Nandang Duryat - 312310233



Tugas Bahasa Pemograman

Fungsi (function) – Buatkan Aplikasi Peminjaman Buku

Nandang Duryat – 312310233 (TI.23.C4)

2023

Tugas Bahasa Pemograman

Nandang Duryat – 312310233

TI.23.C4

Dosen: Hemdani Rahendra Herlianto, S.Kom., M.T.I.

Rabu, 05-Nov-23

Pertemuan ke 7

Fungsi (function) – Buatkan Aplikasi Peminjaman Buku

Tolong buatkan program C++ peminjaman buku di perpustakaan dengan dengan fungsi dan prosedure !

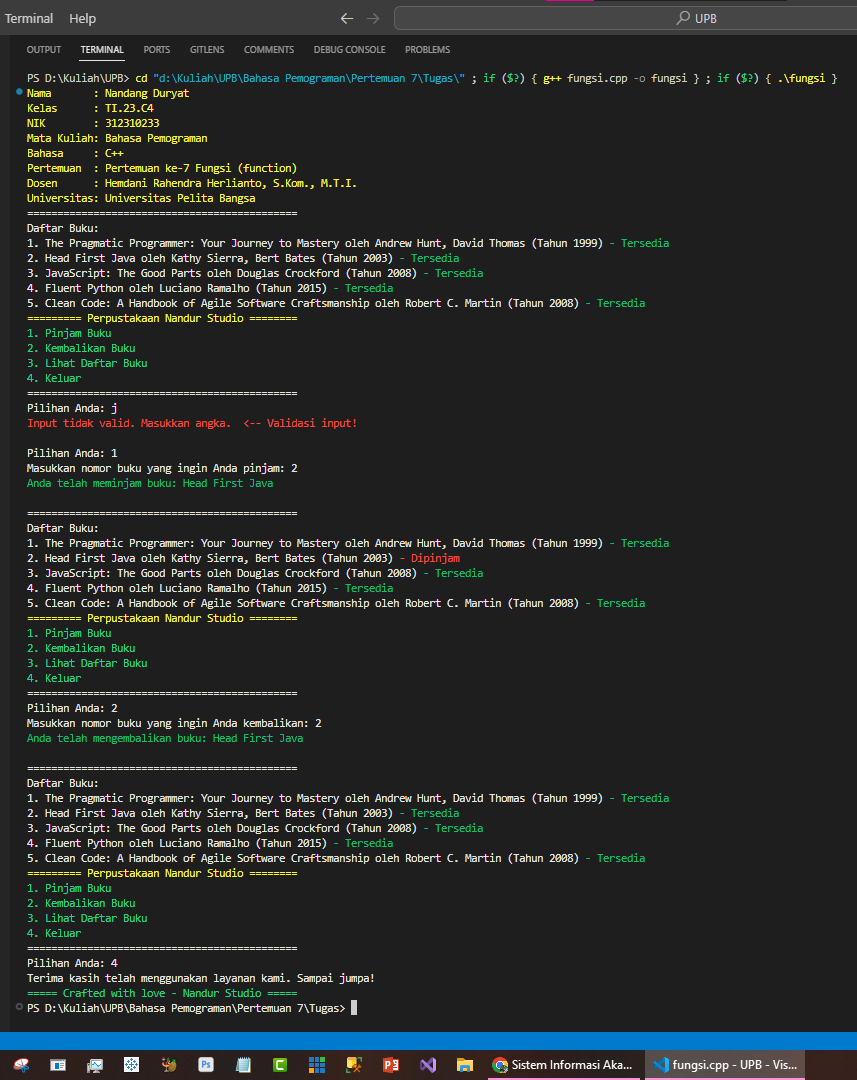
Catatan:

Pastikan untuk menyertakan komentar dalam kode Anda untuk menjelaskan setiap langkah penting.

Selalu validasi input pengguna untuk memastikan bahwa mereka memasukkan data yang sesuai.

