**LAPORAN HASIL PRAKTIKUM**

**PEMROGRAMAN WEBSITE DAN MOBILE I**

****

**NAMA : MUHAMMAD ARIFIN**

**NIM : 193020503025**

**KELAS : A**

**MODUL : IV**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS PALANGKA RAYA**

**2021**

**BAB I**

**TUJUAN DAN LANDASAN TEORI**

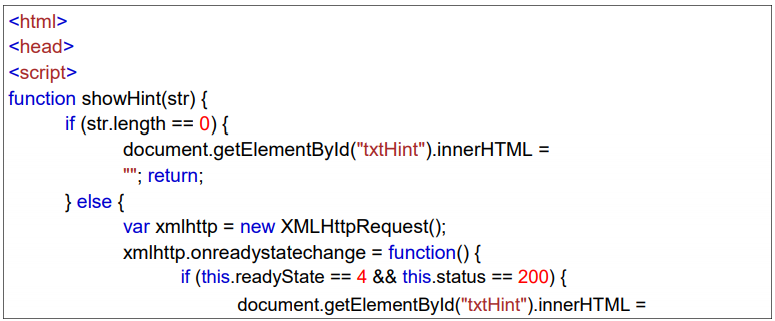
1. **Tujuan Praktikum**
   1. Mahasiswa mampu membuat program dengan menggabungkan fitur PHP dan JavaScript.
   2. Mahasiswa mampu membuat program web yang dinamis.
2. **Landasan Teori**

AJAX digunakan untuk membuat halaman web yang cepat dan dinamis. AJAX mengizinkan halaman web diperbaharui secara asinkron oleh pertukaran data dalam jumlah kecil dengan aktifitas server dibelakang layar. Jadi, pembaharuan halaman web dilakukan tanpa merubah seluruh tampilan halaman. Halaman web yang klasik (tidak menggunakan AJAX) harus memuat seluruh halaman jika kontennya harus berubah. Contoh aplikasi penggunaan AJAX adalah seperti Google Maps, Gmail, Youtube, dan Tab Facebook. AJAX didasari oleh Standar internet dan menggunakan kombinasi dari:

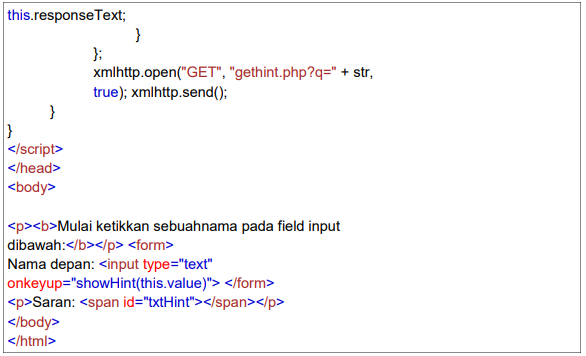
* Objek XMLHttpRequest (untuk merubah data secara asinkron dengan sebuah server)
* JavaScript/DOM (untuk menampilkan atau berinteraksi dengan informasi)
* CSS (untuk memodelkan data)
* XML (sering digunakan sebagai format pengiriman data)

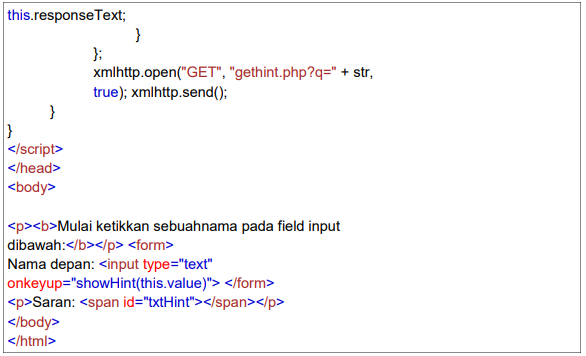
DOM adalah singkatan dari Document Object Model.

Kode program berikut adalah contoh penggunaan AJAX untuk membuat saran otomatis pada text input:



**Gambar 1.1** HTML.

** Gambar 1.2** HTML.

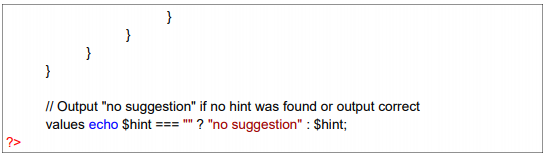
 **Gambar 1.3** HTML.



**Gambar 1.4** HTML.

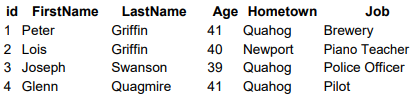


**Gambar 1.5** HTML.



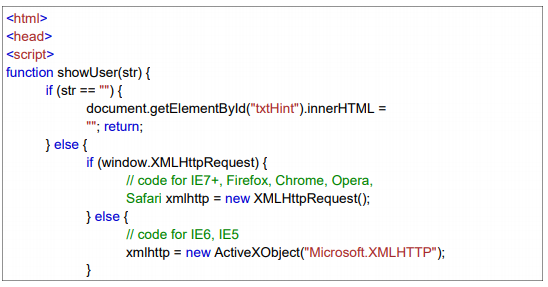
**Gambar 1.6** HTML.

AJAX dan MySQL AJAX bisa digunakan untuk komunikasi interaktif dengan sebuah database. Dibawah ini akan diberikan contoh kode program untuk mengakses database menggunakan AJAX, namun sebelum itu, perlu dibuat tabel “user” pada database “my\_db” dan diisi dengan data sebagai berikut:



**Gambar 1.7** Database.

Berikut adalah contoh program tersebut:



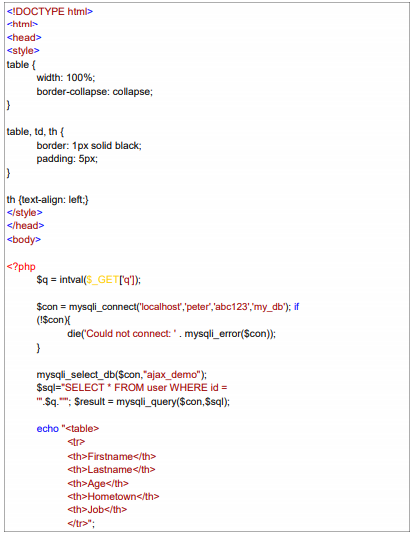
**Gambar 1.8** HTML dan JAVA.



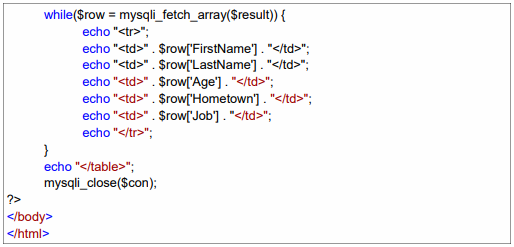
**Gambar 1.9** HTML dan JAVA.

Penjelasan kode: Pertama, memeriksa apakah ada data orang yang terpilih. Jika tidak ada maka (str == “”), bersihkan kontent dari txtHint dan keluar dari fungsi. Jika ada data orang terpilih, maka akan melakukan hal berikut:

* Membuat objek XMLHttpRequest.
* Membuat fungsi untuk dieksekusi ketika jawaban server adalah siap (ready).
* Kirim permintaan off untuk sebuah file pada server.
* Memberitahukan bahwa sebuah parameter (q) ditambahkan ke URL (dengan kontent dropdown list). Contoh program diatas menggunakan file “getuser.php”. File tersebut berisi kode program untuk melakukan query pada database MySQL, Kode program dari file tersebut adalah sebagai berikut:



**Gambar 1.10** HTML dan JAVA.

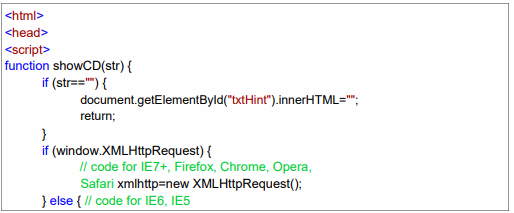


**Gambar 1.11** HTML dan JAVA.

Penjelasan dari kode program: Ketika query dikirimkan dari JavaScript ke file PHP, hal-hal berikut terjadi:

* PHP membuka koneksi ke server MySQL.
* Menemukan data orang.
* Tabel HTML dibuat, menambahkan data ke tabel tersebut, dan mengirimkannya kembali ke placeholder “txtHint”.

AJAX dan XML AJAX bisa digunakan untuk komunikasi interaktif dengan file XML. Berikut adalah contoh program AJAX yang digunakan untuk menampilkan data XML:



**Gambar 1.11** AJAX dan XML.



**Gambar 1.12** AJAX dan XML.

Penjelasan kode program: Fungsi showCD() melakukan hal-hal berikut:

* Memeriksa apakah ada CD yang terpilih.
* Membuat sebuah objek XMLHttpRequest.
* Membuat fungsi yang akan dieksekusi ketika jawaban server adalah siap (ready).
* Mengirimkan permintaan off untuk sebuah file pada server.
* Memberitahukan bahwa sebuah parameter (q) ditambahkan ke URL (dengan kontent dari list dropdown).

Pada kode program tersebut, terdapat file “getcd.php”. Kode program dari file tersebut memuat dokumen XML “cd\_catalog.xml”, menjalankan query terhadap file XML dan mengembalikan hasilnya. File “cd\_catalog.xml” bisa didownload melalui link

https://www.w3schools.com/php/cd\_catalog.xml. Berikut adalah kode program dari file “getcd.php”:



**Gambar 1.13** Fungsi.

Ketika query CD dikirimkan dari JavaScript ke halaman PHP, hal-hal berikut terjadi:

* PHP membuat sebuah objek XML DOM.
* Menemukan semua element yang sesuai dengan nama yang dikirimkan dari JavaScript.
* Mengeluarkan output berupa informasi album (mengirimkan ke placeholder “txtHint”)

**AJAX – Pencarian Langsung (Live Search)**

AJAX bisa digunakan untuk membuat pencarian keyword yang lebih interaktif dan user-friendly. Berikut adalah contoh program untuk melakukan pencarian melalui query pada sebuah file XML “links.xml”. File ini bisa didownload pada link: https://www.w3schools.com/php/links.xml . Kode program dari contoh program pencarian adalah sebagai berikut:

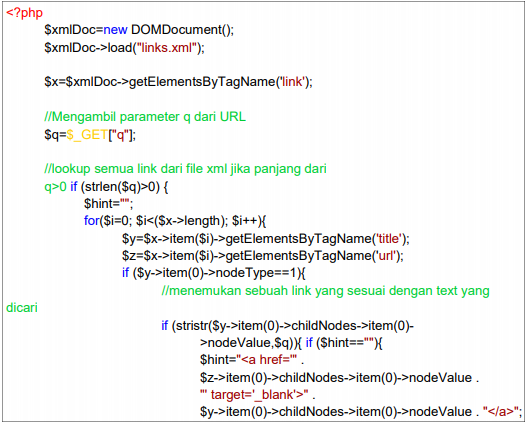


**Gambar 1.14** AJAX.

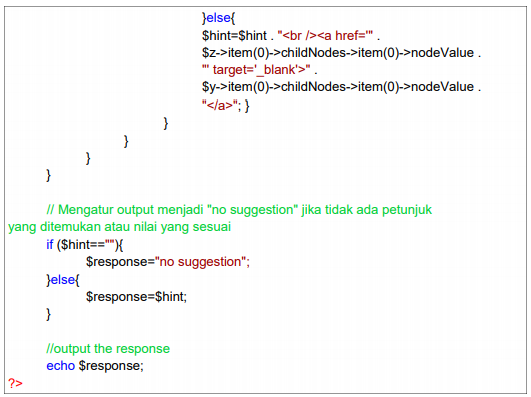
Penjelasan kode program: Jika field input kosong (str.length==0), fungsi akan membersihkan kontent dari placeholder livesearch dan keluar dari fungsi. Jika field input tidak kosong, fungsi showResult() melakukan hal-hal berikut:

* Membuat objek XMLHttpRequest.
* Membuat fungsi yang akan dieksekusi ketika jawaban server adalah siap (ready).
* Mengirimkan permintaan off untuk sebuah file pada server.
* Memberitahukan bahwa sebuah parameter (q) ditambahkan ke URL (dengan kontent field input)

Contoh kode program diatas menggunakan file “livesearch.php”. Kode program dari file tersebut adalah sebagai berikut:



**Gambar 1.15** AJAX.



**Gambar 1.16** AJAX.

Penjelasan dari kode program: Jika ada text yang dikirimkan dari JavaScript (strlen($q) > 0), hal-hal berikut akan dilakukan:

* Memuat file XML menjadi objek XML DOM baru.
* Perulangan melalui semua element untuk menemukan kata yang sesuai dengan text yang dikirimkan dari JavaScript.
* Menugaskan URL yang benar dan judul dalam variabel $response. Jika lebih dari satu yang sesuai, semua yang sesuai akan ditambahkan ke variabel tersebut.
* Jika tidak ada yang sesuai, variabel $response akan dimasukkan nilai text “no suggestion”.

