**DOKUMENTASI AWAL PROYEK PENGEMBANGAN BASIS DATA NILAI SISWA-SISWI SMPN 4 SARADAN**

****

****

**DISUSUN OLEH :**

**KELOMPOK 5**

1. **AMANDA FAHMA PUTRI / 243307065**
2. **LATIFA JAUHARATUL FUADAH / 243307077**
3. **REHANIA PUTRI NUR AINI / 243307087**

**JURUSAN TEKNIK**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MADIUN**

**TAHUN 2024/2025**

**Dokumentasi Awal Proyek Pengembangan Basis Data Nilai**

**Siswa-Siswi SMPN 4 SARADAN**

Nama Klien : Bapak Muh Sholikudin

Alamat : Jl.Waringin Tunggal, RT/RW 21/03, Desa Kebonsari, Kecamatan Kebonsari, Kabupaten Madiun, Jawa Timur, Indonesia

Nama Kontak Utama : +62 895-6304-92854

Jabatan : Guru PAI

Email : sholikudin1@gmail.com

Telepon : +62 895-6304-92854

Nama Kontak Tambahan : Rehania Putri Nur Aini

Jabatan : Mahasiswa

Email : rehania2018putri@gmail.com

Telepon : +62 898-6368-457

**1.PENDAHULUAN**

**1.1. LATAR BELAKANG PROYEK**

Proyek ini bertujuan mengembangkan sistem basis data untuk suatu masalah yang ada di SMPN 4 SARADAN. Masalah tersebut adalah terdapatnya kesulitan dalam pencatatan maupun pengarsipan pada pengolahan data nilai siswa yang masih dalam bentuk kertas. Dari masalah tersebut tentunya akan sangat repot ketika terjadi perubahan dari data nilai siswa. Selain itu kinerja guru juga masih banyak yang kurang mengerti dalam pengembangan teknologi. Maka dari itu, dalam penelitian ini dikembangkan sebuah sistem basis data yang diharapkan mampu menjadi lebih efisien dan akurat. Sehingga mempermudah dalam mengelola data nilai siswa.

**1.2. TUJUAN PROYEK**

* Penggunaan teknologi informasi dapat membantu mengurangi penggunaan kertas sebagai bahan untuk mencetak dokumen.
* Penggunaan system database terkomputerisasi akan membuat pengelolaan data lebih efisien dan mengurangi ruang yang dibutuhkan untuk menyimpan data dan file pelatihan.
* Data lengkap mudah diakses dan mudah dikelola.
* Menghemat biaya pemeliharaan adalah investasi jangka panjang.

**1.3. RUANG LINGKUP**

Yang termasuk :

1. Identifikasi jenis data yang perlu dikelola, seperti data siswa, guru, mata pelajaran, dan nilai.

2. Penyiapan lingkungan kerja yang diperlukan untuk pengembangan database.

Yang tidak termasuk :

1. Pengelolaan Operasi harian Sekolah
2. Pengelolaan dana sekolah
3. Sarana dan Prasarana sekolah

**2. ANALISIS KEBUTUHAN**

**2.1. KEBUTUHAN PENGGUNA**

* Pengguna Utama : Guru Pengajar
* Kebutuhan: Kemampuan untuk menambah, menghapus, dan memperbarui data nilai siswa, melihat laporan nilai akhir siswa. Kemampuan untuk mengembangkan, memperbarui data nilai siswa.

**2.2. KEBUTUHAN SISTEM**

* Teknis : DBMS yang digunakan (MySQL Workbench), kapasitas penyimpanan, kecepatan akses data, keamanan data.
* Fungsional : Pengelolaan data kehadiran siswa, pengelolaan nilai siswa dan pencarian nilai siswa

User Story 1 :

1. Sebagai siswa, saya ingin melihat data nilai
2. Kriteria Penerimaan : Validasi input harus memeriksa kelengkapan data dan format yang benar

User Story 2 :

1. Sebagai guru, saya menginputkan nilai siswa per mata pelajaran sehingga dapat menampilkan nilai
2. Kriteria Penerimaan : menampilkan id nilai, nama siswa, nilai siswa, kelas siswa, mata pelajaran

User Story 3 :

1. Sebagai guru, saya menginputkan data kehadiran siswa sehingga dapat manampilkan informasi data siswa
2. Kriteria penerimaan : menampilkan id kehadiran, nama siswa, NIS, nama kelas, jenis kelamin

**3. DESAIN BASIS DATA**

**3.1. MODEL KONSEPTUAL**

* Entitas: Nilai, mata pelajaran, siswa, guru pengajar,kelas,kehadiran siswa
* Hubungan:

Nilai: siswa memiliki nilai mata pelajaran.

Guru pengajar: guru mengajar mata Pelajaran di setiap kelas

Kehadiran siswa : siswa memiliki absensi kehadiran

**3.2. MODEL LOGIS**

Tabel :

* Siswa(ID Siswa,NIS, Jenis\_Kelamin, Nama)
* Mata Pelajaran(ID Mata Pelajaran, ID\_Guru, Mata\_Pelajaran)
* Nilai(ID Nilai, ID Siswa, ID\_Guru, mata\_Pelajaran,nilai\_mata Pelajaran)
* Guru Pengajaran(ID Guru, NIP, Nama Guru, Jenis\_Kelamin, Alamat)
* Kelas(ID Kelas, ID Guru, ID Kelas, Nama Kelas)
* Kehadiran Siswa(ID Kehadiran, ID Siswa, Tanggal, Status Kehadiran (Hadir, Sakit, Izin, Alpa))

**3.3 MODEL FISIK**

Indeks: Indeks pada Kolom id\_siswa dan id\_pelajaran

Optimasi: Karena pencarian berdasarkan siswa dan mata pelajaran sering dilakukan, indeks pada kedua kolom ini dapat mempercepat proses pencarian nilai untuk siswa tertentu atau untuk mata pelajaran tertentu.

Indeks: Indeks Gabungan (id\_siswa)

Optimasi: Dapat digunakan untuk mempercepat query yang sering kali melibatkan filter berdasarkan siswa dan semester tertentu.