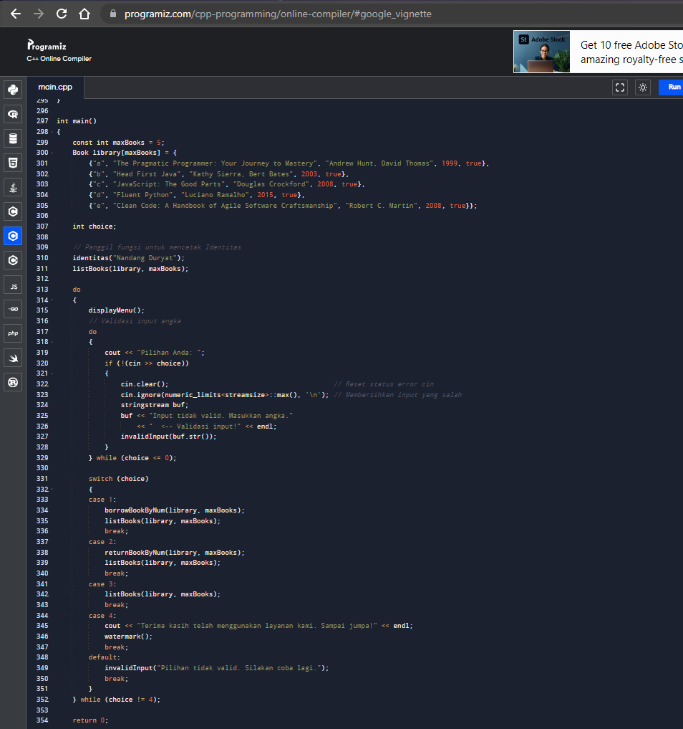
* + - 1. //==============================================================//
      2. // Nama         : Nandang Duryat - 312310233                    //
      3. // Kelas        : TI.23.C4                                      //
      4. // Mata Kuliah  : Bahasa Pemrograman                            //
      5. // Dosen        : Hemdani Rahendra Herlianto, S.Kom., M.T.I.    //
      6. // Kampus       : Universitas Pelita Bangsa                     //
      7. // Tugas        : Pertemuan 7 - Fungsi (function)               //
      8. // Tanggal      : Minggu, 05/11/23                              //
      9. //==============================================================//
      10. // Proyek       : Aplikasi Peminjaman Buku v.1.0.0
      11. // Features     : Advanced features available for windows only!
      12. #include <iostream>
      13. #include <string>
      14. #include <limits>    // Include library limits untuk mengakses numeric\_limits
      15. #include <windows.h> // WinApi header
      16. #include <sstream>
      17. HANDLE hConsole = GetStdHandle(STD\_OUTPUT\_HANDLE);
      18. using namespace std; // Menggunakan namespace std untuk cout, cin, dan endl
      19. // Warna output console khusus Windows
      20. int biru = 10;
      21. int merah = 12;
      22. int kuning = 14;
      23. int putih = 15;
      24. int hijau = 2;
      25. string fromBase64(const string &base64)
      26. {
      27. string base64chars = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/";
      28. string decodedResult = "";
      29. int val = 0, valb = -8;
      30. for (char c : base64)
      31. {
      32. if (isalnum(c))
      33. {
      34. val = (val << 6) + base64chars.find(c);
      35. valb += 6;
      36. if (valb >= 0)
      37. {
      38. decodedResult.push\_back(char((val >> valb) & 0xFF));
      39. valb -= 8;
      40. }
      41. }
      42. }
      43. return decodedResult;
      44. }
      45. // Deklarasi fungsi identitas
      46. void identitas(const string nama\_mahasiswa);
      47. struct Book
      48. {
      49. string num;
      50. string title;
      51. string author;
      52. int year;
      53. bool isAvailable;
      54. };
      55. void displayMenu()
      56. {
      57. SetConsoleTextAttribute(hConsole, kuning);
      58. cout << "========= Perpustakaan Nandur Studio ========" << endl;
      59. SetConsoleTextAttribute(hConsole, biru);
      60. cout << "1. Pinjam Buku" << endl;
      61. cout << "2. Kembalikan Buku" << endl;
      62. cout << "3. Lihat Daftar Buku" << endl;
      63. cout << "4. Keluar" << endl;
      64. SetConsoleTextAttribute(hConsole, putih);
      65. cout << "=============================================" << endl;
      66. }
      67. void userInput(string input)
      68. {
      69. SetConsoleTextAttribute(hConsole, biru);
      70. cout << input;
      71. SetConsoleTextAttribute(hConsole, putih);
      72. }
      73. void invalidInput(string msg)
      74. {
      75. SetConsoleTextAttribute(hConsole, merah);
      76. cout << msg << endl;
      77. SetConsoleTextAttribute(hConsole, putih);
      78. }
      79. void validInput(string msg)
      80. {
      81. SetConsoleTextAttribute(hConsole, hijau);
      82. cout << msg << endl;
      83. SetConsoleTextAttribute(hConsole, putih);
      84. }
      85. int validateBookNumber(int maxBooks)
      86. {
      87. int bookNumber;
      88. while (true)
      89. {
      90. stringstream input;
      91. input << "Masukkan nomor buku (1-" << maxBooks << "): ";
      92. userInput(input.str());
      93. cin >> bookNumber;
      94. if (bookNumber >= 1 && bookNumber <= maxBooks)
      95. {
      96. return bookNumber;
      97. }
      98. else
      99. {
      100. invalidInput("Nomor buku tidak valid. Silakan coba lagi.");
      101. }
      102. }
      103. }
      104. void borrowBook(Book books[], int bookCount)
      105. {
      106. cout << "Daftar Buku yang Tersedia:" << endl;
      107. for (int i = 0; i < bookCount; i++)
      108. {
      109. if (books[i].isAvailable)
      110. {
      111. cout << i + 1 << ". " << books[i].title << " oleh " << books[i].author << " (Tahun " << books[i].year << ")" << endl;
      112. }
      113. }
      114. int bookNumber = validateBookNumber(bookCount);
      115. if (books[bookNumber - 1].isAvailable)
      116. {
