

MODUL 6

PEMROGRAMAN WEB

DATABASE

A. TUJUAN PERKULIAHAN

- a. Mahasiswa mengenali konsep database dalam pemrograman web
- b. Mahasiswa mampu membuat koneksi database dengan aplikasi

B. ALOKASI WAKTU

3 x 50 menit

C. DASAR TEORI

1. DATA

Secara umum, definisi data yaitu sekumpulan fakta atau keterangan yang belum diolah serupa angka, kata-kata, citra/gambar, atau simbol yang diperoleh melalui proses pencarian atau pengamatan dari sumber-sumber tertentu.

Secara bahasa (etimologis), istilah kata “data” berasal dari bahasa latin yaitu “**datum**” yang bermakna sesuatu yang diberikan. Jadi, dapat disimpulkan bahwa pengertian data adalah **kumpulan deskripsi dasar atau keterangan-keterangan dari suatu objek yang didapatkan dari hasil observasi (pengamatan)** serta diolah menjadi bentuk yang mudah untuk dipahami seperti informasi, solusi, database, dan lainnya.

Pada umumnya kaitan antara informasi dengan data saling berhubungan. **Informasi didapatkan dengan cara mengolah data yang telah dikumpulkan sebelumnya.** Sebaliknya, sekumpulan data menjadi sumber pembentuk sebuah *informasi*. Walaupun saling berkaitan dan melengkapi tetapi keduanya memiliki definisi yang sangat berbeda.

Data bersifat mentah, jadi di dalam membuat sebuah informasi diperlukan beberapa data. Pengaplikasian data banyak kita jumpai pada saat mempelajari ilmu eksakta, ilmu

komputer, dan ilmu statistik. Semua bidang ilmu pengetahuan yang berkaitan dengan statistik maka akan menggunakan data dalam jumlah yang tidak sedikit.

Contoh data: data penjualan sebuah perusahaan diolah dalam aplikasi komputer akan menghasilkan laporan penjualan, yang akan mencantumkan besar kecilnya penjualan yang didapat per periodenya.

Contoh informasi: Pihak kepolisian jakarta menyelidiki kasus dengan mengumpulkan berbagai data di TKP, hasilnya akan diperoleh informasi mengenai siapa tersangka sebenarnya.

Data	Informasi
Fakta yang disimpan	Fakta yang diungkap
Ada tapi tidak aktif	Keberadaannya aktif
Berdasarkan teknologi	Berdasarkan bisnis
Dikumpulkan dari berbagai sumber	Ditransformasikan dari data

2. DATABASE

Database atau basis data adalah **kumpulan data yang dikelola sedemikian rupa berdasarkan ketentuan tertentu yang saling berhubungan sehingga mudah dalam pengelolaannya**. Melalui pengelolaan tersebut pengguna dapat memperoleh kemudahan dalam mencari informasi, menyimpan informasi dan membuang informasi.

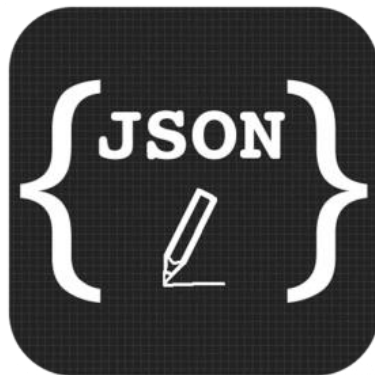
Adapun pengertian lain dari database adalah sistem yang berfungsi sebagai mengumpulkan *file*, tabel, atau arsip yang terhubung dan disimpan dalam berbagai media elektronik. Berikut beberapa jenis database yang biasa terdapat di dalam sistem komputer

a. Operational Database

Operational Database atau biasa disebut dengan *database OLTP (On Line Transaction Processing)*, berguna untuk mengelola data yang dinamis secara langsung atau *real-time*. Jenis ini memungkinkan para pengguna dapat melakukan, melihat, dan memodifikasi data. Modifikasi tersebut bisa berupa mengubah, menambah, menghapus data secara langsung melalui perangkat keras yang digunakan.

1. JSON

JSON (*JavaScript Object Notation*) adalah format *file* yang menggunakan teks untuk pengiriman data. Format ini sangat umum digunakan pengguna untuk pertukaran data layaknya berkomunikasi cepat dengan melalui web browser dan web server. Sinkronisasi data bisa dilakukan secara *real-time*.



Format JSON sendiri berasal dari pemrograman JavaScript, melalui pembuatannya JSON memiliki format bahasa yang berbeda dari lainnya. Diketahui bahwa semua file JSON selalu menggunakan ekstensi khusus berupa “.**json**”.

```
personal.json x
1 {
2   "$schema": "personal-schema.json",
3   "personnel": {
4     "person": [
5       {
6         "id": "Big.Boss",
7         "name": {
8           "family": "Boss",
9           "given": "Big"
10        },
11        "email": "chief@oxygenxml.com",
12        "link": {
13          "subordinates": [
14            "one.worker",
15            "two.worker",
16            "three.worker",
17            "four.worker",
18            "five.worker"
19          ]
20        }
21      }
22    ]
23  }
24 }
```

2. XML

XML (*Extensible Markup Language*) adalah bahasa program *markup* yang memiliki aturan untuk memberikan dua kode dokumen berbeda yang bisa dibaca oleh manusia dan dibaca oleh komputer. Melalui XML, akan menghasilkan format data berupa teks yang dapat digunakan untuk merepresentasikan struktur basis data. Selain itu, sinkronisasi data dapat dilakukan secara *real-time* oleh pengguna.



Bahasa program ini sangat cocok digunakan untuk menangani basis data pada web browser dan web server. Struktur yang digunakan XML dinilai banyak memiliki kesamaan dengan format JSON.

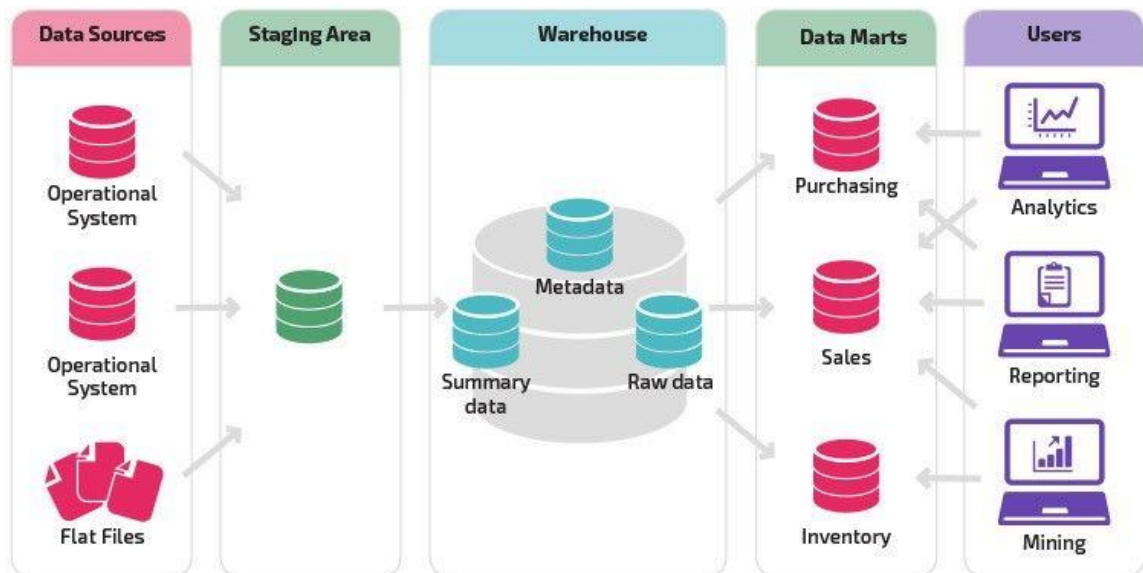
```

- <Parts>
- <Part>
  <Id>4478</Id>
  <Part_Name>1000 Ohm Resistor</Part_Name>
  <Total_Available>25000</Total_Available>
  <Price>0.01</Price>
</Part>
- <Part>
  <Id>3328</Id>
  <Part_Name>15000 Ohm Resistor</Part_Name>
  <Total_Available>75000</Total_Available>
  <Price>0.02</Price>
</Part>
- <Part>
  <Id>4725</Id>
  <Part_Name>555 Timer IC</Part_Name>
  <Total_Available>1500</Total_Available>
  <Price>0.25</Price>
</Part>
</Parts>