**AJAX Polling**

AJAX juga baik digunakan untuk menampilkan secara langsung hasil polling. Berikut adalah contoh kode program untuk melakukan polling

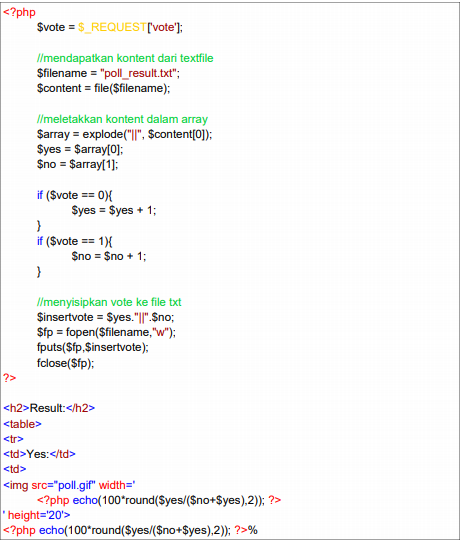


**Gambar 1.17** AJAX.

Fungsi getVote() melakukan hal-hal sebagai berikut:

* Membuat sebuah objek XMLHttpRequest
* Membuat fungsi untuk dieksekusi ketika jawaban server adalah siap (ready).
* Mengirimkan permintaan off untuk sebuah file pada server.
* Memberitahukan bahwa parameter (vote) ditambahkan ke URL (dengan nilai opsi yes atau no)

Kode program diatas menggunakan file “poll\_vote.php”. Kode program dari file tersebut adalah sebagai berikut:



**Gambar 1.18** AJAX.

Nilai dikirim dari JavaScript dan hal-hal berikut akan terjadi:

* Mengambil konten dari file “poll\_result.txt”.
* Meletakkan konten ke dalam variabel dan menambahkan satu ke variabel yang terpilih.
* Menuliskan hasil ke file “poll\_result.txt”
* Mengeluarkan representasi grafis dari hasil polling.

File text (poll\_result.txt) adalah tempatn menyimpan data polling. Data tersebut disimpan dalam bentuk berikut: 0||0

Bilangan pertama merepresentasikan voting “Yes” sedangkan bilangan kedua merepresentasikan voting “No”. Jangan lupa untuk mengizinkan web browser merubah text file.

. Dalam contoh ini, kita memperhatikan file ini memproses data form. Jika pengguna ingin menggunakan file PHP lain untuk memproses data form, menggantinya dengan nama file yang dipilih, maka gunakan variabel super global $\_REQUEST untuk mengumpulkan field nilai input.

**BAB II**

**PEMBAHASAN**

**Tugas Praktikum**

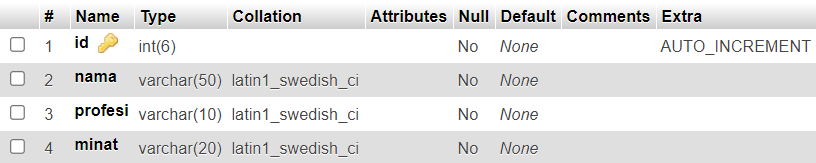
1. Buatlah program web untuk menyimpan data survei yang disimpan di dalam database. Carilah library javascript di internet untuk menampilkan grafik, kemudian rangkum data-data hasil survei kedalam bentuk grafik.
2. Misalkan anda seorang programmer yang diminta untuk membuat program web untuk melakukan pemungutan suara pada PEMILU. Jumlah calon yang akan dipilih adalah lima orang. Buatlah program web untuk pemilihan tersebut, dimana setiap pemilih hanya bisa memilih satu kali.

**Tugas 1 : Database**

Dibuat 4 field yang digunakan untuk menampung beberapa data dari mahasiswa dan pelajar terkait minat atau hobi.

1. Field id digunakan untuk menampung data id dari peminat dengan tipe data int dengan maksimal 6 karakter sekaligus sebagai primary key.
2. Field nama digunakan untuk menampung data nama dari peminat dengan tipe data varchar dan maksimal 50 karakter.
3. Field profesi digunakan untuk menampung data jenis profesi dari peminat dengan tipe data varchar dan maksimal 10 karakter.
4. Field minat digunakan untuk menampung data jenis minat dari peminat dengan tipe data varchar dan maksimal 20 karakter.

Berikut adalah tampilan struktur tabel “peminat” dari database “dataminat” pada server localhost.



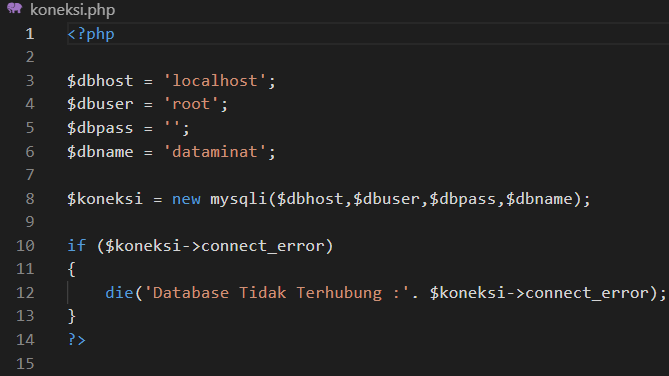
**Gambar 2.1** Tabel peminat.

**Tugas 1 : Project file**

**File koneksi.php**

Untuk mengkoneksikan php dengan database, dibuat terlebih dulu string beberapa string yang akan menyimpan data dari server yang digunakan, username dan password dari server tersebut, serta database yang akan dikoneksikan dengan file php. file php lalu dihubungkan dengan database menggunakan sintaks *mysqli\_connect* yang kemudian memanggil string yang sudah dibuat sebelumnya.

Terdapat sebuah kondisi if yang akan menampilkan sebuah pesan jika database tidak terhubung dengan file project. Berikut adalah full sintaks yang digunakan untuk mengkoneksikan php ke database.

****

**Gambar 2.2** File koneksi.

**File index.php**

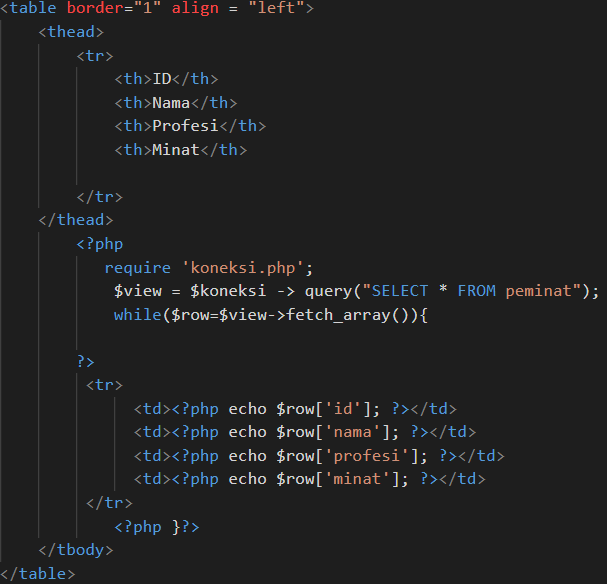
File ini merupakan home atau halaman pertama yang secara otomatis terbuka jika website survei diakses. Terdapat button link menuju halaman selanjutnya yaitu halaman survei yang ditampilkan melalui chart.

****

**Gambar 2.3** Button link ke chart.php.

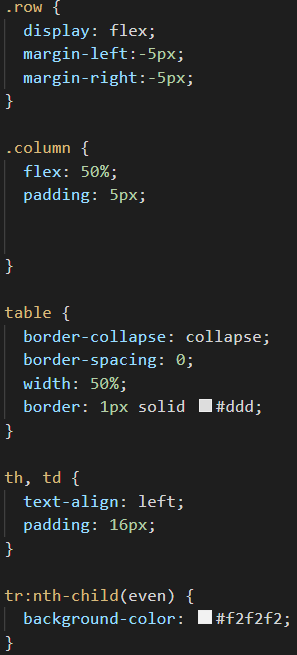
Data yang ada dalam database ditampilkan ke dalam bentuk tabel pada halaman website. Untuk menampilkan data yang ada pada database, terdapat *mysql\_query* yang terhubung pada tabel dalam database menggunakan query SELECT dengan perantara koneksi.

Data ditampilkan sesuai dengan tabel heading masing-masing, data ditampung dalam string data dan dimasukkan ke dalam tabel data menggunakan tag *<td>.* Berikut adalah sintaks yang digunakan untuk menampilkan data ke dalam bentuk tabel.

****

**Gambar 2.4** File index.

Berikut adalah sintaks yang digunakan untuk melakukan styling pada chart halaman index.



**Gambar 2.5** Styling tabel halaman index.

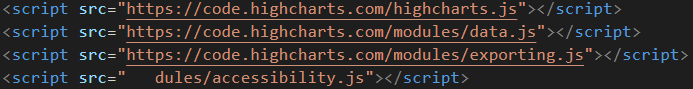
**File chart.php**

File ini merupakan halaman yang menampilkan data pada halaman sebelumnya ke dalam bentuk chart. Terdapat button link menuju halaman selanjutnya yaitu halaman survei yang ditampilkan melalui home.

