

Pengenalan Web Server & NodeJs

Sio Jurnal Pipin, S.Kom.



Hosting

- Tempat penyimpanan data atau file computer
- Memiliki bentuk fisik yang biasa disebut dengan server. Seperti gambar, video, audio dan sebagainya.
- File-file tersebut akan disimpan dalam sebuah tempat yang dinamakan server hosting.
- Server hosting merupakan tempat penyimpanan data, dimana nantinya server akan menampilkan data-data tersebut pada web client (browser) seperti Chrome, Mozilla, Safari, dan Opera.

Jenis Server Hosting

- **Share Hosting**

Dapat digunakan secara bersama - sama atau disebut juga di sharing.

Dalam server ini biasanya terdapat beberapa nama domain dengan kepemilikan yang berbeda.

- **Virtual Private Server**

Merupakan sebuah server yang dibagi - bagi menjadi sebuah virtual mesin, jadi dalam satu computer server terdapat beberapa system operasi server, tentunya secara kepemilikan antara system operasi satu dengan yang lainnya berbeda.

Jenis Server Hosting

- **Dedicated Server**

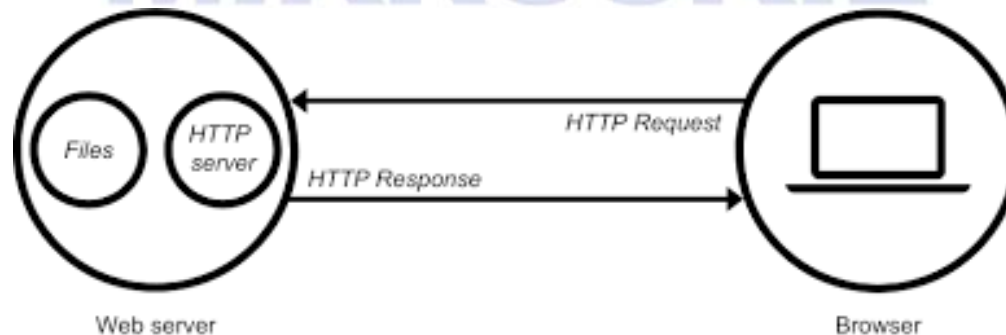
Server yang satu ini biasanya digunakan untuk penyimpanan aplikasi yang lebih besar dari pada Share hosting maupun VPS, jadi pengguna menyewa secara keseluruhan dari computer server yang disediakan oleh webhost.

- **Colocation Server**

Sebuah server yang digunakan untuk keperluan web hosting

Web Server

- Software yang memberikan layanan data yang mempunyai fungsi untuk menerima permintaan HTTP atau HTTPS yang dikirim oleh klien melalui web browser dan mengirimkan Kembali hasilnya dalam bentuk halaman web (HTML / JS).
- Berguna sebagai tempat aplikasi web dan sebagai penerima request dari client.



Node.js

- Sebuah platform software yang dipakai untuk membangun aplikasi – aplikasi serverside yang fleksibel di sebuah jaringan aplikasi.
- Menggunakan javascript sebagai basis bahasa pemrogramannya.
- Memiliki skalabilitas yang sangat tinggi, berbasis **event** (event driven programming), dan memiliki konsep **asynchronous** yang sangat bagus sehingga membuatnya ringan dan efisien.

Node.js

- Platform perangkat lunak yang berjalan pada sisi-server dan aplikasi jaringan.
- Platform ini terdiri dari 2 hal, yaitu **runtime environment** dan **script library**.
- Memiliki pustaka **server HTTP** sendiri sehingga memungkinkan untuk menjalankan webserver tanpa menggunakan program webserver seperti apache atau lighttpd.
- Menggunakan Engine Javascript dari Google bernama V8

Install Node.js

- Download Node.js versi LTS di <https://nodejs.org/en/download/>
- Install dengan klik “**Next**” hingga proses **instalasi** dimulai.
- Verifikasi Instalasi menggunakan command prompt atau PowerShell dengan perintah:
node -v dan **npm -v**

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.19042.844]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\SIO>node -v
v14.15.3