      117. cout << "Anda telah meminjam buku: " << books[bookNumber - 1].title << endl;
      118. books[bookNumber - 1].isAvailable = false;
      119. }
      120. else
      121. {
      122. invalidInput("Buku ini sudah dipinjam. Silakan coba buku lain.");
      123. }
      124. }
      125. void returnBook(Book books[], int bookCount)
      126. {
      127. cout << "Buku yang Anda pinjam:" << endl;
      128. for (int i = 0; i < bookCount; i++)
      129. {
      130. if (!books[i].isAvailable)
      131. {
      132. cout << i + 1 << ". " << books[i].title << " oleh " << books[i].author << " (Tahun " << books[i].year << ")" << endl;
      133. }
      134. }
      135. int bookNumber = validateBookNumber(bookCount);
      136. if (!books[bookNumber - 1].isAvailable)
      137. {
      138. cout << "Anda telah mengembalikan buku: " << books[bookNumber - 1].title << endl;
      139. books[bookNumber - 1].isAvailable = true;
      140. }
      141. else
      142. {
      143. invalidInput("Buku ini tidak dalam daftar peminjaman Anda. Silakan coba lagi.");
      144. }
      145. }
      146. void borrowBookByNum(Book books[], int bookCount)
      147. {
      148. int bookNum;
      149. // Validasi input angka
      150. do
      151. {
      152. cout << "Masukkan nomor buku yang ingin Anda pinjam: ";
      153. if (!(cin >> bookNum))
      154. {
      155. cin.clear();                                         // Reset status error cin
      156. cin.ignore(numeric\_limits<streamsize>::max(), '\n'); // Membersihkan input yang salah
      157. stringstream buf;
      158. buf << "Input tidak valid. Masukkan angka."
      159. << "  <-- Validasi input!" << endl;
      160. invalidInput(buf.str());
      161. }
      162. } while (bookNum <= 0);
      163. string pilihan;
      164. switch (bookNum)
      165. {
      166. case 1:
      167. pilihan = "a";
      168. break;
      169. case 2:
      170. pilihan = "b";
      171. break;
      172. case 3:
      173. pilihan = "c";
      174. break;
      175. case 4:
      176. pilihan = "d";
      177. break;
      178. case 5:
      179. pilihan = "e";
      180. break;
      181. default:
      182. invalidInput("Pilihan tidak valid. Silakan coba lagi.");
      183. break;
      184. }
      185. bool found = false;
      186. for (int i = 0; i < bookCount; i++)
      187. {
      188. if (books[i].isAvailable && books[i].num == pilihan)
      189. {
      190. found = true;
      191. stringstream buf;
      192. buf << "Anda telah meminjam buku: " << books[i].title << endl;
      193. validInput(buf.str());
      194. books[i].isAvailable = false;
      195. break;
      196. }
      197. }
      198. if (!found)
      199. {
      200. stringstream buffer;
      201. buffer << "Buku dengan nama '" << bookNum << "' tidak ditemukan atau sudah dipinjam. Silakan coba lagi." << endl;
      202. invalidInput(buffer.str());
      203. }
      204. }
      205. void returnBookByNum(Book books[], int bookCount)
      206. {
      207. int bookNum;
      208. // Validasi input angka
      209. do
      210. {
      211. cout << "Masukkan nomor buku yang ingin Anda kembalikan: ";
      212. if (!(cin >> bookNum))
      213. {
      214. cin.clear();                                         // Reset status error cin
      215. cin.ignore(numeric\_limits<streamsize>::max(), '\n'); // Membersihkan input yang salah
      216. stringstream buf;
      217. buf << "Input tidak valid. Masukkan angka."
      218. << "  <-- Validasi input!" << endl;
      219. invalidInput(buf.str());
      220. }
      221. } while (bookNum <= 0);
      222. string pilihan;
      223. switch (bookNum)
      224. {
      225. case 1:
      226. pilihan = "a";
      227. break;
      228. case 2:
      229. pilihan = "b";
      230. break;
      231. case 3:
      232. pilihan = "c";
      233. break;
      234. case 4:
      235. pilihan = "d";
      236. break;
      237. case 5:
      238. pilihan = "e";
      239. break;
      240. default:
      241. invalidInput("Pilihan tidak valid. Silakan coba lagi.");
      242. break;
      243. }
      244. bool found = false;
      245. for (int i = 0; i < bookCount; i++)
      246. {
      247. if (!books[i].isAvailable && books[i].num == pilihan)
      248. {
      249. found = true;
      250. stringstream buf;
      251. buf << "Anda telah mengembalikan buku: " << books[i].title << endl;
