**LAPORAN PRAKTIKUM PEMROGRAMAN WEB**

**PERTEMUAN 9**

**POST DAN GET REQUEST**

****

**2411102441052**

**Angga Maulana Saputra**

**FAKULTAS SAINS DAN TEKNOLOGI**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

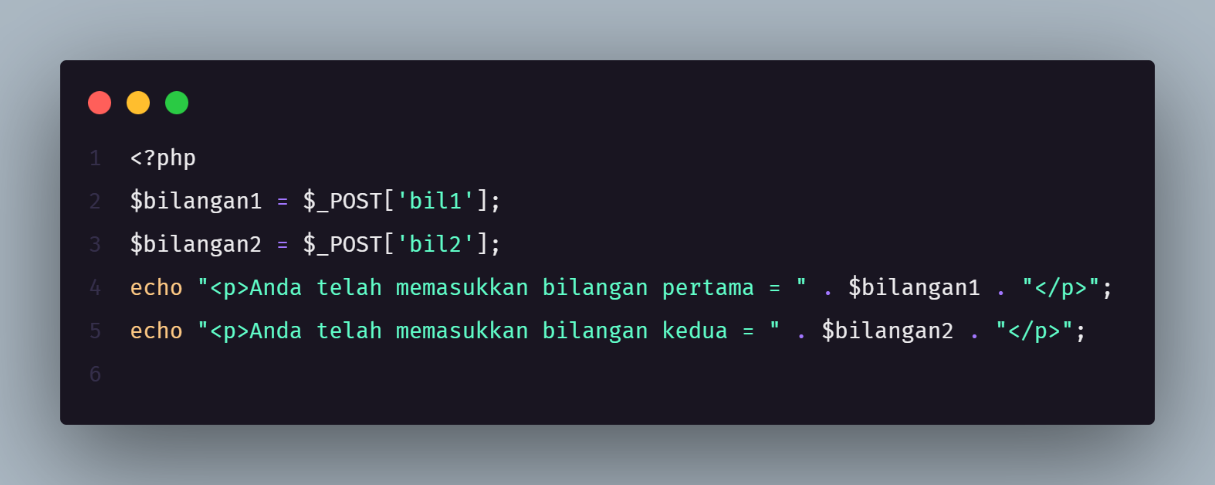
**UNIVERSITAS MUHAMMADIYAH KALIMANTAN TIMUR**

## Latihan 9.1

**input.html**



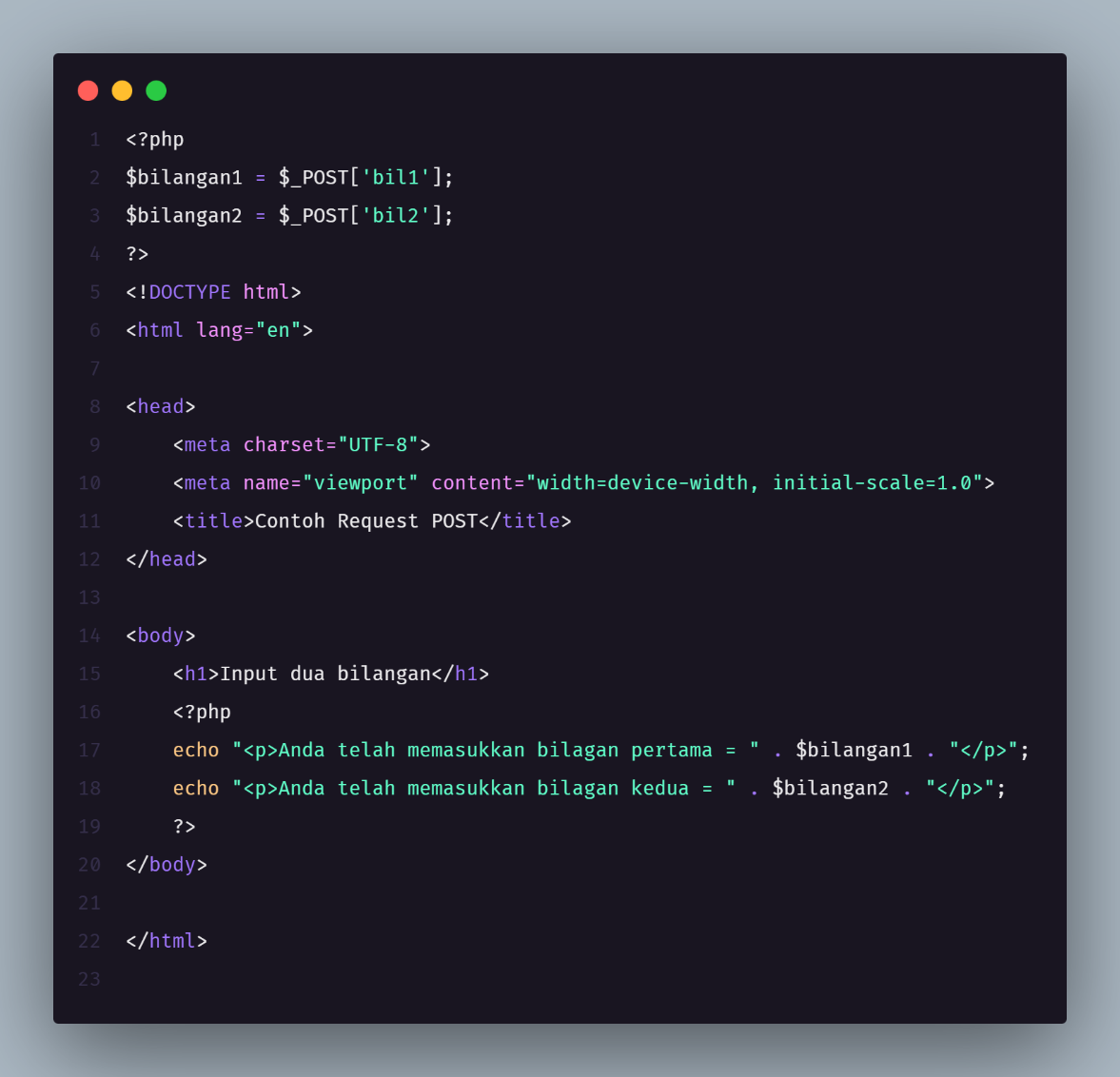
**proses1.php**



File input.html berisi form sederhana yang meminta pengguna untuk memasukkan dua bilangan melalui dua elemen input teks. Setelah pengguna mengisi kedua bilangan dan menekan tombol Submit, data tersebut dikirim ke file proses1.php menggunakan metode POST. Pada file proses1.php, data yang dikirim dari form diterima menggunakan variabel $\_POST, kemudian ditampilkan kembali kepada pengguna dalam bentuk paragraf menggunakan perintah echo.

## Latihan 9.2

**proses2.php**

****

Di dalam file ini, data yang dikirim dari form input.html diterima oleh proses2.php melalui variabel $\_POST['bil1'] dan $\_POST['bil2']. Nilai dari kedua bilangan tersebut kemudian ditampilkan kembali di halaman web menggunakan perintah echo.