C:\Users\SIO>npm -v
6.14.9
```


Module HTTP Node.js

- Module yang digunakan untuk **membuat server**
- Modul HTTP membuat server HTTP yang dengan port server dan memberikan respons kembali ke klien.
- Dapat digunakan untuk mengambil **URL routing**, dan **Query String**.
- **Dokumentasi:** <https://nodejs.org/api/http.html>

Membuat Web Server dengan HTTP

- Buat file **index.js** pada folder M01-HttpModule
- Edit file **index.js** seperti gambar di bawah
- Pada folder M01-HttpModule run terminal / CMD dengan perintah **node index.js**

```
SIQ@DESKTOP-U2DMK91 MINGW64 ~/OneDrive - mikroskil.ac.id/Mikroskil/Mata Kuliah/6. Pemrograman Web Lanjutan/Webserver/M01-HttpModule  
$ node index.js
```

```
index.js x  
M01-HttpModule > index.js > ...  
1  const http = require("http");  
2  
3  //buat object server dengan listen port 3000  
4  http  
5  .createServer(function (req, res) {  
6    //Respon ke browser / client  
7    res.write("Module HTTP Mobile and Web");  
8  
9    //Respon diakhiri  
10   res.end();  
11 }  
12 ).listen(3000);  
13
```

Buka browser dengan alamat
Localhost:3000

← → ↻ ⓘ localhost:3000

Module HTTP Mobile and Web

HTTP Header – Jenis Konten

- Fungsi `writeHead(status_code, server_response)` memiliki 2 parameter, yakni:
 1. HTTP Status code, bila nilainya 200 itu berarti OK. Nilai status code lainnya adalah 400 yang berarti Bad Request, dll | Baca lainnya <https://developer.mozilla.org/id/docs/Web/HTTP/Status>.
 2. Server Response/Response header yang berisi sebuah nilai keterangan seperti **'Content-Type': 'text/html'**. | baca lainnya disini <https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers>

Jenis lainnya seperti:

```
response.writeHead(200, {'Content-Type': 'application/json'}); // untuk JSON
```

```
response.writeHead(200, {'Content-Type': 'application/pdf'}); // untuk PDF
```

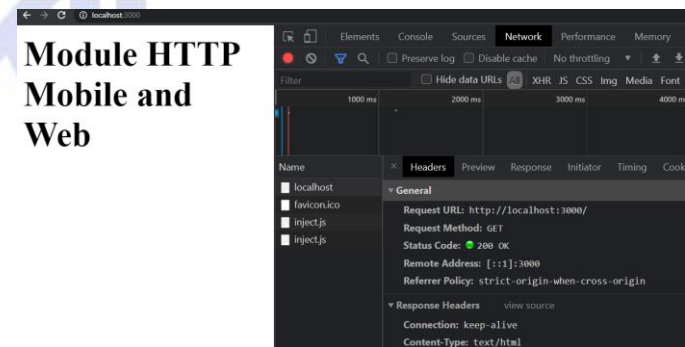
```
response.writeHead(200, {'Content-Type': 'application/xml'}); // untuk XML
```

HTTP Header - WriteHead

- Tambahkan **baris 7** dan ubah **baris 11** dengan HTML **H1** seperti pada gambar.
- Stop server di CMD dengan **CTRL + C**, lalu jalankan ulang server dengan perintah **node index.js**

```
index.js x
M01-HttpModule > .\index.js > ...
1  const http = require("http");
2
3  //buat object server dengan listen port 3000
4  http
5  .createServer(function (req, res) {
6    //http Header
7    res.writeHead(200, { "Content-Type": "text/html" });
8
9    //Respon ke browser / client
10   res.write("<H1>Module HTTP Mobile and Web</H1>");
11
12   //Respon diakhiri
13   res.end();
14 }
15 .listen(3000);
16
```

Cek dengan inspek element, pada **Bagian Network**, pilih **localhost**. Terdapat status code **200** dan **Content-Type: Text/html**



Latihan

- Install NodeJs pada perangkat kamu
- Buatlah web server menggunakan dengan listen post 3400 untuk menampilkan pesan dengan formatting HTML.