      252. validInput(buf.str());
      253. books[i].isAvailable = true;
      254. break;
      255. }
      256. }
      257. if (!found)
      258. {
      259. stringstream buffer;
      260. buffer << "Buku dengan nama '" << bookNum << "' tidak ditemukan atau sudah dipinjam. Silakan coba lagi." << endl;
      261. invalidInput(buffer.str());
      262. }
      263. }
      264. void listBooks(Book books[], int bookCount)
      265. {
      266. cout << "=============================================" << endl;
      267. cout << "Daftar Buku:" << endl;
      268. for (int i = 0; i < bookCount; i++)
      269. {
      270. cout << i + 1 << ". " << books[i].title << " oleh " << books[i].author << " (Tahun " << books[i].year << ")";
      271. if (books[i].isAvailable)
      272. {
      273. validInput(" - Tersedia");
      274. }
      275. else
      276. {
      277. invalidInput(" - Dipinjam");
      278. }
      279. }
      280. }
      281. void watermark()
      282. {
      283. // Watermark
      284. string base64Encoded = "Q3JhZnRlZCB3aXRoIGxvdmUgLSBOYW5kdXIgU3R1ZGlv";
      285. string decodedText = fromBase64(base64Encoded);
      286. SetConsoleTextAttribute(hConsole, biru);
      287. cout << "===== " << decodedText << " =====" << endl;
      288. SetConsoleTextAttribute(hConsole, putih);
      289. }
      290. int main()
      291. {
      292. const int maxBooks = 5;
      293. Book library[maxBooks] = {
      294. {"a", "The Pragmatic Programmer: Your Journey to Mastery", "Andrew Hunt, David Thomas", 1999, true},
      295. {"b", "Head First Java", "Kathy Sierra, Bert Bates", 2003, true},
      296. {"c", "JavaScript: The Good Parts", "Douglas Crockford", 2008, true},
      297. {"d", "Fluent Python", "Luciano Ramalho", 2015, true},
      298. {"e", "Clean Code: A Handbook of Agile Software Craftsmanship", "Robert C. Martin", 2008, true}};
      299. int choice;
      300. // Panggil fungsi untuk mencetak Identitas
      301. identitas("Nandang Duryat");
      302. listBooks(library, maxBooks);
      303. do
      304. {
      305. displayMenu();
      306. // Validasi input angka
      307. do
      308. {
      309. cout << "Pilihan Anda: ";
      310. if (!(cin >> choice))
      311. {
      312. cin.clear();                                         // Reset status error cin
      313. cin.ignore(numeric\_limits<streamsize>::max(), '\n'); // Membersihkan input yang salah
      314. stringstream buf;
      315. buf << "Input tidak valid. Masukkan angka."
      316. << "  <-- Validasi input!" << endl;
      317. invalidInput(buf.str());
      318. }
      319. } while (choice <= 0);
      320. switch (choice)
      321. {
      322. case 1:
      323. borrowBookByNum(library, maxBooks);
      324. listBooks(library, maxBooks);
      325. break;
      326. case 2:
      327. returnBookByNum(library, maxBooks);
      328. listBooks(library, maxBooks);
      329. break;
      330. case 3:
      331. listBooks(library, maxBooks);
      332. break;
      333. case 4:
      334. cout << "Terima kasih telah menggunakan layanan kami. Sampai jumpa!" << endl;
      335. watermark();
      336. break;
      337. default:
      338. invalidInput("Pilihan tidak valid. Silakan coba lagi.");
      339. break;
      340. }
      341. } while (choice != 4);
      342. return 0;
      343. }
      344. void identitas(const string nama\_mahasiswa)
      345. {
      346. string kelas = "TI.23.C4";
      347. string nik = "312310233";
      348. string makul = "Bahasa Pemograman";
      349. string bahasa = "C++";
      350. string pertemuan = "Pertemuan ke-7 Fungsi (function)";
      351. string dosen = "Hemdani Rahendra Herlianto, S.Kom., M.T.I.";
      352. string universitas = "Universitas Pelita Bangsa";
      353. SetConsoleTextAttribute(hConsole, kuning);
      354. cout << "Nama       : " << nama\_mahasiswa << "\n"; // Gunakan parameter yang diberikan
      355. cout << "Kelas      : " << kelas << "\n";
      356. cout << "NIK        : " << nik << "\n";
      357. cout << "Mata Kuliah: " << makul << "\n";
      358. cout << "Bahasa     : " << bahasa << "\n";
      359. cout << "Pertemuan  : " << pertemuan << "\n";
      360. cout << "Dosen      : " << dosen << "\n";
      361. cout << "Universitas: " << universitas << "\n";
      362. SetConsoleTextAttribute(hConsole, putih);
      363. }