**3.4 FITUR DAN QUERY :**

1. Sebagai siswa, saya ingin melihat data nilai

**Menampilkan nilai siswa**

Query :

SELECT ID Nilai, ID Siswa, ID\_Guru, mata\_Pelajaran,nilai\_mata Pelajaran

FROM tbl\_nilai

WHERE id\_siswa = '{id\_siswa}'

ORDER BY tanggal DESC;

1. Sebagai guru, saya menginputkan nilai siswa per mata pelajaran sehingga dapat menampilkan nilai

**Input nilai siswa**

Query :

INSERT INTO tbl\_nilai (ID Nilai, ID Siswa, ID\_Guru, mata\_Pelajaran,nilai\_mata Pelajaran)

VALUES ({ID Nilai}, '{ID Siswa}', '{ID Guru}', '{mata\_pelajaran}', {nilai\_mata\_pelajaran});

1. Sebagai guru, saya menginputkan data kehadiran siswa sehingga dapat manampilkan informasi data siswa

**Input kehadiran**

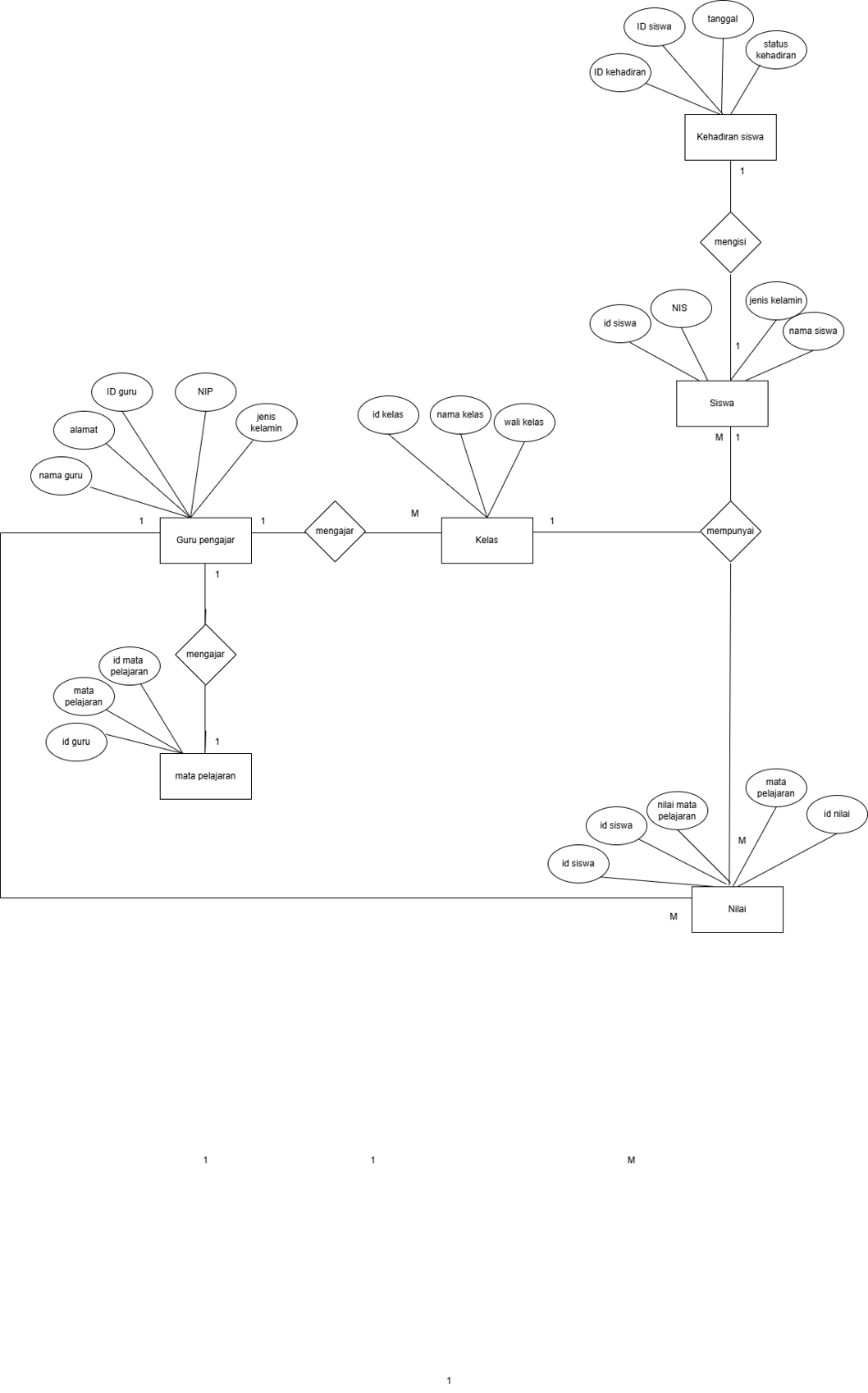
Query :

INSERT INTO tbl\_kehadiran (ID Kehadiran, ID Siswa, Tanggal, Status Kehadiran)

VALUES ('{ID Kehadiran}', '{ID siswa}', '{tanggal}', '{status\_kehadiran}');

**4.PERANCANGAN DATABASE**

**4.1. KESESUAIAN ERD DENGAN KEBUTUHAN SISTEM**

****

User Story 1 :

1. Sebagai siswa, saya ingin melihat data nilai
2. Kriteria Penerimaan : Validasi input harus memeriksa kelengkapan data dan format yang benar

User Story 2 :

1. Sebagai guru, saya menginputkan nilai siswa per mata pelajaran sehingga dapat menampilkan nilai
2. Kriteria Penerimaan : menampilkan id nilai, nama siswa, nilai siswa, kelas siswa, mata pelajaran

User Story 3 :

1. Sebagai guru, saya menginputkan data kehadiran siswa sehingga dapat manampilkan informasi data siswa
2. Kriteria penerimaan : menampilkan id kehadiran, nama siswa, NIS, nama kelas, jenis kelamin

**4.2. IMPLEMENTASI NORMALISASI DATABASE**

**A. TABEL SEBELUM NORMALISASI**

1. **TABEL GURU PENGAJAR**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID guru | NIP | Nama guru | Jenis kelamin | Alamat |
| 08900 | 2430090 | P. nanang | L | Geger |
| 08901 | 2430091 | P. tri | L | Madiun |
| 08902 | 2430092 | B. nisa | P | Caruban |
| 08903 | 2430093 | B. nia | P | Kebonsari |

1. **TABEL SISWA**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID siswa | NIS | Nama | Nama kelas | Jenis kelamin |
| 241A | 00620 | Amanda | A | P |
| 241B | 00621 | Latifa | B | P |
| 241C | 00622 | Rehania | C | P |
| 241D | 00623 | Marselino | D | L |

1. **TABEL MATA PELAJARAN**

|  |  |  |
| --- | --- | --- |
| ID guru | ID mata Pelajaran | Mata Pelajaran |
| 08900 | MA301 | Matematika |
| 08901 | BI302 | Bahasa Indonesia |
| 08902 | GR303 | Bahasa inggris |
| 08903 | PK304 | PPKN |