## Latihan 9.3

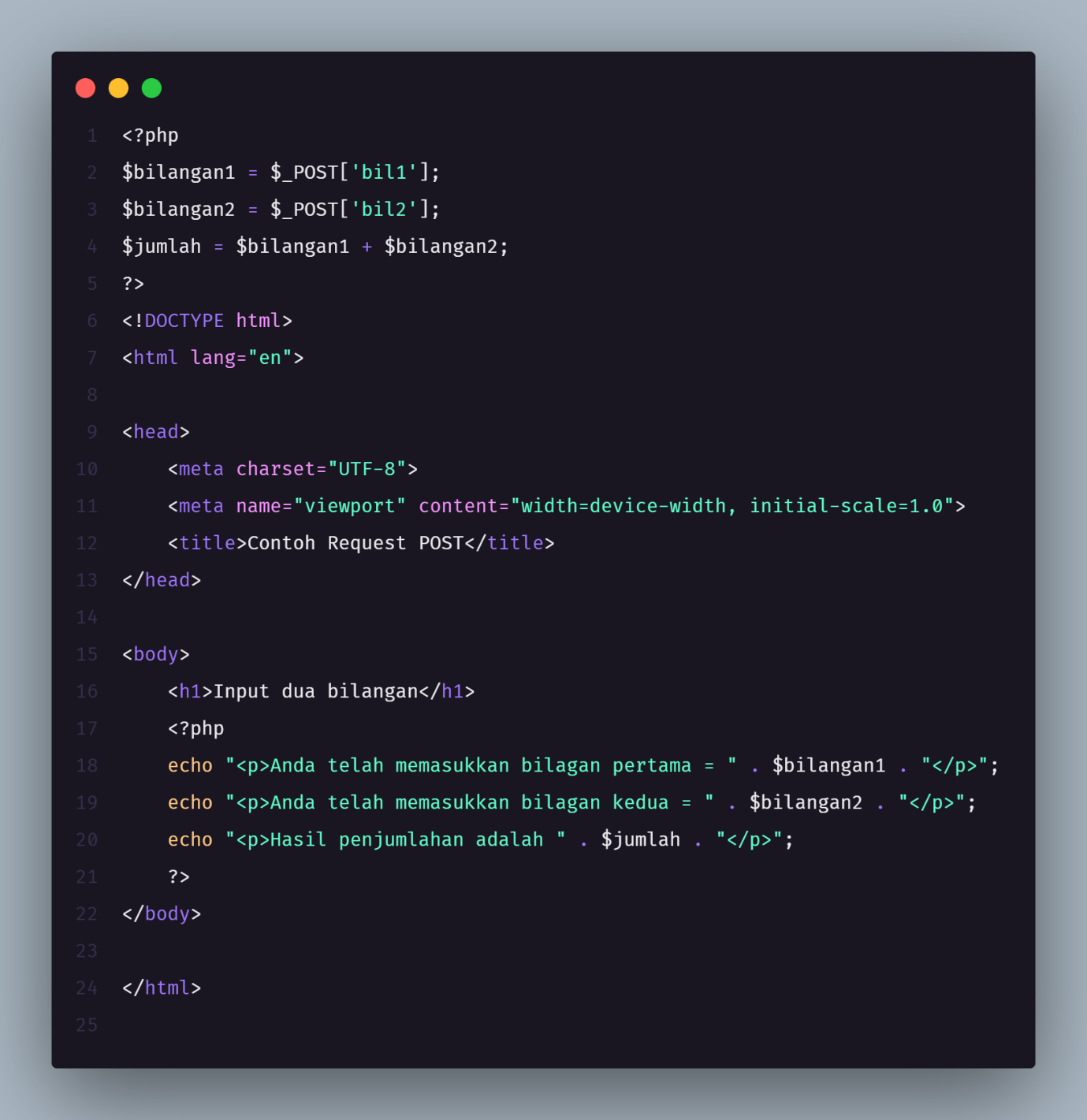
**proses3.php**



Di dalam file proses3.php, data yang dikirim dari form diterima secara langsung melalui variabel $\_POST['bil1'] dan $\_POST['bil2'] tanpa perlu disimpan terlebih dahulu ke dalam variabel terpisah. Nilai kedua bilangan tersebut kemudian ditampilkan di halaman web menggunakan perintah echo dalam tag HTML paragraf <p>.

## Latihan 9.4

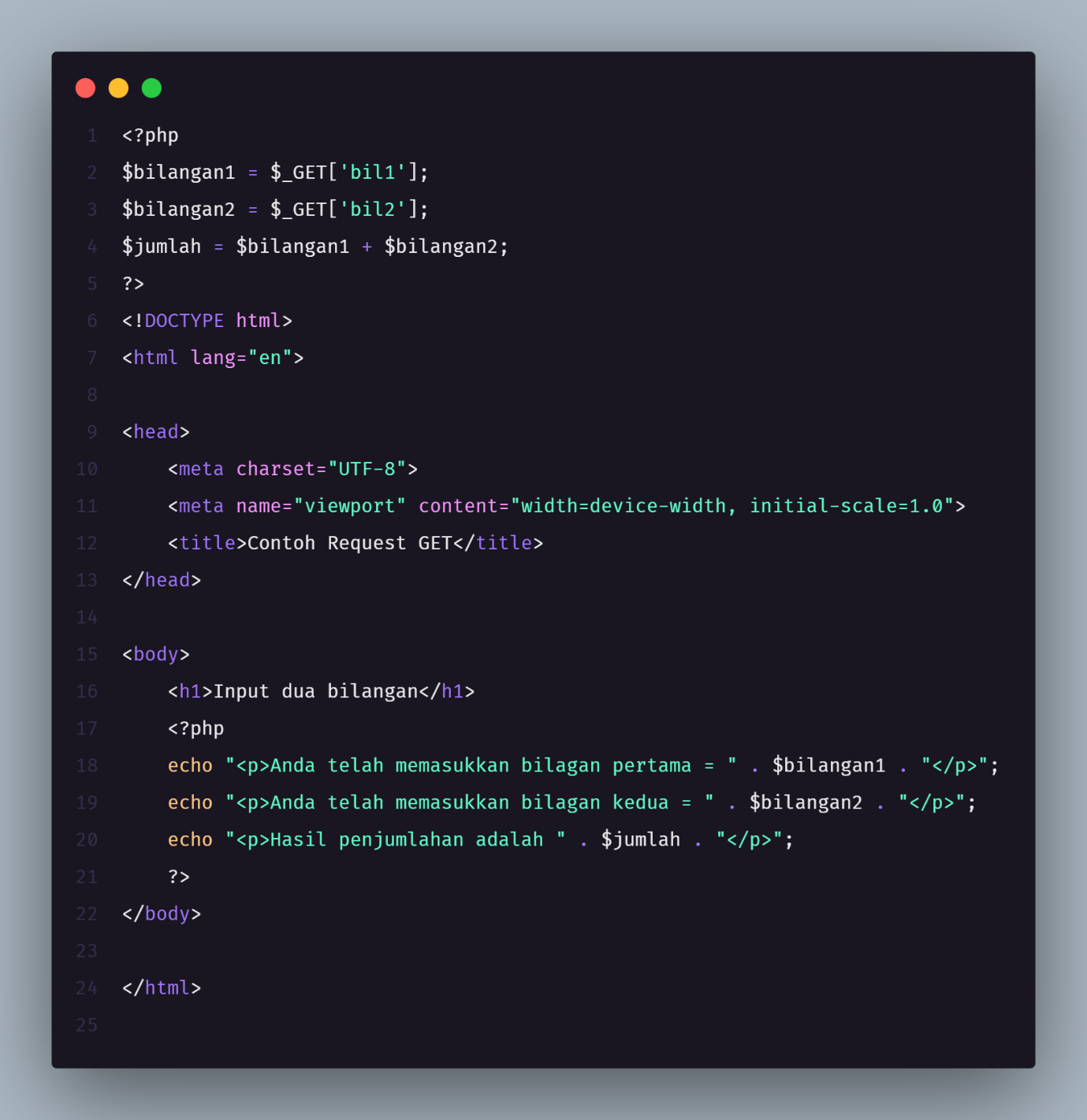
**proses4.php**



Pada halaman input.html, user diminta memasukkan dua bilangan, lalu ketika tombol Submit ditekan, data tersebut dikirim ke proses4.php melalui metode POST. Di dalam file ini, data dari form diterima melalui variabel $\_POST['bil1'] dan $\_POST['bil2'], kemudian dilakukan proses penjumlahan menggunakan operasi $jumlah = $bilangan1 + $bilangan2;. Hasil perhitungan tersebut ditampilkan bersama dengan nilai kedua bilangan yang dimasukkan pengguna menggunakan perintah echo.