```

b. Database Warehouse

Database Warehouse adalah sistem basis data yang biasa digunakan untuk pelaporan dan analisis data. Sistem ini dianggap sebagai komponen inti dari *business intelligence*. *Database Warehouse* merupakan repositori sentral data yang terpadu dari satu atau lebih sumber yang berbeda. *Database* tersebut juga menyimpan data terkini dan historis dengan satu tempat yang digunakan untuk membuat laporan analisis.



Data yang tersimpan di *warehouse* awalnya diunggah dari sistem operasi. Data bisa melewati penyimpanan operasional dan memungkinkan untuk pembersihan data. Proses tersebut menjadi operasi tambahannya dan dapat memastikan kualitas data sebelum digunakan di *warehouse* sebagai pelaporannya. Contoh dari database warehouse adalah **Microsoft SQL Server**.



Microsoft SQL Server adalah sistem basis data yang dibangun oleh Microsoft. Sebagai *server database*, sistem ini merupakan produk perangkat lunak yang berfungsi menyimpan dan mengambil data sesuai permintaan aplikasi lainnya. Hal tersebut memungkinkan dapat berjalan baik melalui komputer yang sama atau komputer lainnya melalui jaringan internet.

Setidaknya Microsoft pernah memasarkan 12 edisi yang berbeda sistem Microsoft SQL Server ini. Hal itu ditujukan untuk memberikan pilihan kepada pengguna dan untuk kebutuhan yang berbeda juga.

c. **Distributed Database**

Distributed Database adalah basis data yang perangkat penyimpanannya tidak terpasang pada perangkat komputer yang sama. Basis data tersebut disimpan di beberapa perangkat komputer yang terletak di tempat yang sama atau tersebar melalui jaringan komputer lainnya yang saling berhubungan.

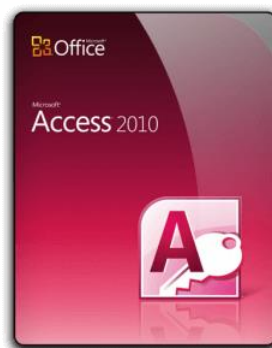
Distributed databases vs. centralized databases

Here are 5 main differences between distributed databases and centralized database systems.

Distributed database	Centralized database
Consists of multiple database files located at different sites	Consists of a single, central database file
Allows multiple users to access and manipulate data	Bottlenecks when multiple users access same file simultaneously
Files delivered quickly from location nearest the user	Files may take longer to deliver to users
If one site fails, data is retrievable	Single site means downtime in cases of system failures
Multiple files from dispersed databases must be synchronized	Simpler to update and manage data in single, central system

Sistem ini tidak sama dengan sistem paralel yang menggabungkan erat dan bersistem data tunggal. Sistem ini terdistribusi melalui situs yang tergabung dan tidak memiliki komponen fisik.

Melalui administrator, basis data dapat mendistribusikan sekumpulan data di beberapa lokasi yang berada di server jaringan terorganisir. Karena sistem yang begitu unik, basis data terdistribusi bisa meningkatkan kinerja bagi *end user* dengan membiarkan transaksi melalui proses mesin yang banyak sehingga tidak fokus pada satu mesin saja. Contoh distributed database adalah **Microsoft Access (Office)**

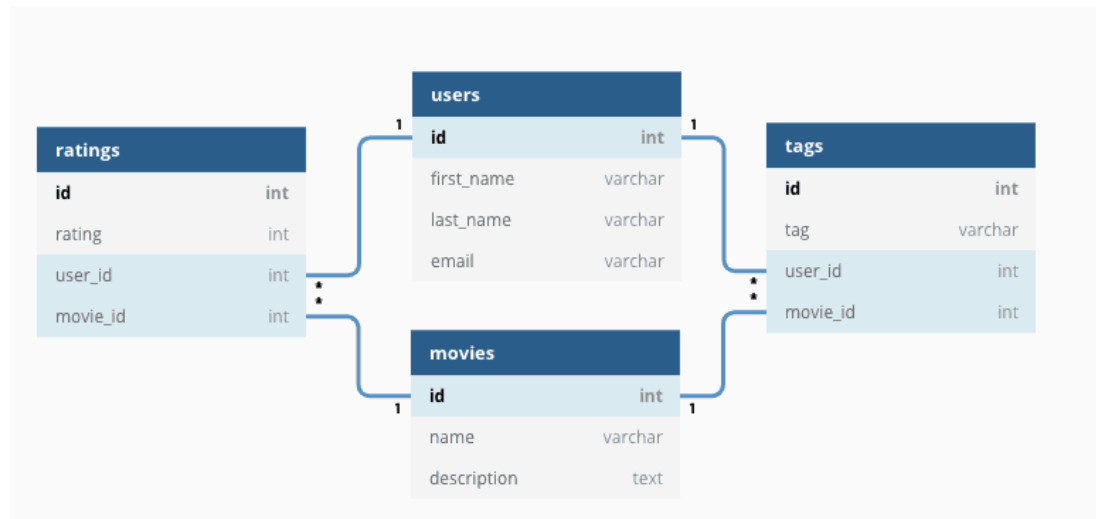


Microsoft Access adalah sistem DBMS yang menggabungkan Microsoft Jet Database Engine dengan alat pengembang perangkat lunak. Microsoft Access menyimpan data dengan formatnya sendiri. Melalui *software* ini pengguna dapat mengimpor atau menghubungkan langsung ke data yang tersimpan di *database* lainnya.

Microsoft Access sangat cocok digunakan pada sistem informasi dengan *distributed database*. Karena penyimpanan *file* tidak memerlukan *server database* aktif sehingga bersifat *portable*.

d. Relational Database

Relational Database atau basis data relasional adalah basis data yang mengorganisir berdasarkan model hubungan data. Banyak sekali perangkat lunak yang menggunakan sistem ini untuk mengatur dan memelihara basis data melalui hubungan setiap data. Umumnya, semua sistem menggunakan *Structured Query Language* (SQL) sebagai bahasa pemrograman untuk pemeliharaan basis data dan *query*.



e. End-User Database

End user database adalah database pengguna akhir yang terdiri dari dari berbagai jenis file yang dikelola dan dikembangkan oleh end user workstation. Contoh database end user ini

adalah dokumen spreadsheet, word processing bahkan file download juga termasuk dari end user database. Contoh dari end-user database adalah **SQLite**.



SQLite adalah sistem manajemen basis data yang ada pada library pemrograman C. Berbeda dengan sistem lainnya, SQLite bukan merupakan mesin *database client server*. SQLite tertanam ke dalam program akhir sehingga cocok digunakan dalam mendukung penyimpanan data akhir *end user*.

SQLite sangat populer digunakan sebagai perangkat lunak *database* untuk penyimpanan lokal / klien melalui perangkat lunak aplikasi seperti peramban web. Sistem ini merupakan sistem yang paling banyak digunakan melalui sistem operasi, peramban web, dan sistem *embedded* yang lebih luas seperti ponsel.