1. **TABEL NILAI**

|  |  |  |  |
| --- | --- | --- | --- |
| ID nilai | ID siswa | Mata pelajaran | Nilai mata  pelajaran |
| 100M | 241A | Matematika | 95 |
| 101B | 241B | Bahasa Indonesia | 90 |
| 102G | 241C | Bahasa inggris | 85 |
| 103P | 241D | PPKN | 80 |

1. **TABEL KELAS**

|  |  |  |
| --- | --- | --- |
| ID kelas | Nama kelas | Wali kelas |
| 001A | A | P. nanang |
| 001B | B | P. tri |
| 001C | C | B. nisa |
| 001D | D | B. nia |

1. **TABEL KEHADIRAN SISWA**

|  |  |  |  |
| --- | --- | --- | --- |
| ID kehadiran | ID siswa | Tanggal | Status kehadiran |
| 020 | 241A | 24 Mei 2024 | Hadir |
| 021 | 241B | 24 Mei 2024 | Hadir |
| 022 | 241C | 24 Mei 2024 | Hadir |
| 023 | 241D | 24 Mei 2024 | Hadir |

**B. TABEL SETELAH NORMALISASI**

1. **TABEL GURU PENGAJAR**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID guru | NIP | Nama guru | Jenis kelamin | Alamat |
| 08900 | 2430090 | P. nanang | L | Geger |
| 08901 | 2430091 | P. tri | L | Madiun |
| 08902 | 2430092 | B. nisa | P | Caruban |
| 08903 | 2430093 | B. nia | P | Kebonsari |

1. **TABEL SISWA**

|  |  |  |  |
| --- | --- | --- | --- |
| ID siswa | NIS | Jenis kelamin | Nama |
| 241A | 00620 | P | Amanda |
| 241B | 00621 | P | Latifa |
| 241C | 00622 | P | Rehania |
| 241D | 00623 | L | Marselino |

1. **TABEL MATA PELAJARAN**

|  |  |  |
| --- | --- | --- |
| ID mata Pelajaran | ID guru | Mata Pelajaran |
| MA381 | 08900 | Matematika |
| BI312 | 08901 | Bahasa Indonesia |
| GR323 | 08902 | Bahasa inggris |
| PK334 | 08903 | PPKN |

1. **TABEL NILAI**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID nilai | ID siswa | ID guru | Mata pelajaran | Nilai mata  pelajaran |
| 100M | 241A | 08900 | Matematika | 95 |
| 101B | 242B | 08901 | Bahasa Indonesia | 90 |
| 102G | 243C | 08902 | Bahasa Inggris | 85 |
| 103P | 244D | 08903 | PPKN | 80 |

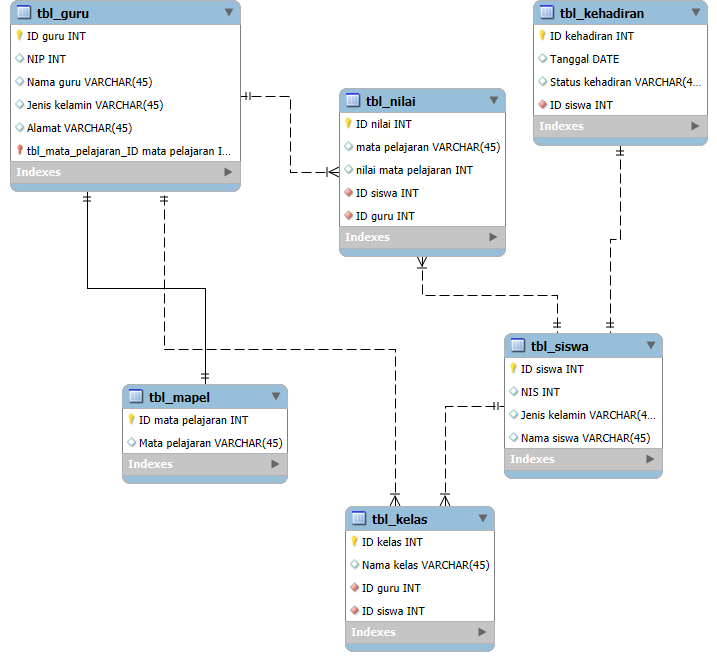
1. **TABEL KELAS**

|  |  |  |  |
| --- | --- | --- | --- |
| ID kelas | ID guru | ID siswa | Nama kelas |
| 001A | 08900 | 241A | A |
| 002B | 08901 | 242B | B |
| 003C | 08902 | 243C | C |
| 004D | 08903 | 244D | D |

1. **TABEL KEHADIRAN SISWA**

|  |  |  |  |
| --- | --- | --- | --- |
| ID kehadiran | ID siswa | Tanggal | Status kehadiran |
| 020 | 241A | 24 Mei 2024 | Hadir |
| 021 | 242B | 24 Mei 2024 | Hadir |
| 022 | 243C | 24 Mei 2024 | Hadir |
| 023 | 244D | 24 Mei 2024 | Hadir |

