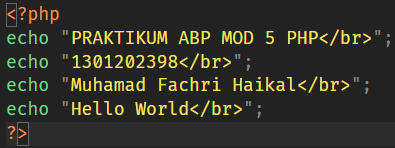
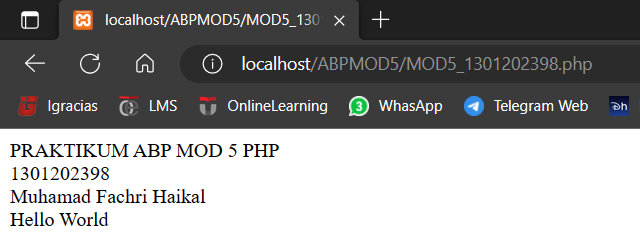
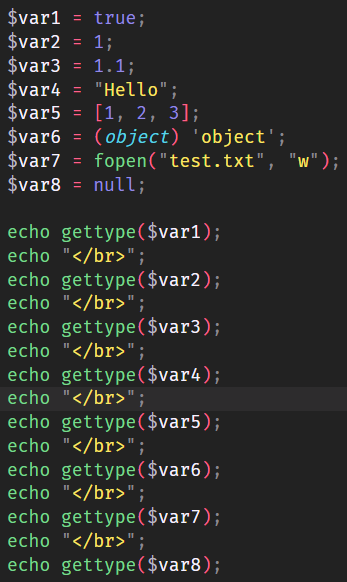
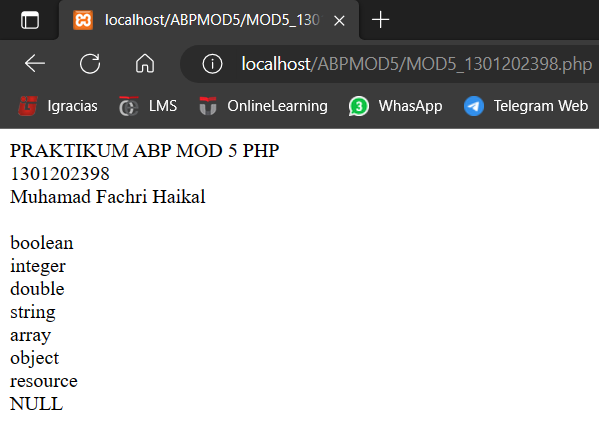
* 1. **Web Server dan Server Side Scripting**

Web server adalah perangkat lunak yang berfungsi untuk melayani permintaan dari klien melalui protokol HTTP (Hypertext Transfer Protocol) atau HTTPS (Hypertext Transfer Protocol Secure) pada jaringan internet. Web server mengelola dan memproses permintaan klien untuk mengakses berbagai jenis sumber daya web seperti halaman web, gambar, video, file, atau aplikasi web. Contohnya adalah Apache Web Server.

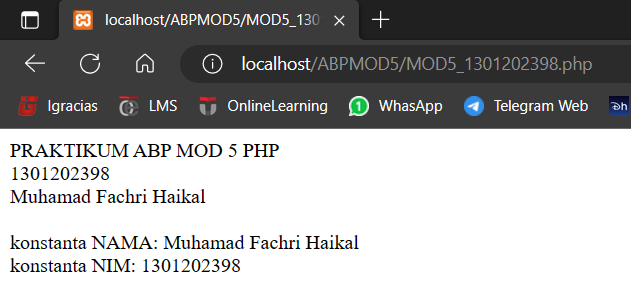
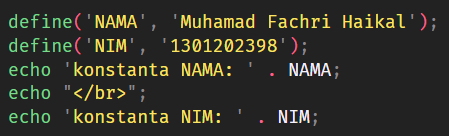
Server-side scripting adalah kode program yang dijalankan pada sisi server dalam sebuah aplikasi web. Dalam hal ini, kode program ditulis dalam bahasa pemrograman tertentu dan dijalankan pada server, kemudian hasilnya dikirimkan ke browser pengguna sebagai bagian dari respon dari server. Contohnya adalah PHP.