3. MANFAAT

A. Kecepatan dan kemudahan

Sistem *database* memberikan kemampuan dalam seleksi data menjadi satu kelompok yang terurut dengan cepat. Instrumen tersebut menghasilkan pencarian informasi yang dibutuhkan ditemukan dengan cepat. Kecepatannya juga dipengaruhi oleh jenis *database* yang digunakan. Setiap jenis *database* memberikan kemampuan yang berbeda-beda.

B. Multi-user

Database memberikan kemudahan akses bagi banyak pengguna dalam waktu yang bersamaan. Sistem tersebut memungkinkan akses suatu dokumen ke lebih dari satu pengguna. Sehingga

kinerja mesin dan jaringan dimudahkan melalui *multi-user* karena penyimpanan hanya terdiri satu unit yang dapat diakses secara bersamaan.

C. Keamanan data

Sistem *database* melalui bahasa pemrogramannya telah dibuat secara *safety*. Melalui instrumen *password* membuat data tersebut hanya bisa diakses kepada pihak yang diijinkan. Manajemen tersebut telah diterapkan pada hampir seluruh jenis sistem *database*. Sehingga menjadikan keamanan data merupakan hal prioritas bagi layanan sistem *database*.

D. Penghematan biaya perangkat

Memiliki satu *database* terpusat sudah cukup bagi perusahaan besar yang membutuhkan pengumpulan data secara ringkas. Hal ini membuat perusahaan tidak memerlukan ruang penyimpanan di tiap tempat yang berbeda. Melalui jaringan internet, cabang perusahaan di daerah terpencil pun bisa melakukan akses data yang ada di pusat.

E. Kontrol data terpusat

Database tidak memerlukan *server* lebih dari satu dalam penggunaannya. Cukup satu *server* terpusat untuk menyimpan data sehingga data tersebut bisa diakses oleh banyak pengguna. Hal ini memberikan harga yang murah bagi perusahaan untuk investasi ruang penyimpanan data penting perusahaan.

Seperti kantor perusahaan tidak perlu membuat suatu data di tiap divisi jobnya. Setiap divisi bisa mengumpulkan data khusus melalui satu *server* yang ditentukan sehingga laporan untuk atasan menjadi ringkas.

F. Mudah membuat aplikasi

Melalui kaitannya terhadap perusahaan jika perusahaan membutuhkan aplikasi *input* data yang baru, programmer tidak perlu membuat ulang struktur *database*. Menggunakan struktur *database* yang dibuat sebelumnya sudah cukup untuk mengenali aplikasi input data yang baru.

4. DBMS

1. MySQL



MySQL adalah sebuah sistem untuk manajemen basis data relasional. Banyak sekali produk-produk IT yang dibuat dengan menggunakan komponen utama MySQL. Beberapa aplikasi seperti WordPress, Google, Flickr, Youtube, Facebook, Joomla, phpBB, Drupal, dan MODx menggunakan sistem ini untuk manajemen basis data relasional mereka.

2. PostgreSQL



Sistem kedua yang merupakan sistem pengelolaan basis data relasional adalah PostgreSQL. Sistem ini berfungsi untuk menyimpan data secara aman dan dapat mengembalikan data tersebut sebagai respon atas *request* dari aplikasi lainnya. PostgreSQL dapat bekerja melalui aplikasi mesin tunggal kecil hingga aplikasi internet besar beserta pengguna yang banyak secara bersamaan. Sistem ini biasa digunakan pada sistem operasi mac OS server, karena pengaturannya sudah tersedia secara *default*. Sistem operasi lainnya seperti Windows dan Linux juga dapat ditemukan dengan mengubah pengaturannya.

3. MariaDB



MariaDB adalah sistem yang dikembangkan dari MySQL. Pengembangan ini bertujuan untuk mempertahankan kompatibilitas yang tinggi dari MySQL dan cocok dengan API MySQL beserta perintah-perintahnya. MariaDB memiliki mesin penyimpanan XtraDB untuk mengganti InnoDB.

Sistem ini dikembangkan langsung oleh beberapa pengembang asli MySQL dengan pihak yang bercabang. Tujuannya adalah agar tidak diakuisisi oleh Oracle Corporation yang merupakan kompetitornya.

4. MongoDB



MongoDB adalah *software database* yang berorientasi pada dokumen *cross platform* dan *open source*. MongoDB menggunakan dokumen yang mirip dengan skema JSON, oleh karena itu sistemnya diklasifikasikan ke dalam program basis data NoSQL.

5. Oracle Database



Oracle Database adalah sistem *relation database* selanjutnya yang diproduksi dan dipasarkan oleh perusahaan Oracle. Dalam penggunaannya, sistem Oracle mengacu pada struktur memori *server-side* sebagai sistem area globalnya. Sistem area global dapat menyimpan *cache*, perintah SQL, dan informasi pengguna. Selain itu, sistem ini memungkinkan untuk menyimpan riwayat transaksional seperti *redo log online*.

6. SAP HANA



SAP HANA adalah sistem lain yang berorientasi pada kolom dan hubungan antar tabel. Sistem ini memiliki fungsi utama sebagai *database server* yang menyimpan dan mengambil data sesuai permintaan aplikasi. Selain fungsi tersebut, SAP HANA juga dapat melakukan analisis lanjutan seperti analisis prediksi, pemrosesan data spasial, analisis teks, analisis *streaming*, pencarian teks, dan pemrosesan data grafik. SAP HANA ini merupakan sistem yang dikembangkan oleh SAP SE.

7. IBM Db2



IBM Db2 adalah sistem lainnya yang dikembangkan oleh perusahaan IBM. Melalui sistem ini, model relasional dapat terdukung. Hanya saja versi terbarunya IBM Db2 memiliki multi-fungsi yang mendukung fitur relasional dan non relasional seperti JSON dan XML.

Melalui sejarahnya, IBM Db2 merupakan sistem yang dibuat untuk produk DB2 pada masing-masing sistem operasi utamanya. Pada tahun 1990-an, Perusahaan IBM mulai membuat produk DB2 universal yang kodenya dapat digunakan untuk sistem operasi lainnya.

8. MemSQL



MemSQL adalah sistem manajemen basis data SQL terdistribusi dan *in-memory*. Sistem ini juga termasuk ke dalam sistem manajemen basis data relasional (RDBMS). MemSQL berfungsi untuk mengkompilasi SQL ke dalam kode mesin melalui proses pembuatan kode atau *code generation*.

9. Firebird



Sistem basis data relasional terakhir adalah Firebird. Firebird adalah sistem SQL yang *open source* dan berjalan di sistem operasi mac OS X, Windows, Linux, dan sistem operasi lainnya yang unik. Basis data Firebird juga merupakan cabang dari *open source* Borland Interbase pada tahun 2000. Hanya saja sejak versi Firebird 1.5 kode yang dibuat sebagian besar telah ditulis ulang.

5. PHPMyAdmin

a. PHPMyAdmin

Pada dasarnya, mengelola basis data dengan MySQL harus dilakukan dengan cara mengetikkan baris-baris perintah yang sesuai (command line) untuk setiap maksud tertentu. Jika seseorang ingin membuat basis data (database), ketikkan baris perintah yang sesuai untuk membuat basis data. Jika seseorang menghapus tabel, ketikkan baris perintah yang sesuai untuk menghapus tabel. Hal tersebut tentu saja sangat menyulitkan karena seseorang harus hafal dan mengetikkan perintahnya satu per satu

Saat ini banyak sekali perangkat lunak yang dapat dimanfaatkan untuk mengelola basis data dalam MySQL, salah satunya adalah phpMyAdmin. Dengan phpMyAdmin, seseorang dapat membuat database, membuat tabel, mengisi data, dan lain-lain dengan mudah, tanpa harus menghafal baris perintahnya.