**4.3. OPTIMASI STRUKTUR TABEL DAN RELASI**

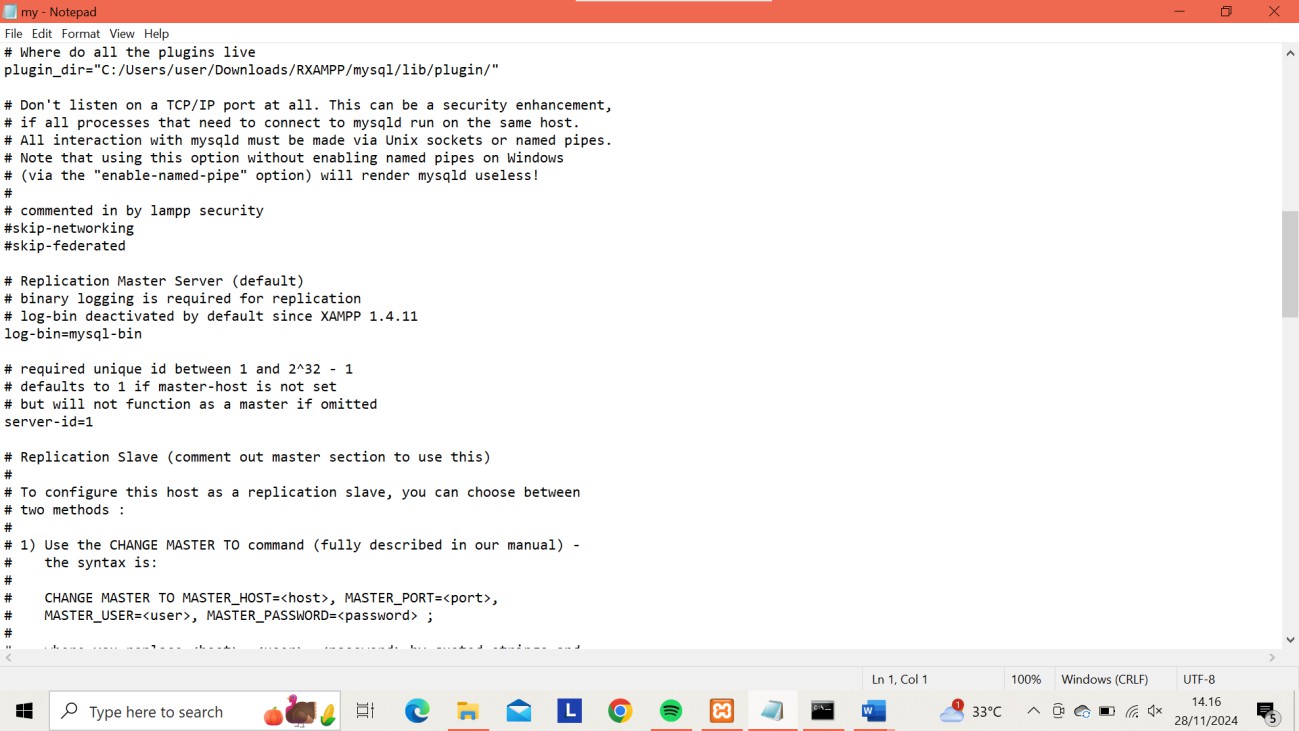
****

**5.IMPLEMENTASI SISTEM**

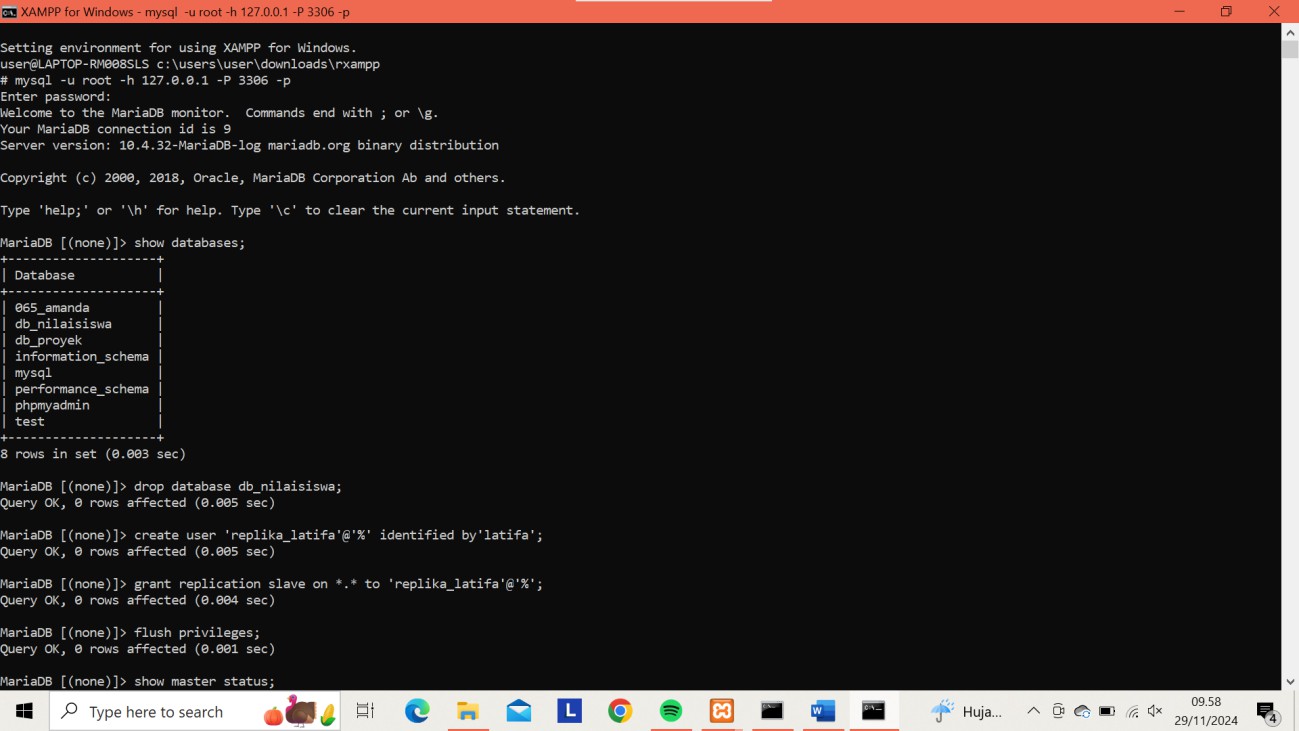
