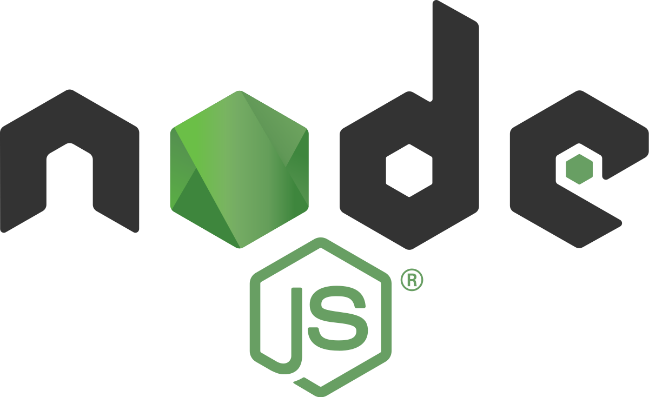
**MySQL Node.js Module**





1. **Pembuka**

Puji Tuhan YME, telah memberikan barokah dan kelancaran untuk membuat **Module MySQL** **Node.js** ini, semoga dapat memberikan manfaat bagi yang membaca. Dan bilamana ada kesalahan pengetikan, sebagai pengetik modul ini, diucapkan mohon maaf yang sebesar-besarnya karena pengetik hanya manusia biasa, dan pasti kesalahan-pun itu ada.

Untuk pembaca, silahkan gunakan modul dengan baik, dan semoga penjelasan di dalam modul ini cukup jelas untuk dimengerti.

Terima kasih, dan selamat belajar!

1. **Pengertian MySQL**

[MySQL](file:///C:\Users\WINDOWS%2010\AppData\Roaming\Microsoft\Word\mysql.com) adalah sebuah perangkat lunak sistem basis data yang bermultialur, multipengguna, dengan instalasi sekitar 6 juta di dunia, terlisensikan dengan lisensi GPL (Komersial).

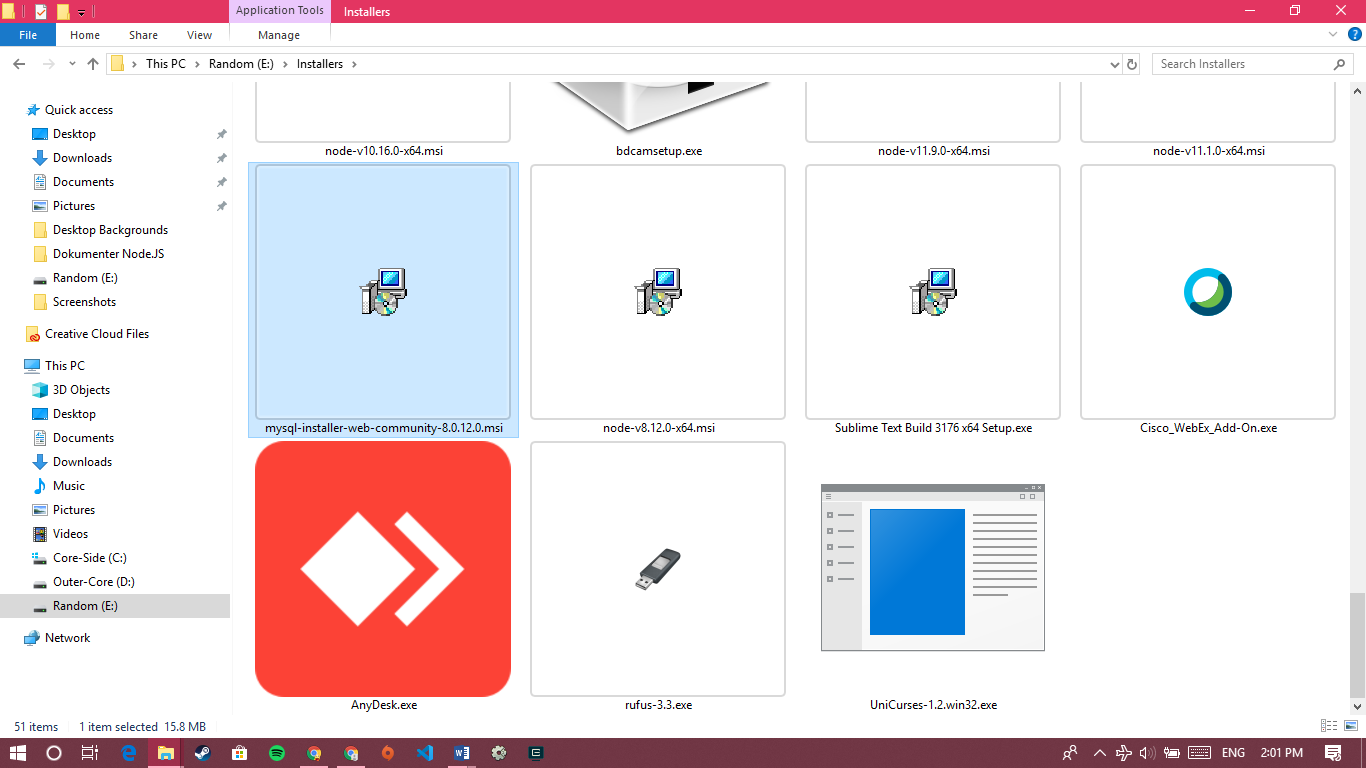
1. **Instalasi MySQL**

Untuk instalasi MySQL dapat dilakukan di sistem operasi manapun. Berikut adalah cara instalasi lewat sis-op Windows.

1. Silahkan menuju situs resmi pengunduhan [MySQL](https://dev.mysql.com/downloads/installer/).
2. Setelah itu silahkan unduh *installer* MySQL.



1. Silahkan untuk klik file *installer* MySQL dua kali, dan *install*.



1. **Instalasi XAMPP**

Apa itu XAMPP? XAMPP adalah sebuah perangkat lunak bebas, yang berfungsi sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, database MySQL, dan penerjemah bahasa PHP dan Perl.

Untuk instalasi XAMPP di sistem operasi Windows, silahkan ikuti instruksi dibawah:

1. Silahkan akses website [XAMPP](https://www.apachefriends.org/index.html).
2. Silahkan klik tombol “XAMPP for Windows.”



1. Setelah itu, XAMPP secara otomatis akan terunduh.
2. Sehabis terunduh, install XAMPP sesuai instruksi [ini](https://www.wikihow.com/Install-XAMPP-for-Windows).
3. **Node.js Login**

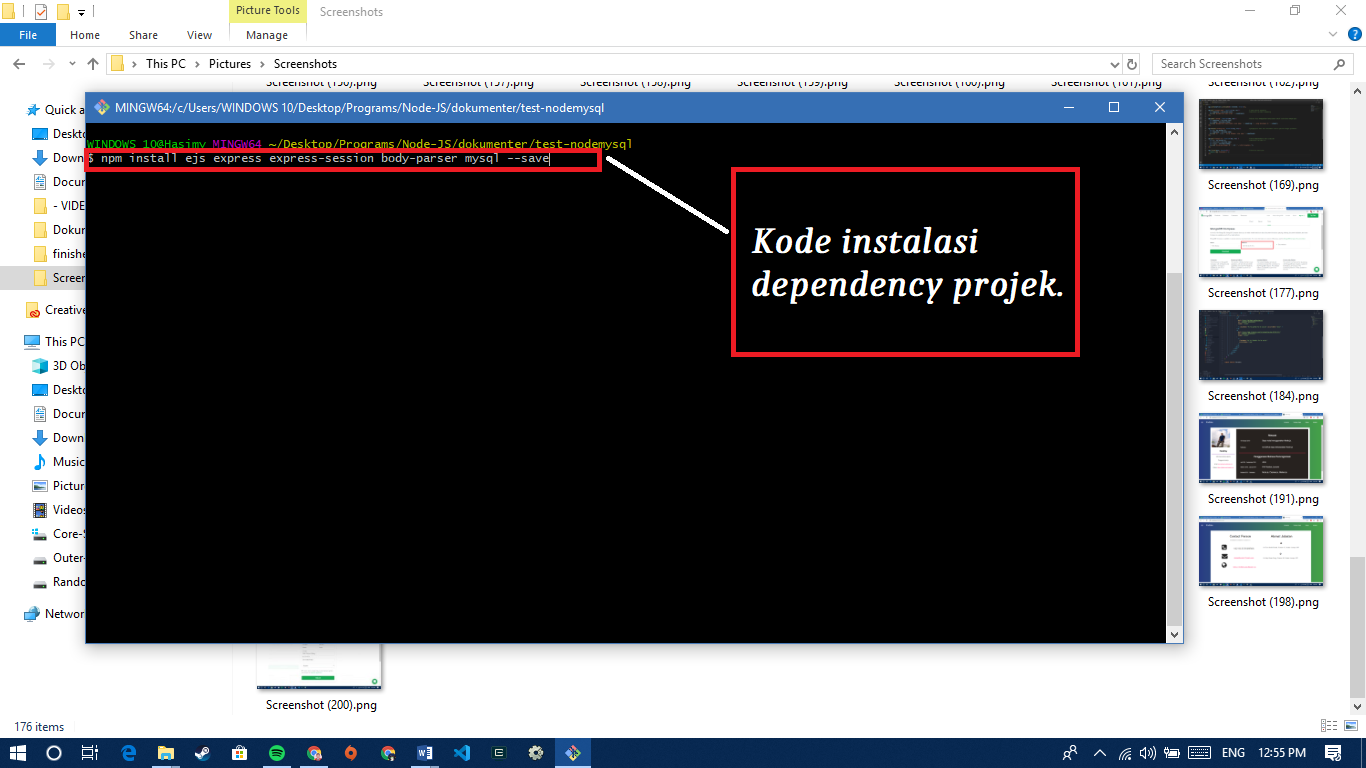
Kali ini kita akan membuat aplikasi *login* dengan Node.js dan menggunakan sistem manajemen database [SQL](https://id.wikipedia.org/wiki/SQL) yaitu [MySQL](https://id.wikipedia.org/wiki/MySQL). Baiklah, mari kita mulai membuat aplikasi ini, dengan awalnya persiapan folder, dan pembuatan *database*!

1. Silahkan buat folder baru.
2. Silahkan jalankan kode: “npm init --yes” di CLI/Terminal anda yang sudah pindah direktori ke folder baru anda untuk membuat file “package.json”.

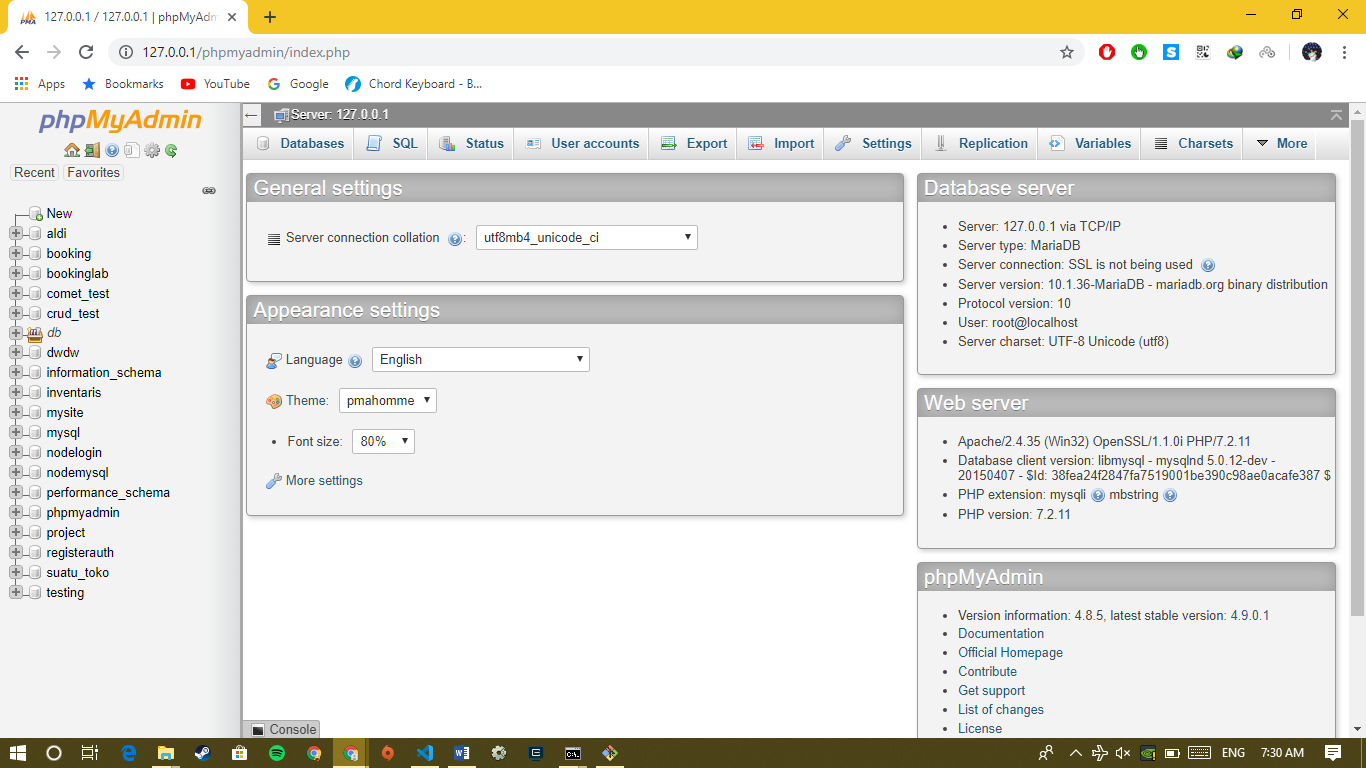
Di projek ini, kita memerlukan beberapa *dependencies*, yaitu EJS, Express.js, [express-session](https://www.npmjs.com/package/express-session) (untuk membuat sesi *login/out*), [body-parser](https://www.npmjs.com/package/body-parser) (untuk menerima *request* variabel *post* dengan mudah sebagai suatu parameter).