****

**Gambar 2.6** Button link ke index.php.

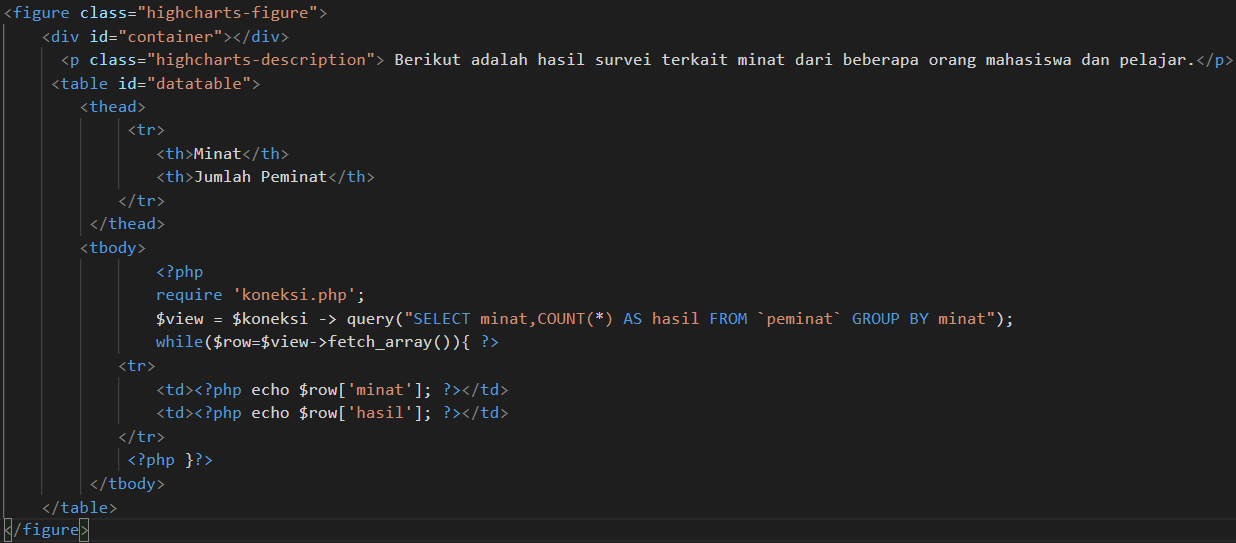
Chart di import dari luar atau dari website dengan javascript menggunakan sintaks *src*. Berikut adalah source dari chart yang digunakan pada halaman chart.php.

****

**Gambar 2.7** Source chart.

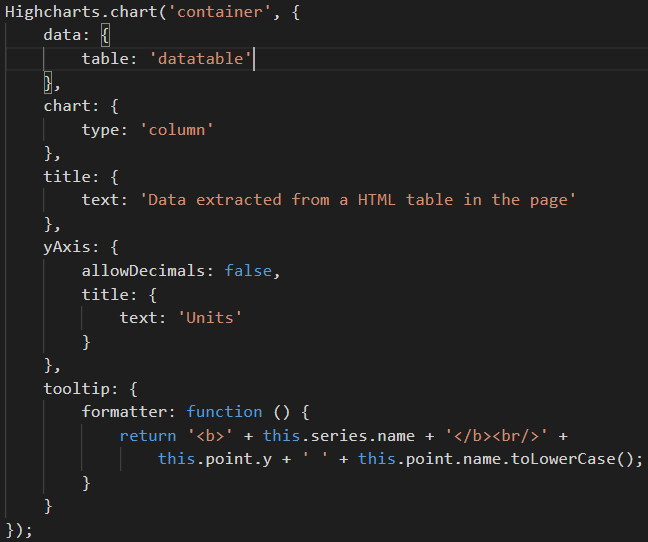
Data-data yang akan ditampilkan ke dalam chart diambil melalui database dengan perantara koneksi. Terdapat query sql *"SELECT minat,COUNT(\*) AS hasil FROM `peminat` GROUP BY minat"*. Query ini berfungsi untuk menampilkan jumlah dari minat masing-masing peminat. Penjumlahan dilakukan menggunakan fungsi COUNT yang hasilnya ditampilkan pada field hasil.

Selain dalam bentuk chart, data juga ditampilkan ke dalam bentuk tabel dimana terdapat 2 field yaitu minat dan jumlah peminat. Untuk field minat mengambil data dari field minat dan untuk field jumlah peminat mengambil data field hasil dari pemrosesan query sql sebelumnya. Pengambilan data dari field database menggunakan sintaks *<?php echo $row['nama\_field']; ?.*



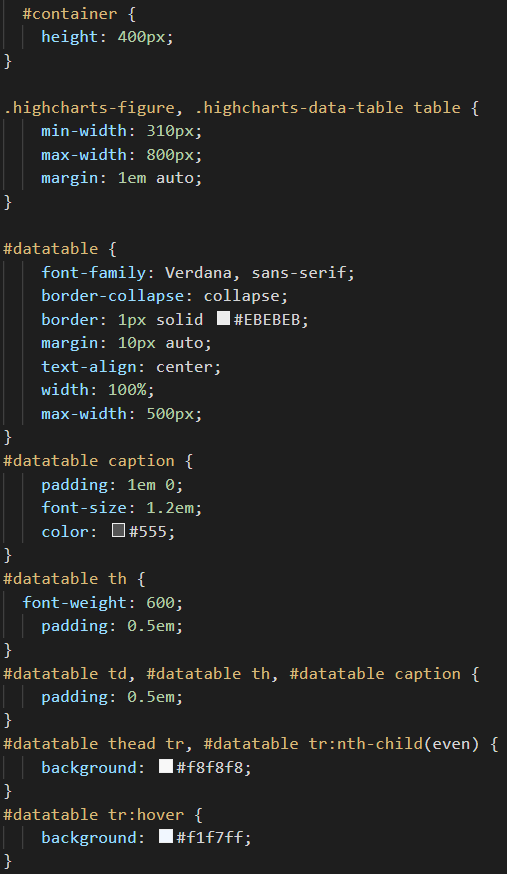
**Gambar 2.8** Chart dan tabel halaman chart.

Berikut adalah sintaks yang digunakan untuk melakukan styling pada chart halaman chart.



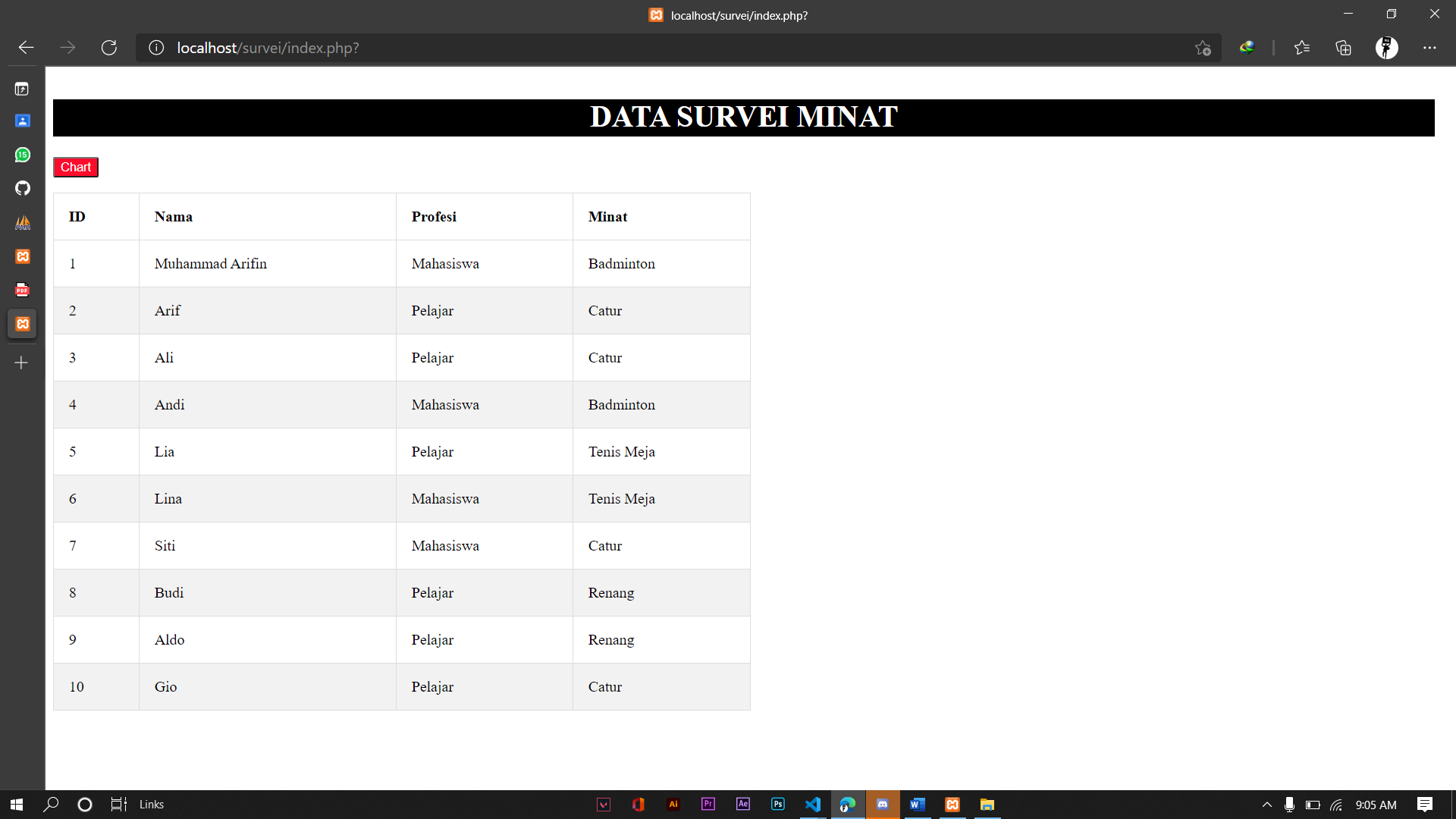
**Gambar 2.9** Styling chart halaman chart.

Berikut adalah sintaks yang digunakan untuk melakukan styling pada tabel halaman chart.