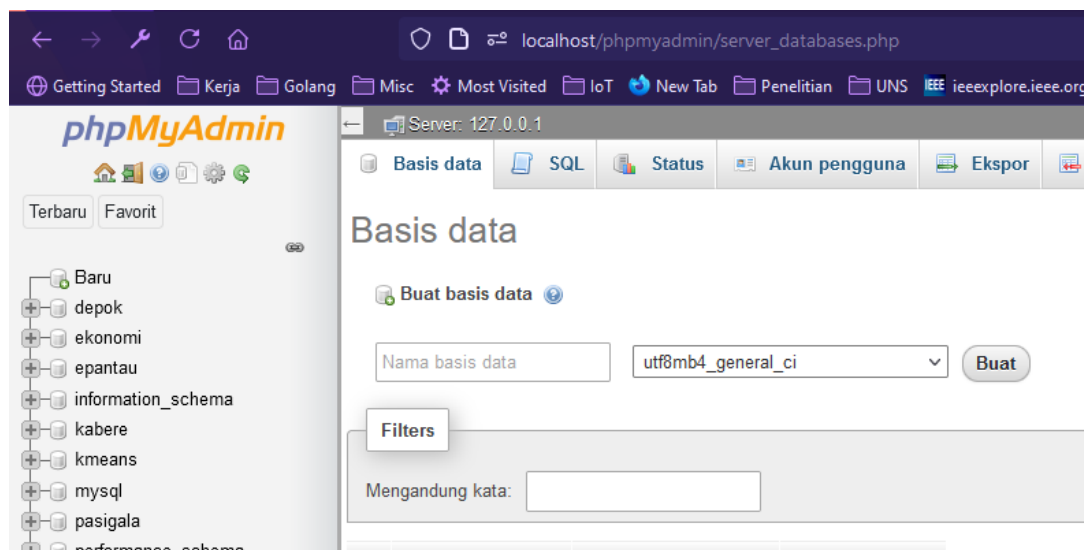


phpMyAdmin merupakan bagian untuk mengelola basis data MySQL yang ada di komputer. Untuk membukanya, buka browser lalu ketikkan alamat **<http://localhost/phpmyadmin>**, maka akan muncul halaman phpMyAdmin. Di situ nantinya seseorang bisa membuat (create) basis data baru, dan mengelolanya

b. Membuat Database

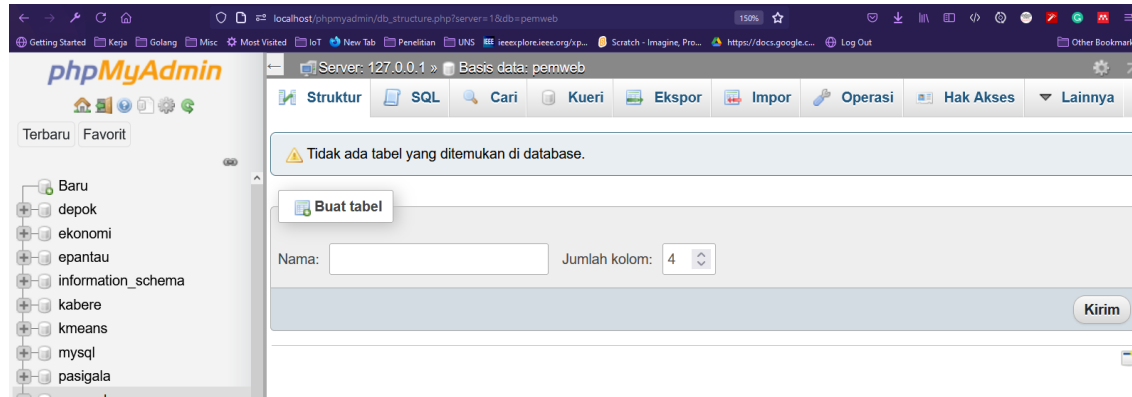
Untuk membuat database, kita dapat menuliskan nama database pada field “**nama basis data**” di bagian tab “**basis data**”. Pilih character set sesuai dengan yang diinginkan.

Pemilihan character set ini berkaitan dengan **collation**, yakni **sekumpulan rule atau aturan yang digunakan oleh database untuk untuk membandingkan karakter yang ada pada sebuah character set**.

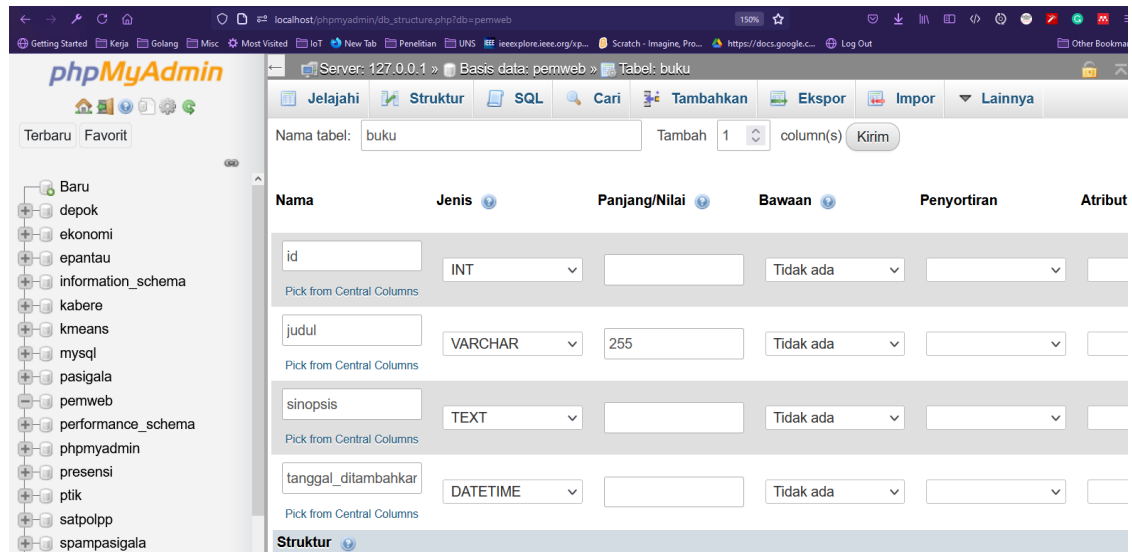


c. Membuat Tabel dan Menambah Field

Masuk ke dalam database yang diinginkan, lalu tulis nama tabel pada bagian “**buat tabel**” di tab Struktur. Pastikan jumlah kolom kita sesuai dengan field yang akan ditambahkan



Tulis nama field beserta jenis dan panjang karakter dari field tersebut. Pastikan bahwa panjang field harus dapat dibuat seefisien mungkin, karena hal ini akan berkaitan pula dengan performa database kita.



d. Membuat Relasi Tabel

Untuk membuat relasi tabel, kita butuh minimal 2 tabel. Pada gambar terlihat tiga buah tabel, yakni buku, rak dan buku-rak. Asumsinya, setiap rak memiliki buku sehingga perlu adanya penanda bahwa suatu buku tersimpan di dalam rak tertentu. Hal ini dapat ditangani dengan merujuk id_buku dan id_rak pada sebuah tabel penengah, buku-rak. Pasikan field tersebut bersifat INDEX.