**5.1. IMPLEMENTASI SQL (DDL, DML, DCL)**

**- SEBAGAI ANGGOTA 2(MASTER)**

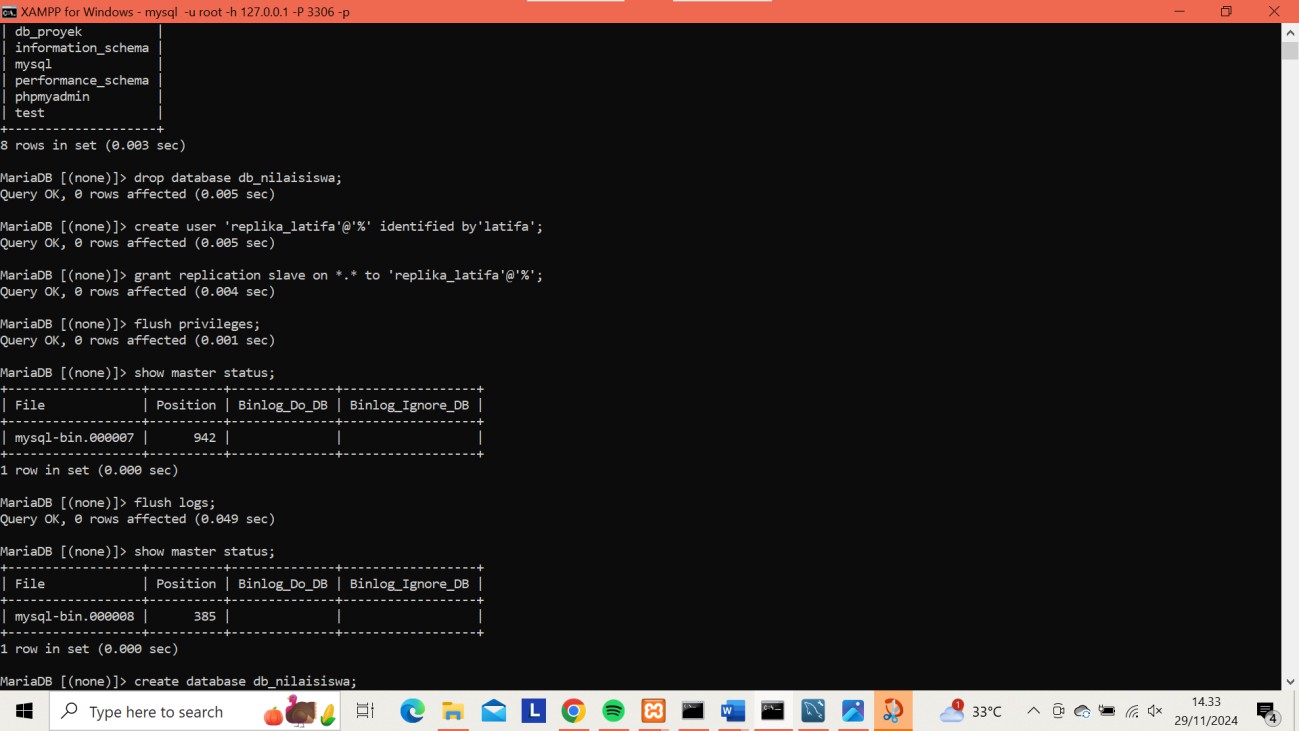
**1. Sebagai master hapus # pada log-bin=mysql-bin**