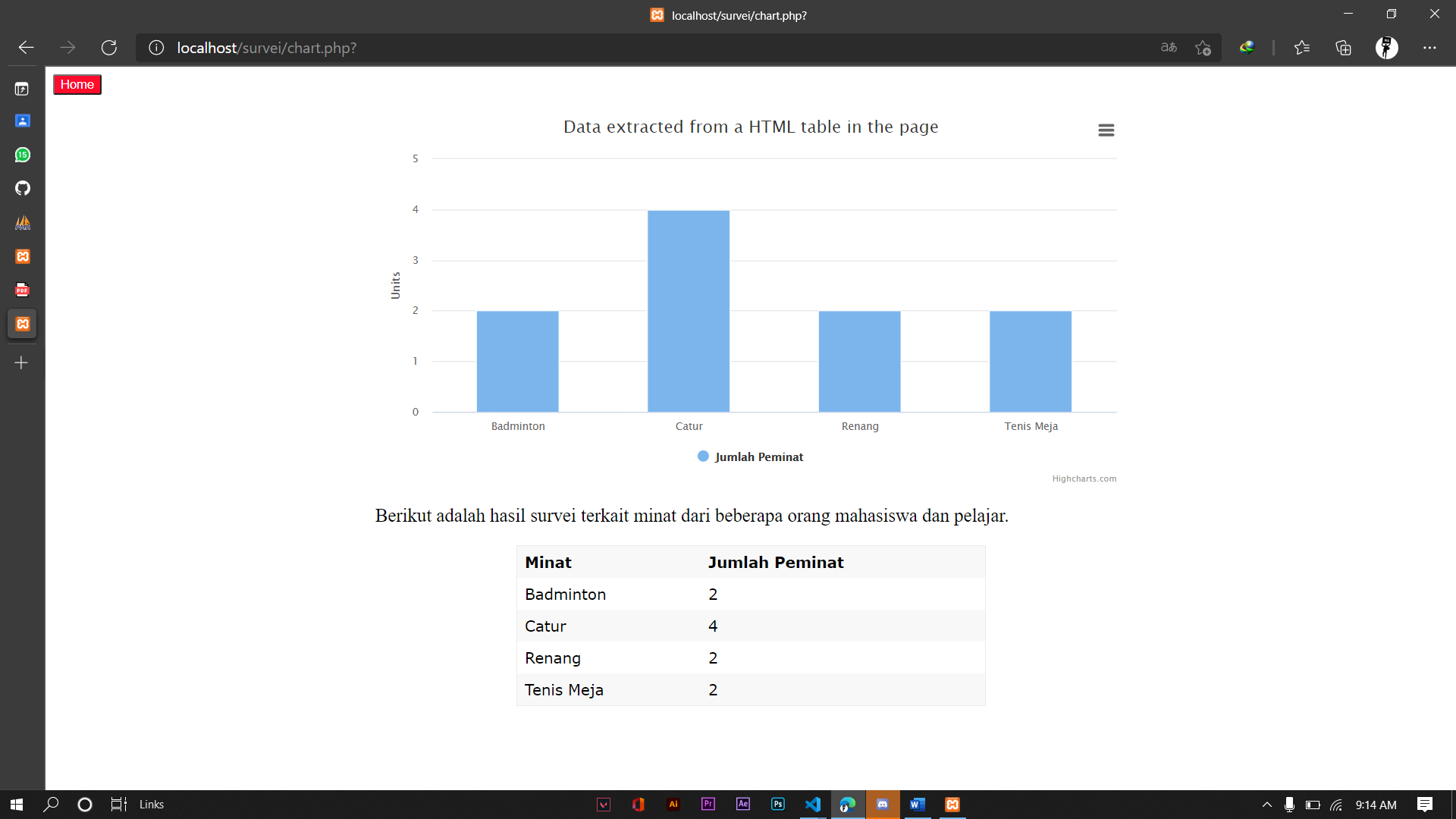


**Gambar 2.10** Styling tabel halaman chart.

Berikut adalah tampilan halaman index dan chart setelah selesai dibuat.



**Gambar 2.11** Halaman index.



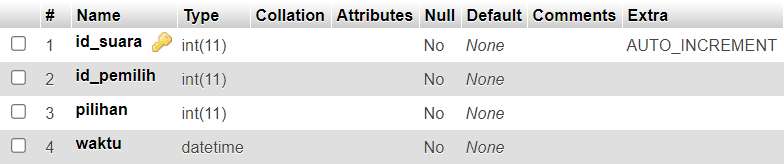
**Gambar 2.12** Halaman chart.

**Tugas 2 : Database**

Dibuat 4 field yang digunakan untuk menampung beberapa data dari mahasiswa dan pelajar terkait minat atau hobi.

1. Field id\_suara digunakan untuk menampung data identitas dari suara sebagai sebagai primary key yang mencegah duplikasi data voting dengan tipe data int dan maksimal 11 karakter.
2. Field id\_pemilih digunakan untuk menampung data identitas dari pemilih dengan tipe data int dan maksimal 11 karakter.
3. Field pilihan digunakan untuk menampung data calon yang akan dipilih dengan tipe data int dan maksimal 11 karakter.

Berikut adalah tampilan struktur tabel “peminat” dari database “dataminat” pada server localhost.

****

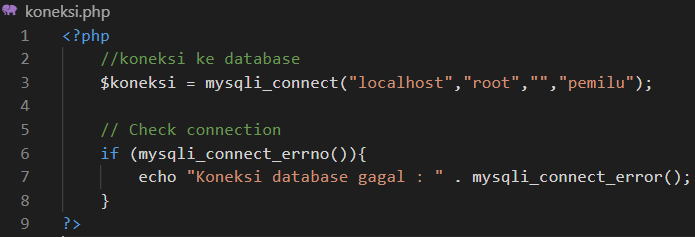
**Gambar 2.13** Tabel pemilu.

**Tugas 2 : Project file**

**File koneksi.php**

Untuk mengkoneksikan php dengan database, dibuat terlebih dulu string beberapa string yang akan menyimpan data dari server yang digunakan, username dan password dari server tersebut, serta database yang akan dikoneksikan dengan file php. file php lalu dihubungkan dengan database menggunakan sintaks *mysqli\_connect* yang kemudian memanggil string yang sudah dibuat sebelumnya.

Terdapat sebuah kondisi if yang akan menampilkan sebuah pesan jika database tidak terhubung dengan file project. Berikut adalah full sintaks yang digunakan untuk mengkoneksikan php ke database.



**Gambar 2.14** File koneksi.

**File index.php**

Berikut adalah sintaks yang digunakan untuk melakukan voting calon, dibuat kolom id\_pemilih untuk melakukan input id dari pemilih, dengan lebar 20% dari tanda “:”, dan tanda “:” itu sendiri berjarak 2% dari field input. Untuk input type atau jenis input pada field ini adalah number seperti pada database yang menggunakan tipe data int.

Terdapat 5 daftar calon ketua yang dibuat ke dalam option. Dengan adanya option, calon bisa dipilih dengan klik pada nama calon tanpa harus mengetikkan nama calon itu sendiri. jenis input pada field ini adalah submit yang berarti data akan dikirim jika button diklik.



**Gambar 2.15** Field input.

Data voting yang sudah di input akan ditampilkan pada tabel yang dibuat menggunakan sintaks seperti di bawah. Tabel memiliki *border* sebesar *1px* dengan *align* di posisi kiri atau *left*.

Data ditampilkan ke dalam tabel menggunakan query sql *"SELECT pilihan,COUNT(\*) AS hasil FROM `suara` GROUP BY pilihan"* dengan perantara koneksi. Query ini berfungsi untuk menampilkan jumlah dari total suara yang didapat masing-masing calon. Penjumlahan dilakukan menggunakan fungsi COUNT yang hasilnya ditampilkan pada field hasil.

Terdapat 2 field yaitu calon dan total. Untuk field calon mengambil data dari field pilihan dan untuk field total mengambil data field hasil dari pemrosesan query sql sebelumnya. Pengambilan data dari field database menggunakan sintaks *<?php echo $row['nama\_field']; ?.*



**Gambar 2.16** Tabel output.

Pada bagian ini, button kirim yang ada pada field input memiliki metode *$\_POST.* Data yang sudah di input pada field input membuat sebuah antrian data yang kemudian ditangkap menggunakan *$\_POST* dan dikirim ke database.

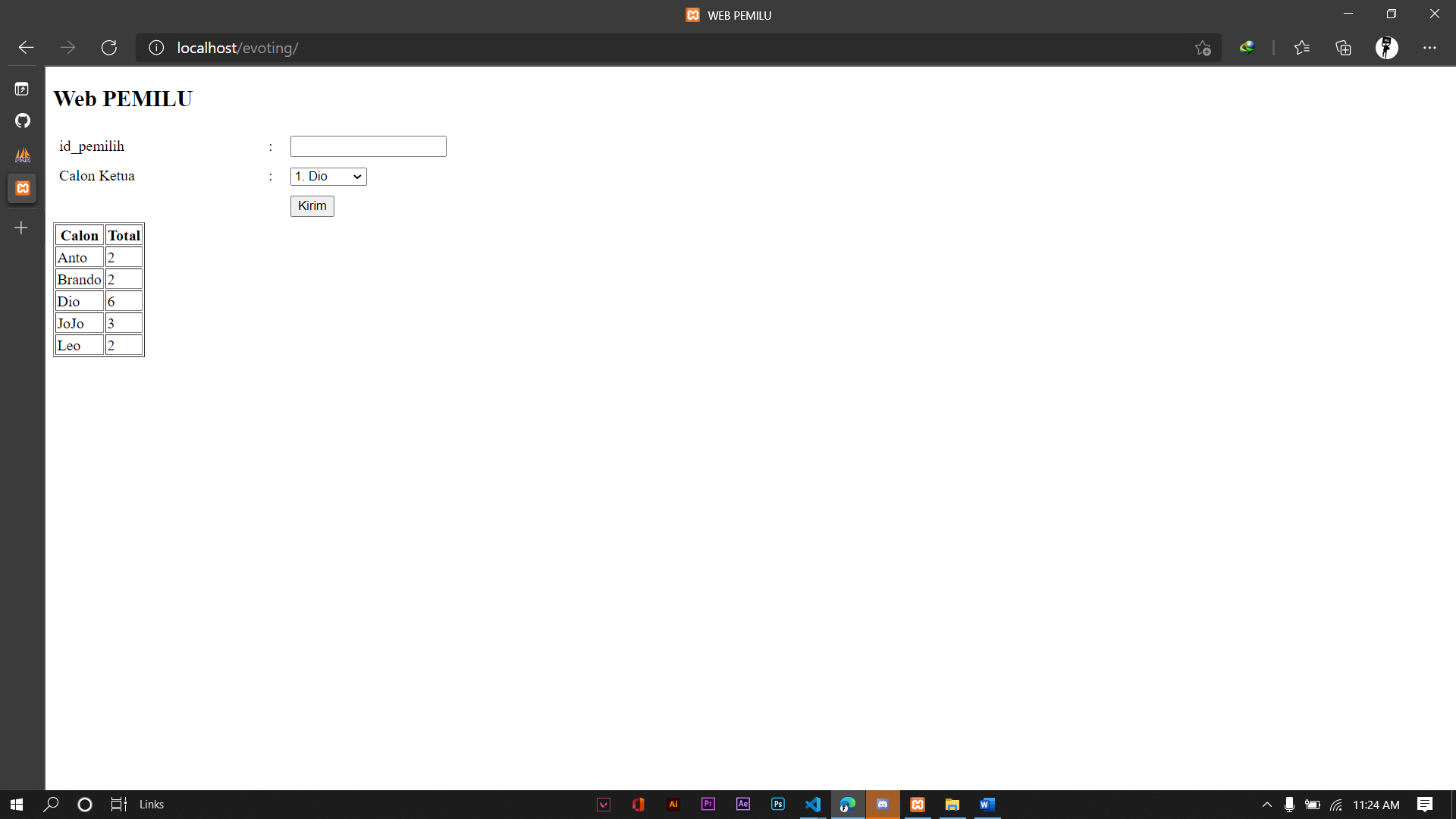
Data yang di input akan diperiksa menggunakan kondisi *if* yang jika *$data* bernilai kurang dari atau sama dengan 0 terpenuhi, maka input akan dilakukan karena data yang akan di input adalah data baru lalu dijalankan query sql *INSERT* ke dalam field-field yang ada pada database dan akan muncul alert atau pemberitahuan bahwa *“Data berhasil ditambah”*.

Jika kondisi *$data* bernilai lebih dari atau sama dengan 0, maka input tidak bisa dilakukan karena sudah ada data yang memiliki primary key serupa dengan input yang akan dikirim. Alert akan muncul untuk memberitahukan bahwa id sudah digunakan.