Tabel	Tindakan	Baris	Jenis	Penyortiran	Ukuran	Beban
<input type="checkbox"/> buku	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/> buku-rak	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/> rak	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	16.0 KB	-
3 tabel	Jumlah	0	InnoDB	utf8mb4_general_ci	48.0 KB	0 B

Pada menu database, pilih tab desainer



Sebuah halaman baru akan muncul, pada menu sidebar sebelah kiri, pilih create relationship,

Struktur
SQL
Cari
Kueri
Ekspor
Impor
Operasi
Hak Akses

Belum ada nama *

Show/Hide tables list
View in fullscreen
Tambahkan tabel dari database lainya
New page
Buka halaman
Simpan halaman
Simpan halaman sebagai
Delete pages
Buat tabel
Create relationship
Pilih kolom untuk ditampilkan
Muat ulang
Bantuan
Tautan angular / Tautan langsung
Lekatkan kepada kisi
Small/Big All
Toggle small/big
Toggle relationship lines
Export schema
Pindahkan Menu
Sematkan teks

pemweb buku

id_buku : int(11)
judul : varchar(255)
sinopsis : text
tanggal_ditambahkan : datetime

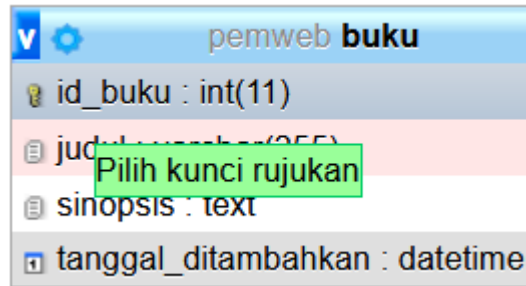
pemweb buku-rak

id_buku_rak : int(11)
id_buku : int(11)
id_rak : int(11)

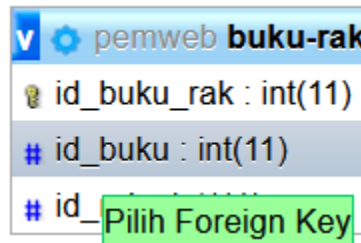
pemweb rak

id_rak : int(11)
kategori : varchar(255)
kapasitas : int(11)

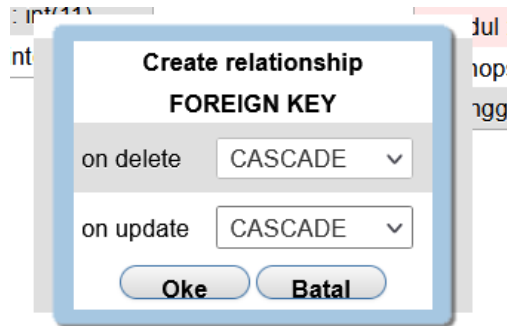
Arahkan kursor ke primary key dari tabel utama untuk memilih kunci rujukan



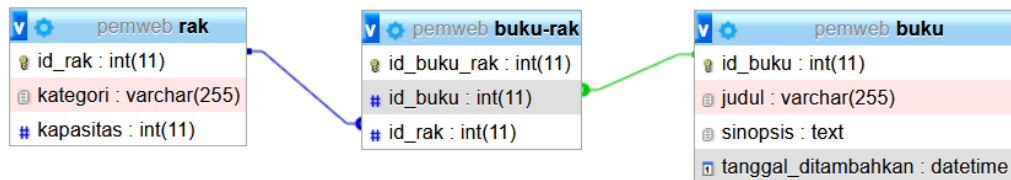
Arahkan kursor ke primary key dari tabel utama untuk memilih foreign key



Pilih jenis relasi, apakah CASCADE, SET NULL, DO NOTHING

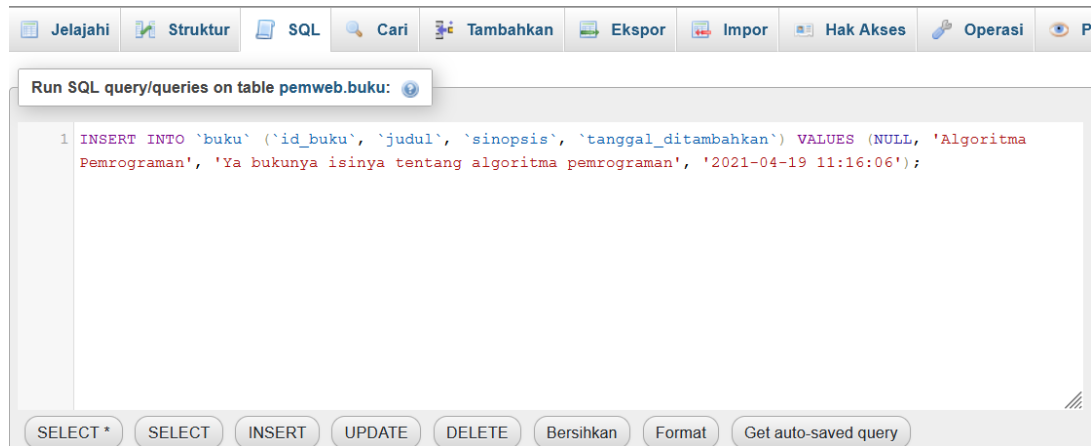


Hasil yang terbentuk apabila seluruh tabel telah diberi relasi. Hal inilah yang menjadi ciri khas dari RDBMS.

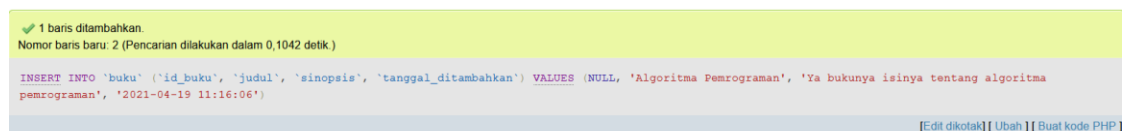


e. Memanggil Custom SQL Query

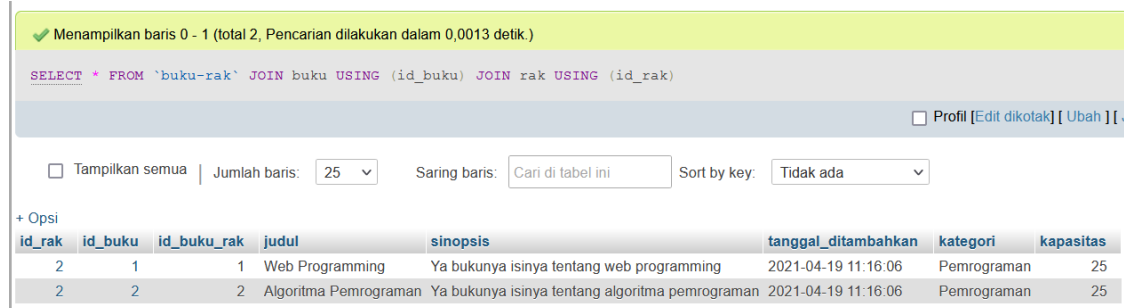
Untuk menulis custom SQL query, masuk pada tab kueri. Pada bagian textarea, Anda bisa menuliskan SQL query yang diinginkan lalu klik submit/kirim. Pada contoh akan menambahkan data buku ke dalam tabel.



Apabila berhasil dijalankan, maka akan muncul notif berikut

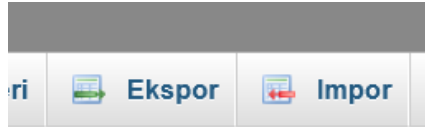


Semua query bisa dilalakan disini, tidak hanya insert, update, delete. Penggunaan join juga dapat langsung menampilkan hasilnya sesuai dengan penulisan query.



f. Ekspor dan Impor

Ekspor data dilakukan untuk mendistribusikan database / tabel ke dalam bentuk file yang dapat dibuka pada perangkat atau aplikasi lain. Sementara import data dilakukan untuk mengunggah database / tabel ke dalam sistem database kita sehingga data dapat dibuka di dalam phpmyadmin. Proses ekspor dan impor dapat dilakukan dengan mengakses tab **ekspor/impor**



Proses ekspor dapat dilakukan tiga cara, yakni ekspor database, ekspor tabel, atau ekspor hasil query.