* 1. **Pengenalan PHP**

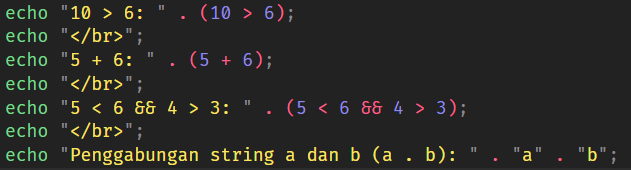
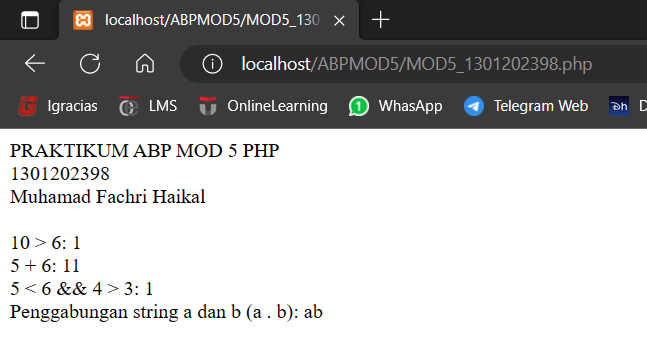
Pengenalan PHP dilakukan dengan echo atau print suatu teks seperti yang ada di atas, yaitu judul praktikum, nim, nama, dan hello world.

* 1. **Variabel**

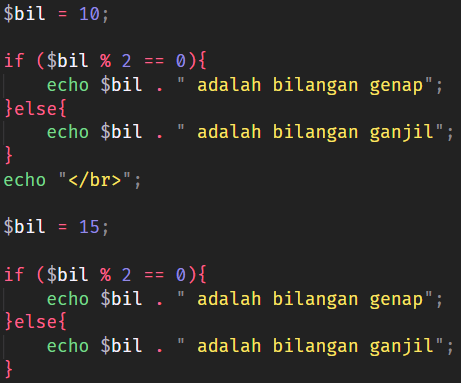
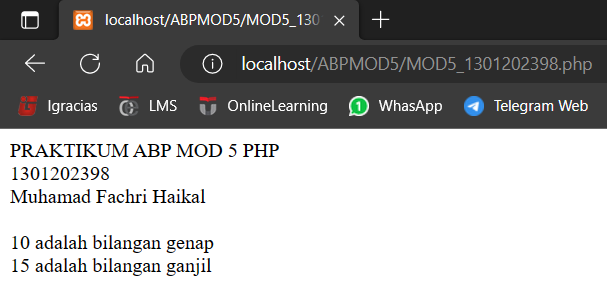
Terdapat 8 tipe data primitif pada PHP sesuai dengan output di atas yaitu, var1:Boolean, var2:integer, var3:float, var4:string, var5:array, var6;object, var7:resource, var8:NULL

* 1. **Konstanta**

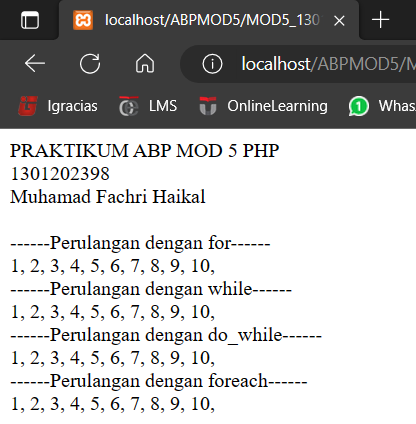
Pendefinsiian konstanta pada PHP dibuat menggunakan function define(). Contohnya adalah code di atas yaitu membuat konstanta NAMA dan NIM.

* 1. **Operator dalam PHP**

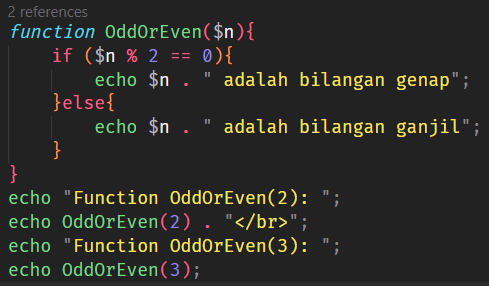
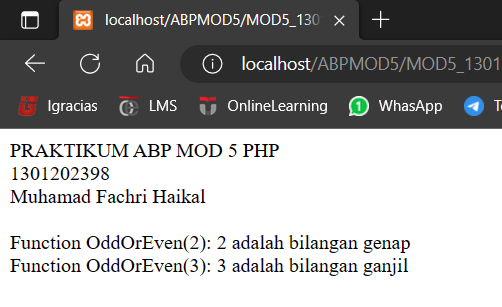
Operator dalam PHP pada umumnya sama dengan Bahasa pemrograman lainnya. Sebagai contoh di atas ada operator perbandingan(>), aritmatika(+), logika(&&), dan penggabungan string (.).

