

ProsesPemesanan.php

1. Script ini dimulai dengan mengecek apakah permintaan (request) adalah metode POST dengan menggunakan `$_SERVER['REQUEST_METHOD'] === 'POST'`. Hal ini menunjukkan bahwa script ini akan dijalankan ketika ada data yang dikirimkan melalui formulir POST.
2. Kemudian, script mengambil data yang dikirimkan melalui POST request seperti **nama**, **mail**, dan **no_hp**. Ini adalah informasi yang diisi oleh pengguna pada formulir pemesanan.
3. Selanjutnya, script mendefinisikan daftar harga alat camping dalam bentuk asosiatif array dengan nama **\$equipment_prices**. Daftar ini mencakup harga untuk setiap item seperti "tenda_4", "gas", "kompor", dan lain-lain.
4. Selanjutnya, script mendefinisikan jumlah barang yang tersedia dari masing-masing item dalam variabel seperti **\$jumlah_barang_tenda_4**, **\$jumlah_barang_gas**, dan seterusnya. Ini digunakan untuk melakukan iterasi dan mencari kombinasi optimal.
5. Kemudian, ada dua loop bersarang. Loop pertama digunakan untuk menghitung kombinasi optimal dari barang-barang camping. Setiap iterasi melalui loop akan mencoba semua kombinasi jumlah barang yang mungkin untuk setiap item dan menghitung total pendapatan (revenue) yang dihasilkan dari kombinasi tersebut.
6. Setiap kali kombinasi baru menghasilkan pendapatan lebih besar dari pendapatan maksimum sebelumnya (**\$revenue > \$max_revenue**), maka kombinasi tersebut disimpan dalam **\$optimal_combination**, dan nilai maksimum pendapatan diperbarui dengan nilai yang baru.
7. Setelah kombinasi optimal ditemukan, script membuat koneksi ke database MySQL menggunakan objek **mysqli**. Informasi koneksi seperti **\$servername**, **\$username**, **\$password**, dan **\$database** telah ditentukan sebelumnya.
8. Script kemudian membangun query SQL **INSERT INTO orders** untuk menyimpan data pemesanan ke database. Data yang disimpan termasuk nama, email, nomor HP, dan total harga pemesanan.
9. Query SQL dieksekusi menggunakan **\$conn->query(\$sql)**. Jika query berhasil dieksekusi, script akan memberikan pesan konfirmasi kepada pengguna dengan menampilkan nama dan total harga pemesanan. Jika terjadi kesalahan, script akan menampilkan pesan error.
10. Terakhir, script menutup koneksi database dan melakukan pengalihan (redirect) ke halaman "index.php".

Pada file index.php

1. Metadata HTML: Bagian <head> dari halaman ini berisi informasi metadata dan tautan ke file CSS dan font yang diperlukan untuk tampilan halaman web yang responsif dan rapi.
2. Navigasi: Halaman ini memiliki navigasi yang terletak di atas, disebut "Topbar." Ini mencakup tombol untuk membuka/menutup sidebar, pencarian, notifikasi, dan profil pengguna. Ini biasanya ditempatkan di bagian atas halaman untuk memberikan akses cepat ke fitur-fitur utama.
3. Sidebar: Navigasi sisi (sidebar) muncul di sebelah kiri halaman dan berisi tautan-tautan ke berbagai bagian aplikasi. Dalam kode ini, pengecekan apakah ada sesi admin dan menampilkan sidebar hanya jika sesi admin telah dimulai.
4. Konten Halaman: Ini adalah area di mana konten utama dari halaman, seperti formulir pemesanan, ditampilkan. Untuk pengguna yang belum masuk, ada juga formulir login admin yang ditampilkan.
5. Formulir Pemesanan: Formulir ini memungkinkan pengguna untuk memasukkan informasi mereka (nama, email, dan nomor telepon) dan memilih jumlah alat camping yang mereka inginkan. Ada pula daftar harga yang diberikan di bawah formulir.
6. Informasi Penjualan Optimal: Bagian ini menampilkan hasil perhitungan dan kombinasi optimal dari alat camping yang dapat dipesan. Di dalam kode ini, kombinasi ini sudah ada secara statis sebagai contoh. Namun, dalam implementasi yang sebenarnya, kombinasi optimal biasanya akan dihasilkan secara dinamis berdasarkan perhitungan, seperti yang terlihat dalam kode PHP.
7. Kalkulasi Harga Total: Ada juga skrip JavaScript yang menghitung total harga berdasarkan input pengguna. Setiap kali input berubah, fungsi calculateTotal dipanggil untuk menghitung harga total.
8. Footer: Bagian ini berisi informasi hak cipta dan biasanya muncul di bagian paling bawah halaman.
9. Logout Modal: Ini adalah modal yang muncul ketika pengguna ingin keluar dari sesi. Modal ini meminta pengguna untuk mengonfirmasi apakah mereka benar-benar ingin keluar.
10. Skrip JavaScript: Ada beberapa skrip JavaScript yang digunakan untuk mengaktifkan fungsi interaktif dalam halaman seperti perhitungan harga total.

Pada content_admin.php

1. **Koneksi ke Database:** Kode pertama adalah untuk menghubungkan ke database MySQL. Informasi koneksi seperti nama host, nama pengguna, kata sandi, dan nama database didefinisikan di sini. Jika koneksi gagal, pesan kesalahan akan ditampilkan.
2. **Penanganan Formulir:** Kode ini menangani dua jenis operasi dengan menggunakan metode POST:
 - **Update Status Order:** Jika `$_POST["update_order"]` tersedia, maka status pesanan diubah dalam database berdasarkan nilai yang dipilih oleh admin.
 - **Hapus Order:** Jika `$_POST["delete_order"]` tersedia, pesanan dihapus dari database.
3. **Query Database:** Query SQL untuk mengambil data dari database adalah: **"SELECT * FROM orders"**. Ini mengambil semua pesanan yang ada dari tabel "orders" dalam database.
4. **Tabel Daftar Pesanan:** Kode HTML menampilkan daftar pesanan dalam bentuk tabel. Informasi seperti nama, email, nomor HP, total harga, status, dan jumlah masing-masing item alat camping ditampilkan dalam kolom-kolom tabel. Setiap baris pesanan termasuk dua formulir:
 - Formulir pertama mengizinkan admin untuk mengubah status pesanan (Diproses, Dikirim, Selesai) dengan mengirimkan data dengan tombol "Update".
 - Formulir kedua memungkinkan admin untuk menghapus pesanan dengan tombol "Delete".
5. **Laporan Laba Rugi:** Setelah tabel pesanan, kode ini juga mencakup sebagian tambahan yang menghitung laporan laba rugi dari pesanan. Ini menghitung total pendapatan (jumlah total harga dari pesanan) dan total biaya operasional (jumlah total item alat camping yang terjual), dan kemudian menghitung laba bersih dengan mengurangi biaya operasional dari pendapatan.
6. **Tutup Koneksi Database:** Terakhir, kode ini menutup koneksi ke database setelah selesai mengambil data dan menghitung laporan laba rugi.