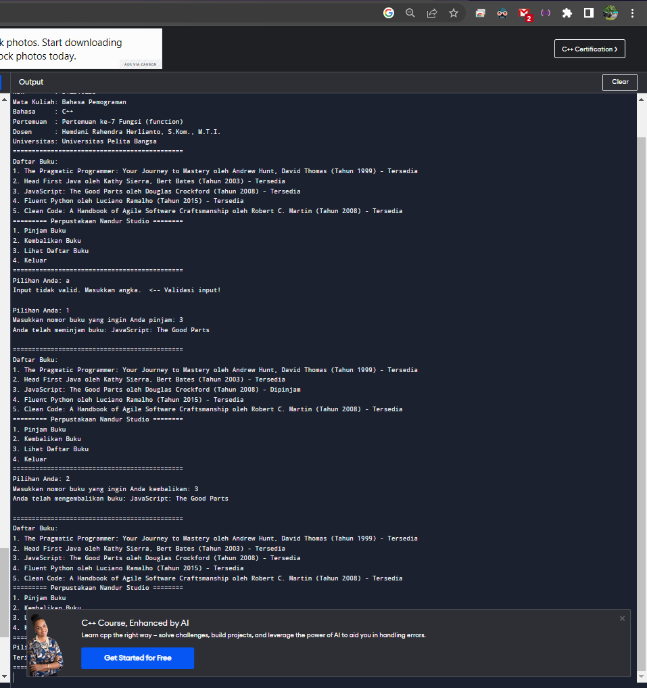
Output pada console VSCode Windows 10 Pro x64