* 1. **Struktur Kondisi**

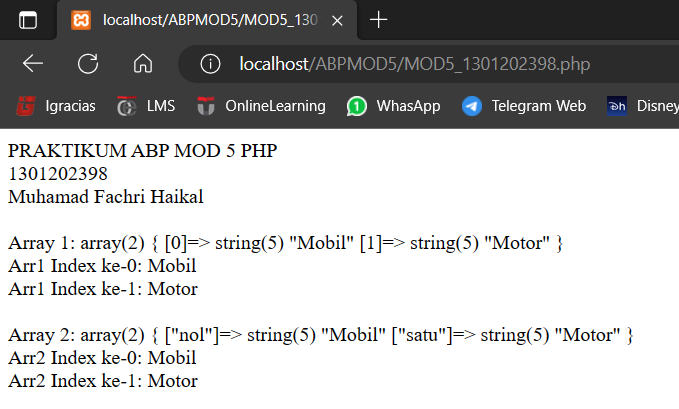
Struktur pada PHP dapat diterapkan menggunakan IF-ELSE seperti contoh di atas yaitu pengondisian bilangan genap atau ganjil.

* 1. **Perulangan**

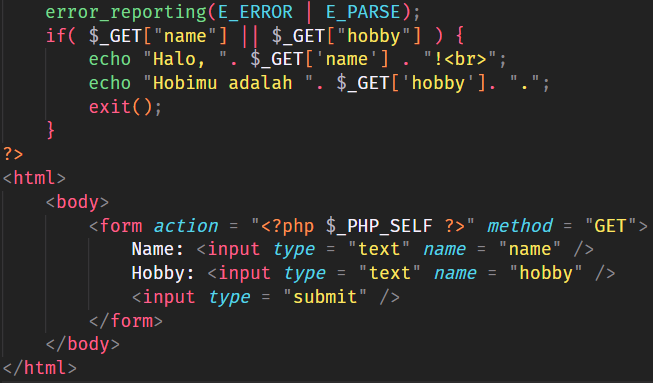
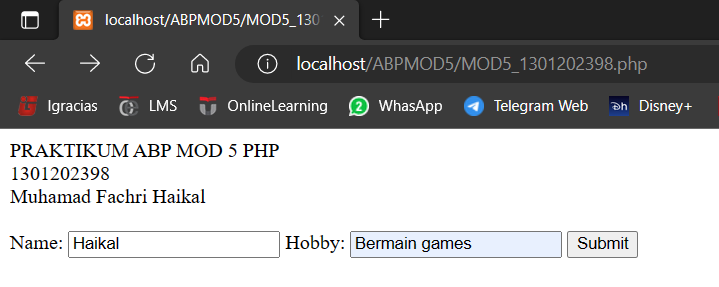
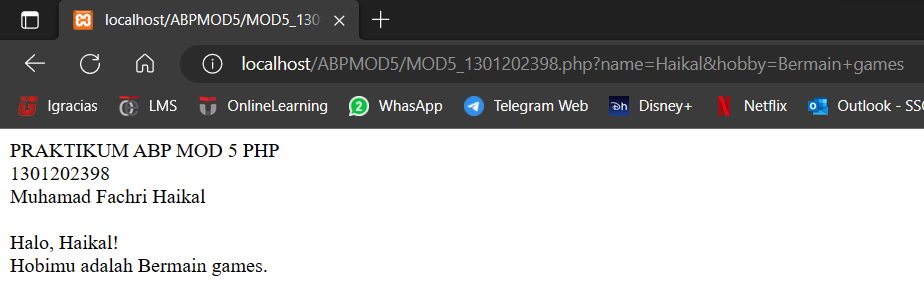
Perulangan pada PHP umumnya juga mirip dengan Bahasa pemrograman lain seperti contoh di atas terdapat 4 loop yang digunakan yaitu, FOR, WHILE, DO-WHILE, dan FOREACH.