1. Maka dari itu, silahkan jalankan kode ini di CLI/Terminal anda untuk meng*install* *dependencies* yang diperlukan:

“npm install ejs express express-session body-parser mysql --save”

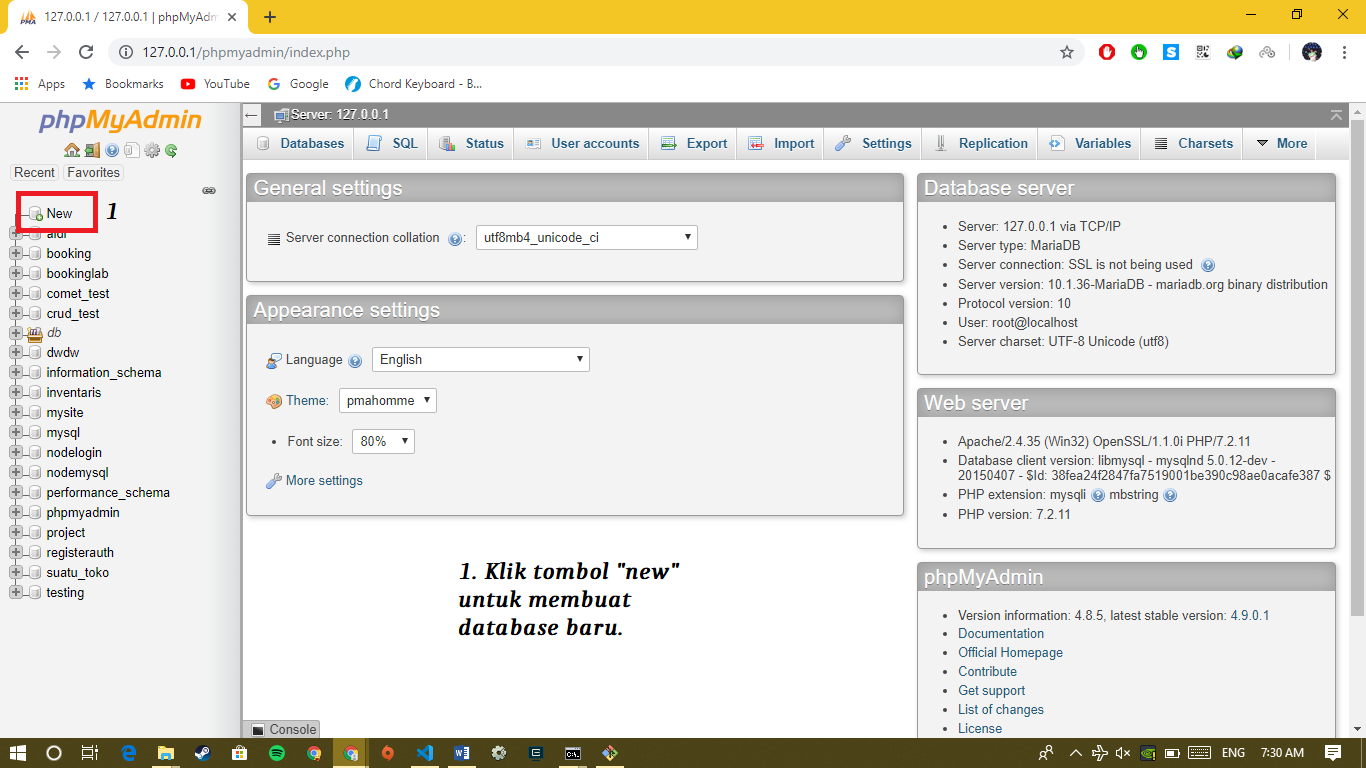


1. Setelah itu, pastikan anda telah meng*install* [PHPMyAdmin](https://www.phpmyadmin.net/) atau alat *software* lainnya yang dapat mengendalikan administrasi MySQL di dalam web.

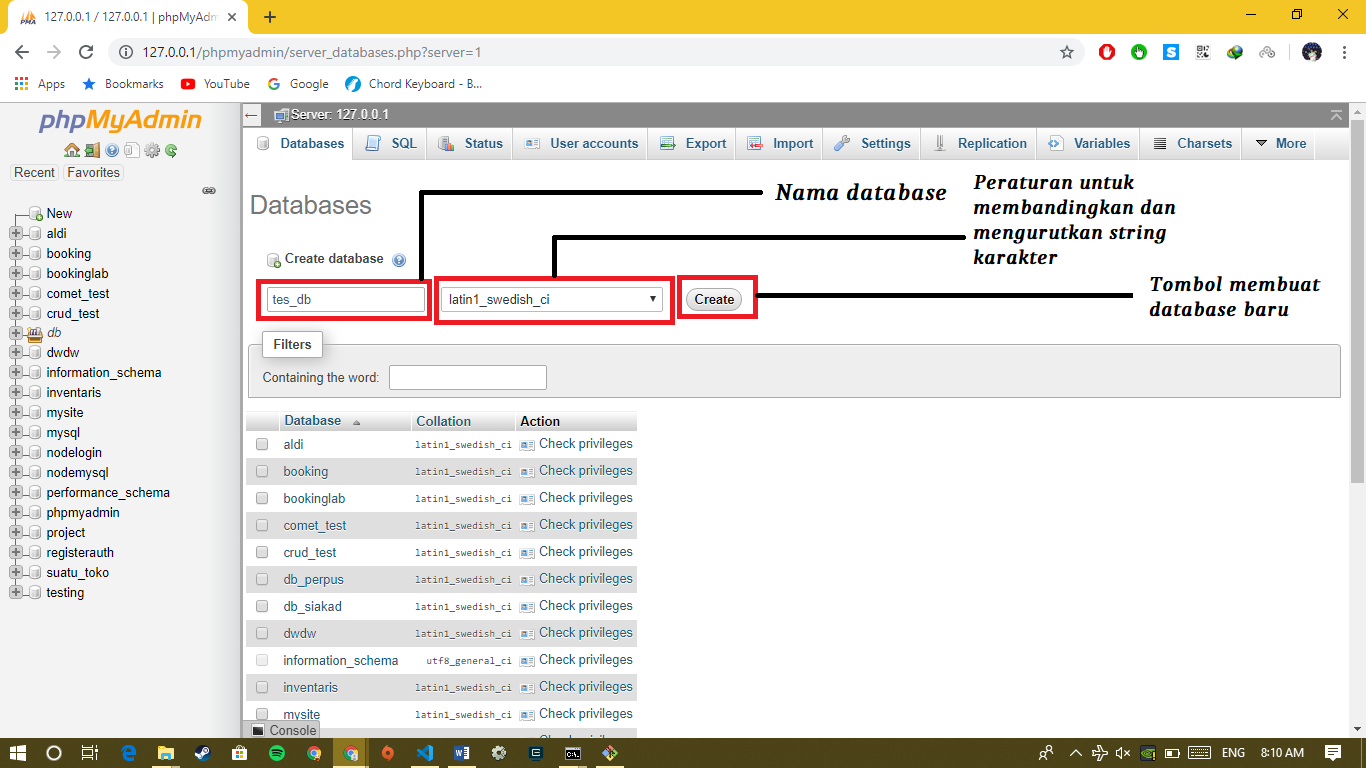


Di PHPMyAdmin anda dapat membuat/edit *database* MySQL.

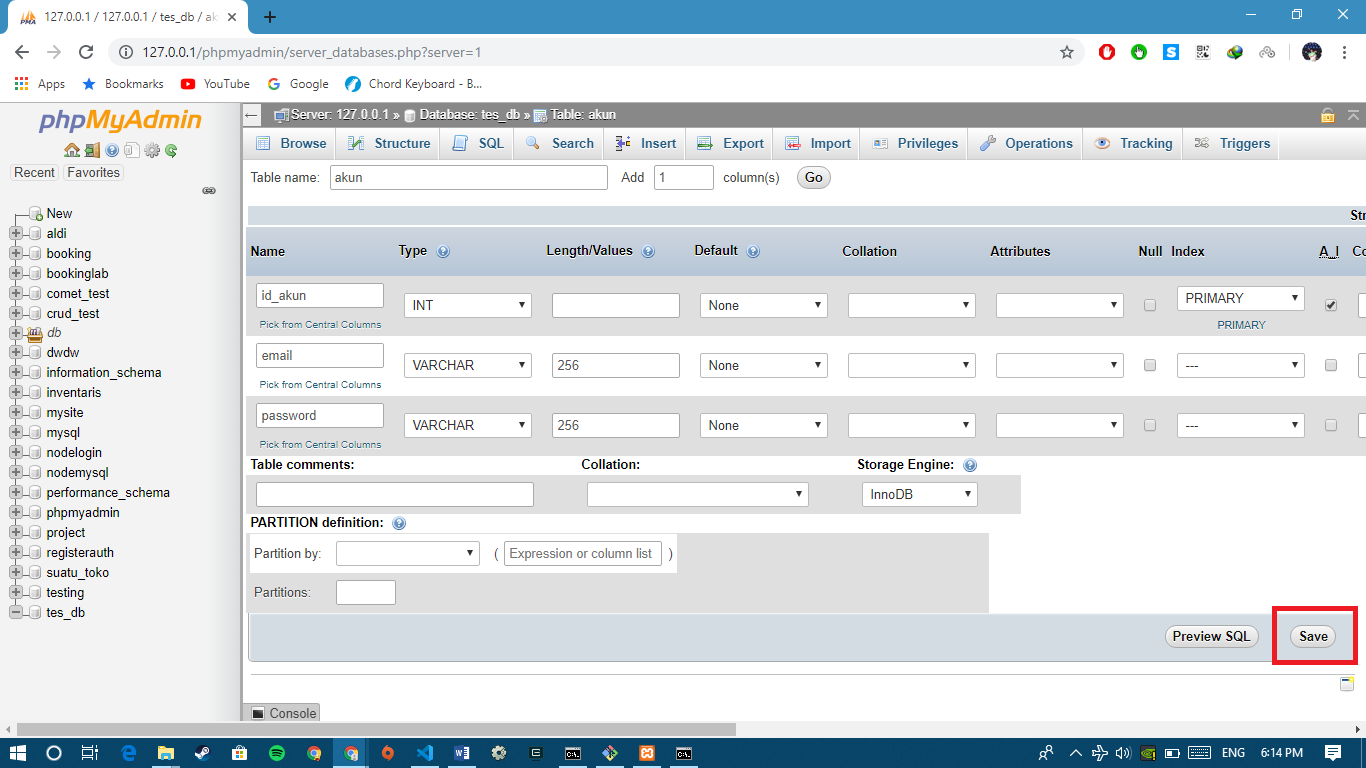
1. Sehabis buka PHPMyAdmin, silahkan membuat *database* baru dengan klik tombol new di daftar *database* MySQL di sebelah kiri.



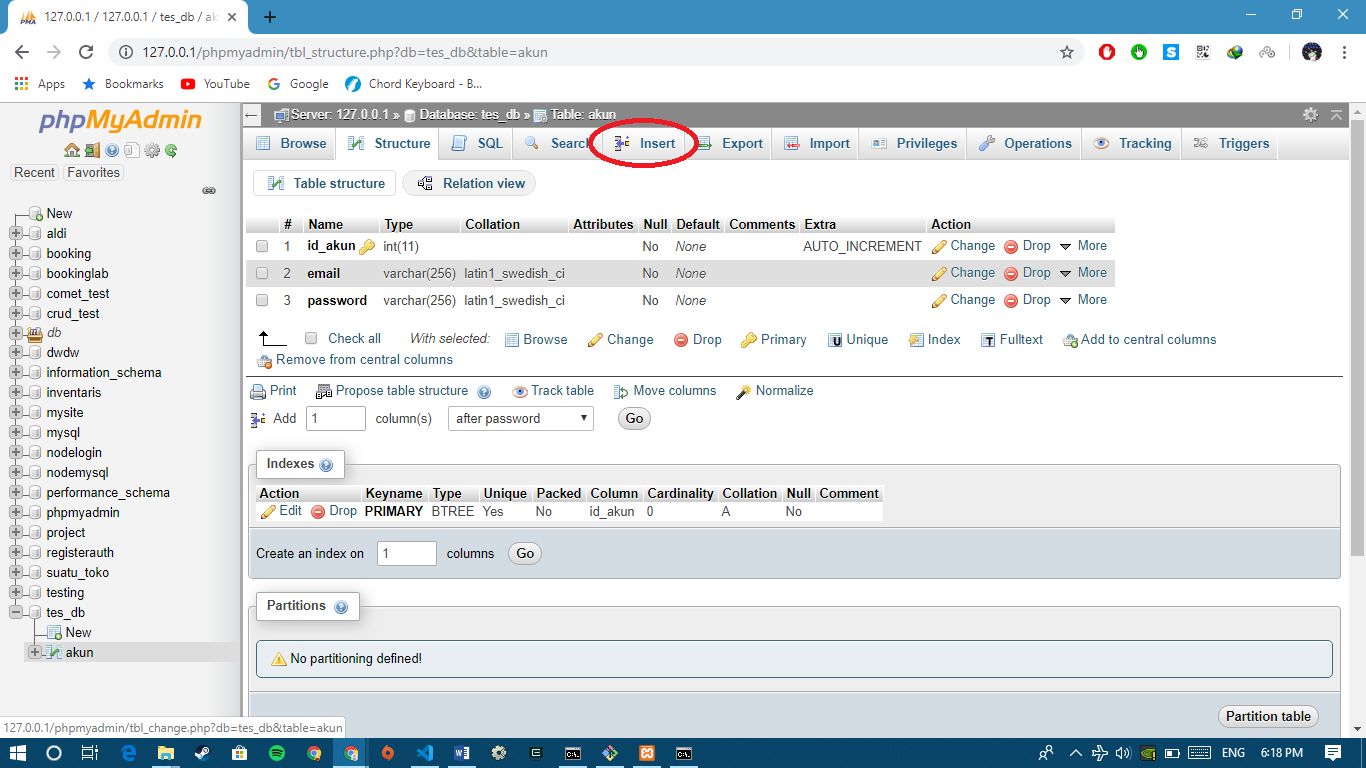
1. Selepas itu, silahkan mengisikan nama baru untuk *database* anda, contoh nama seperti dibawah ini: (setelah itu klik tombol “*create*”);



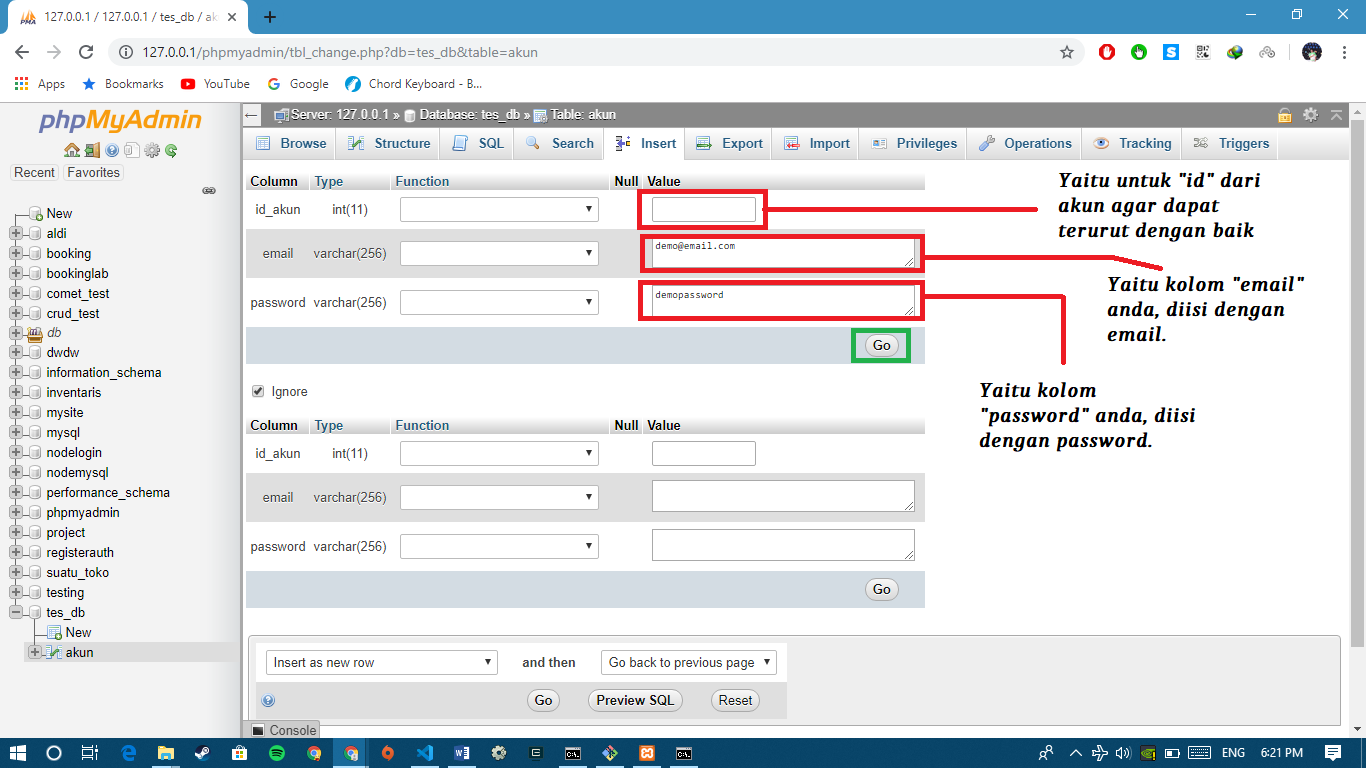
1. Setelah membuat *database* baru, silahkan membuat *database*-nya dengan format seperti dibawah ini: (setelah sudah, silahkan klik tombol “*save*”).