Server: 127.0.0.1 » Basis data: pemweb

Struktur SQL Cari Kueri Ekspor Impor Operasi Hak Akses

Mengekspor tabel dari basis data "pemweb"

Template ekspor:

Template Baru: Nama template

Template yang tersedia: Template: -- Pilih template --

Metode ekspor:

☒ Cepat - menampilkan opsi minimum

☐ Kustom - menampilkan semua opsi

Format:

SQL

Server: 127.0.0.1 » Basis data: pemweb » Tabel: buku

Jejak Struktural SQL Cari Tambahkan Ekspor Impor Hak Akses

Mengekspor baris dari tabel "buku"

Template ekspor:

Template Baru: Nama template

Template yang tersedia: Template: -- Pilih template --

Metode ekspor:

☒ Cepat - menampilkan opsi minimum

☐ Kustom - menampilkan semua opsi

Format:

SQL

Baris:

☐ Dump beberapa baris

Jumlah baris:

Baris untuk memulai:

☒ Dump semua row

Proses impor dapat dilakukan dua jenis, yakni impor database atau impor tabel tertentu.

Server: 127.0.0.1 » Basis data: pemweb » Tabel: buku

Jejak Struktural SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi Pelacakan Trigger

Mengimpor ke dalam tabel "buku"

Berkas untuk impor:

Dapat berupa berkas terkompresi (gzip, bzip2, zip) atau tidak.
A compressed file's name must end in `.[format].[compression]`. Example: `.sql.zip`

Telusuri komputer Anda: No file selected. (Batas ukuran: 500MB)

Anda juga dapat drag dan drop sebuah file pada halaman manapun.

Set karakter berkas: utf-8

Impor sebagian:

☒ Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. (This might be a good way to import large files, however it can break transactions.)

Lewati beberapa kueri (untuk SQL) mulai dari yang pertama: 0

Opsis lain:

☒ Perbolehkan cek foreign key

Format:

SQL

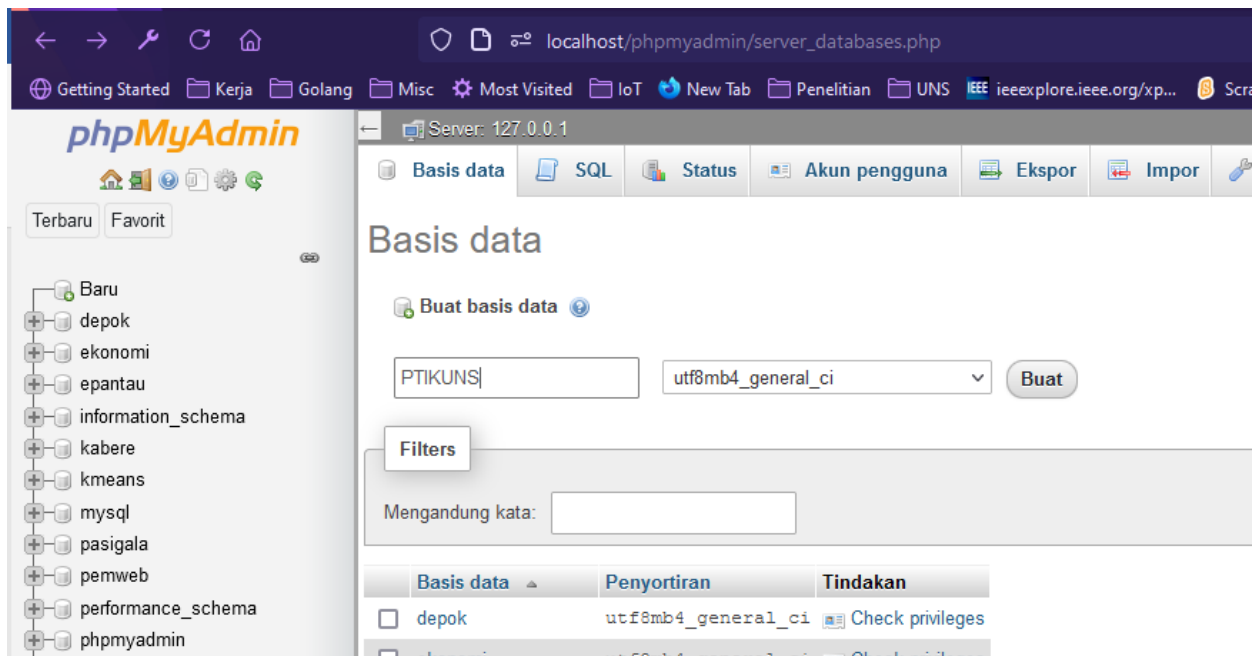
Opsis format-spesifik:

D. PRAKTIKUM : Koneksi Database

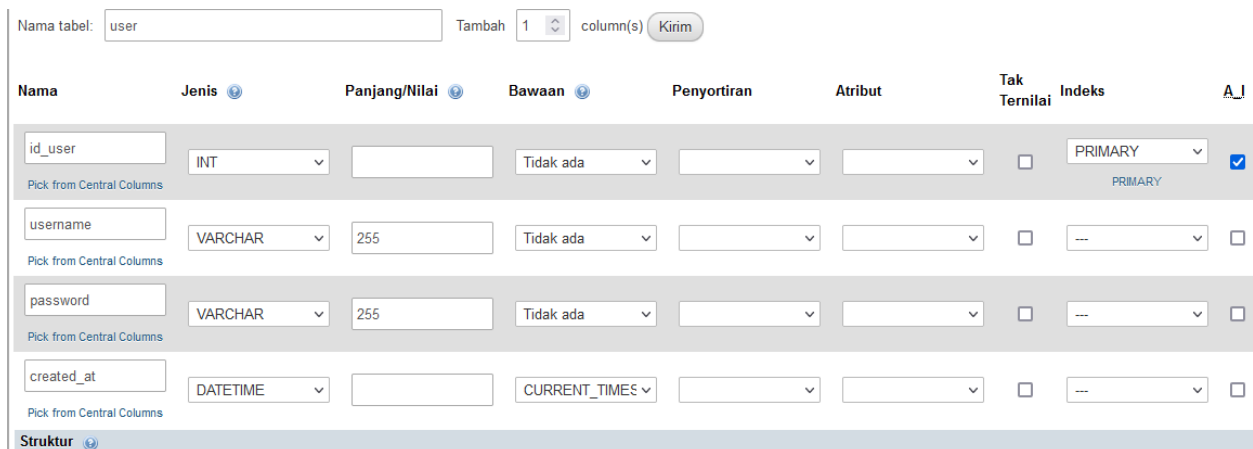
Kegiatan 1 – Create Database

1. Membuat Database Tabel User dan Mahasiswa

- Masuk ke dalam PHPMyAdmin, lalu buatlah table dengan nama “PTIKUNS”.



- Masuk ke dalam database tersebut lalu buatlah table dengan nama “user”. Isikan table “user” dengan field berikut.