* 1. **Function**

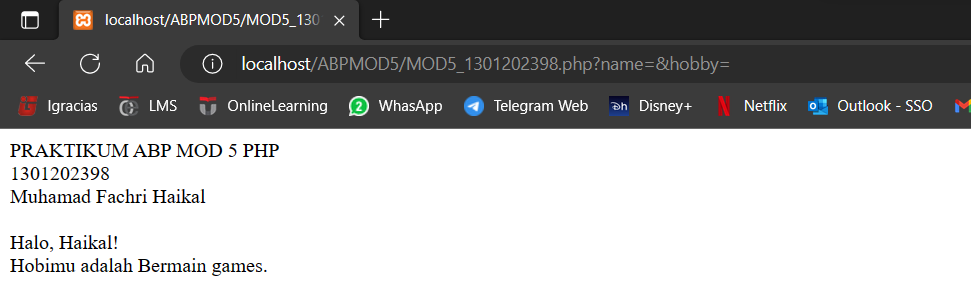
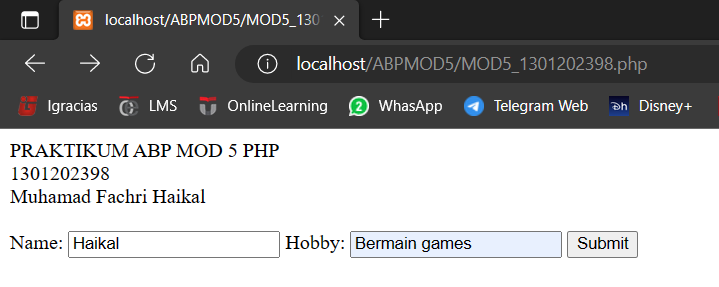
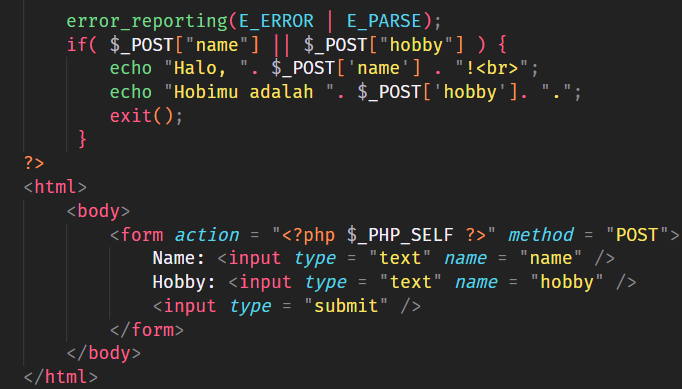
Pada PHP juga terdapat function seperti contoh di atas yaitu function untuk menentukan suatu bilangan termasuk genap atau ganjil.

* 1. **Array**

Pada PHP terdapat array untuk menyimpan kumpulan data dalam 1 struktur data. Array pada PHP dapat menggunakan indeks angka seperti arr1 dan juga indeks string seperti arr2 di atas.

* 1. **GET dan POST**
     1. **Metode GET**

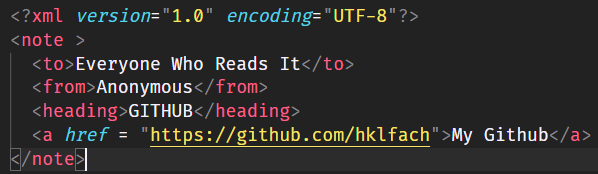
Metode Get digunakan untuk mendapatkan data atau nilai dari form yang bisa dilihat langsung pada URL. Contoh kode di atas adalah mendapatkan data nama dan hobi kemudian ditampilkan hasilnya.

* + 1. ******Metode POST**

Metode POST digunakan untuk mendapatkan data atau nilai dari form tetapi tidak bisa dilihat secara langsung pada URL sehingga lebih aman. Contoh kode di atas adalah mendapatkan data nama dan hobi kemudian ditampilkan hasilnya.