1. Setelah sudah, silahkan klik tombol “*insert*” untuk mulai memasukan data ke dalam tabel dari *database*.



1. Silahkan isi kolom data dengan format seperti dibawah, dan jika selesai silahkan klik tombol “*go*” dibawah kolom data.

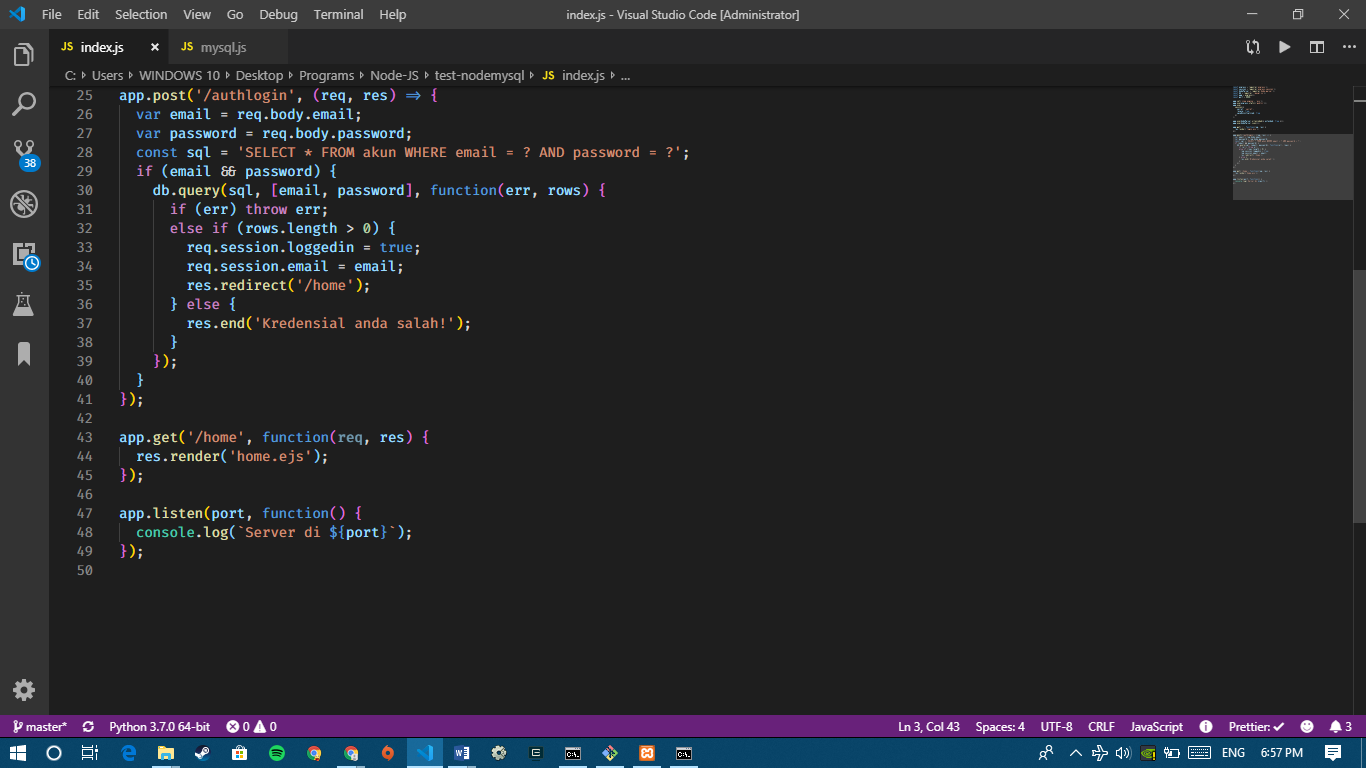
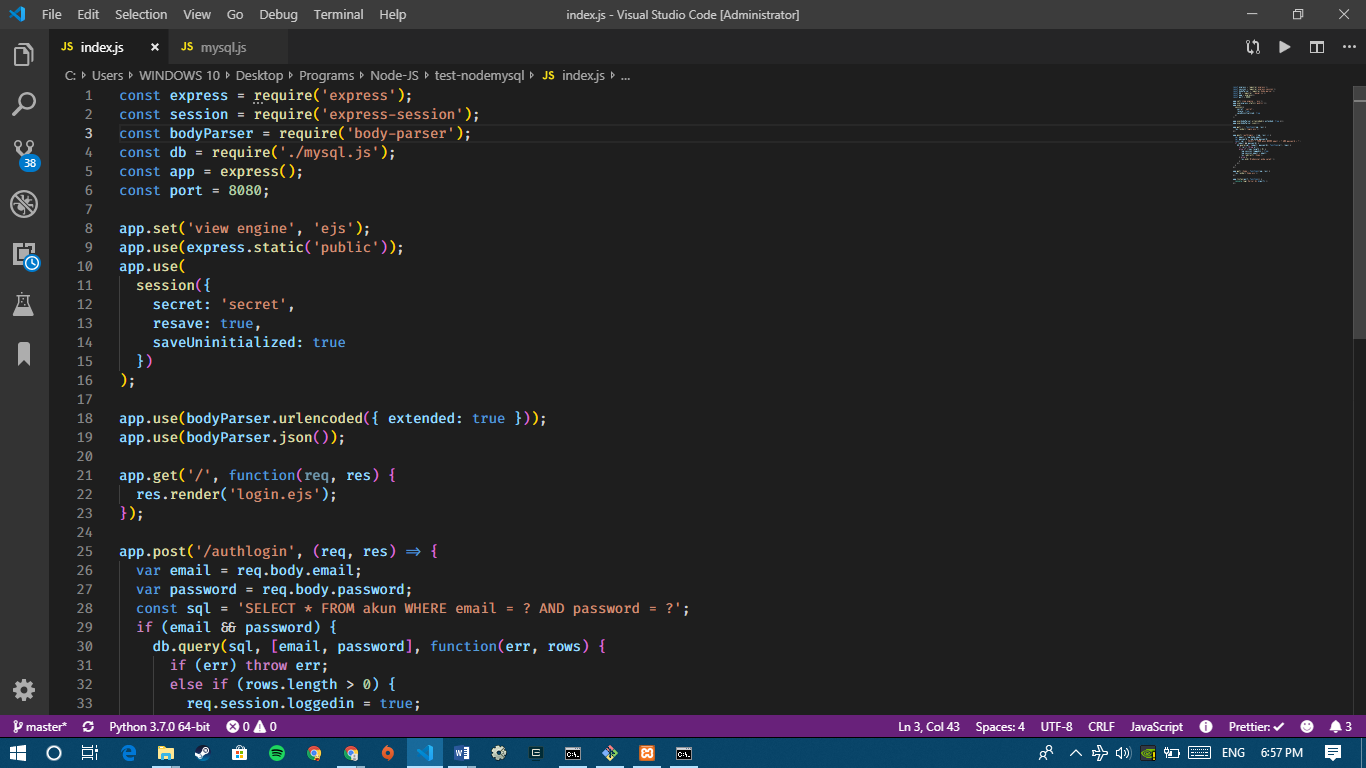


Mungkin ada pertanyaan mengapa kolom “id\_akun” itu dikosongi, yaitu karena pada pembuatan *database* baru tadi, untuk kolom “id\_akun” kita sudah atur dengan *setting* **AI**.

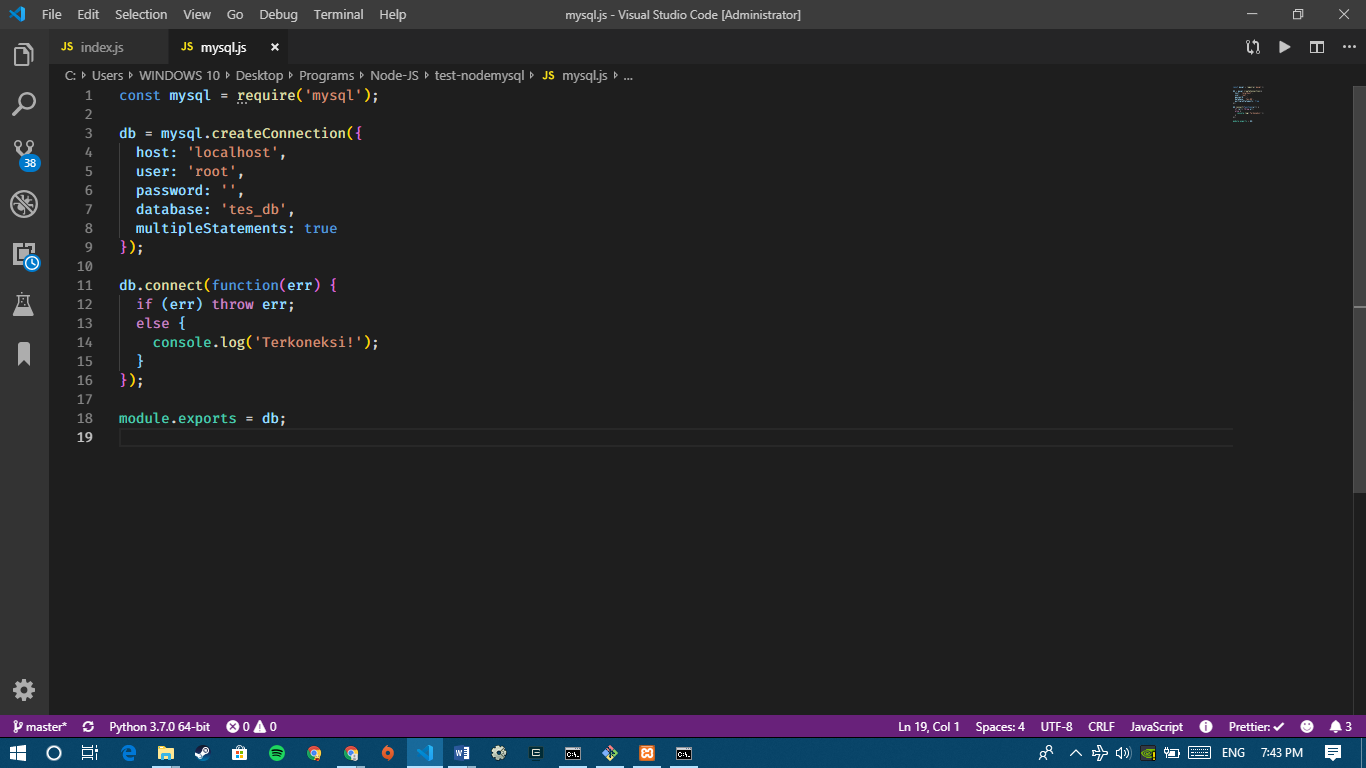
Arti **AI** disini yaitu auto-increment, yang berfungsi secara otomatis menambahkan nilai bila terjadi penambahan isi baris pada tabel dimana bidang itu ada.

Baiklah, setelah penjelasan tersebut mari kita lanjut di *coding*-nya sekarang!

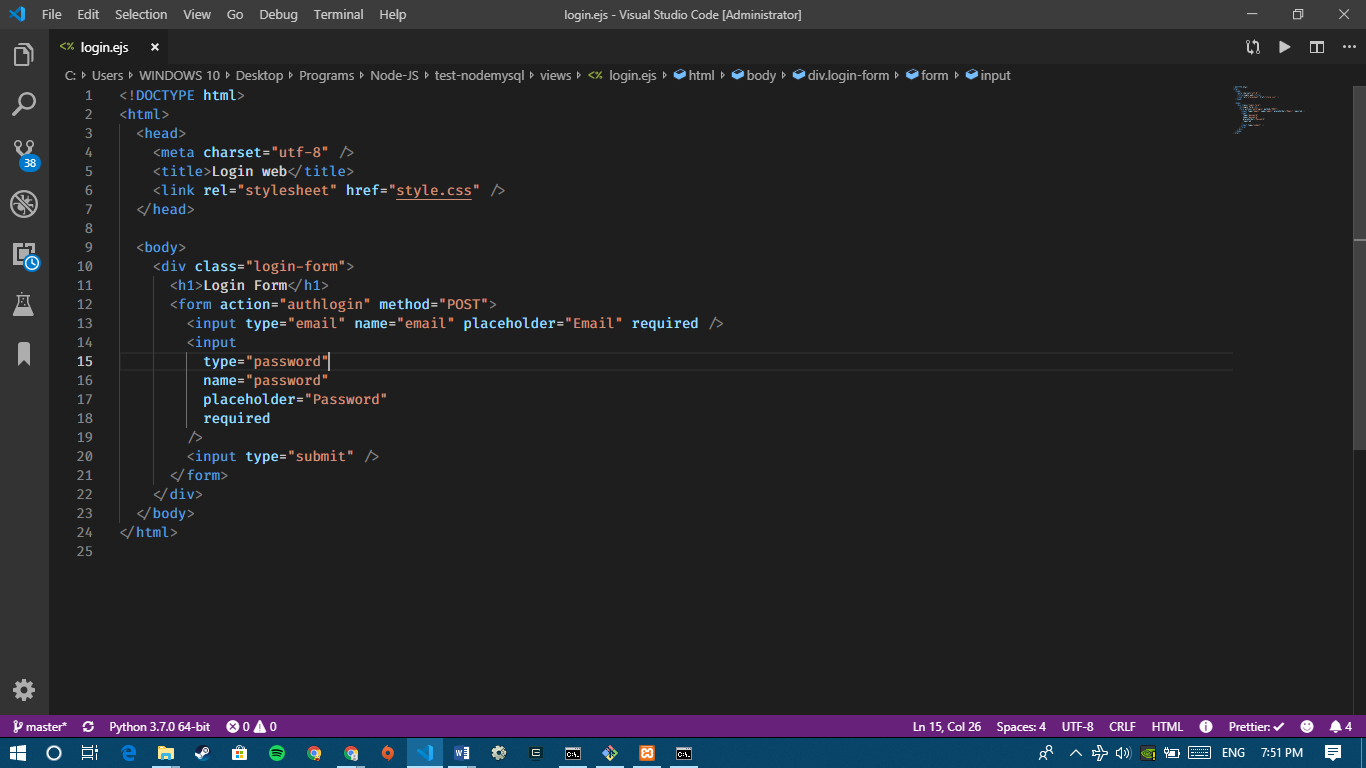
1. Pada folder projek, buatlah file baru bernama “index.js”.
2. Di file “index.js”, silahkan ketikan kode-kode seperti dibawah:



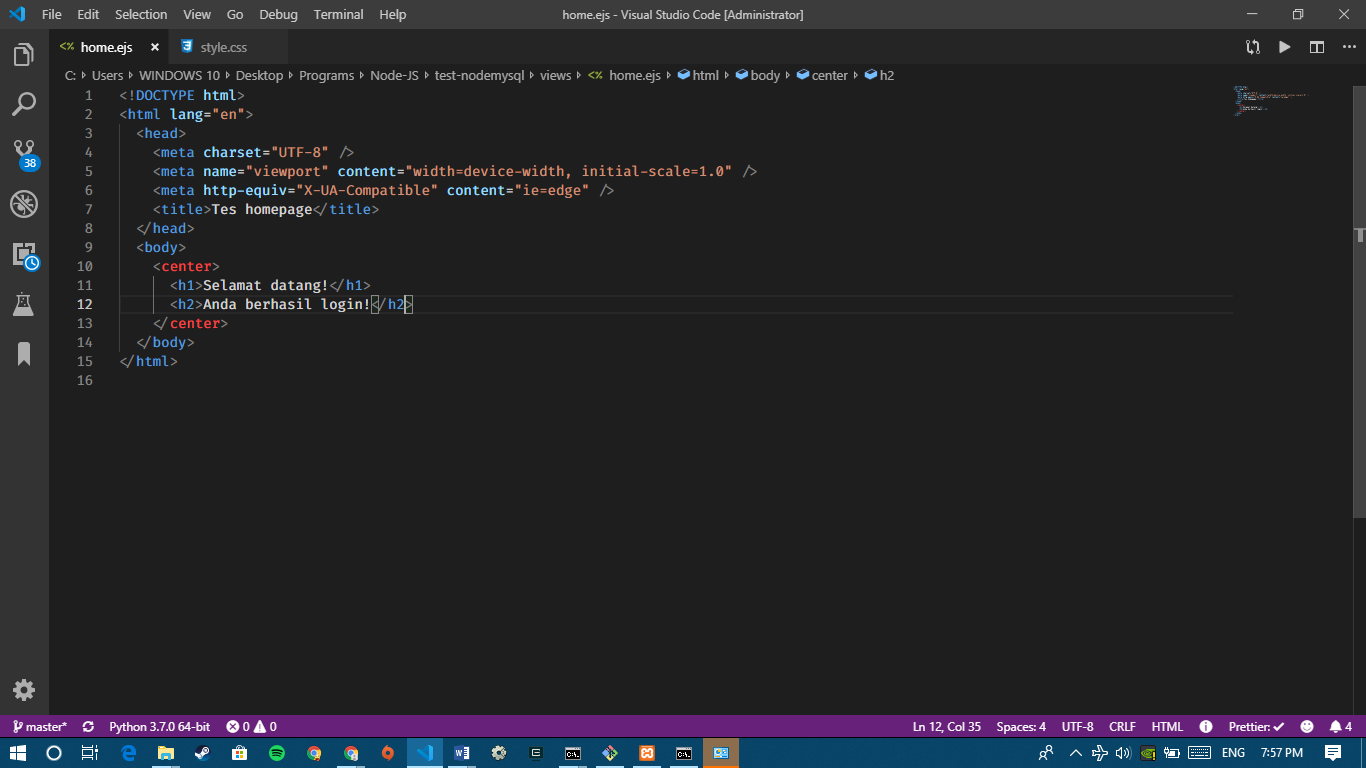
1. Setelah itu buatlah file baru bernama “mysql.js”, dan ketikan kode-kode dibawah:



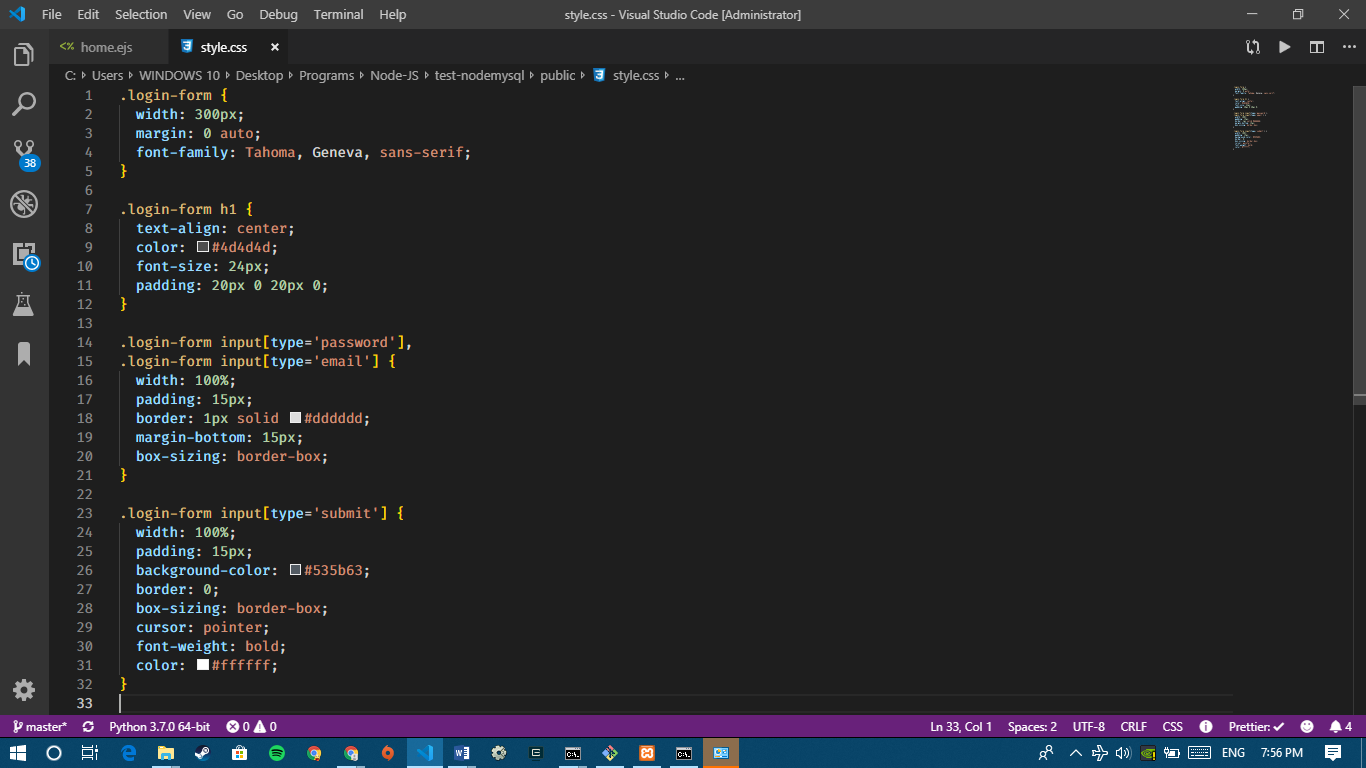
1. Jika sudah, langsung buat 2 folder baru bernama “public” dan “views”.
2. Di folder “views”, buatlah 2 file baru bernama ‘login.ejs’ dan ‘home.ejs’.
3. Silahkan buka ‘login.ejs’, dan ketikan kode HTML dibawah:



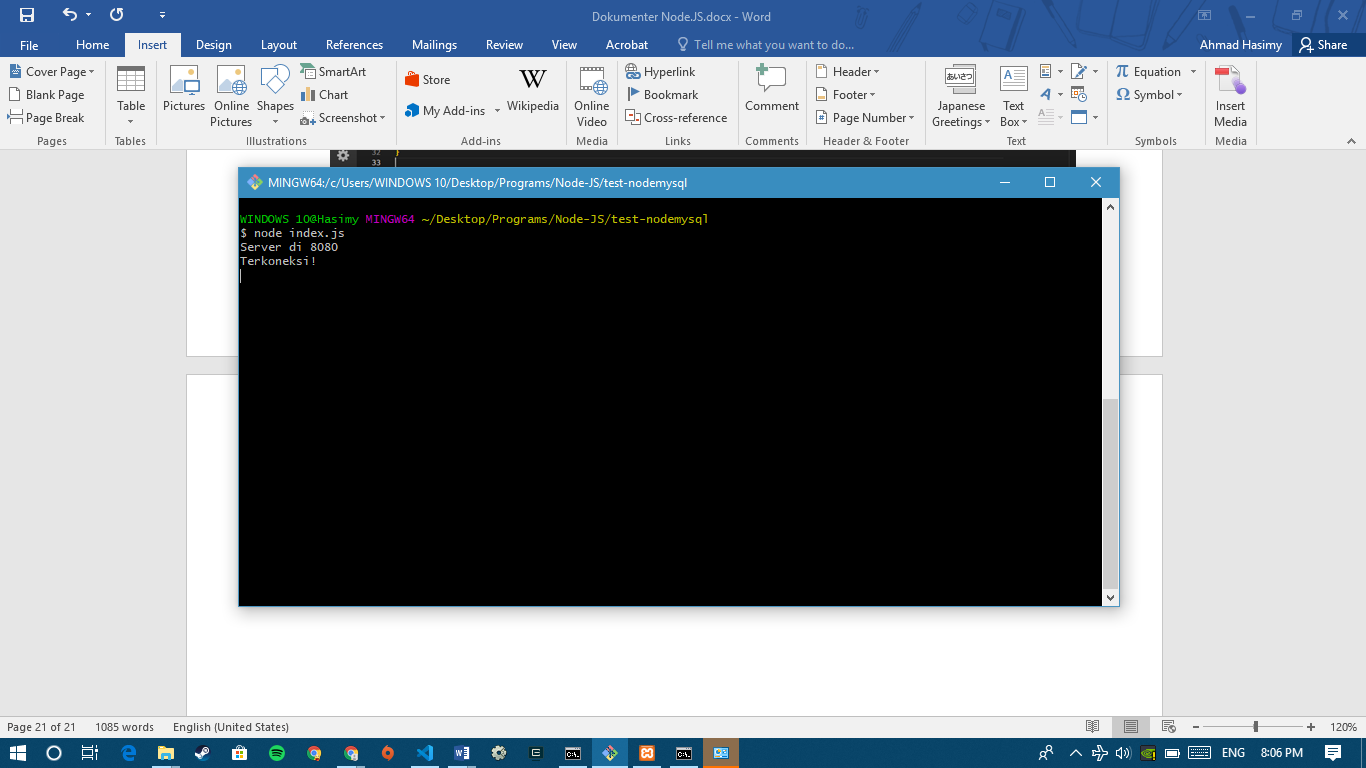
1. Untuk ‘home.ejs’, silahkan isi dengan kode HTML berikut:



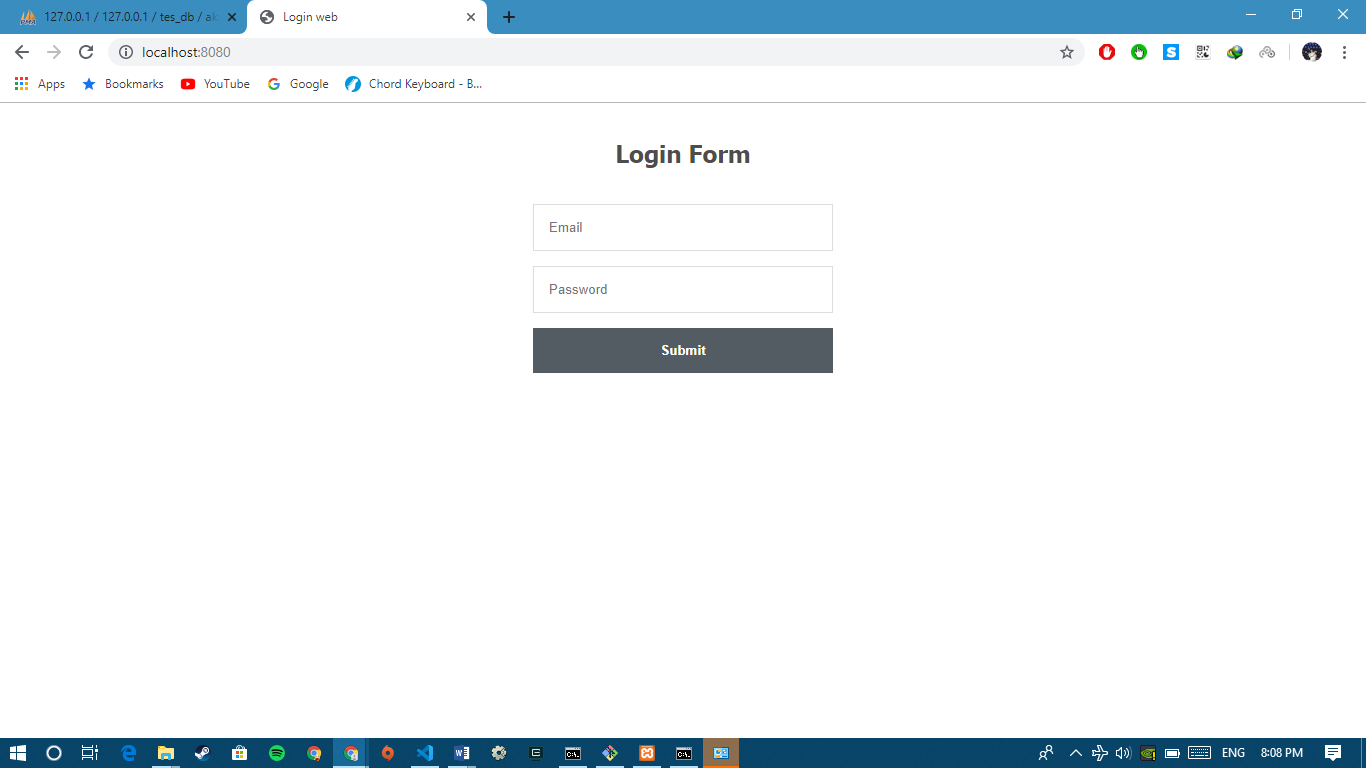
1. Setelah selesai, silahkan pergi ke folder “public” dan buat file baru bernama ‘style.css’ untuk desain *form login*, dan isikan dengan kode berikut:



1. Jika sudah, silahkan buka kembali CLI/Terminal anda di folder projek, dan jalankan aplikasinya, seperti contoh dibawah:



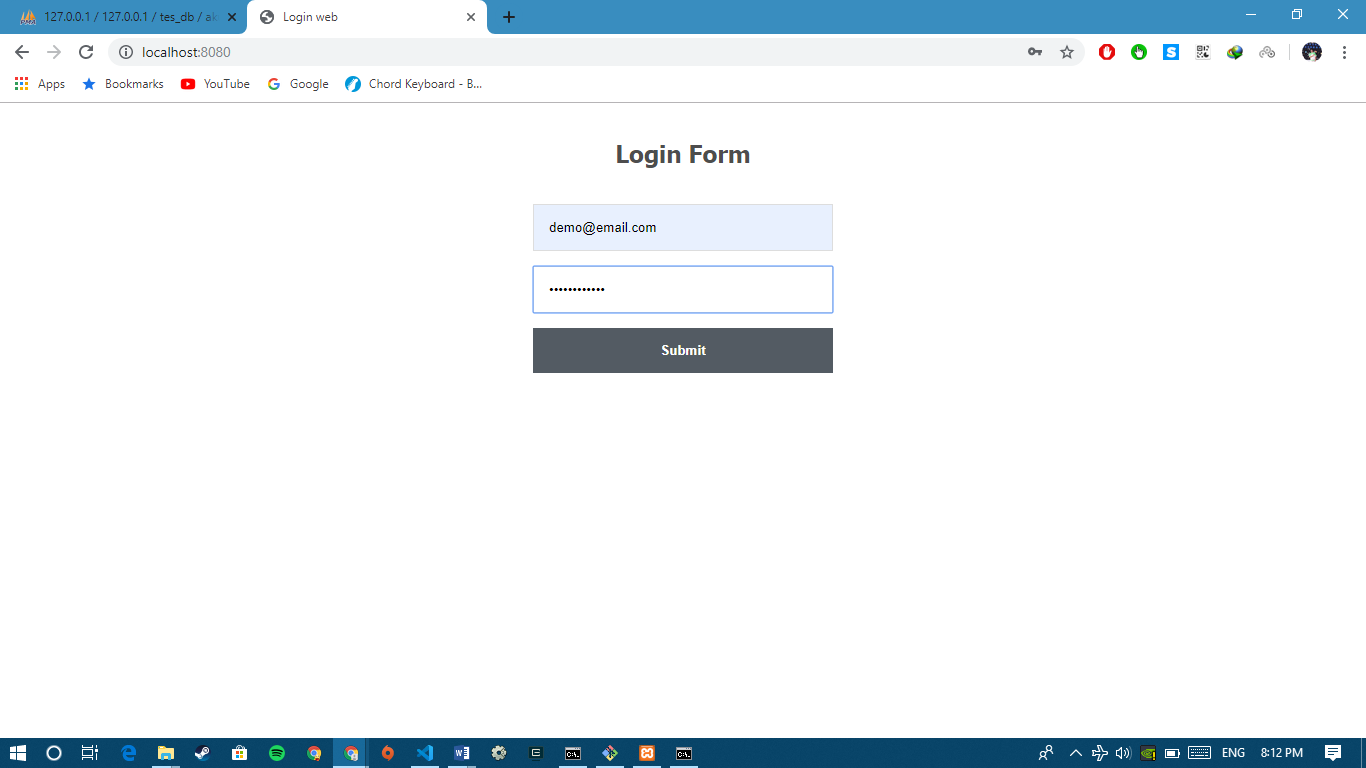
1. Bukalah browser anda, dan ketikan “localhost:8080/” untuk mengakses aplikasi.