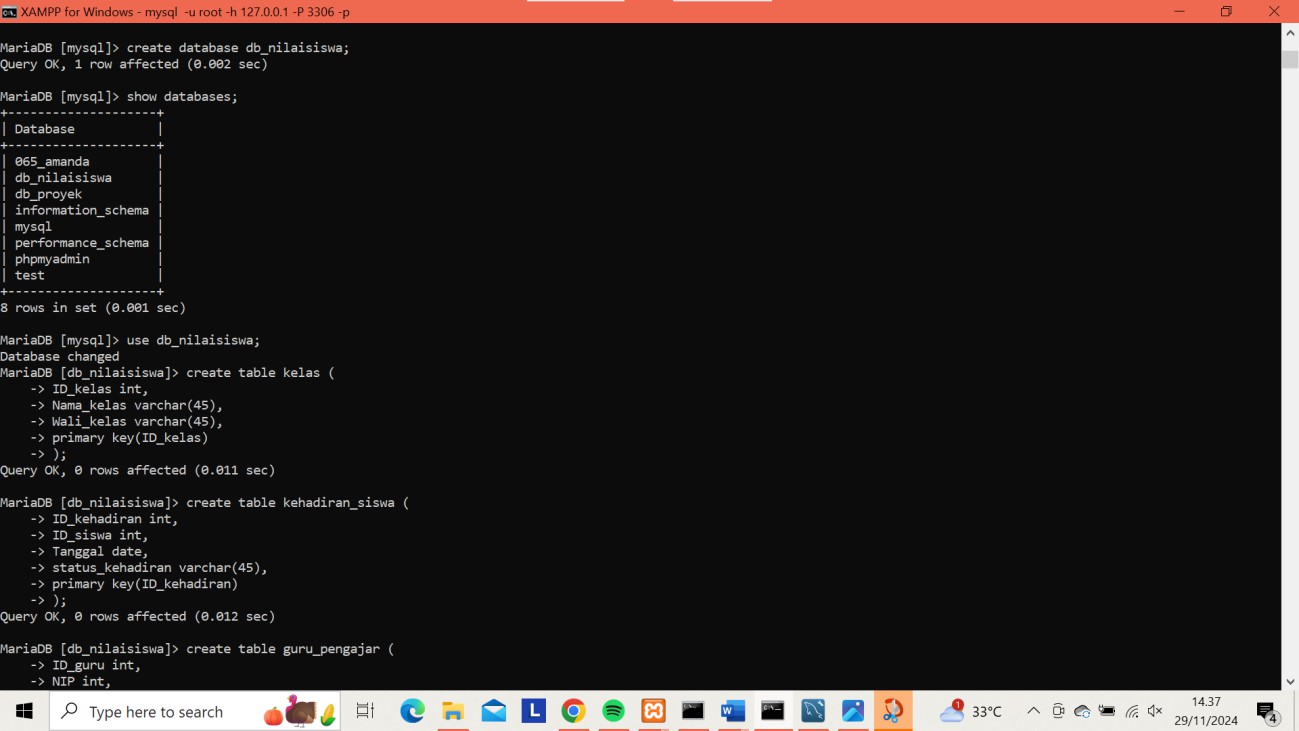


**2. Buat user, grant, dan show master status**

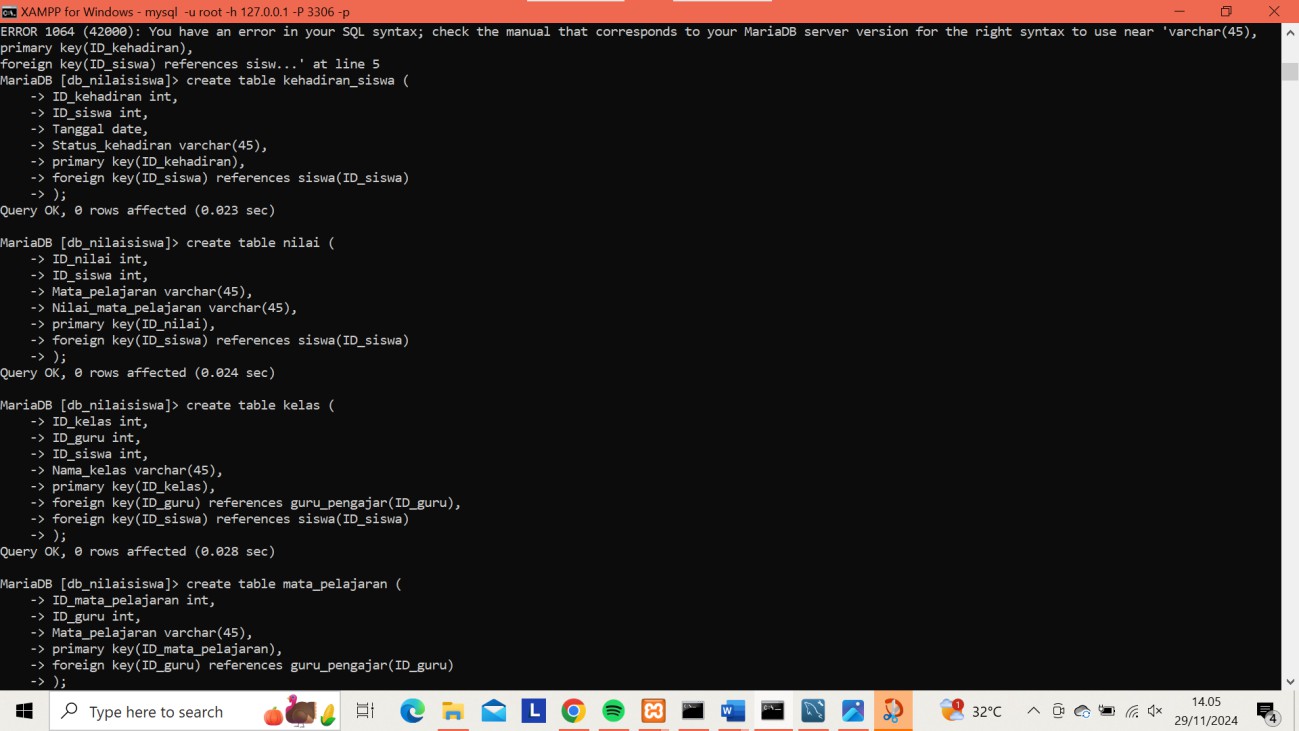
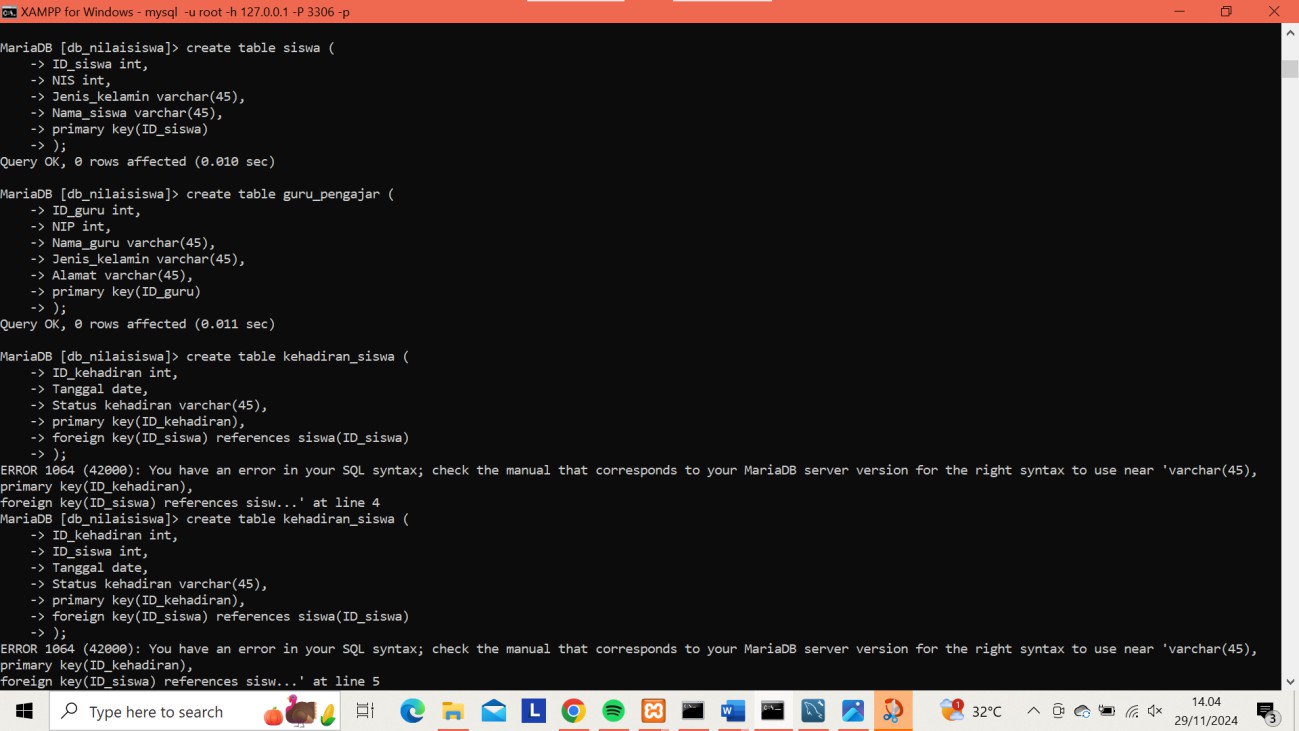