* 1. **XML**
     1. **Pengertian XML**

XML atau Extensible Markup Language adalah sebuah bahasa markup yang digunakan untuk membuat aturan sintaks dan struktur dokumen yang konsisten dan terstruktur. XML memungkinkan pengguna untuk membuat tag atau elemen yang dapat disesuaikan dalam dokumen untuk pertukaran data antar sistem dan platform yang berbeda. XML sering digunakan dalam berbagai aplikasi seperti web, basis data, dan sistem manajemen dokumen, serta digunakan dalam pertukaran data antar sistem dan platform yang berbeda.

* + 1. **Sintaks XML**

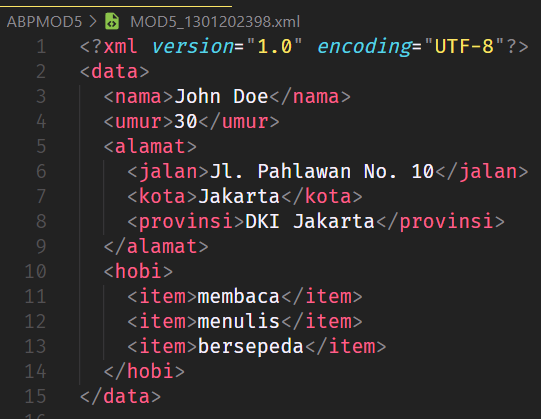
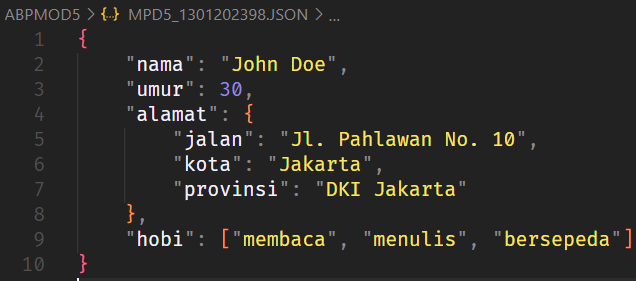
Pendeklarasian XML ada pada line paling atas . Secara umum XML memiliki kemiripan dengan HTML yaitu sama-sama menggunakan tag dan elemen, seperti contoh di atas yaitu <note> . XML juga dapat memiliki atribut seperti href pada tag <a>.

* 1. **JSON**
     1. **Pengertian JSON**

JSON adalah sebuah format data yang ringan dan mudah dimengerti oleh mesin serta manusia, dan digunakan untuk pertukaran data antar aplikasi, terutama di dalam pengembangan aplikasi web. Data JSON terdiri dari pasangan kunci-nilai yang dipisahkan oleh titik dua dan dipisahkan oleh koma. Data JSON dimulai dengan tanda kurung kurawal dan diakhiri dengan tanda kurung kurawal. Contoh data JSON sederhana termasuk pasangan kunci-nilai seperti nama dan umur, alamat yang berisi objek lain dan hobi yang berisi array.

* + 1. **Perbedaan JSON dan XML**

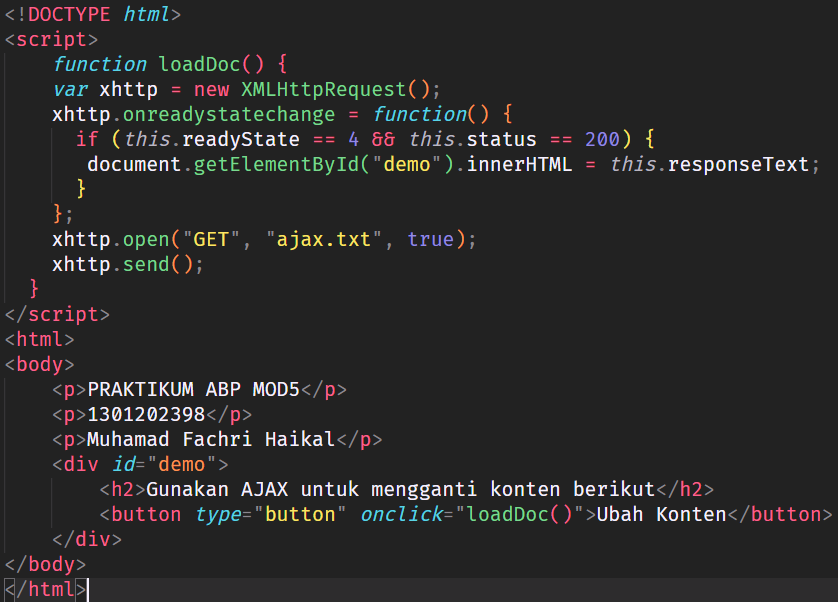
Berikut adalah beberapa perbedaan antara JSON dan XML:

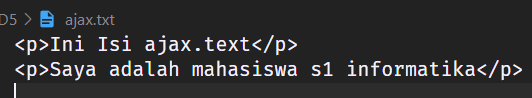
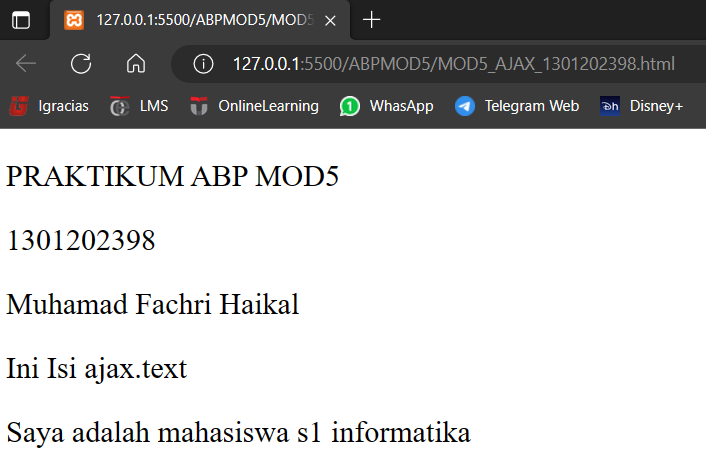
1. Struktur data: JSON menggunakan struktur data berbasis pasangan kunci-nilai (key-value pairs), sedangkan XML menggunakan struktur data berbasis pohon hierarki (tree-based).
2. Ringan atau berat: JSON biasanya lebih ringan (lightweight) dan lebih mudah dimengerti oleh manusia dan mesin, sedangkan XML lebih berat dan kompleks.
3. Suatu objek pada JSON dinotasikan dengan simbol kurung kurawal ({}) sedangkan kumpulan dari beberapa objek dapat dihimpun dalam notasi kurung siku ([]).
   1. **AJAX**
      1. **Apa Itu AJAX**

AJAX (Asynchronous JavaScript and XML) adalah teknologi pengembangan web yang memungkinkan komunikasi antara aplikasi web dan server tanpa memuat ulang halaman web. Dalam penggunaannya, AJAX menggabungkan bahasa pemrograman JavaScript dan teknologi XML atau format data lainnya seperti JSON, untuk mengirim permintaan ke server secara asinkron dan menerima responsnya tanpa harus memuat ulang halaman.

* + 1. **Cara Kerja AJAX**

Berikut adalah cara kerja AJAX dalam pengembangan aplikasi web:

1. User mengirim permintaan: Ketika pengguna melakukan tindakan pada halaman web, seperti mengklik tombol atau mengisi form, maka permintaan akan dikirim ke server.
2. XMLHttpRequest (XHR) dibuat: Setelah permintaan dikirim, JavaScript akan membuat objek XMLHttpRequest (XHR) yang bertanggung jawab untuk mengirim permintaan ke server dan menerima responsnya.
3. Permintaan dikirim ke server: Objek XHR mengirim permintaan ke server dengan menggunakan metode HTTP, seperti GET atau POST, dan menyertakan data yang diperlukan dalam permintaan.
4. Server merespons permintaan: Server akan memproses permintaan yang diterima dari objek XHR dan mengirimkan responsnya ke objek XHR.
5. Respons ditampilkan di halaman web: Setelah menerima respons dari server, objek XHR akan memperbarui halaman web dengan informasi yang diterima, seperti menambahkan data baru atau memperbarui data yang sudah ada.
   * 1. **Event Handling**

****