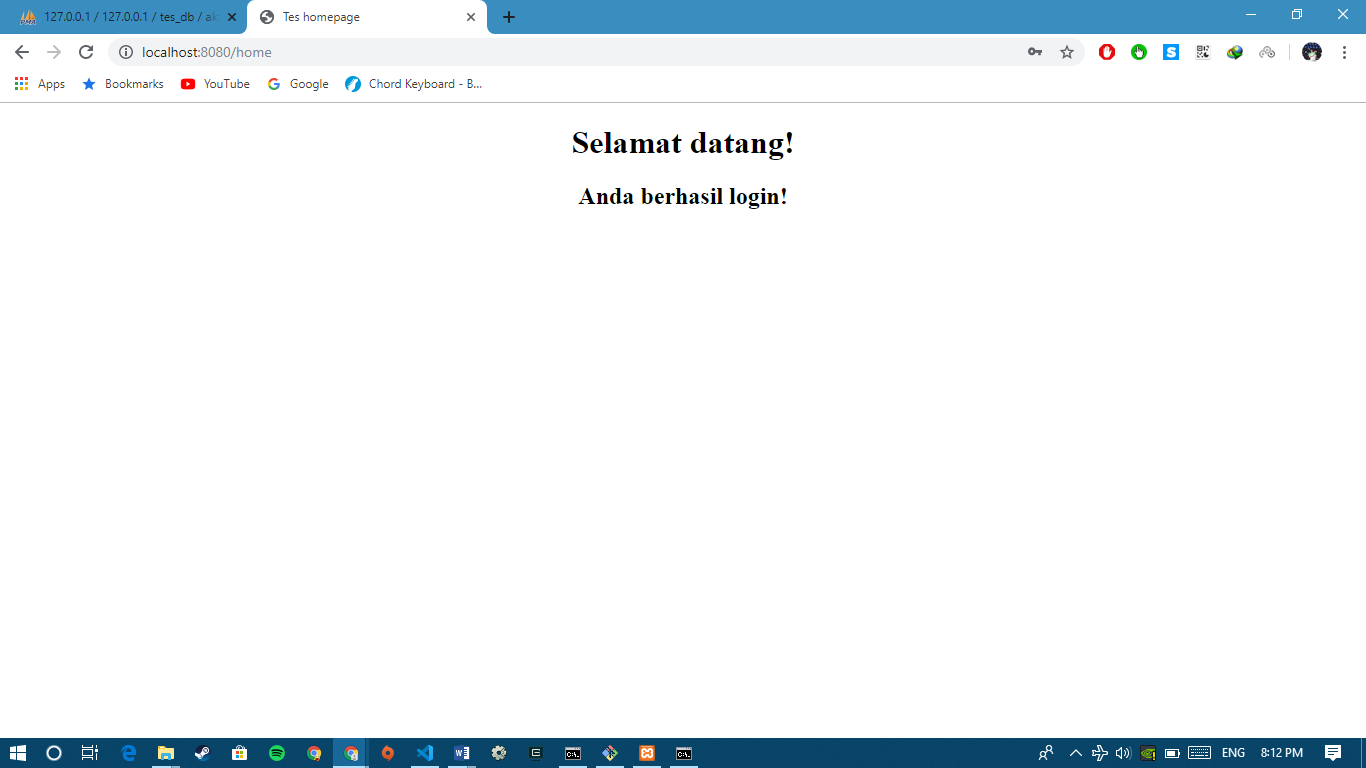
Setelah aplikasi terbuka, berarti aplikasi anda sudah bisa berjalan dengan lancar!

Jangan lupa, jikalau ada **error**, silahkan di cek kembali *coding*-an anda, jikalau ada typo karena itu biasanya penyebab dari **SyntaxError**.

1. Untuk menuju *homepage*, silahkan masukan data *email* dan *password* kalian, sama seperti yang anda masukan ke dalam *database* MySQL sebelumnya. Contoh seperti dibawah:



1. Jika *homepage*-pun terakses, berarti sistem login dari web anda telah berhasil!



1. **Node.js Register/Login Web**

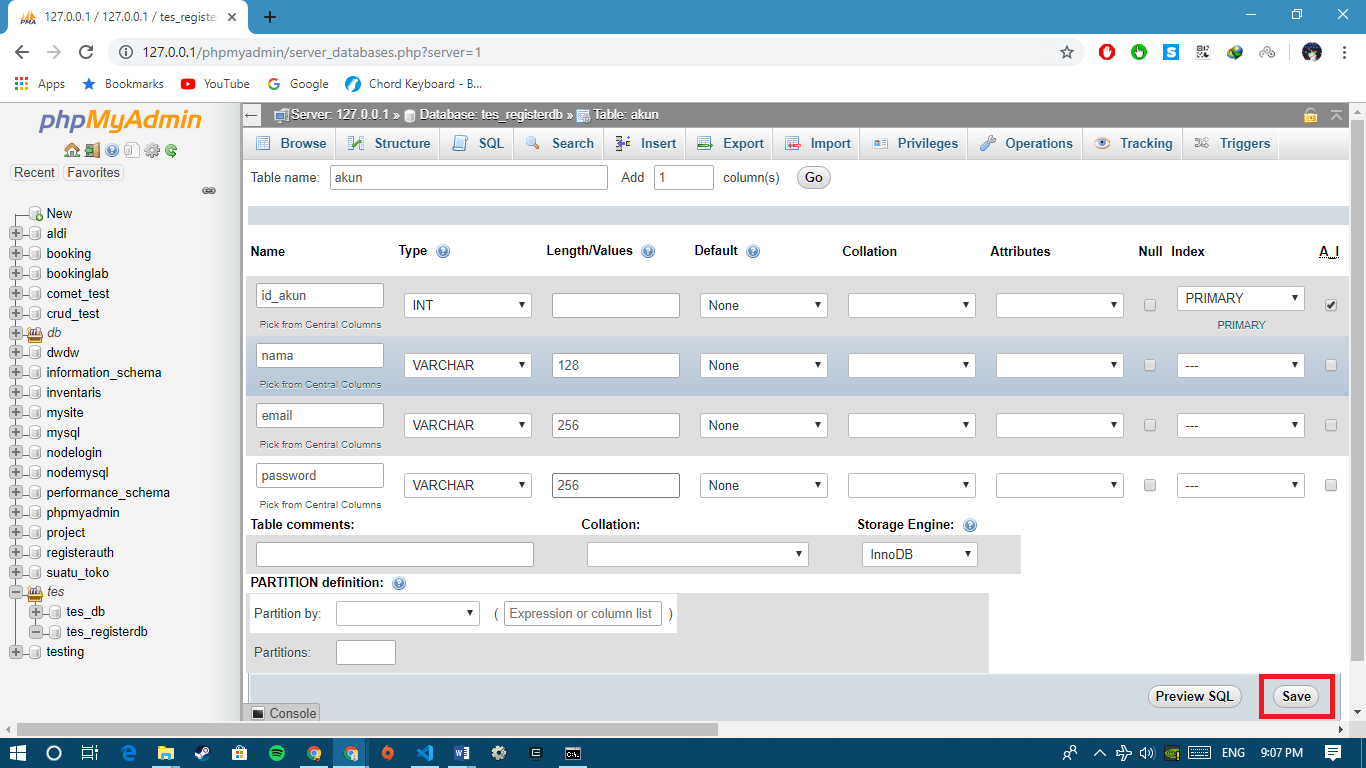
Pada kali ini, kita akan membuat aplikasi *Register* dan *Login* dengan Node.js, dan sistem *database* MySQL. Proses pembuatan ini hampir sama dengan proses pembuatan aplikasi *Login* di atas, dengan perbedaan struktur tabel di *database*, dan sintaks untuk memasukkan data ke *database*. Mari kita mulai!

1. Silahkan buat folder baru.
2. Jalankan kode “npm init --yes” di CLI/Terminal anda di folder projek.
3. Silahkan *install* *dependencies* yang diperlukan dengan jalankan kode ini di CLI/Terminal anda:

“npm install ejs express express-session body-parser --save”

*dependencies sama seperti di projek* ***Login****.*

1. Jika sudah, silahkan buka PHPMyAdmin anda, dan buatlah *database* baru, dengan format seperti berikut: (setelah selesai, klik “save”).