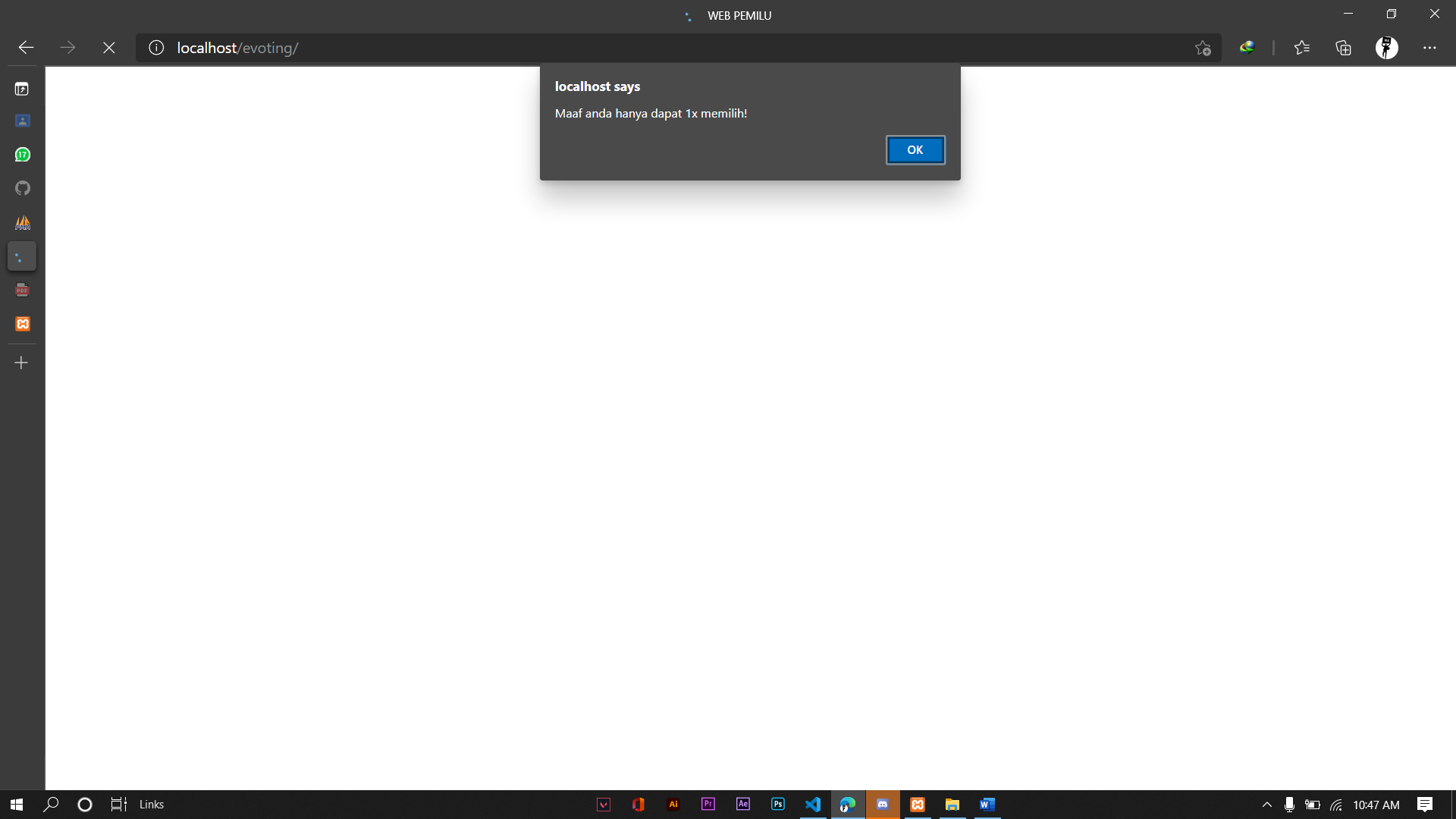
**Gambar 2.17** Metode input, kondisi, dan alert.

Berikut adalah tampilan halaman website setelah selesai dibuat.



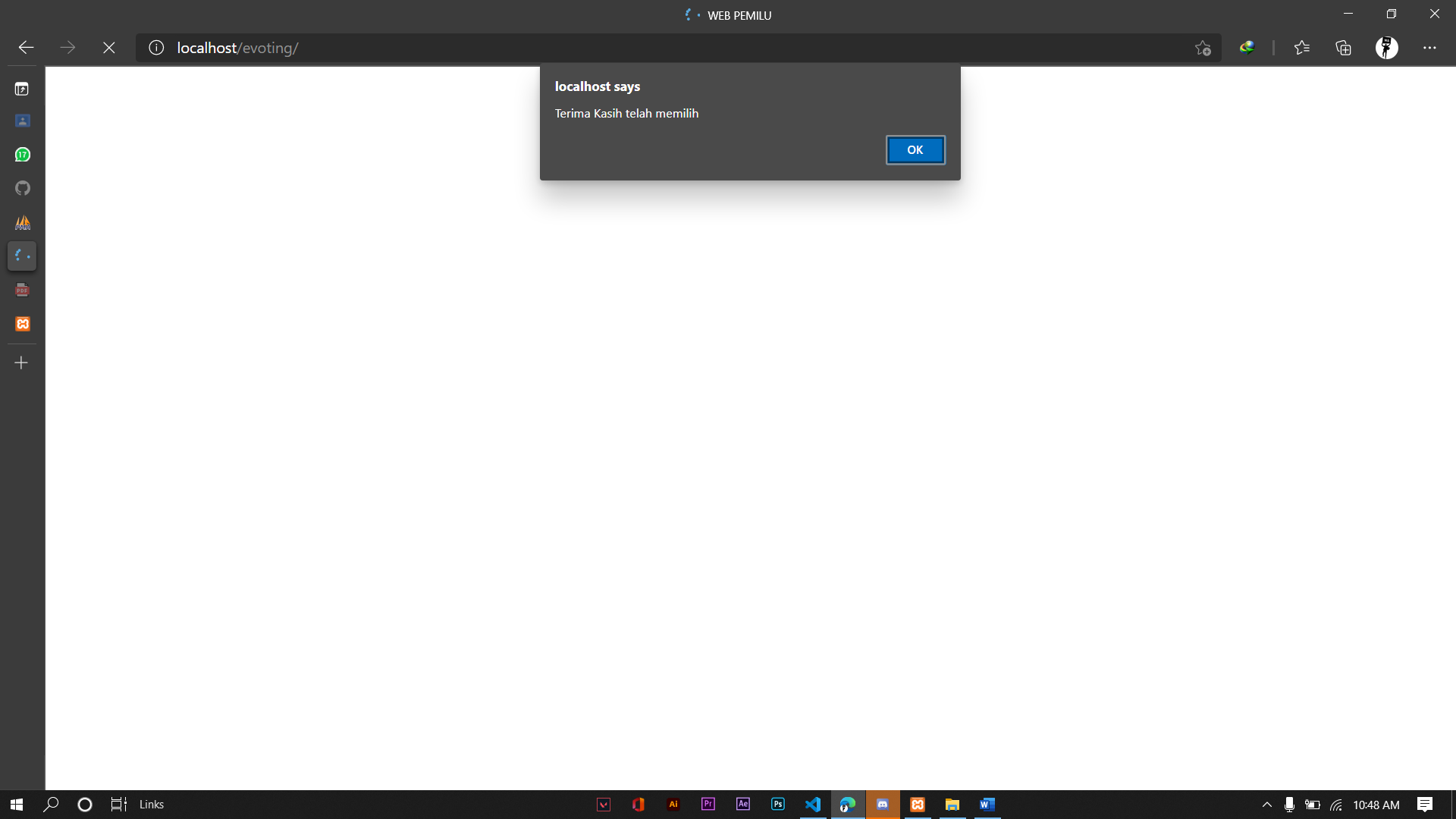
**Gambar 2.18** Halaman pemilihan.

Berikut adalah kondisi jika terjadi duplikasi id\_suara.



**Gambar 2.19** Input tidak bisa dilakukan.

Berikut adalah kondisi jika voting berhasil dilakukan.



**Gambar 2.20** Input bisa dilakukan.

**BAB III**

**KESIMPULAN**

Di dalam Website, SQL akan sangat erat hubunganya dengan PHP, karena PHP merupakan struktur pemrograman pada sisi server sehingga server membutuhkan koneksi ke dalam database, dan SQL digunakan untuk menghubungkan website (dengan struktur bahasa PHP) ke dalam Database menggunakan bahasa SQL.

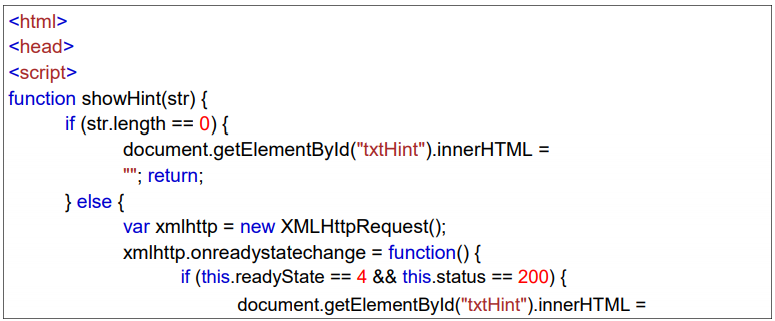
Untuk penggunaan javascript, javascript tidak dapat berdiri sendiri dan harus didasari oleh HTML atau PHP. Perbedaan Javascript dengan CSS adalah javascript mengatur logika seperti validasi untuk membuat tampilan website lebih dinamis dan CSS mengatur tampilan dari website tersebut seperti gambar, warna, font, dan lain-lain.

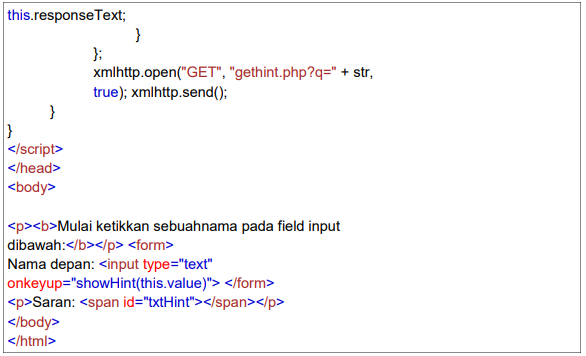
Pada praktikum kali ini, javascript digunakan untuk membuat alert dan untuk mengimport chart dari luar.

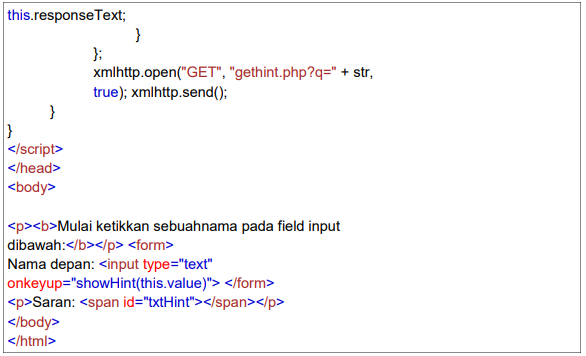
DAFTAR PUSTAKA

Praktikum, K. (n.d.). *MODUL PRAKTIKUM PEMROGRAMAN WEB I Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya*.

**LAMPIRAN**

**Gambar 1.1** HTML.

**Gambar 1.2** HTML.

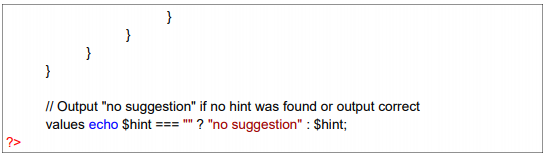
**Gambar 1.3** HTML.