- buatlah table dengan nama “mahasiswa”. Isikan table “mahasiswa” dengan field berikut

Nama tabel: mahasiswa Tambah 1 column(s) Kirim

Nama	Jenis	Panjang/Nilai	Bawaan	Penyortiran	Atribut	Tak Terilai	Indeks	A_I_K
id_mahasiswa <small>Pick from Central Columns</small>	INT		Tidak ada			<input type="checkbox"/>	PRIMARY	<input checked="" type="checkbox"/>
nama <small>Pick from Central Columns</small>	VARCHAR	255	Tidak ada			<input type="checkbox"/>	---	<input type="checkbox"/>
nim <small>Pick from Central Columns</small>	VARCHAR	10	Tidak ada			<input type="checkbox"/>	---	<input type="checkbox"/>
email <small>Pick from Central Columns</small>	VARCHAR	255	Tidak ada			<input type="checkbox"/>	---	<input type="checkbox"/>
telepon <small>Pick from Central Columns</small>	VARCHAR	15	Tidak ada			<input type="checkbox"/>	---	<input type="checkbox"/>
created_at <small>Pick from Central Columns</small>	DATETIME		CURRENT_TIMES			<input type="checkbox"/>	---	<input type="checkbox"/>

Struktur




2. Mengisi Tabel User dan Mahasiswa

- Sebelum mengisi tabel user, tambahkan sebuah field, yakni id_mahasiswa pada tabel user. Hal ini dapat digunakan untuk “mengakali” penggunaan tabel penengah antara user dan mahasiswa. Pilih tambahkan 1 kolom setelah id_user.


Cetak
 Usulkan struktur tabel
 Lacak tabel
 Move columns
 Normalisasi

Tambahkan 1 kolom setelah id_user Kirim

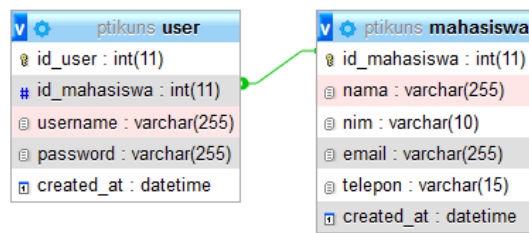
- Tambahkan field **id_mahasiswa**

Nama **Jenis**  **Panjang/Nilai**  **Bawaan** 



Pick from Central Columns

Struktur 

- Tambahkan relasi apabila diperlukan



- Isikan data ke dalam tabel user. Pastikan bagian password sudah diberi enkripsi (MD5). Tambahkan data sebanyak-banyaknya sesuai yang Anda inginkan.


 

Kolom	Jenis	F
id_user	int(11)	
id_mahasiswa	int(11)	
username	varchar(255)	
password	varchar(255)	
created_at	datetime	

AES_DECRYPT
 AES_ENCRYPT
 BIN
 CHAR
 COMPRESS
 CURRENT_USER
 DATABASE
 DAYNAME
 DES_DECRYPT
 DES_ENCRYPT
 ENCRYPT
 HEX
 INET6_NTOA
 INET_NTOA
 LOAD_FILE
 LOWER
 LTRIM
MD5

Tambahkan Ekspor Impor Hak Akses Operasi Pelacakan Trigger

Tak	Terilai	Nilai
	1	
		20509051
		12345678

created_at datetime current_timestamp() 

Kirim

- Isikan data ke dalam tabel mahasiswa. Tambahkan data sebanyak-banyaknya sesuai yang Anda inginkan.

Kolom	Jenis	Fungsi	Tak Termini	Nilai
id_mahasiswa	int(11)			
nama	varchar(255)			Murcayya Pradana
nim	varchar(10)			M0509051
email	varchar(255)			n.ptp@uns.ac.id
telepon	varchar(15)			085647430099
created_at	datetime			current_timestamp()

Kirim

- Hasil insert data

✓ Menampilkan baris 0 - 2 (total 3, Pencarian dilakukan dalam 0,0007 detik.)

`SELECT * FROM `user``

☐ Tampilkan semua |
 Jumlah baris: 25 |
 Saring baris: Cari di tabel ini |
 Sort by key: Tidak ada

+ Opsi

	id_user	id_mahasiswa	username	password	created_at
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	1	1	mhs1	25d55ad283aa400af464c76d713c07ad	2021-04-26 13:17:12
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	2	2	mhs2	25d55ad283aa400af464c76d713c07ad	2021-04-26 13:18:48
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	3	3	mhs3	25d55ad283aa400af464c76d713c07ad	2021-04-26 13:18:55

3. Membuat File Koneksi

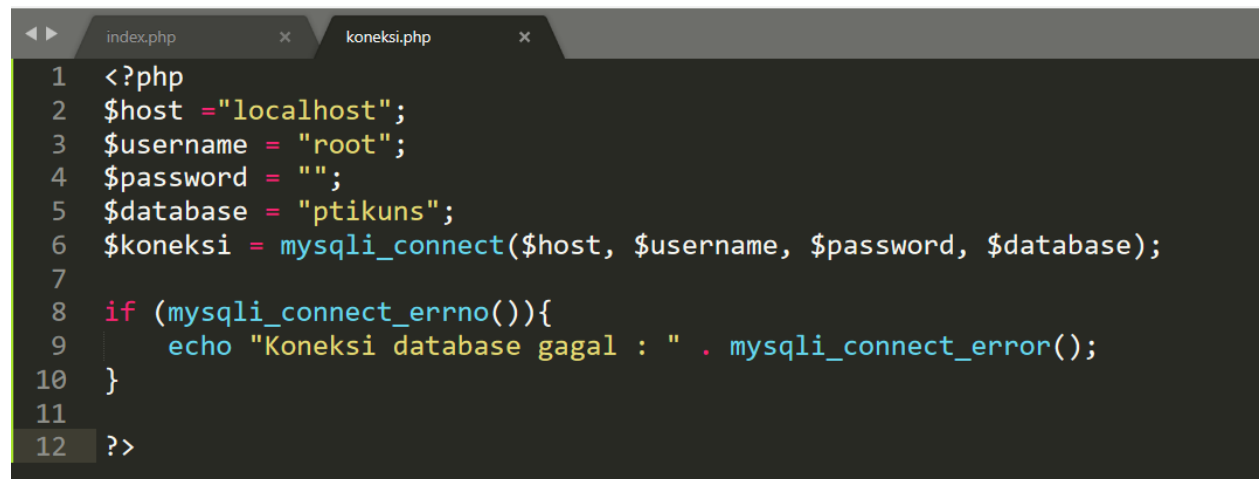
File koneksi berperan penting sebagai penghubung antara aplikasi dengan database. File ini berisi konfigurasi database, tergantung dari driver database apa yang dimiliki oleh komputer Anda. Ada 2 cara menghubungkannya: MySQL/MySQLi dan PDO (PHP Data Object)

Buka

<https://www.youtube.com/watch?v=bmA5OqgmD3Zs&list=PL3NJYF3IrMkVdRs0X4Gy2RVLbZokXpTIY&index=5>

Untuk mempelajari cara koneksi PHP ke MySQL lebih lanjut

Pada praktikum ini, kita coba menggunakan konfigurasi dengan ekstensi MySQLi.

A screenshot of a code editor with two tabs: 'index.php' and 'koneksi.php'. The 'koneksi.php' tab is active, showing the following PHP code:

```
1 <?php
2 $host = "localhost";
3 $username = "root";
4 $password = "";
5 $database = "ptikuns";
6 $koneksi = mysqli_connect($host, $username, $password, $database);
7
8 if (mysqli_connect_errno()){
9     echo "Koneksi database gagal : " . mysqli_connect_error();
10 }
11
12 ?>
```

4. Membuat Halaman Mahasiswa

Halaman mahasiswa kita anggap sebagai homepage, maka buat halaman mahasiswa pada file index.php.