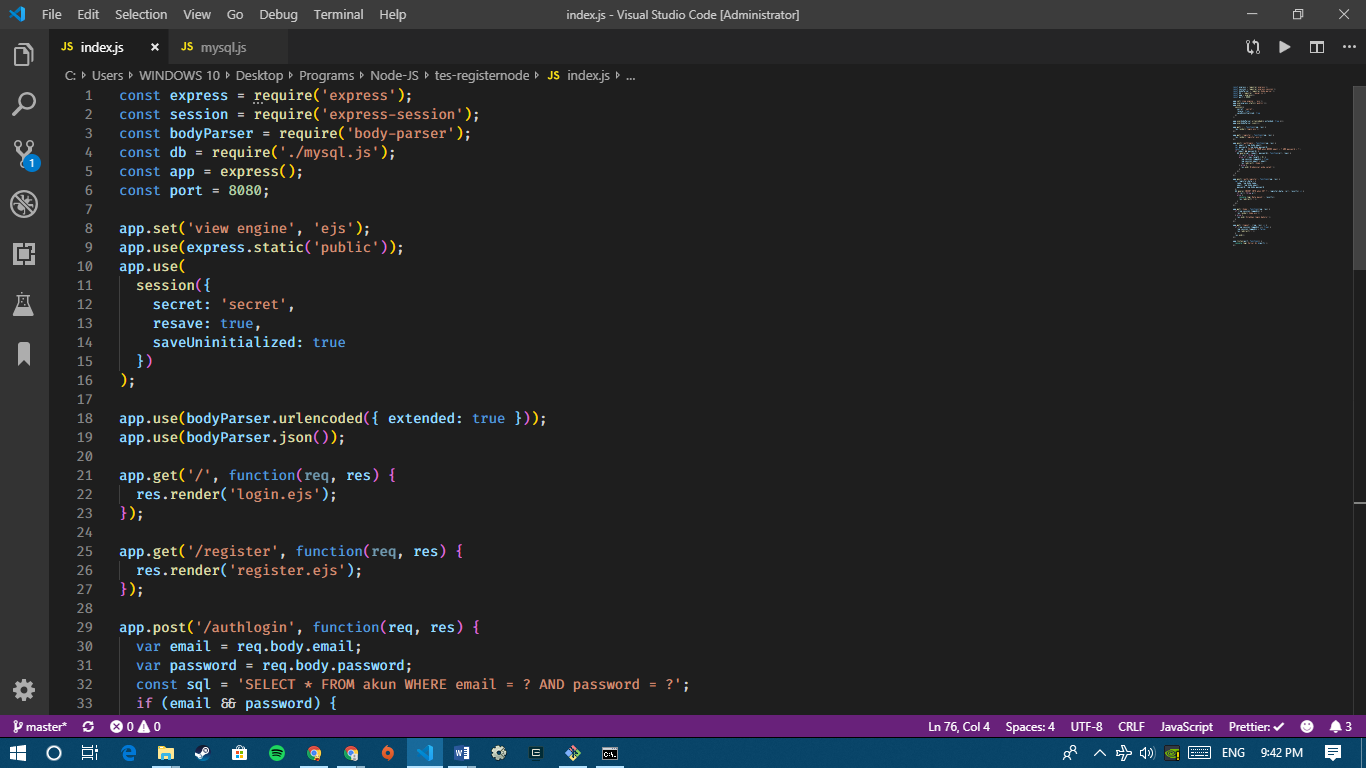


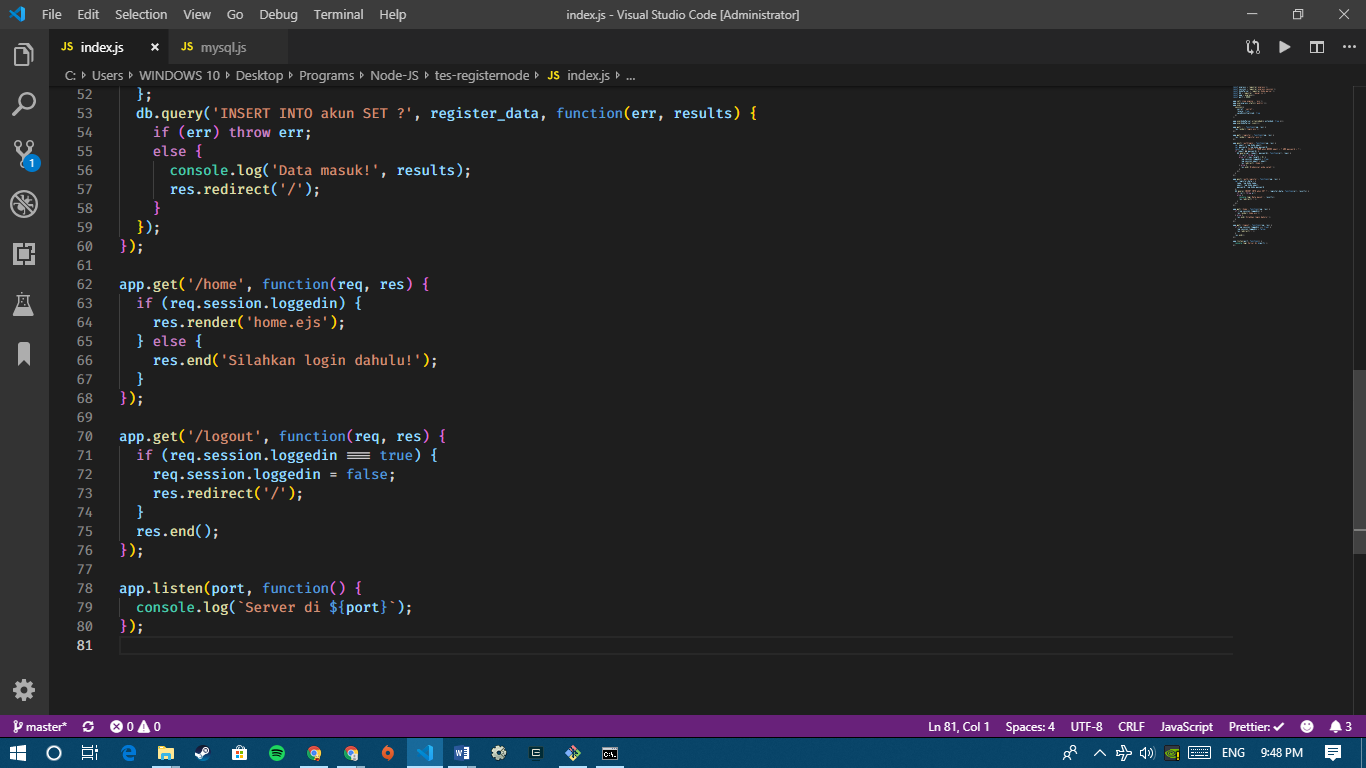
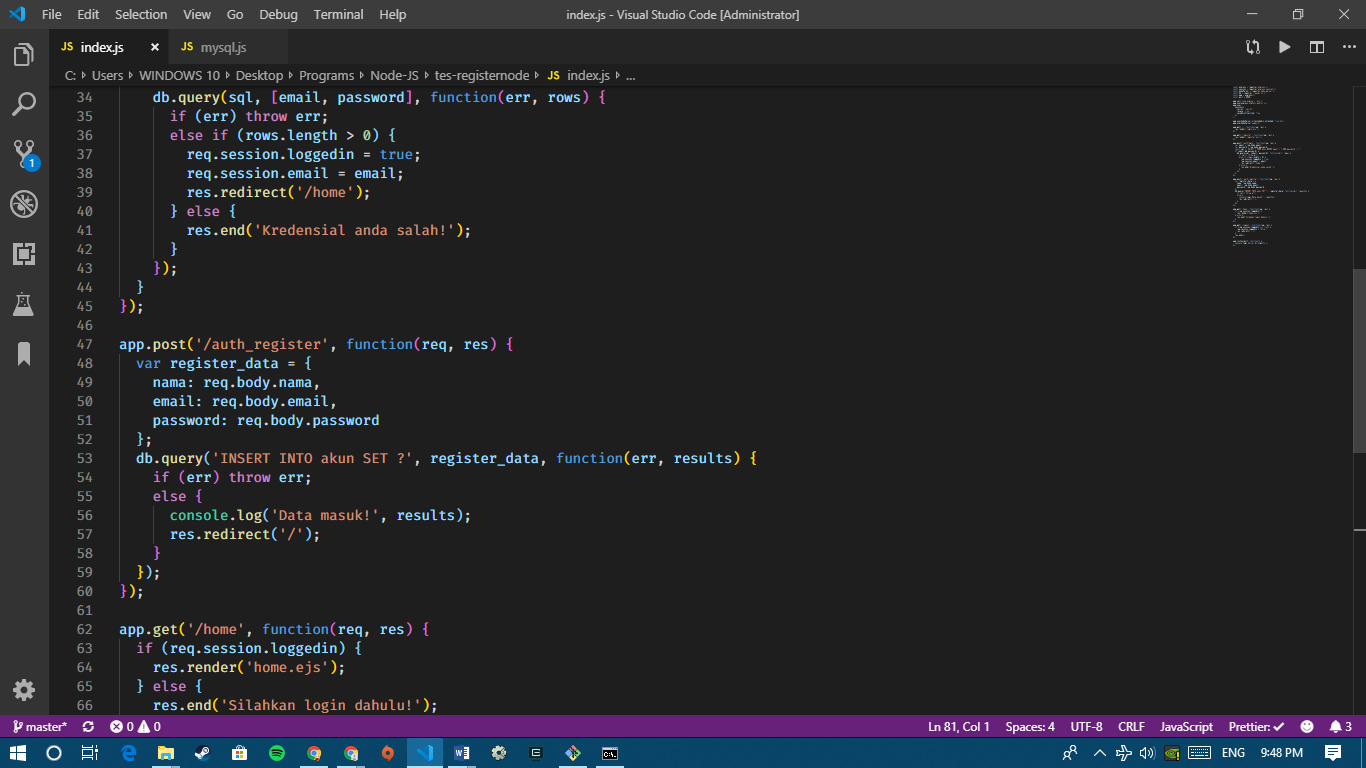
Di tabel *database* ini sama dengan tabel *database* *Login* tadi, hanya ditambahkan kolom nama yang nanti hanya digunakan sebagai pelengkap saat proses *Register*.

Juga setelah proses ini, tidak perlu mencoba *Insert* data lewat PHPMyAdmin, karena kita akan mencobanya langsung dari web yang nanti kita buat!

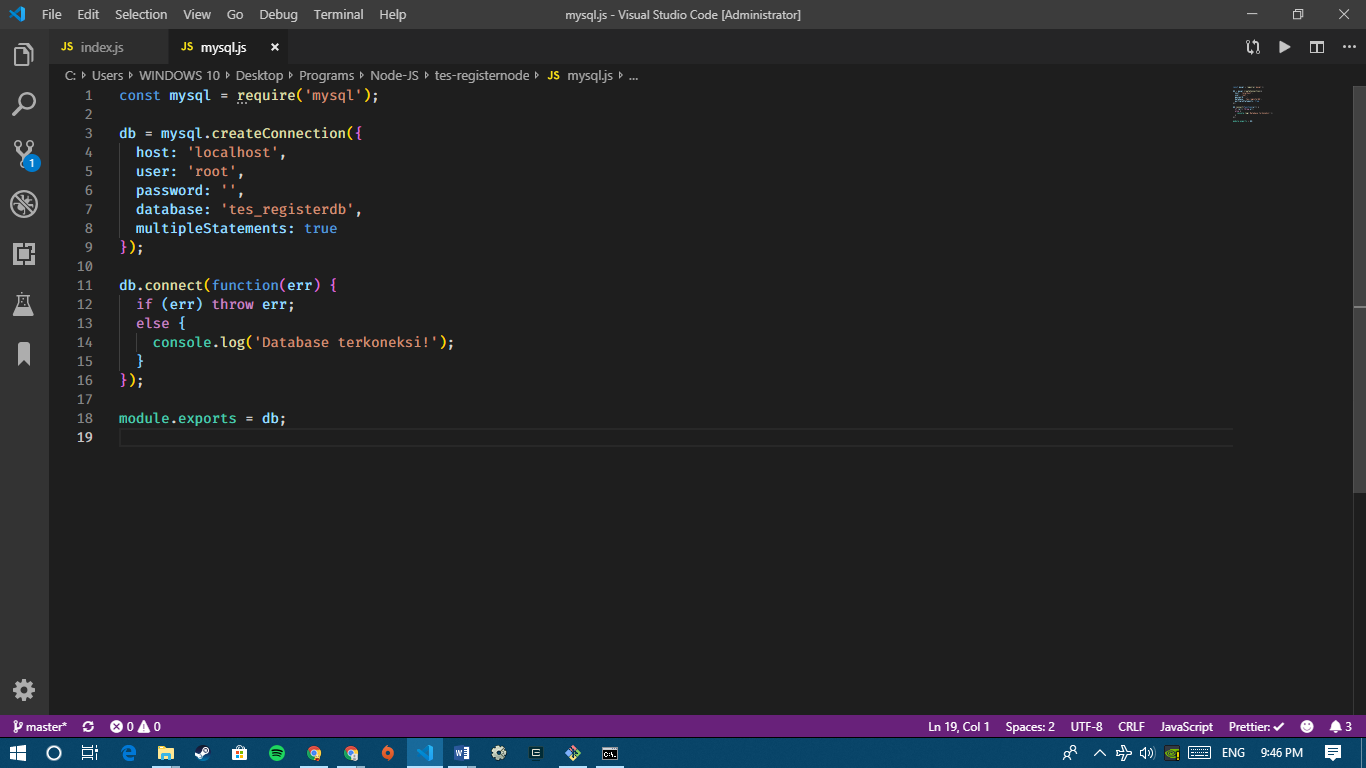
Dibawah ini, kita akan memulai proses *coding*-nya!

1. Buatlah file baru bernama “index.js” di folder projek anda.
2. Ketikan kode-kode dibawah di dalam file “index.js”:

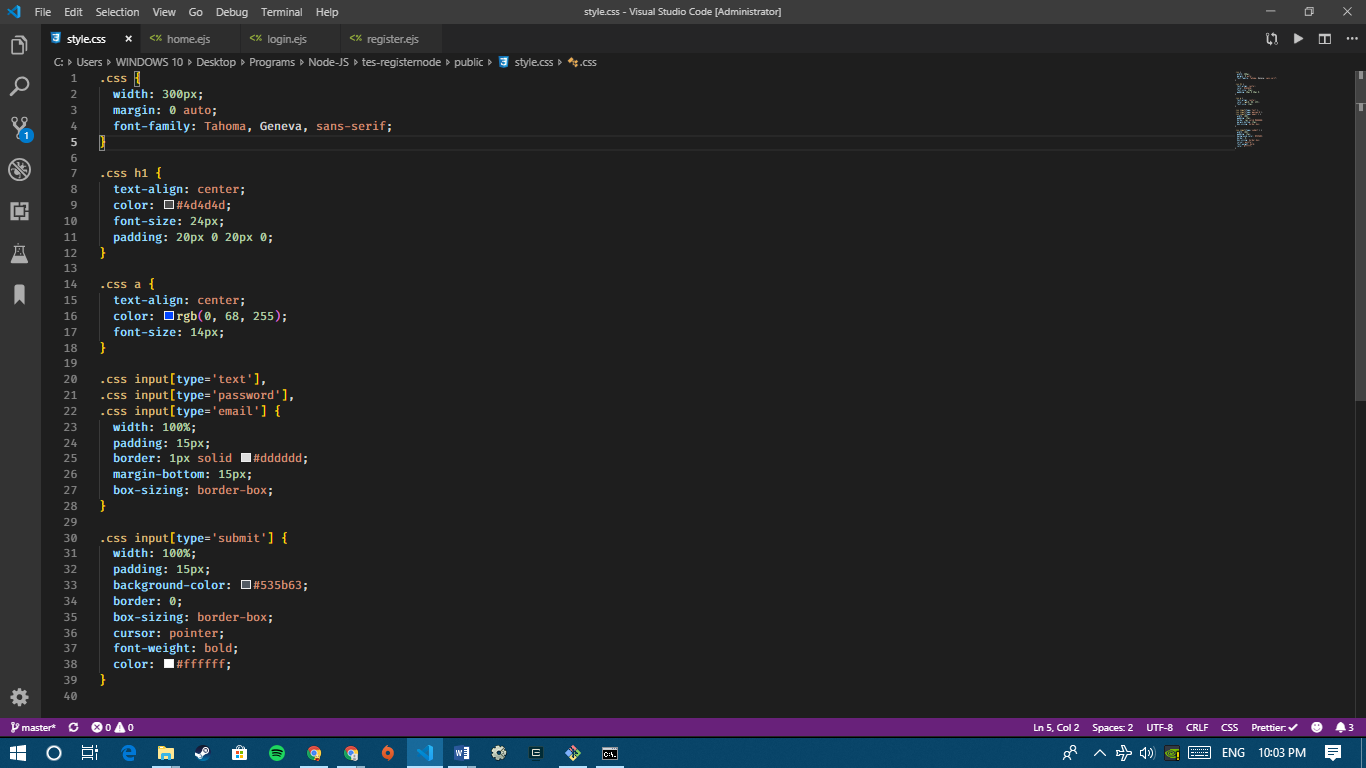




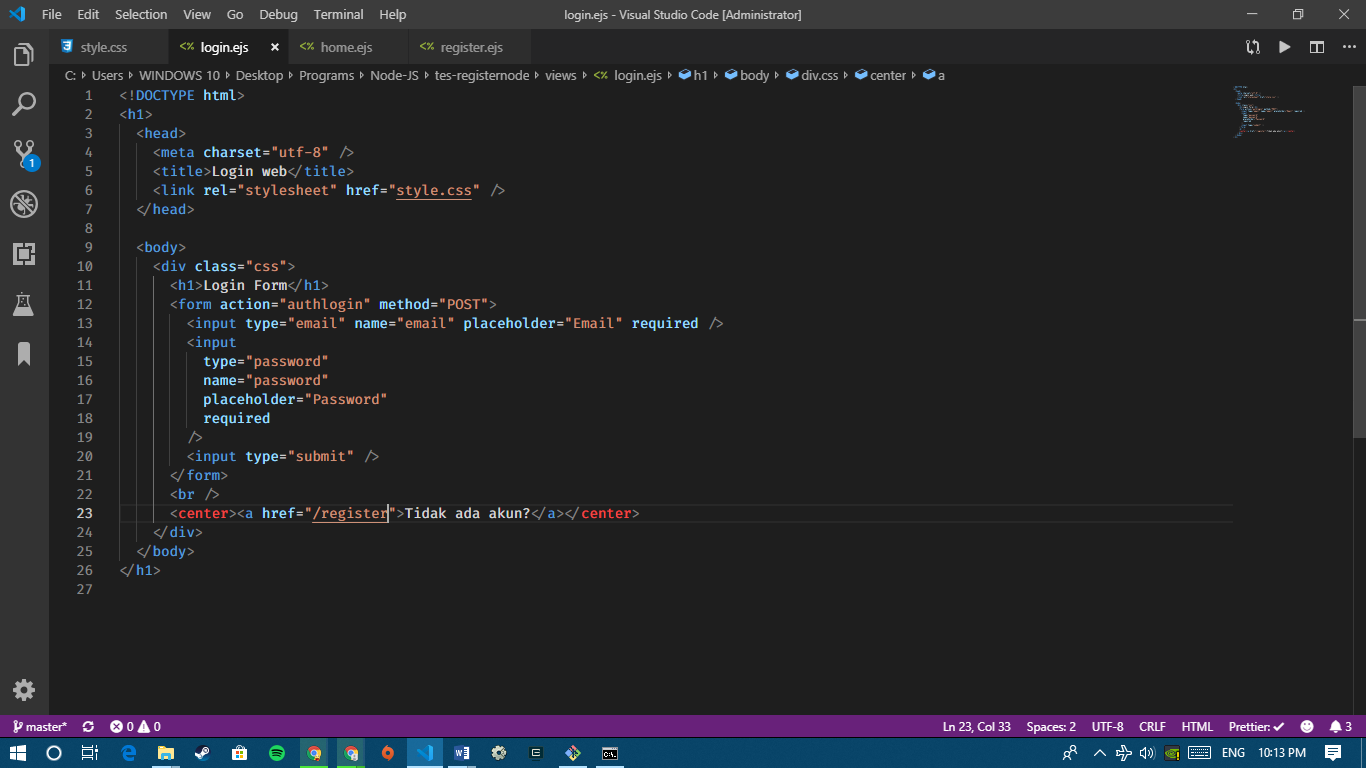
1. Buatlah file baru dengan nama “mysql.js”, dan ketikan kode dibawah:



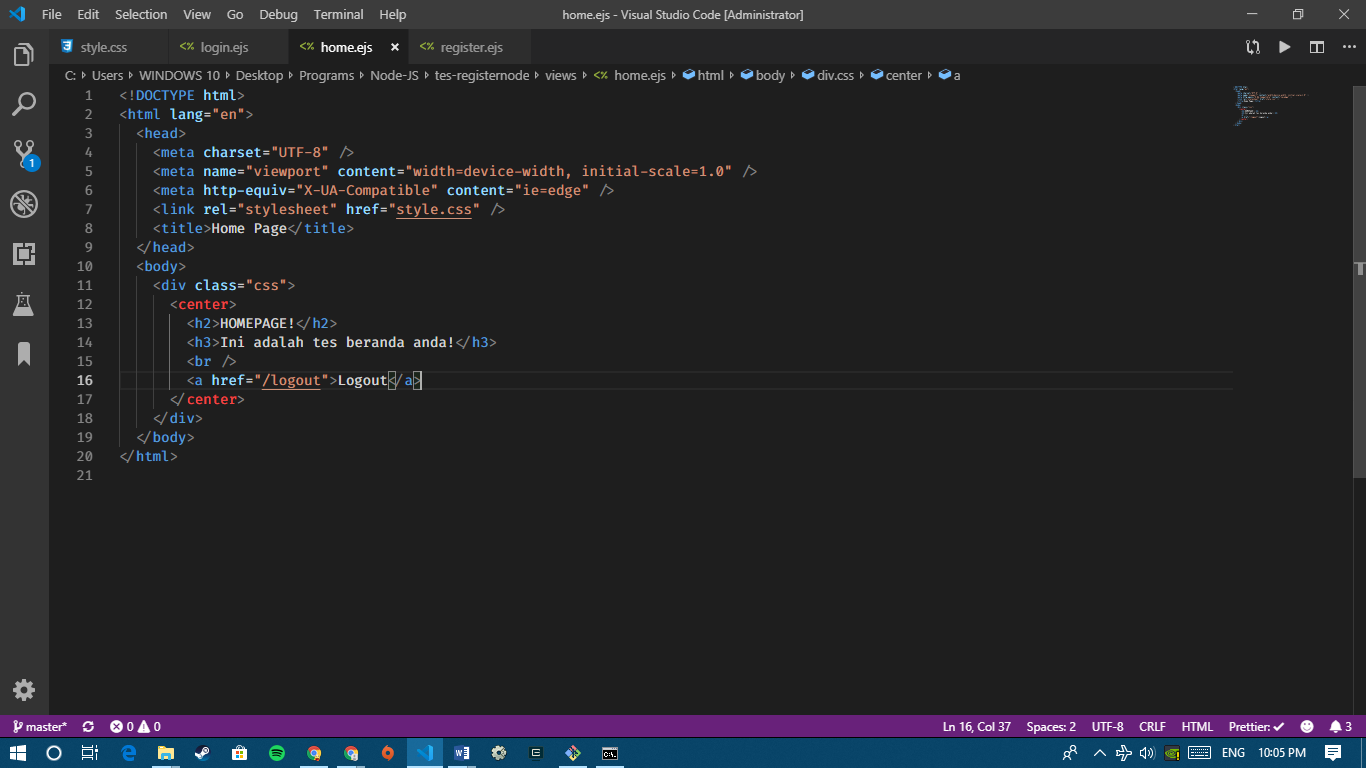
1. Silahkan buat 2 folder baru bernama “public” dan “views.
2. Pergi ke folder public, dan buat file dengan nama “style.css”.
3. Ketikan kode berikut di “style.css”:



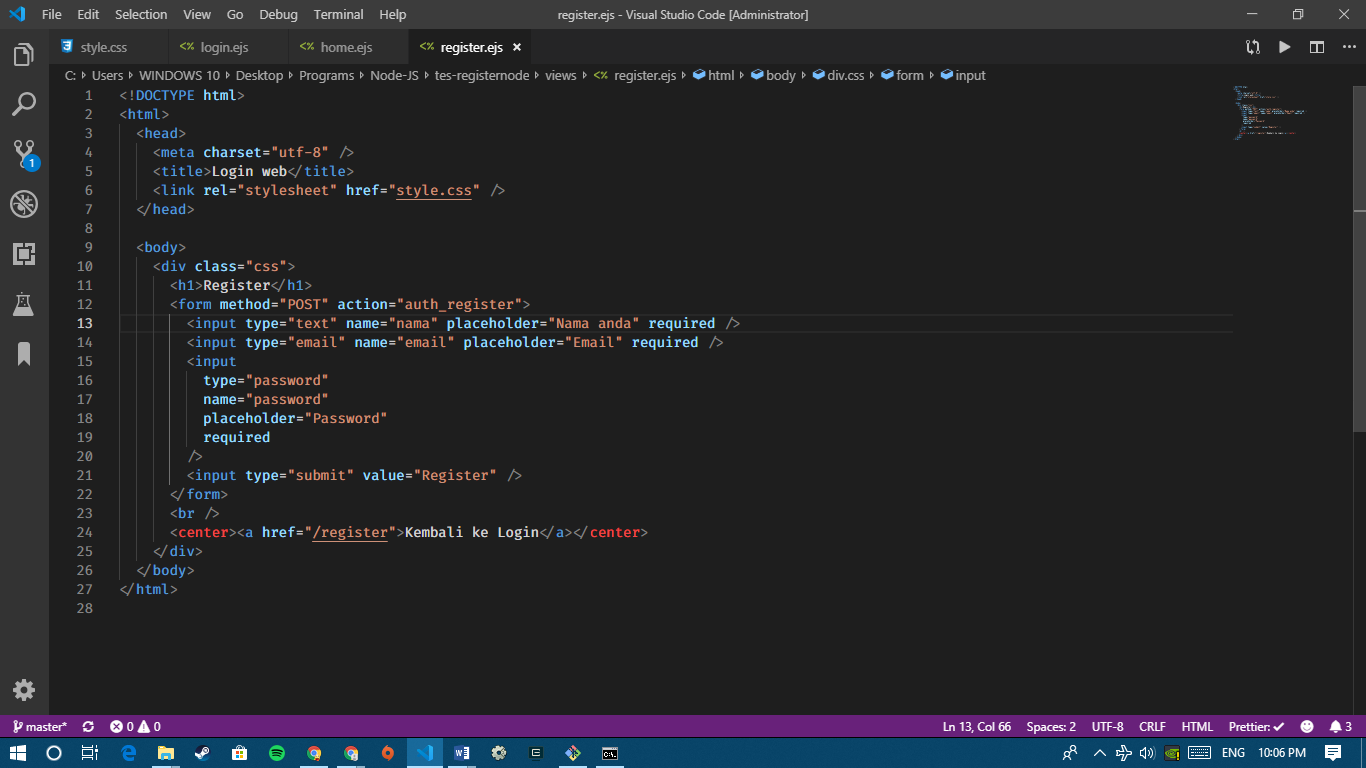
1. Pergi ke folder views, dan buat 3 file baru, “login.ejs, “home.ejs”, dan “register.ejs”.
2. Di “login.ejs”, ketikan kode HTML berikut ini:



1. Di “home.ejs”, ketikan kode HTML berikut:



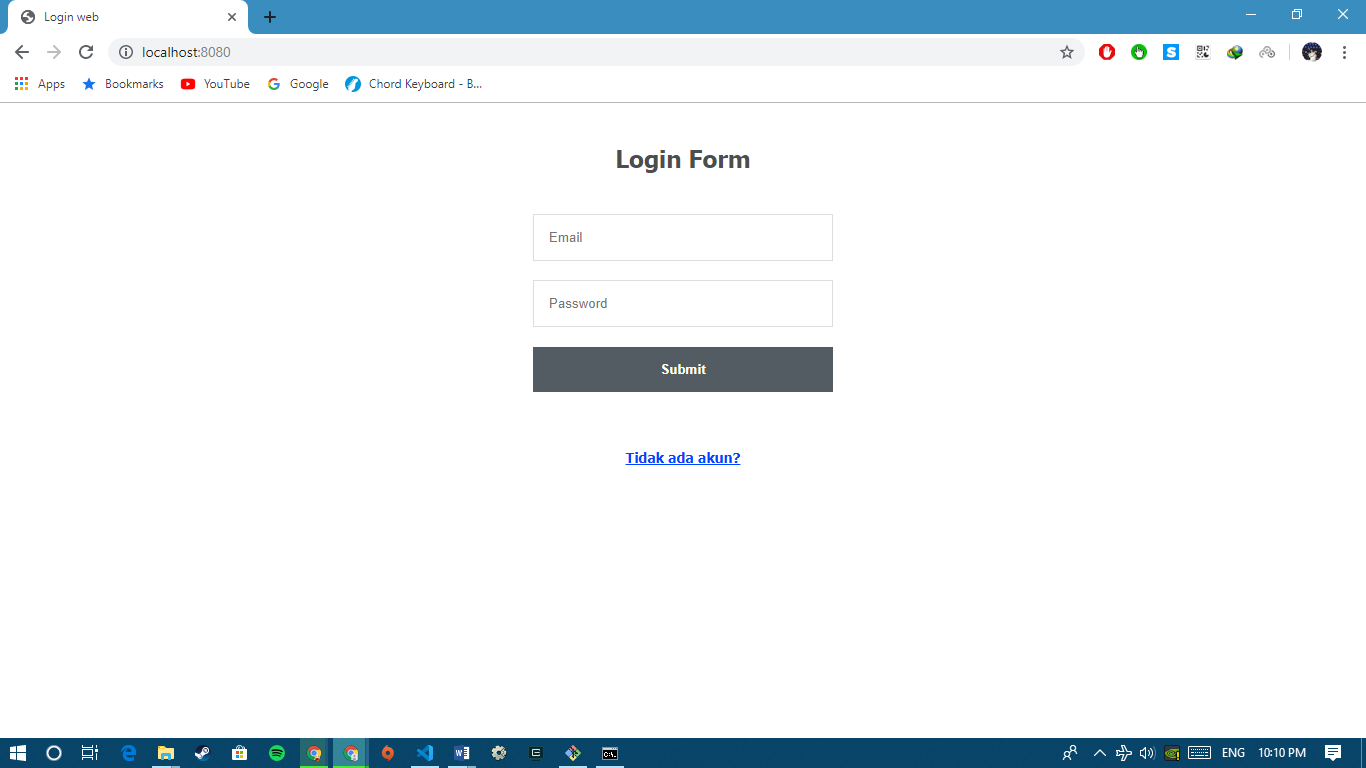
1. Di “register.ejs”, ketikan kode HTML berikut:



Setelah sudah, maka bagian *coding*-pun selesai!

Kitapun sudah dapat mencoba aplikasi ini:

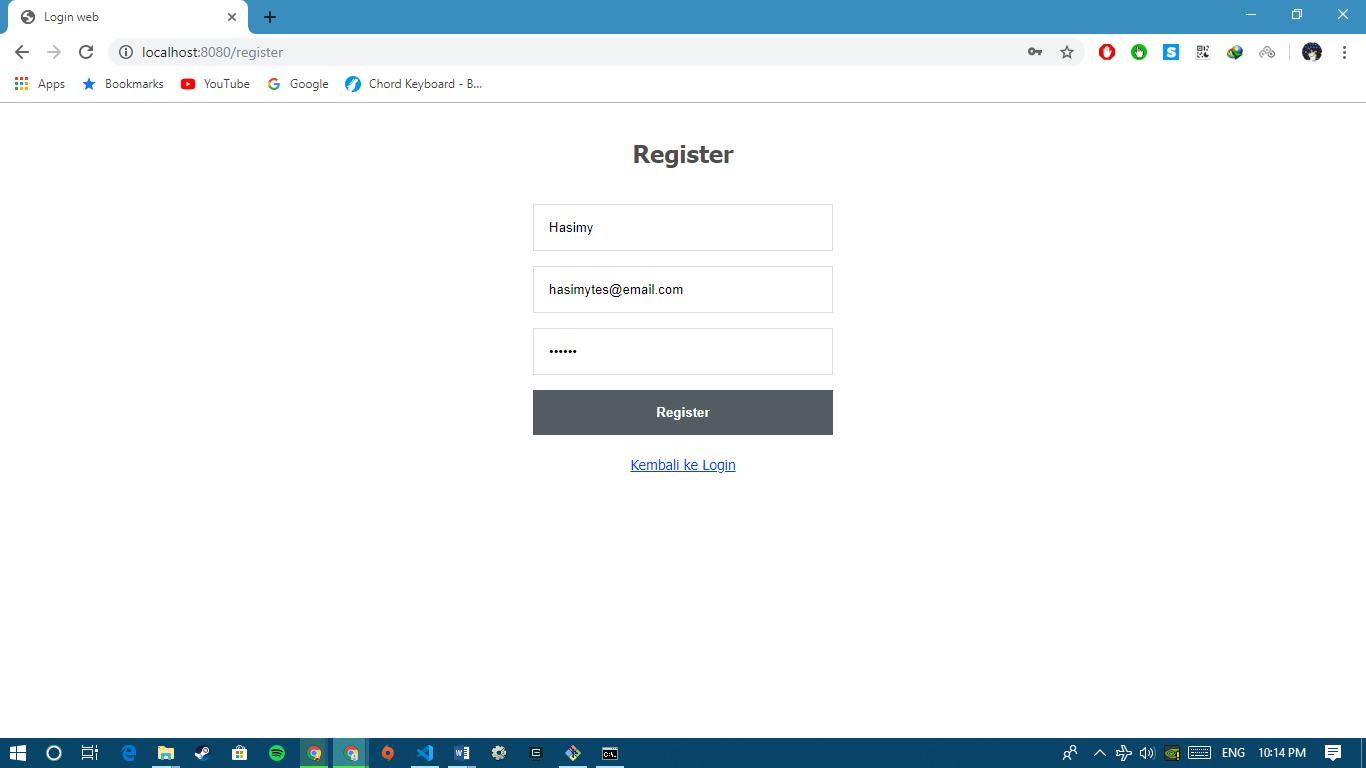
1. Pergi ke folder projek, dan ketikan “node index.js” di CLI/Terminal anda.
2. Jika sudah, buka browser anda, dan ketikan “localhost:8080/”.
3. Maka hasilnya akan seperti dibawah:



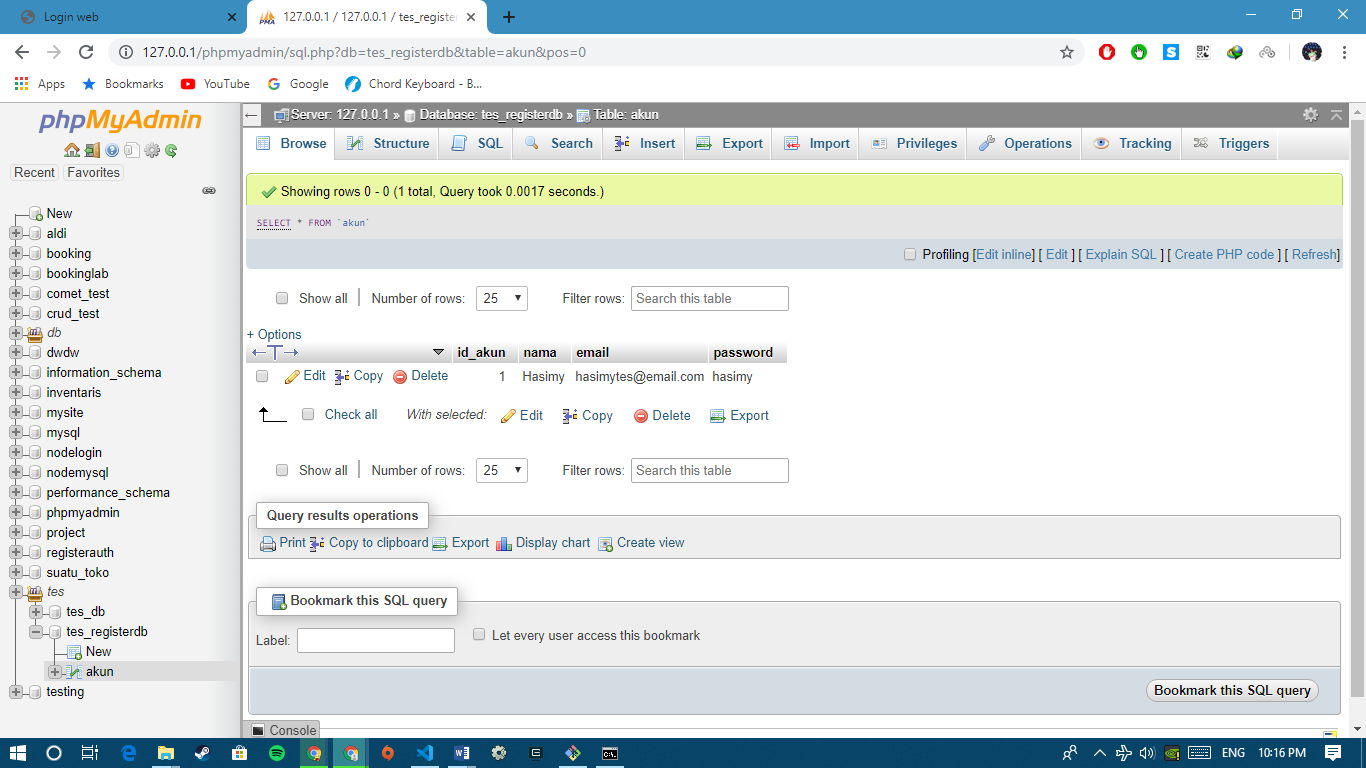
(Karena kita tidak *insert* data ke *database*, maka kita harus masukan lewat sini, mari kita testing!)