## Latihan 9.5

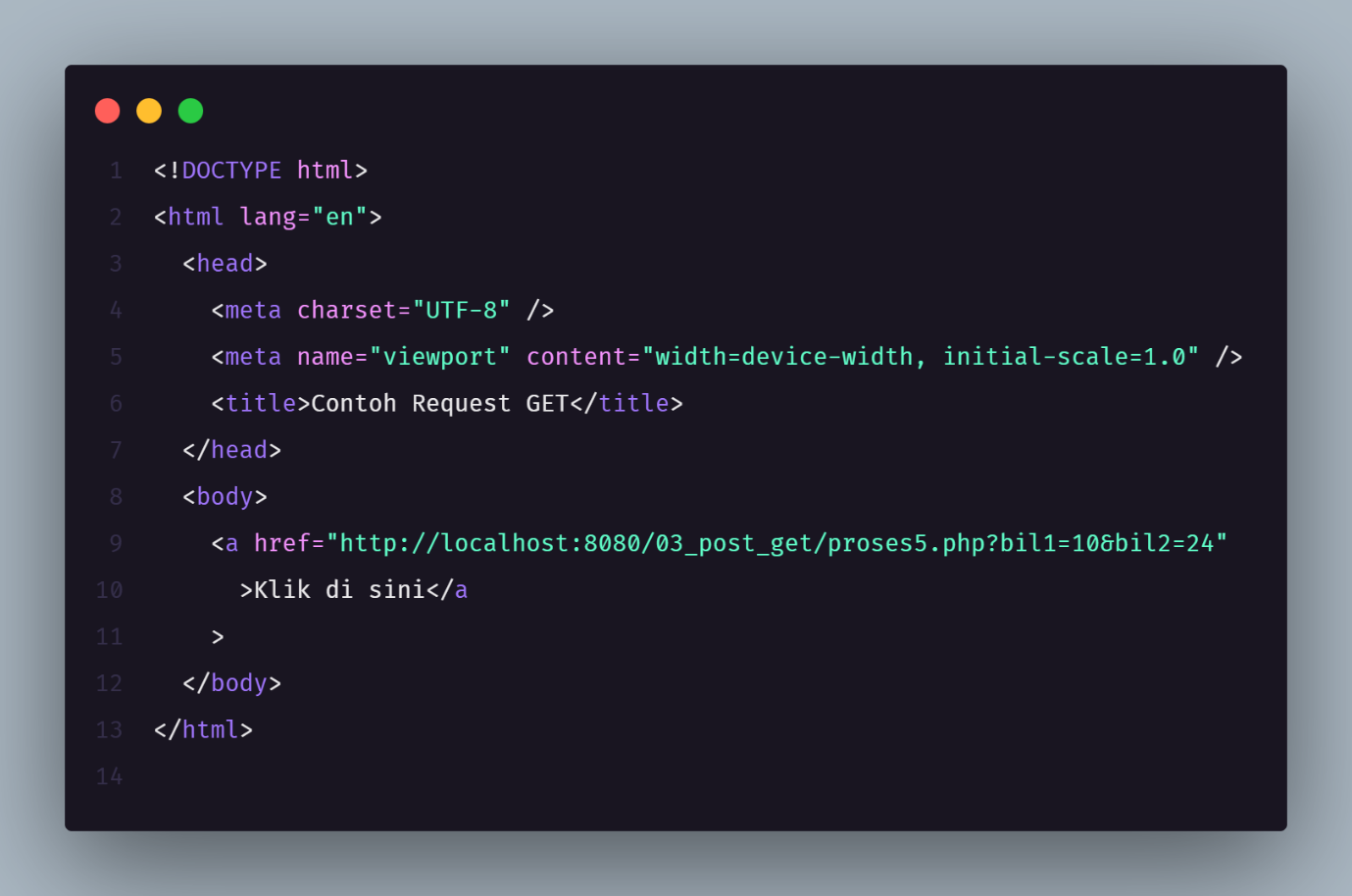
**proses5.php**

****

Di dalam file ini, data yang dikirim melalui URL diterima menggunakan variabel $\_GET['bil1'] dan $\_GET['bil2']. Nilai dari kedua bilangan tersebut kemudian dijumlahkan menggunakan operasi $jumlah = $bilangan1 + $bilangan2;, dan hasilnya ditampilkan di halaman web menggunakan perintah echo. Berbeda dengan metode POST, metode GET menampilkan data yang dikirim langsung pada URL, sehingga cocok digunakan untuk pengiriman data yang bersifat tidak rahasia.