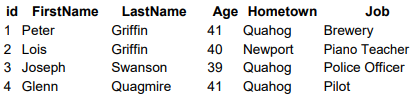


**Gambar 1.4** HTML.

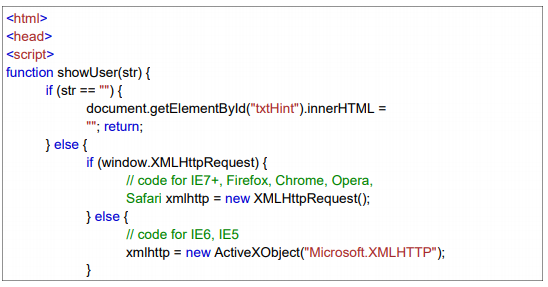


**Gambar 1.5** HTML.

**Gambar 1.6** HTML.



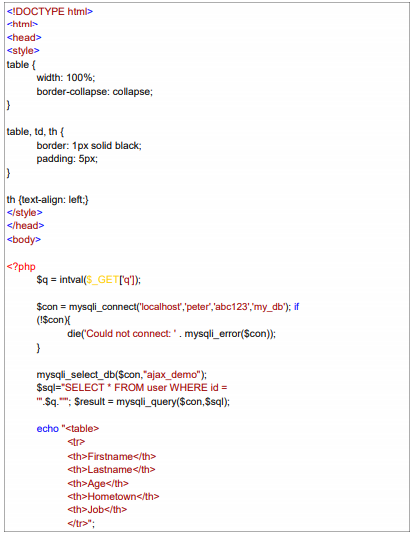
**Gambar 1.7** Database.



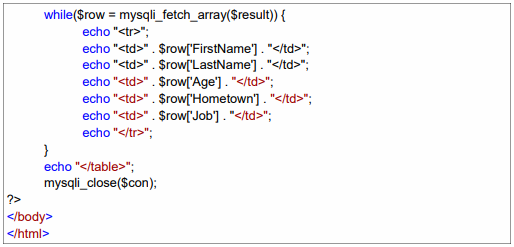
**Gambar 1.8** HTML dan JAVA.



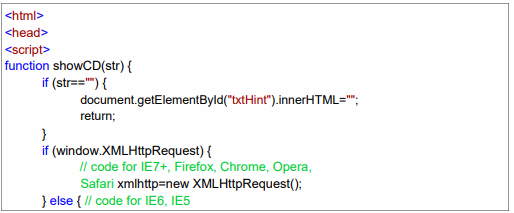
**Gambar 1.9** HTML dan JAVA.



**Gambar 1.10** HTML dan JAVA.



**Gambar 1.11** HTML dan JAVA.



**Gambar 1.11** AJAX dan XML.



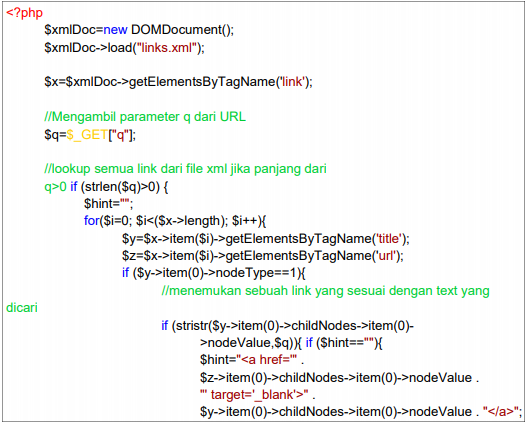
**Gambar 1.12** AJAX dan XML.



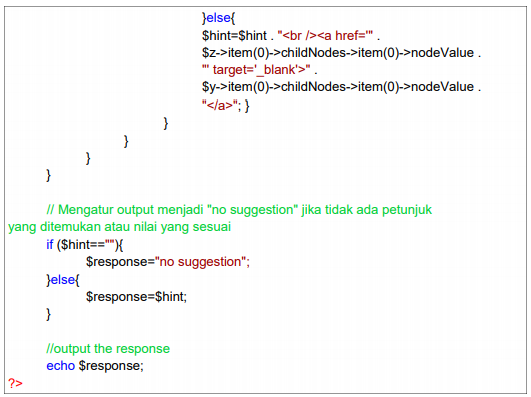
**Gambar 1.13** Fungsi.



**Gambar 1.14** AJAX.



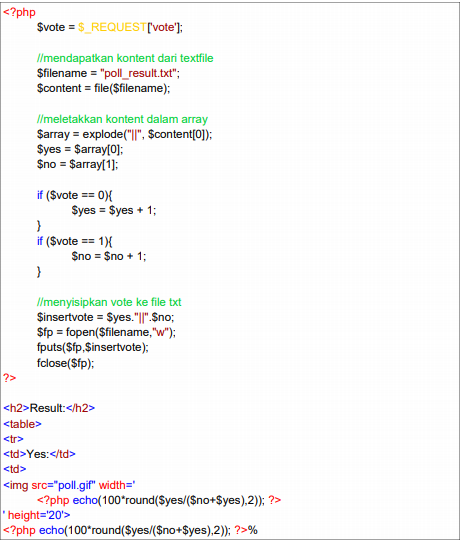
**Gambar 1.15** AJAX.



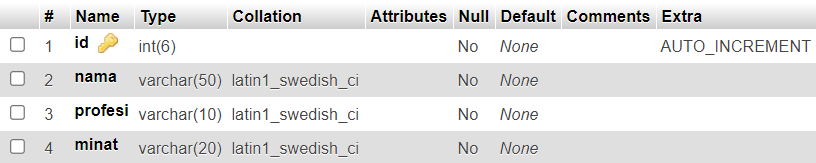
**Gambar 1.16** AJAX.



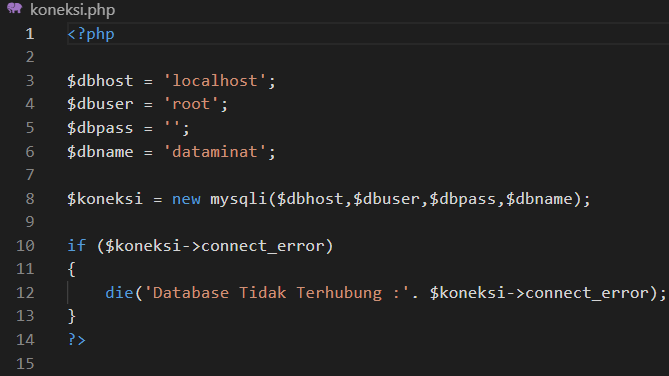
**Gambar 1.17** AJAX



**Gambar 1.18** AJAX.



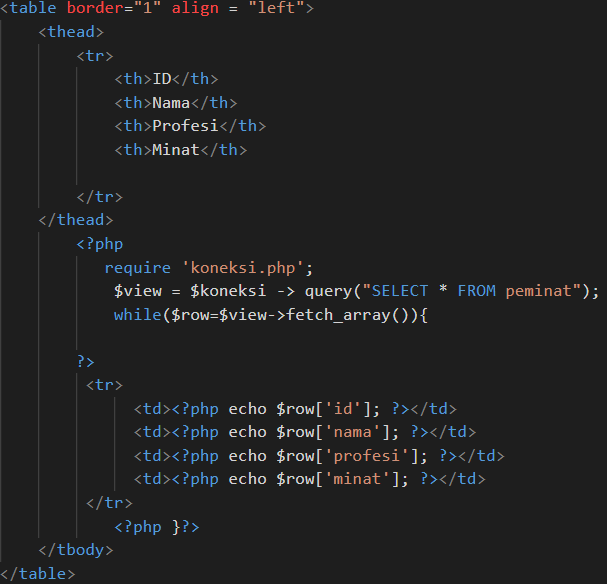
**Gambar 2.1** Tabel peminat.

****

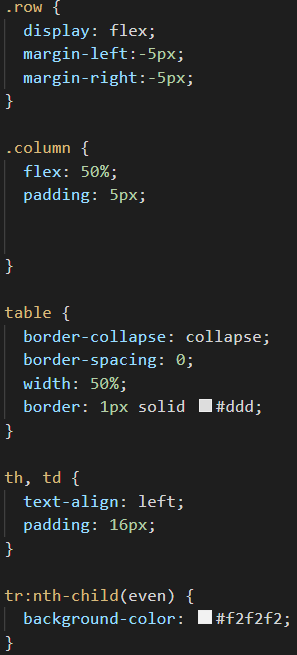
**Gambar 2.2** File koneksi.

****

**Gambar 2.3** Button link ke chart.php.

****

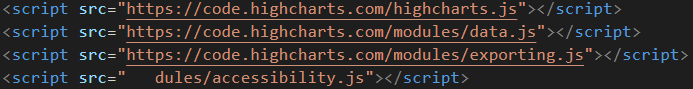
**Gambar 2.4** File index.



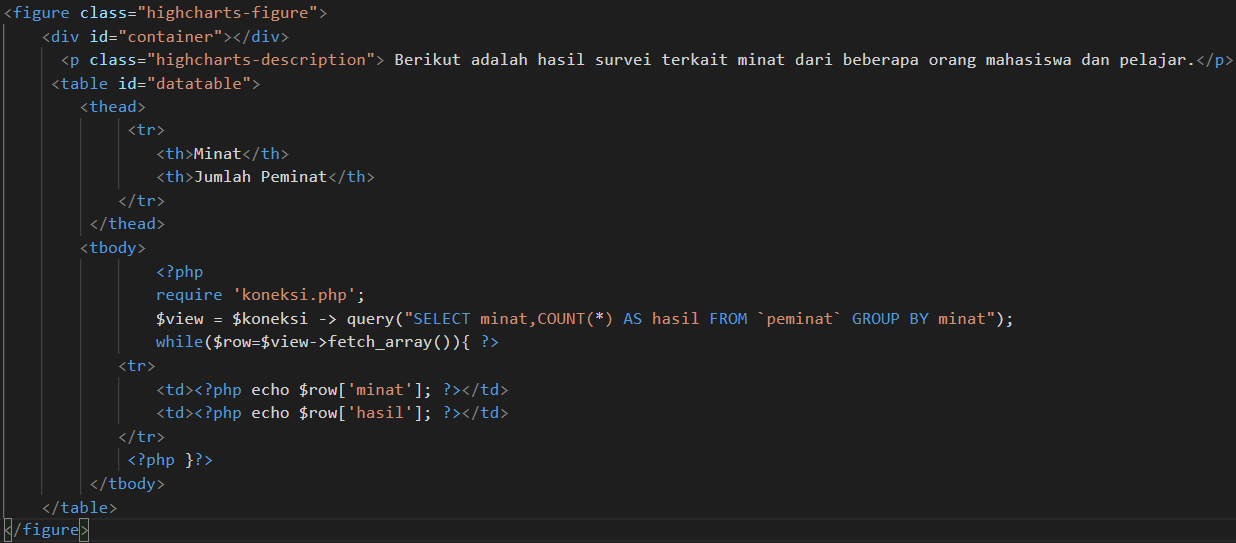
**Gambar 2.5** Styling tabel halaman index.

****

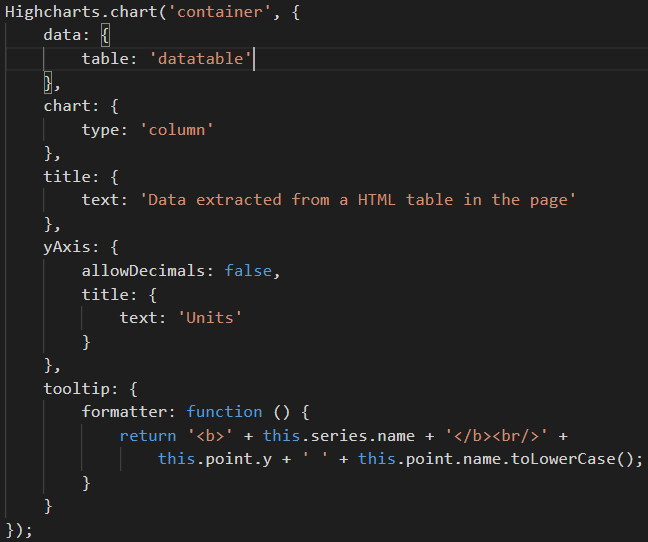
**Gambar 2.6** Button link ke index.php.

