
113. for buku in keranjang:
114. subtotal = buku['harga'] \* buku['qty\_beli']
115. total += subtotal
116. print(f"| {buku['judul']:<19} | Rp. {buku['harga']:<8,} |   {buku['qty\_beli']:<6} | Rp. {subtotal:<8,}|\n")
117. print("------------------------------------------------------------")
118. print(f"Total Harga yang Harus Dibayar: Rp. {total:,}")
120. # Meminta uang bayar dari pengguna
121. while True:
122. try:
123. uang\_bayar = int(input('Masukkan jumlah uang yang dibayarkan: Rp. '))
124. if uang\_bayar <= 0:
125. print("Error: Uang yang dibayarkan harus lebih besar dari 0.")
126. elif uang\_bayar < total:
127. print(f"Uang yang dibayarkan kurang. Anda masih perlu Rp. {total - uang\_bayar}.")
128. else:
129. kembalian = uang\_bayar - total
131. # Kurangi stok buku di DataBuku
132. for buku in keranjang:
133. kode\_buku = buku['kode']
134. if kode\_buku in DataBuku:
135. DataBuku[kode\_buku]['qty'] -= buku['qty\_beli']
136. # Pastikan stok tidak negatif
137. if DataBuku[kode\_buku]['qty'] < 0:
138. DataBuku[kode\_buku]['qty'] = 0
140. # Simpan perubahan ke file
141. simpan\_data\_buku()
142. clear\_os()
143. # Cetak struk
144. print('\n============================================================')
145. print('|                    STRUK PEMBELIAN                       |')
146. print('============================================================')
147. print(f'| Tanggal Pembelian : { datetime.datetime.now().strftime("%d - %m - %Y | %H : %M : %S          |")}')
148. print('============================================================')
149. for buku in keranjang:
150. subtotal = buku['harga'] \* buku['qty\_beli']
151. print(f" {buku['judul']:<20} x {buku['qty\_beli']:<3} @Rp. {buku['harga']:<8,} = Rp. {subtotal:,}")
152. print("------------------------------------------------------------")
153. print(f"Total Belanja       : Rp. {total:,}")
154. print(f"Uang Dibayarkan     : Rp. {uang\_bayar:,}")
155. print(f"Kembalian           : Rp. {kembalian:,}")
156. print('============================================================')
157. print("Terima kasih telah berbelanja di toko kami!")
159. keranjang.clear()  # Kosongkan keranjang setelah pembayaran
160. break
161. except ValueError:
162. print("Error: Masukkan angka yang valid untuk uang bayar.")
164. input('Tekan [Enter] untuk kembali ke menu pembeli...')
165. pembeli()

# BAB IV

# PENUTUP

## Kesimpulan

Aplikasi "Toko Buku Sederhana" berhasil dirancang dan diimplementasikan sebagai solusi untuk membantu pengelolaan data buku dan transaksi penjualan secara digital. Aplikasi ini menyediakan fitur lengkap untuk admin, seperti pengelolaan data buku (CRUD), serta mempermudah pembeli dalam melihat daftar buku, menambahkan ke keranjang, dan melakukan transaksi. Dengan sistem berbasis terminal, aplikasi ini ringan, mudah digunakan, dan cocok untuk toko buku skala kecil.

Meskipun demikian, aplikasi ini memiliki keterbatasan, seperti belum mendukung antarmuka grafis (GUI) dan pengelompokan buku berdasarkan kategori. Pengembangan lebih lanjut dapat dilakukan untuk meningkatkan fungsi dan fleksibilitas sistem, seperti penambahan fitur laporan penjualan atau integrasi dengan database berbasis SQL.

## Saran

* Tambahkan antarmuka gragis (GUI) agar lebih user-frendly.
* Implementasikan fitur kategori buku untuk mempermudah pencarian.
* Tingkatkan keamanan dengan enkripsi pada proses login admin.
* Integrasikan dengan database untuk pengelolaan data yang lebih baik.
* Kembangkan aplikasi berbasis web atau mobile untuk jangkauan pengguna yang lebih luas.

# DAFTAR PUSTAKA

**Buku**:

Van Rossum, Guido. *Python Programming Language*. 1991.

**Artikel**:

"Pengertian dan Kelebihan Bahasa Pemrograman Python." *Jurnal Teknologi Informasi dan Komunikasi*, vol. 5, no. 2, 2023, pp. 45-58.

**Sumber daring**:

"Flowchart: Pengertian dan Simbol-Simbolnya." *https://www.teknologi.com/flowchart-pengertian*. Diakses pada 7 Desember 2024.