**3.Buat database db\_nilai\_siswa**

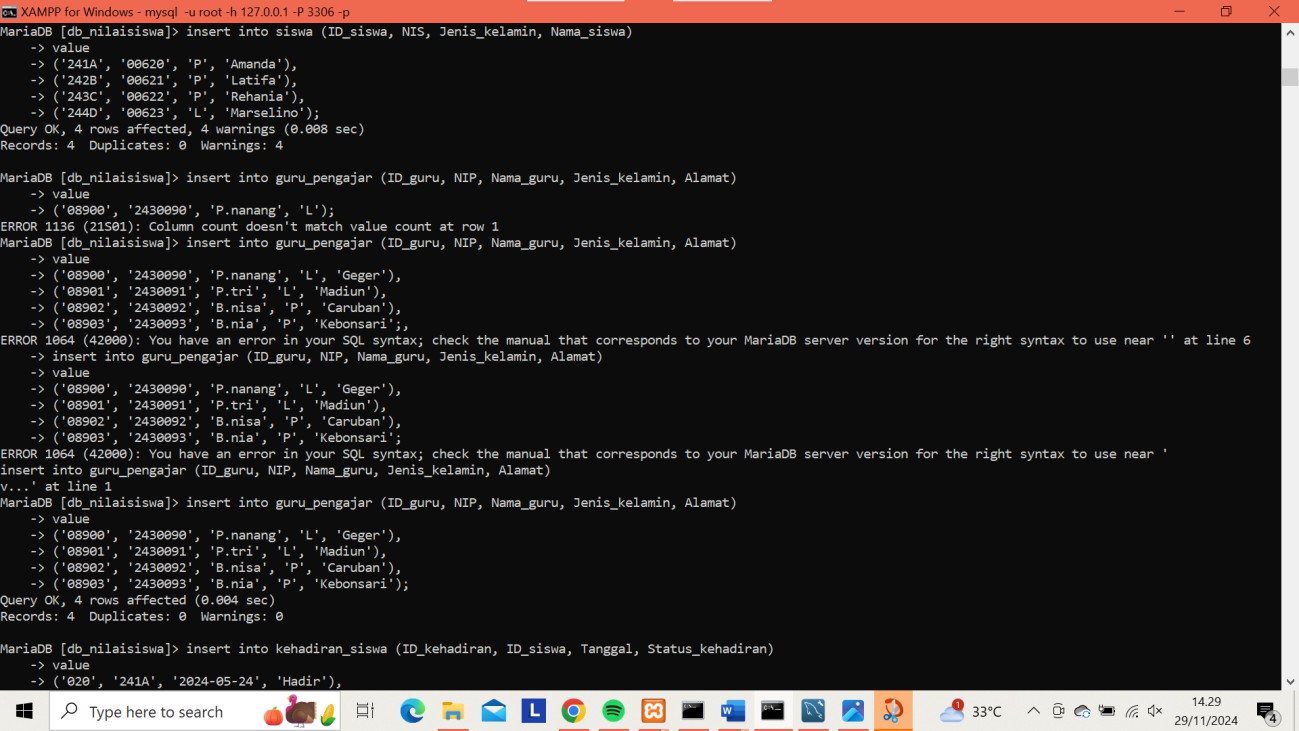


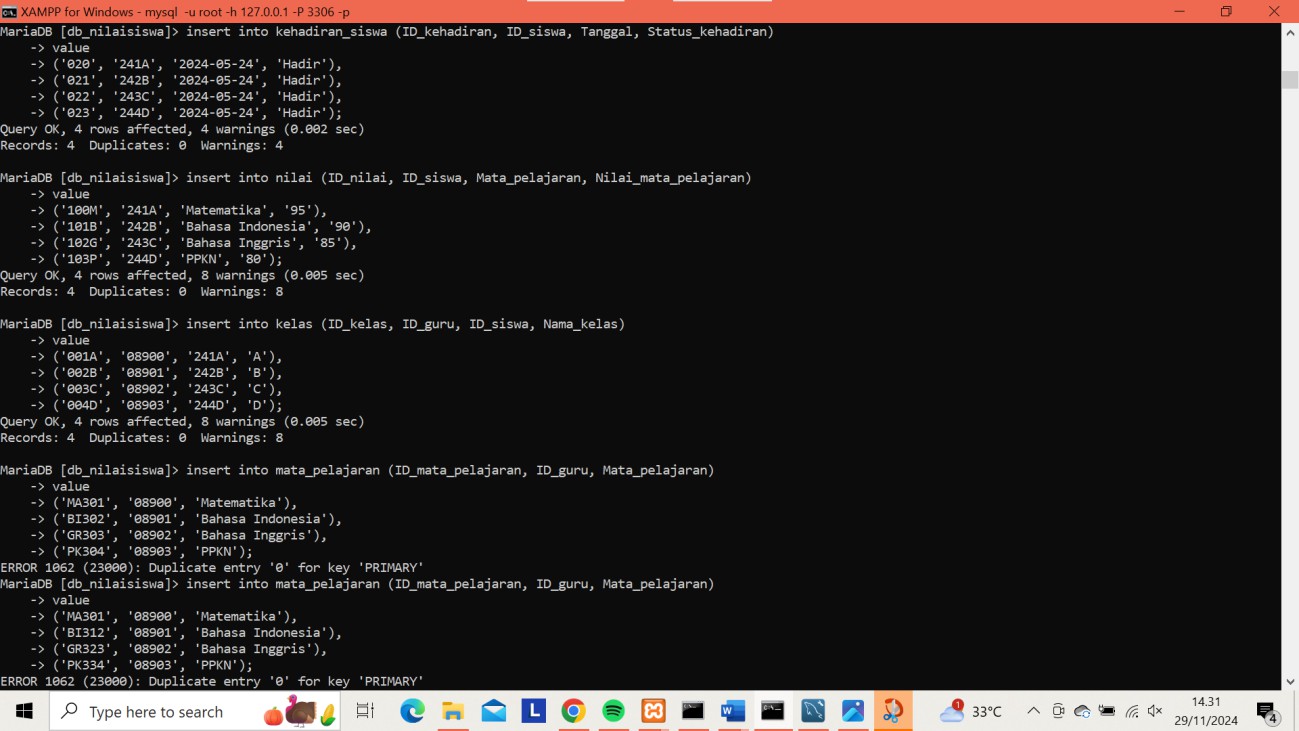


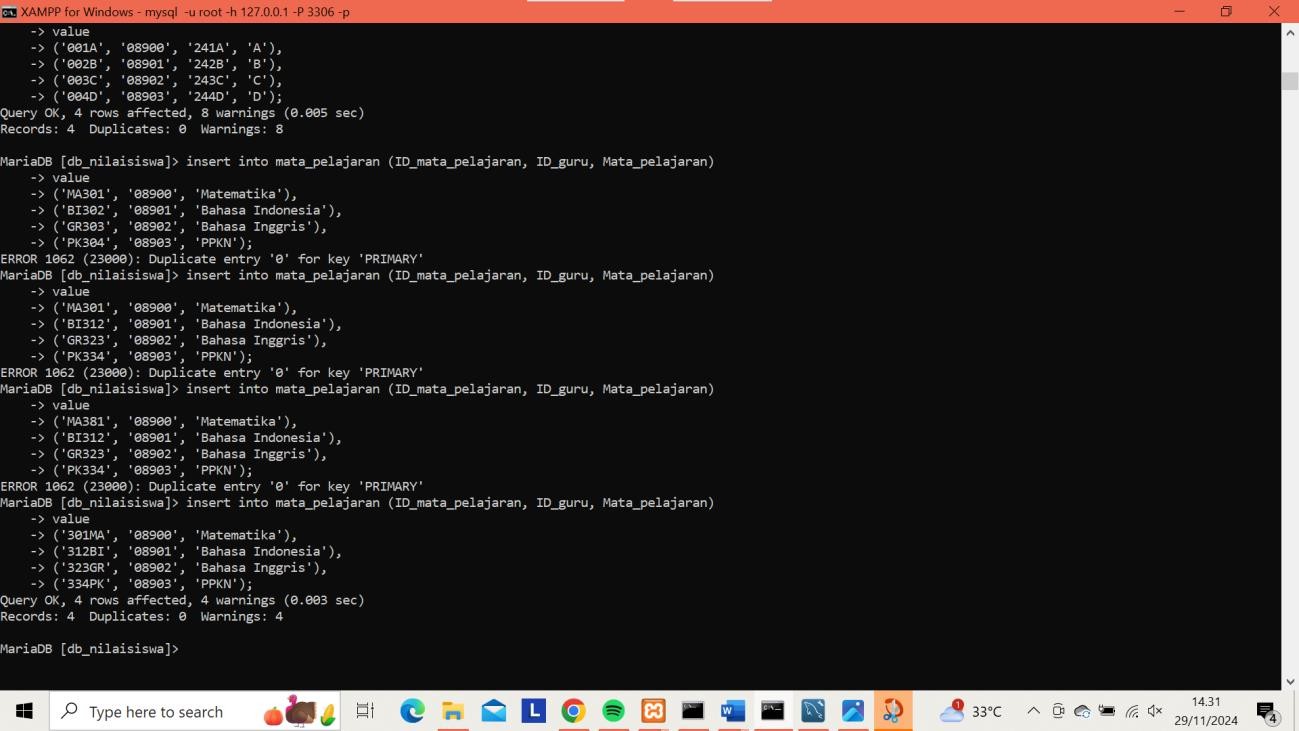
1. Create tabel
   1. Tabel siswa
   2. Tabel guru pengajar
   3. Tabel kehadiran siswa
   4. Tabel nilai
   5. Tabel Kelas
   6. Tabel Mata Pelajaran