****

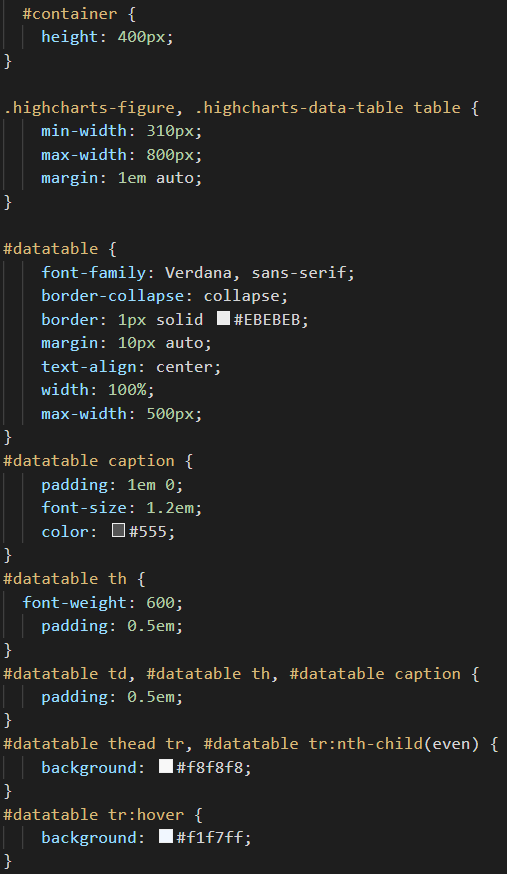
**Gambar 2.7** Source chart.



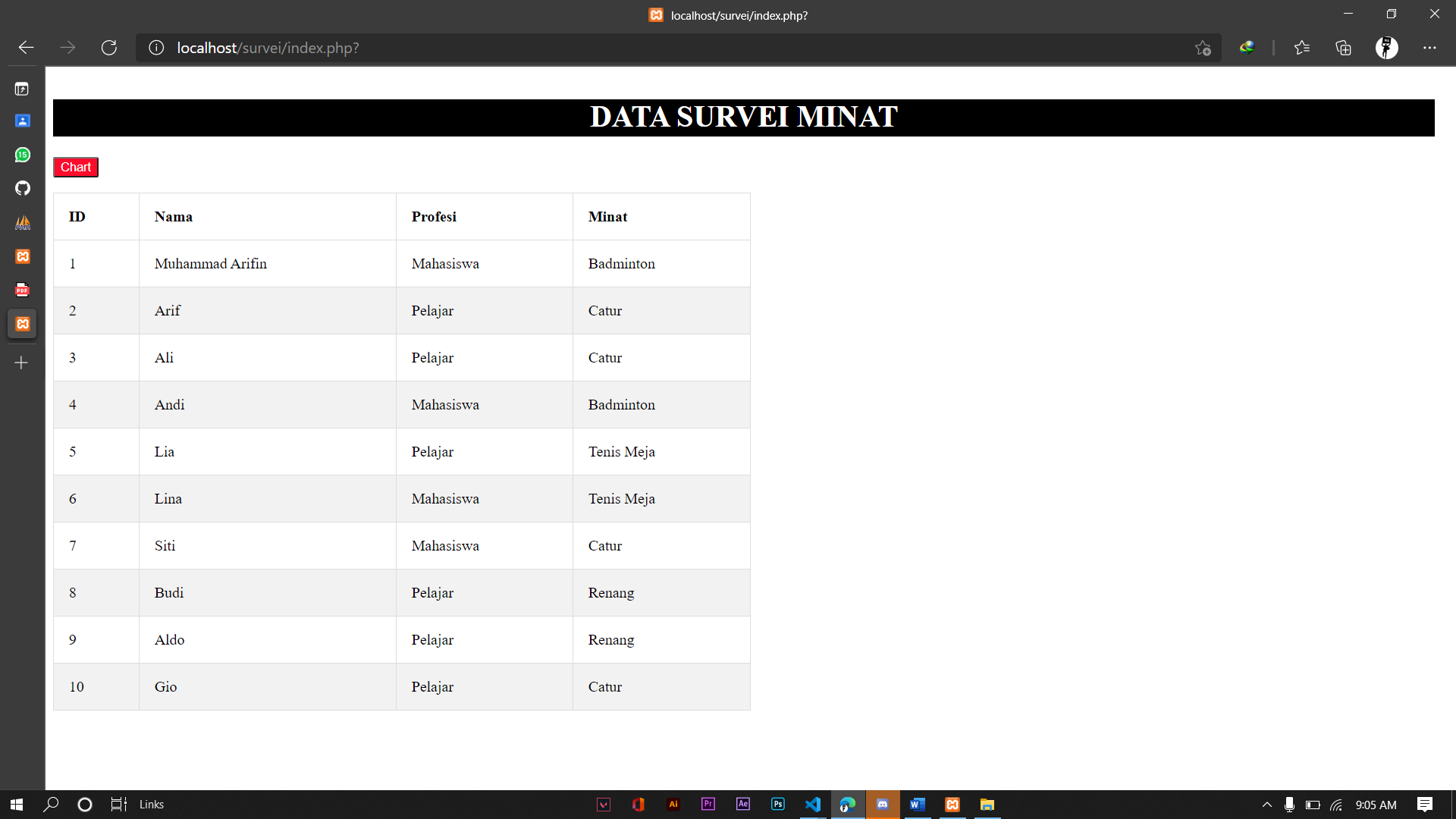
**Gambar 2.8** Chart dan tabel halaman chart.



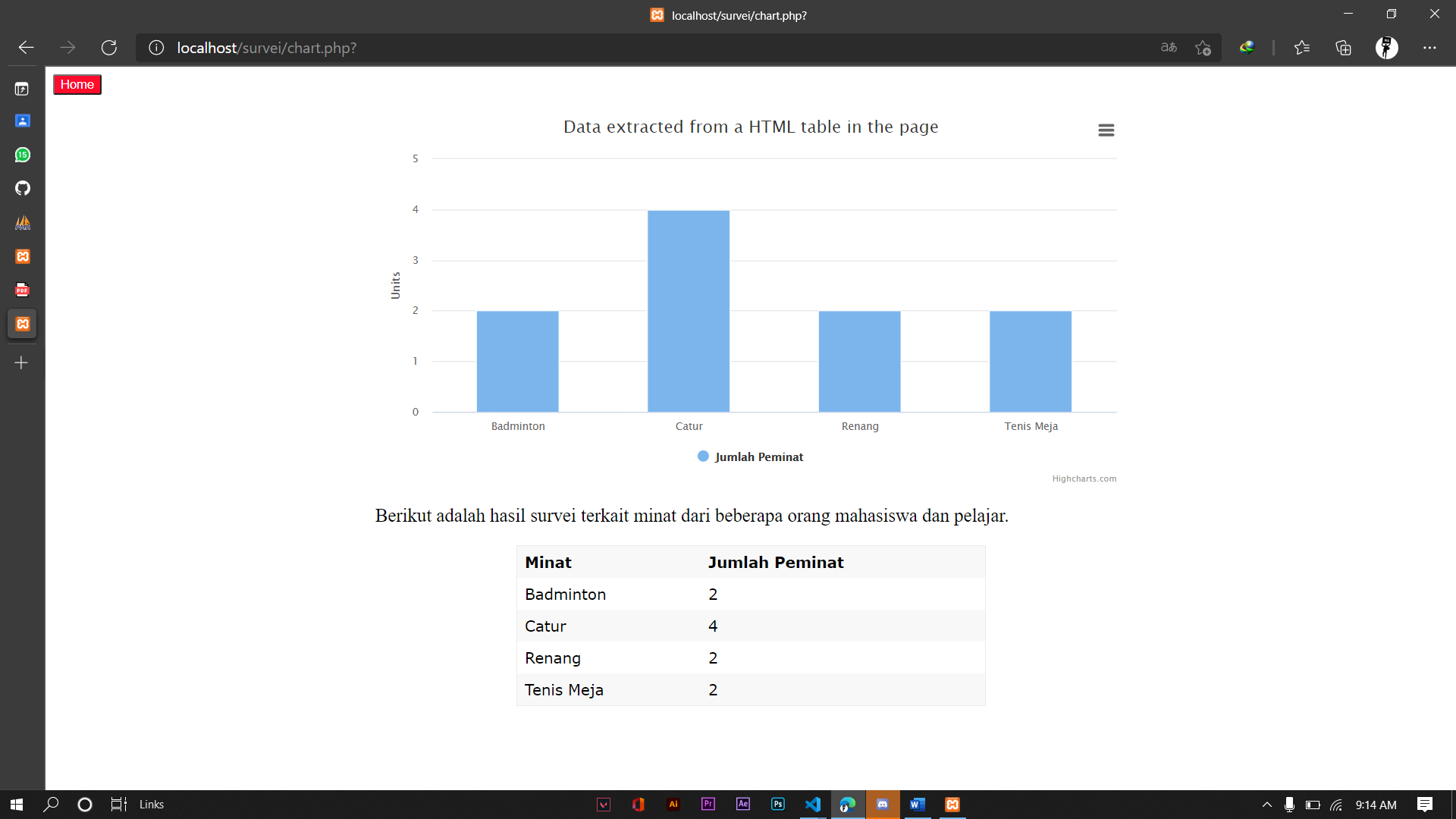
**Gambar 2.9** Styling chart halaman chart.



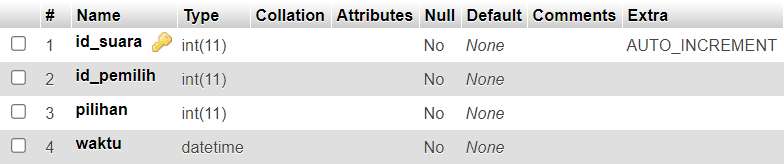
**Gambar 2.10** Styling tabel halaman chart.



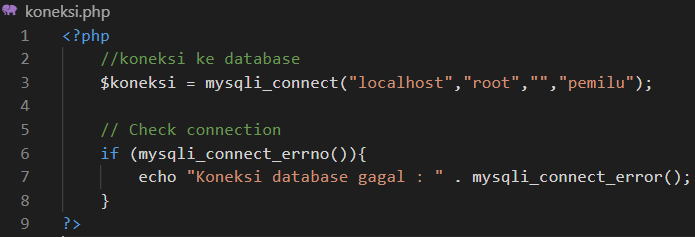
**Gambar 2.11** Halaman index.