1. Klik tombol “Tidak ada akun?” untuk menuju ke form registrasi.
2. Jika sudah, anda dipersilahkan untuk *input* data seperti nama, email, dan password.

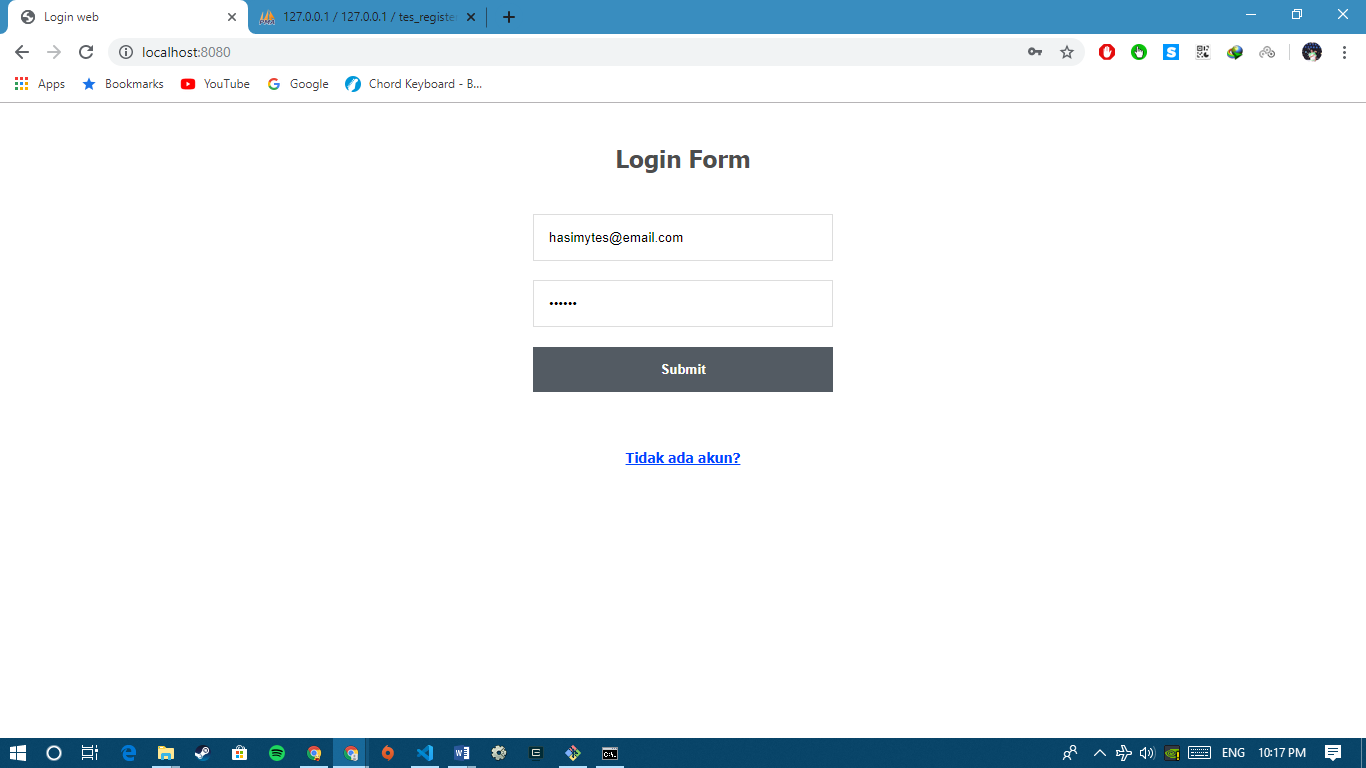
Contoh dibawah:



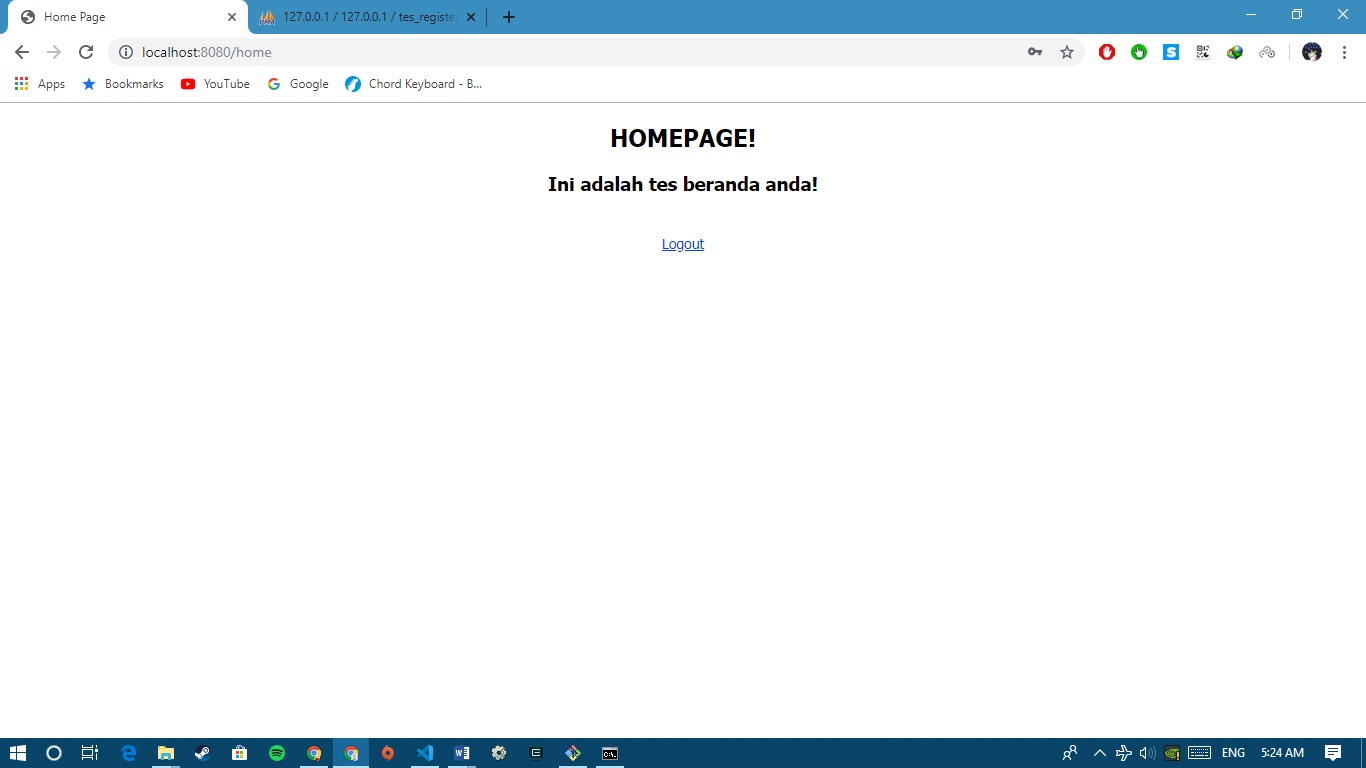
1. Jika sudah, klik “Register”. Dari sini pula, untuk memastikan data masuk ke *database*, silahkan buka PHPMyAdmin, dan menuju *database* anda, dan tabelnya.



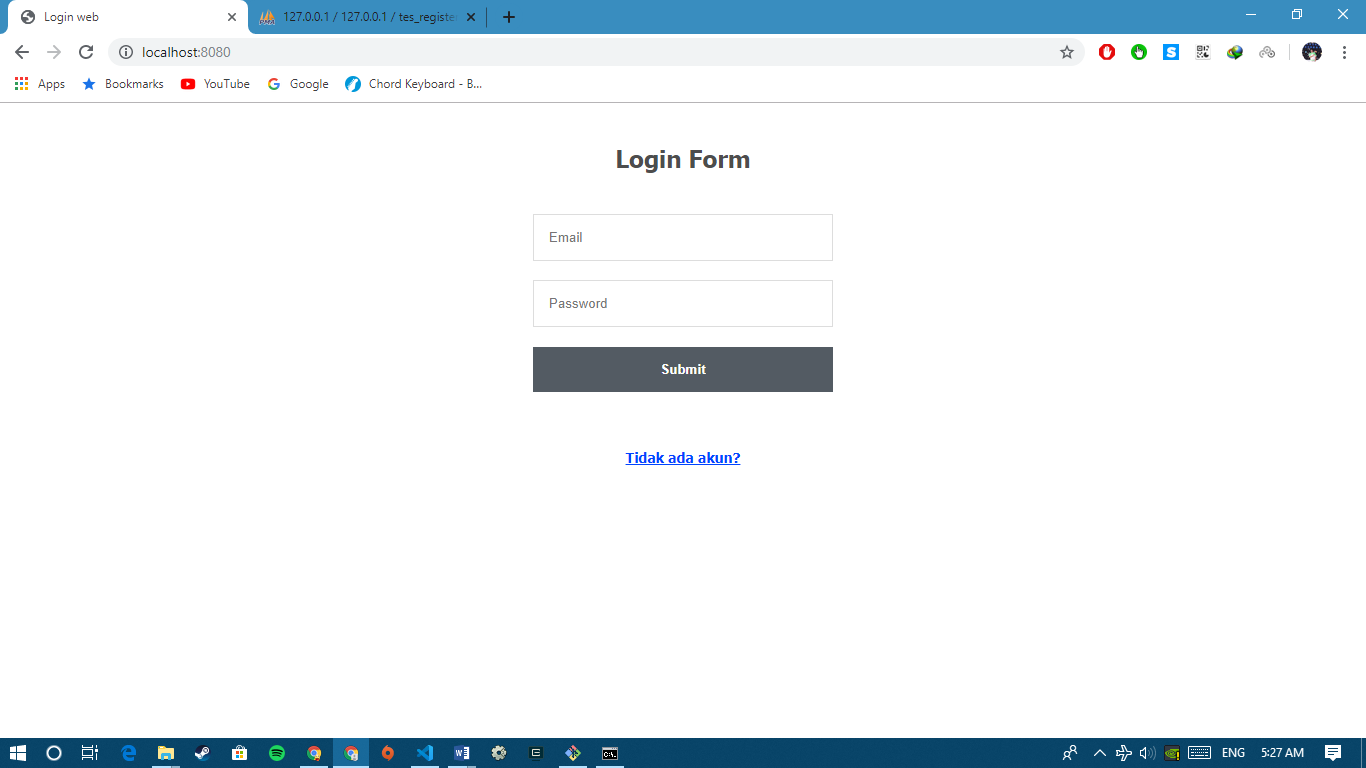
1. Setelah dari situ, kembali ke laman web projek, dan cobalah login dengan akun dari *database* yang anda registrasikan tadi.



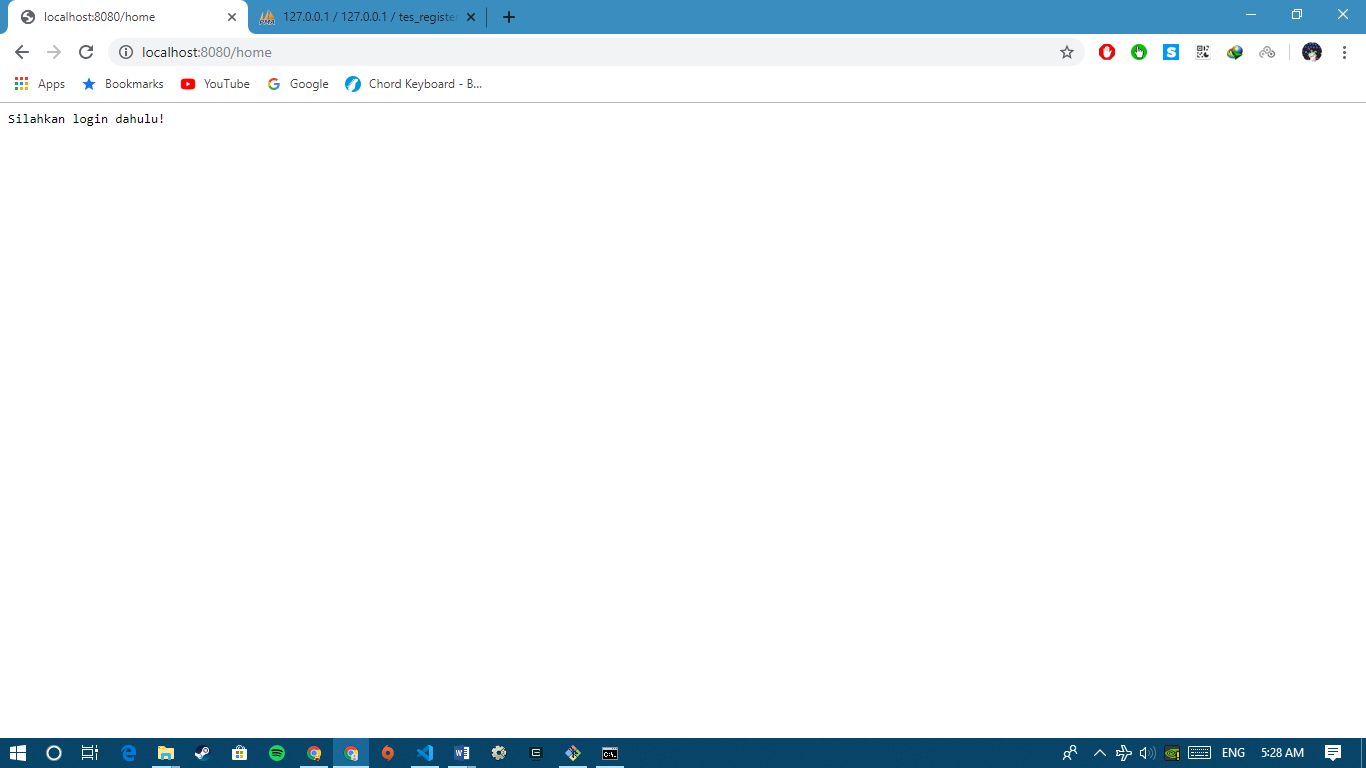
1. Sistem *login* anda akan berhasil jika sudah memasuki laman berandanya.



1. Untuk mencoba sistem *logout,* silahkan klik tombol “*Logout*”.
2. Jika sistem *logout* berhasil, maka laman web akan kembali ke bagian *Login* lagi.



1. Untuk memastikan, silahkan coba akses kembali bagian beranda laman web anda dengan link “localhost:8080/home”, Jikalau ada peringatan harus login terlebih dahulu, maka sistem sesi *login/logout* anda berhasil!



Itulah yang disebut sistem sesi *Login*/*out*, dimana setelah *login*, akan dibuat sesi akun *user* agar dapat mengakses bagian laman beranda. Jika tombol *Logout* telah diklik, maka sesi akun *user* akan dihilangkan, *user*-pun kembali ke laman *login*, dan untuk mengakses bagian beranda harus *login* lagi terlebih dahulu.

Itu pula kegunaan daripada *dependency* bernama **express-session** yang kita install.

1. **Penutup**

Puji Tuhan YME, selesailah **Module MySQL Node.js** pada kali ini, semoga bermanfaat bagi kita semua, dan mari bersama kita kembangkan dunia pemrograman Back-End dengan Node.js! Sampai jumpa di Dokumenter Node.JS yang lain!



Sekian, terima kasih.

Salam Sinergi Bangun Negeri!