Coba Jalankan Di Web Editor/Compiler

https://www.programiz.com/cpp-programming/online-compiler/#google\_vignette





*Untuk web compiler ini tidak bisa menggunakan Library* ***#include <windows.h>*** *sehingga harus dihapus terlebih dahulu dari code original yang dijalankan di VSCode*

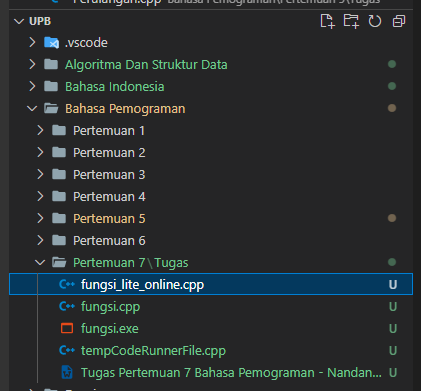
Catatan Program v1.0.0:

Secara garis besar, code program yang saya submit ini hanya bisa berjalan di Windows. Dikarenakan saya menggunakan pustaka standard C++ untuk memanggil pustaka Windows ***#include <windows.h>;***

Fungsi tersebut bisa membantu membedakan warna pada console yang mana sering dilakukan para professional programmer dalam implementasi code pustaka mereka.

Terus bagaimana untuk menjalankan pada online compiler?

Saya sudah sederhanakan pada versi ***fungsi\_lite\_online.cpp*** yang mana code tersebut tidak menyertakan pustaka ***<windows.h>*** dan sudah di hosting di github.



FEATURES:

* Bisa menampilkan list Buku dalam Perpustakaan.
* Penerimaan input sudah dihandle aplikasi sehingga jika tidak sesuai arahan, maka program akan menanyakan kembali sampai input tersebut dianggap benar. Cth: input wajib angka tapi user input huruf.
* Windows only\*, console diberikan warna-warna pembeda seperti warna kuning untuk identitas dan merah ketika input invalid.
* Bisa mencetak terus menerus tanpa batas sampai user menghentikan program tersebut baik melalui pilihan ataupun tutup paksa.

Semua kode program sudah di upload ke GitHub sebagai Hosting Repository Tugas Perkuliahan.

[**https://github.com/nandurstudio/UPB**](https://github.com/nandurstudio/UPB)

Created by:

**Nandang Duryat TI.23.C4 – Universitas Pelita Bangsa**

Editor : Visual Studio Code

Extension : C/C++

Compiler : MinGW 64

Total baris : 395

Link : <https://github.com/nandurstudio/UPB/blob/89487b1ab921196fec05f7f95cd0345b4f5b1bd6/Bahasa%20Pemograman/Pertemuan%207/Tugas/fungsi.cpp>