**Gambar 2.12** Halaman chart.

****

**Gambar 2.13** Tabel pemilu.



**Gambar 2.14** File koneksi.



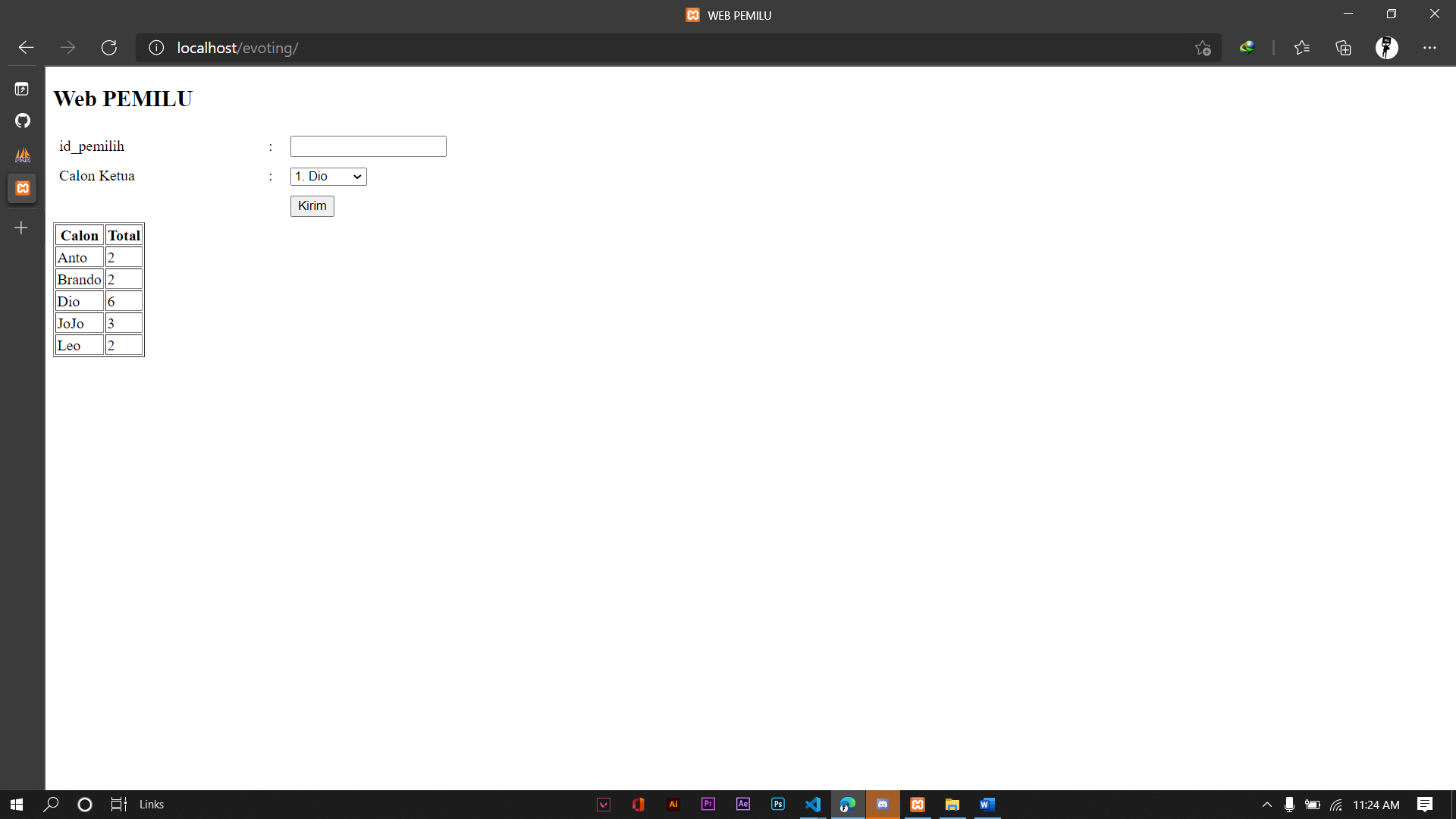
**Gambar 2.15** Field input.



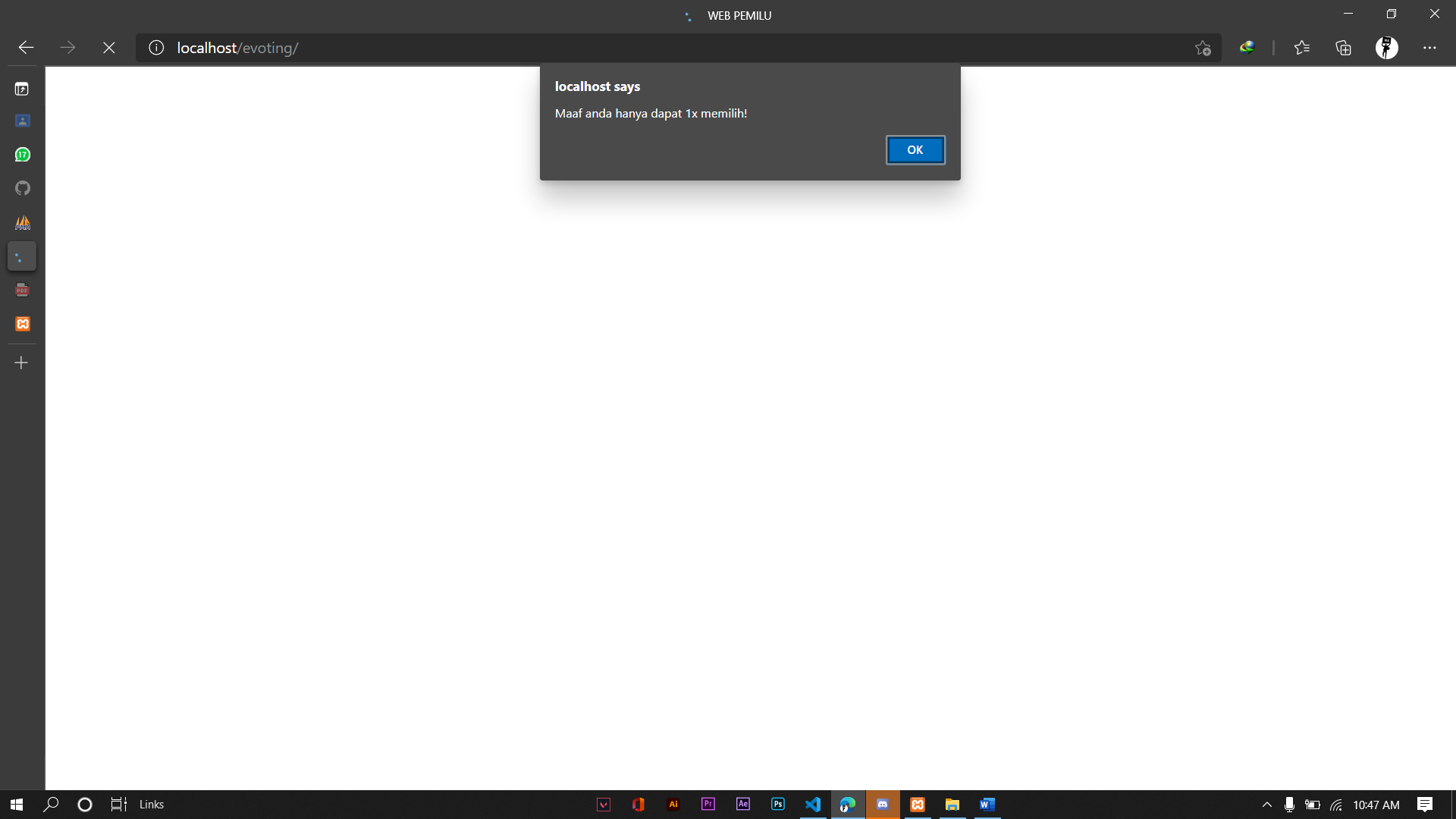
**Gambar 2.16** Tabel output.



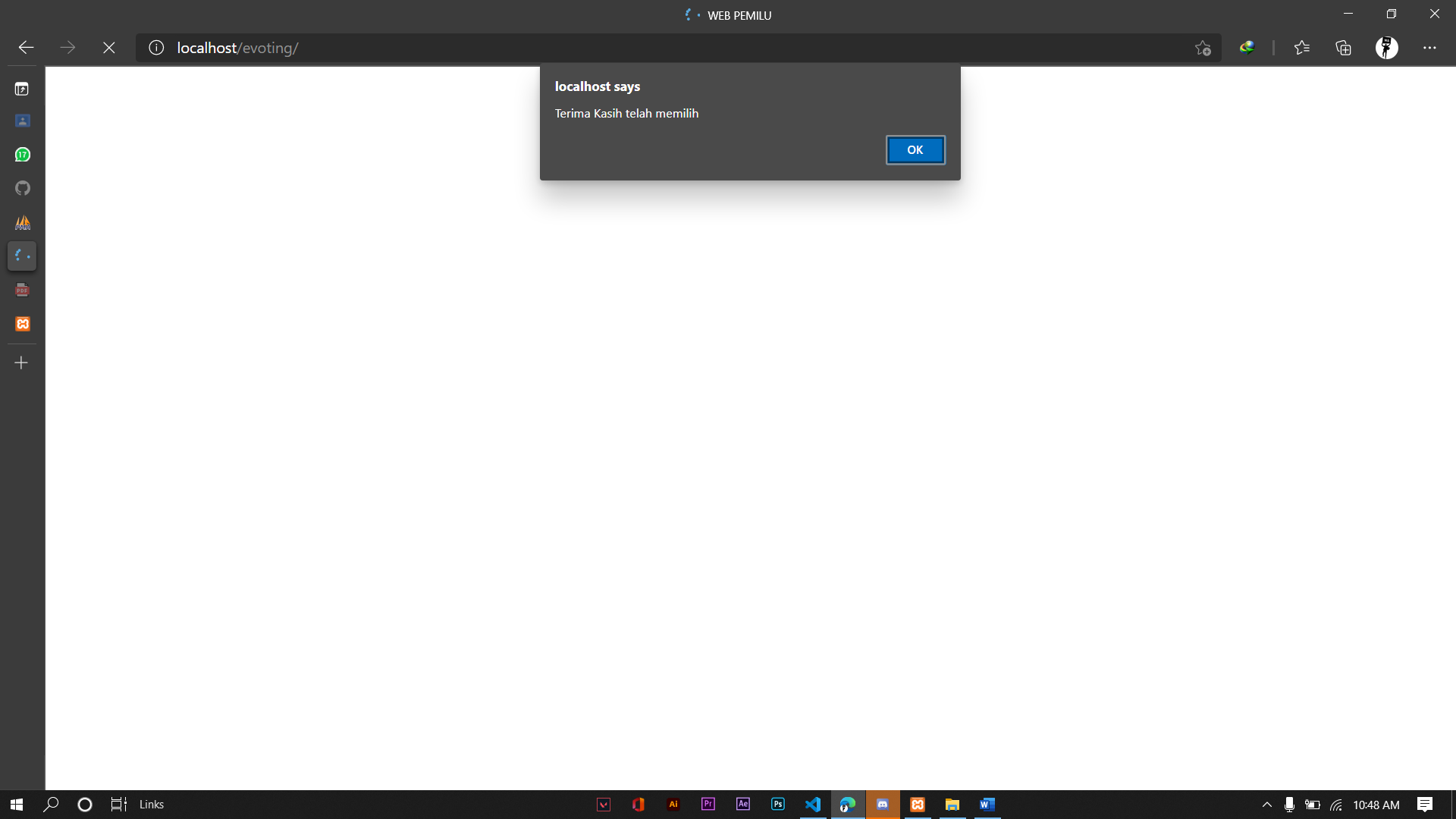
**Gambar 2.17** Metode input, kondisi, dan alert.



**Gambar 2.18** Halaman pemilihan.



**Gambar 2.19** Input tidak bisa dilakukan.



**Gambar 2.20** Input bisa dilakukan.