- a. Isikan file tersebut dengan kode berikut. Kode disadur dari laman form template Bootstrap. <https://getbootstrap.com/docs/5.0/forms/overview/>

```

1  <!doctype html>
2  <html lang="en">
3    <head>
4      <!-- Required meta tags -->
5      <title>Halaman Mahasiswa</title>
6      <meta charset="utf-8">
7      <meta name="viewport" content="width=device-width, initial-scale=1">
8      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel=
        "stylesheet" integrity="sha384-EVSTQN3/
        azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTWfSpd3yD65Vohhpuc0mLASjC" crossorigin="anonymous">
9    </head>
10   <body>
11
12   .....
13
14   <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js
        " integrity="sha384-MrcW6ZMFY1zcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
        crossorigin="anonymous"></script>
15   </body>
16 </html>
17

```

- b. Tambahkan navigasi yang bisa diambil dari template bootstrap, lalu modifikasi seperti berikut

```

12   <div class="container-fluid">
13     <nav class="navbar navbar-expand-lg navbar-light bg-light">
14       <div class="container-fluid">
15         <a class="navbar-brand" href="#">Pertemuan06</a>
16         <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
            data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
            aria-expanded="false" aria-label="Toggle navigation">
17           <span class="navbar-toggler-icon"></span>
18         </button>
19         <div class="collapse navbar-collapse" id="navbarSupportedContent">
20           <ul class="navbar-nav me-auto mb-2 mb-lg-0">
21             <li class="nav-item">
22               <a class="nav-link" aria-current="page" href="index.php">Mahasiswa</a>
23             </li>
24             <li class="nav-item">
25               <a class="nav-link" aria-current="page" href="akun.php">Akun</a>
26             </li>
27           </ul>
28         </div>
29       </div>
30     </nav>

```

- c. Isi body dengan tabel

```

30 <div class="row">
31   <div class="col-lg-6 offset-lg-3 mt-3">
32     <div class="card">
33       <table class="table table-striped table-bordered">
34         <tr>
35           <th>No</th>
36           <th>Nama</th>
37           <th>NIM</th>
38           <th>email</th>
39         </tr>
40       </table>
41     </div>
42   </div>
43 </div>
44 </div>

```

- d. Isi tabel dengan kode PHP yang mengambil data secara langsung dari dalam database.

```

40 <?php
41 include 'koneksi.php';
42 $no=1;
43 $query = mysqli_query($koneksi,"select * from mahasiswa order by id_mahasiswa desc");
44 while ($item = mysqli_fetch_array($query)) { ?>
45   <tr>
46     <td><?= $no++; ?></td>
47     <td><?= $item['nama']; ?></td>
48     <td><?= $item['nim']; ?></td>
49     <td><?= $item['email']; ?></td>
50   </tr>
51 <?php } ?>

```

- e. Hasil.

Pertemuan06 Mahasiswa Akun			
No	Nama	NIM	email
1	Noctis Lucis	M0509053	n.lcs@uns.ac.id
2	Solid Snake	M0509052	s.ss@uns.ac.id
3	Nurcahya Pradana	M0509051	n.ptp@uns.ac.id

5. Membuat Halaman Akun

Halaman kedua adalah halaman akun, yang akan digunakan untuk menampilkan data kredensial dari masing-masing mahasiswa. Sama seperti halaman mahasiswa, kode

disadur dari laman starter template Bootstrap

<https://getbootstrap.com/docs/5.0/getting-started/introduction/>.

a. Isikan file tersebut dengan kode berikut. Kode disadur dari laman form template

Bootstrap. <https://getbootstrap.com/docs/5.0/forms/overview/>

```
1 <!doctype html>
2 <html lang="en">
3   <head>
4     <!-- Required meta tags -->
5     <title>Halaman Akun</title>
6     <meta charset="utf-8">
7     <meta name="viewport" content="width=device-width, initial-scale=1">
8     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel=
      "stylesheet" integrity="sha384-EVSTQN3/
      azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTWfSpd3yD65VohhpuuCOmLASjC" crossorigin="anonymous">
9   </head>
10  <body>
11
12  ...|
13
14  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js
    " integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
    crossorigin="anonymous"></script>
15 </body>
16 </html>
17
```

b. Tambahkan navigasi yang bisa diambil dari template bootstrap, lalu modifikasi seperti berikut

```
12 <div class="container-fluid">
13   <nav class="navbar navbar-expand-lg navbar-light bg-light">
14     <div class="container-fluid">
15       <a class="navbar-brand" href="#">Pertemuan06</a>
16       <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
        data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
        aria-expanded="false" aria-label="Toggle navigation">
17         <span class="navbar-toggler-icon"></span>
18       </button>
19       <div class="collapse navbar-collapse" id="navbarSupportedContent">
20         <ul class="navbar-nav me-auto mb-2 mb-lg-0">
21           <li class="nav-item">
22             <a class="nav-link" aria-current="page" href="index.php">Mahasiswa</a>
23           </li>
24           <li class="nav-item">
25             <a class="nav-link" aria-current="page" href="akun.php">Akun</a>
26           </li>
27         </ul>
28       </div>
29     </div>
30   </nav>
31 </div>
```

c. Isi body dengan tabel

```

30     <div class="row">
31         <div class="col-lg-6 offset-lg-3 mt-3">
32             <div class="card">
33                 <table class="table table-striped table-bordered">
34                     <tr>
35                         <th>No</th>
36                         <th>Username</th>
37                         <th>Password</th>
38                         <th>Waktu Pembuatan</th>
39                     </tr>
40                 </table>
41             </div>
42         </div>
43     </div>

```

- d. Isi tabel dengan kode PHP yang mengambil data secara langsung dari dalam database.

```

40 <?php
41 include 'koneksi.php';
42 $no=1;
43 $query = mysqli_query($koneksi,"select * from mahasiswa order by id_mahasiswa desc");
44 while ($item = mysqli_fetch_array($query)) { ?>
45     <tr>
46         <td><?= $no++; ?></td>
47         <td><?= $item['nama']; ?></td>
48         <td><?= $item['nim']; ?></td>
49         <td><?= $item['email']; ?></td>
50     </tr>
51 <?php } ?>

```

- e. Hasil.

Pertemuan06 Mahasiswa Akun			
No	Username	Password	Waktu Pembuatan
1	mhs3	25d55ad283aa400af464c76d713c07ad	2021-04-26 13:18:55
2	mhs2	25d55ad283aa400af464c76d713c07ad	2021-04-26 13:18:48
3	mhs1	827ccb0eea8a706c4c34a16891f84e7b	2021-04-26 13:17:12

Kegiatan 2 – This is the real deal!

- Ubah koneksi aplikasi di atas dengan koneksi menggunakan **PDO**.
- Buat file-file di atas menjadi modular yang terdiri dari file **index.php**, **akun.php**, **config.php**, **header.php**, **footer.php**
- Buat sebuah halaman lagi yang menampilkan tabel gabungan antara mahasiswa dengan akun. Beri nama file tersebut **user.php**. Hasil yang diharapkan adalah seperti pada gambar berikut!
- Simpan file dan database ke dalam sebuah RAR/ZIP dengan nama **p06.rar/zip**

Pertemuan06 Mahasiswa Akun User						
ID Mahasiswa	Nama	NIM	Username	Password		Waktu Pembuatan
3	Noctis Lucis	M0509053	mhs3	25d55ad283aa400af464c76d713c07ad		2021-04-26 13:22:19
2	Solid Snake	M0509052	mhs2	25d55ad283aa400af464c76d713c07ad		2021-04-26 13:20:44
1	Nurcahya Pradana	M0509051	mhs1	827ccb0eea8a706c4c34a16891f84e7b		2021-04-26 13:20:07