## Latihan 9.6

**jumlah.html**



Pada file ini, digunakan elemen hyperlink (<a>) yang memiliki atribut href berisi URL menuju file proses5.php disertai dengan parameter bil1 dan bil2. Ketika tautan diklik, browser akan mengirimkan nilai bil1=10 dan bil2=24 ke proses5.php melalui URL, sehingga data dapat diproses oleh skrip PHP. Di proses5.php, nilai kedua bilangan tersebut dijumlahkan dan hasilnya ditampilkan di halaman web.

# Contoh Studi Kasus

## Praktikum 9.7

**script6-1.php**



**script6-1proses.php**

****

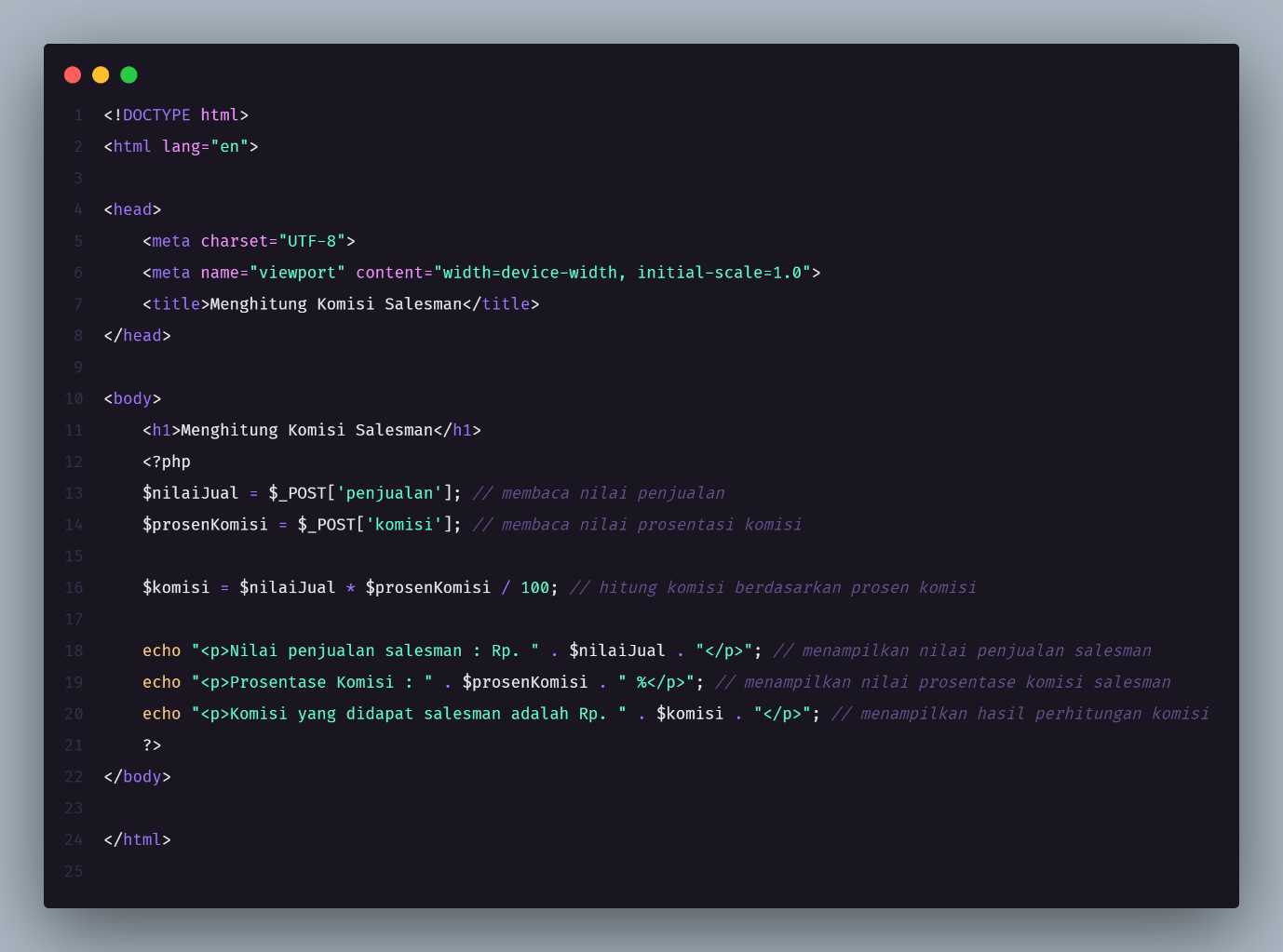
File script6-1.php berfungsi sebagai halaman form yang meminta pengguna untuk mengisi beberapa data, yaitu nama, alamat, jenis kelamin, pekerjaan, dan status pernikahan. Setelah form diisi dan tombol Submit ditekan, data tersebut dikirim ke file script6-1-proses.php menggunakan metode POST. Di dalam script6-1-proses.php, setiap data yang dikirim dari form diterima oleh skrip PHP melalui variabel $\_POST, kemudian ditampilkan kembali dalam bentuk tabel yang rapi di halaman web.

## Praktikum 9.8

**script6-2.php**



**script6-2proses.php**



File script6-2.php berisi form HTML yang meminta pengguna memasukkan jumlah penjualan serta prosentase komisi dalam bentuk angka. Setelah data diisi dan tombol Submit ditekan, nilai tersebut dikirim ke file script6-2proses.php menggunakan metode POST. Di dalam file script6-2proses.php, data yang dikirim dari form dibaca menggunakan variabel $\_POST['penjualan'] dan $\_POST['komisi']. Kemudian, PHP melakukan proses perhitungan komisi dengan rumus $komisi = $nilaiJual \* $prosenKomisi / 100;. Hasil perhitungan tersebut ditampilkan kembali di halaman web berupa nilai penjualan, persentase komisi, dan total komisi yang diperoleh.

## Praktikum 9.9

**script6-3.php**



**script6-3proses.php**



File script6-3.php berisi form HTML yang meminta pengguna untuk memasukkan dua waktu, masing-masing terdiri dari jam, menit, dan detik. Setelah data diisi dan tombol Submit ditekan, nilai-nilai tersebut dikirim ke script6-3proses.php menggunakan metode POST. Di dalam file script6-3proses.php, data dari form dibaca menggunakan variabel $\_POST, kemudian dilakukan proses konversi dari jam, menit, dan detik menjadi total detik untuk masing-masing waktu. Setelah itu, program menghitung selisih antara kedua waktu dalam satuan detik dengan rumus $selisih = $totalDetik1 - $totalDetik2;. Hasil perhitungan ditampilkan di halaman web berupa jumlah selisih detik antara dua waktu yang dimasukkan.

# Soal

## Soal 1

**soal1.html**



**soal1.php**

****

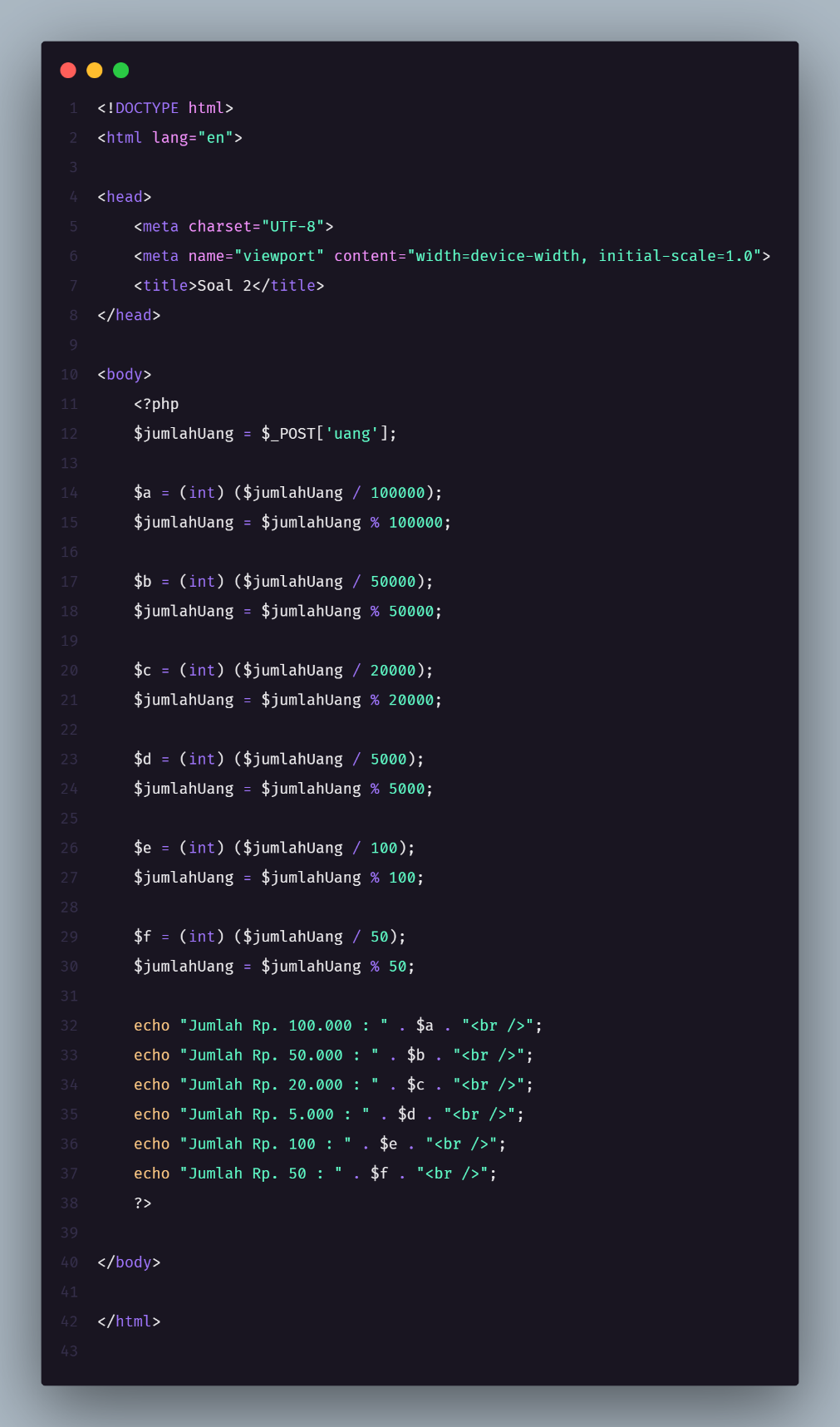
File soal1.html berisi form HTML yang meminta pengguna untuk memasukkan tiga data, yaitu saldo awal, bunga dalam persen, dan jumlah bulan. Setelah data dimasukkan dan tombol Submit ditekan, nilai-nilai tersebut dikirim ke file soal1.php menggunakan metode POST. Di dalam file soal1.php, data dari form diterima menggunakan variabel $\_POST, kemudian dilakukan proses perhitungan saldo akhir dengan menambahkan hasil bunga terhadap saldo awal. Hasil perhitungan tersebut ditampilkan di halaman web dalam format teks yang menunjukkan saldo akhir.

## Soal 2

**soal2.html**



**soal2.php**

****

File soal2.html berisi form yang meminta pengguna untuk memasukkan jumlah uang dalam satuan rupiah. Setelah nilai dimasukkan dan tombol Submit ditekan, data tersebut dikirim ke file soal2.php menggunakan metode POST. Di dalam soal2.php, nilai yang diterima kemudian diproses menggunakan operasi pembagian dan modulus (/ dan %) untuk menghitung berapa lembar atau keping uang yang dibutuhkan dari setiap pecahan, mulai dari Rp100.000 hingga Rp50. Program ini menampilkan hasil berupa jumlah masing-masing pecahan yang diperlukan agar totalnya sama dengan nilai uang yang dimasukkan.

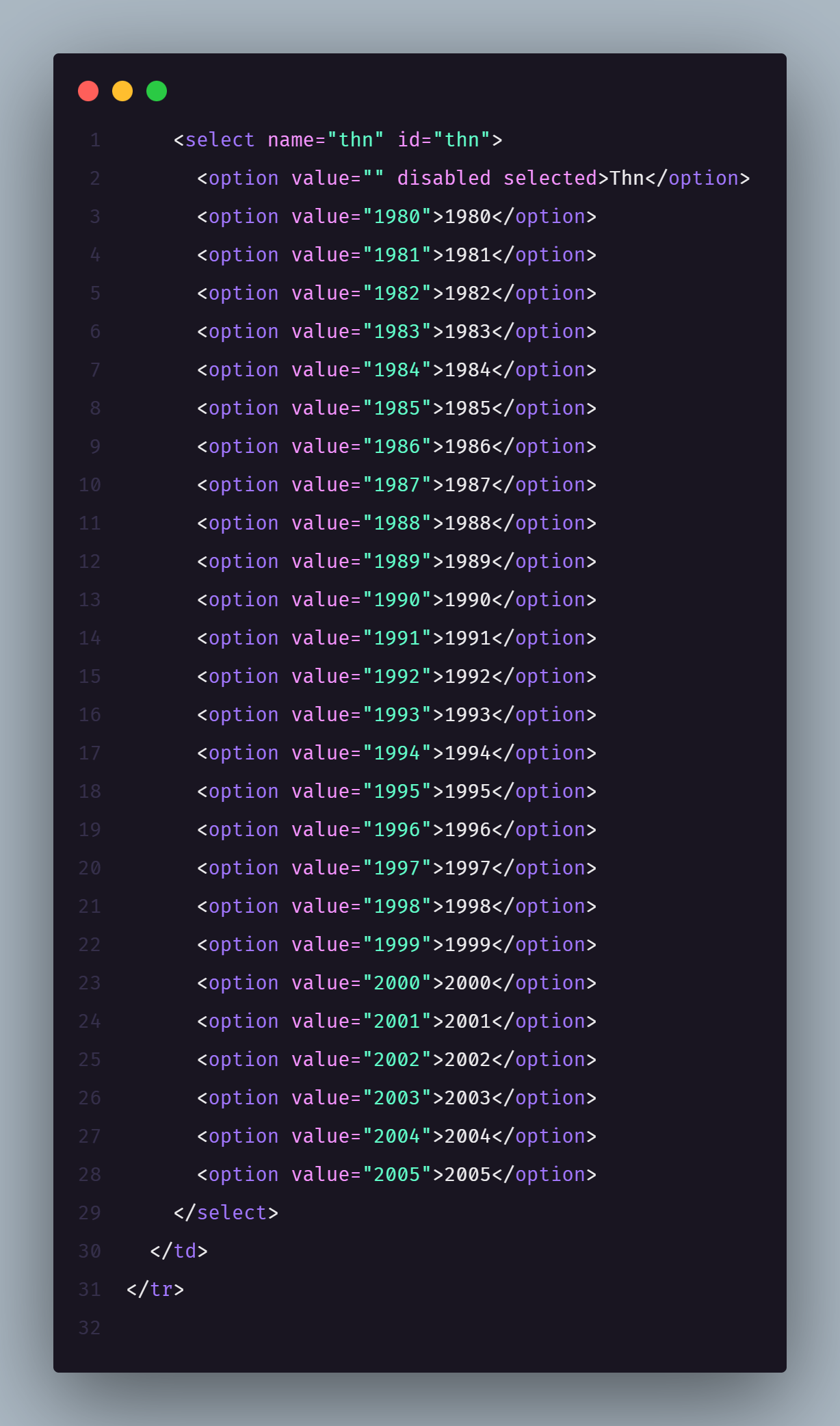
## Soal 3

**soal3.html**

****

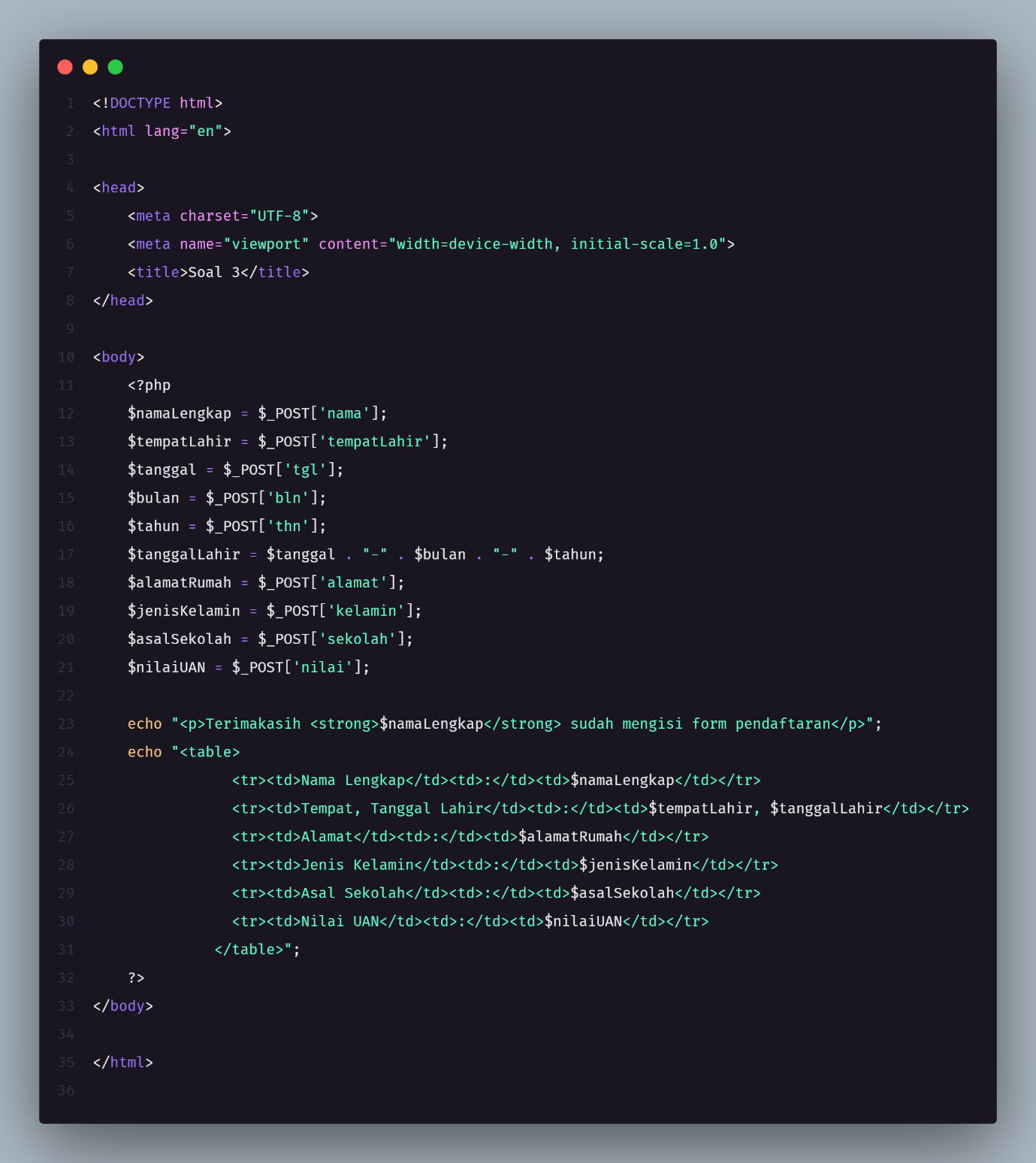
****

****

****

****

**soal3.php**

****

Pada halaman soal3.html, pengguna diminta untuk mengisi berbagai data pribadi seperti nama lengkap, tempat dan tanggal lahir, alamat rumah, jenis kelamin, asal sekolah, serta nilai UAN. Setelah form diisi, data dikirim ke halaman soal3.php melalui metode POST. Di dalam soal3.php, setiap input yang diterima dari pengguna dibaca menggunakan variabel $\_POST dan kemudian ditampilkan kembali dalam bentuk tabel agar lebih rapi dan mudah dibaca.