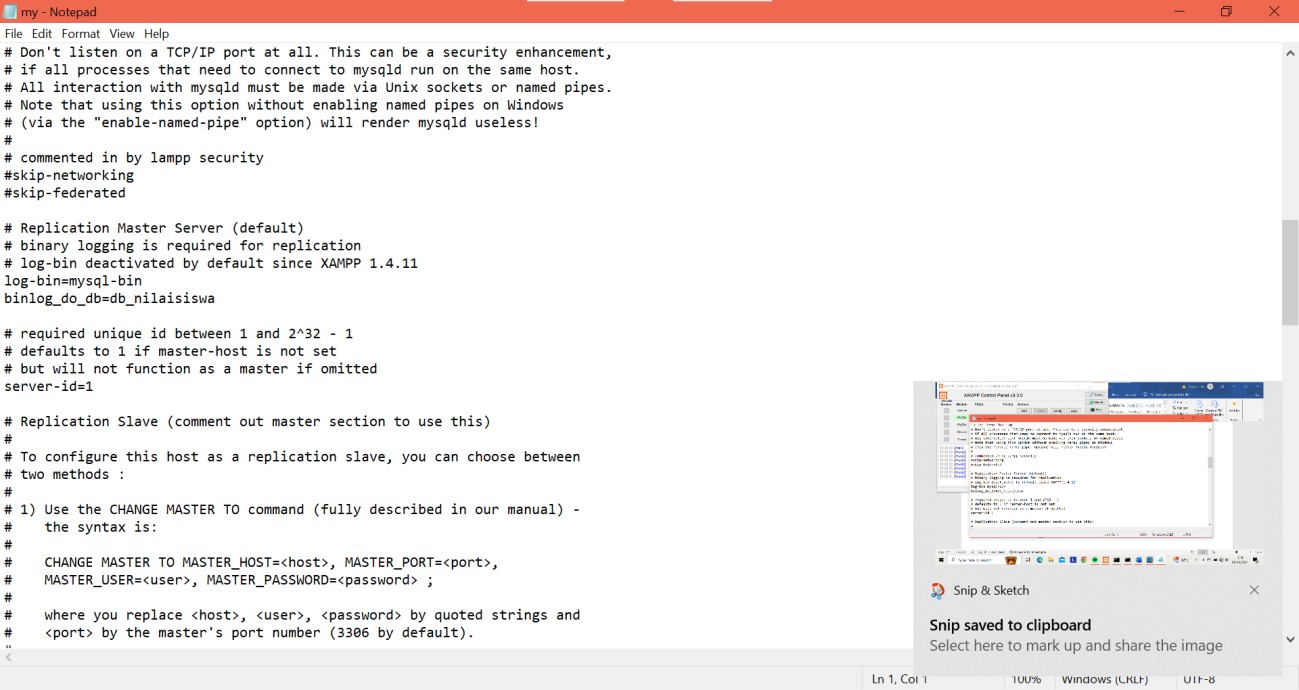
1. Isi tabel
   1. Tabel siswa
   2. Tabel guru pengajar
   3. Tabel kehadiran siswa
   4. Tabel nilai
   5. Tabel kelas
   6. Tabel mata Pelajaran



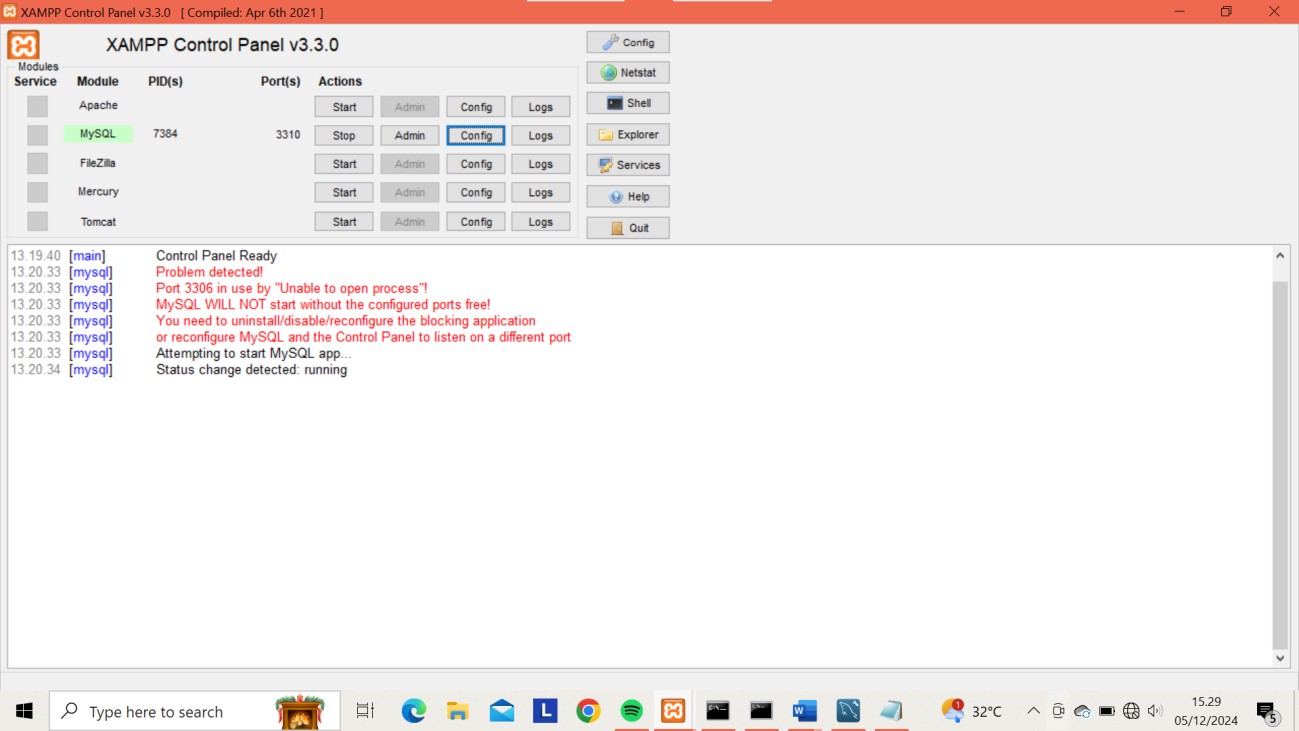




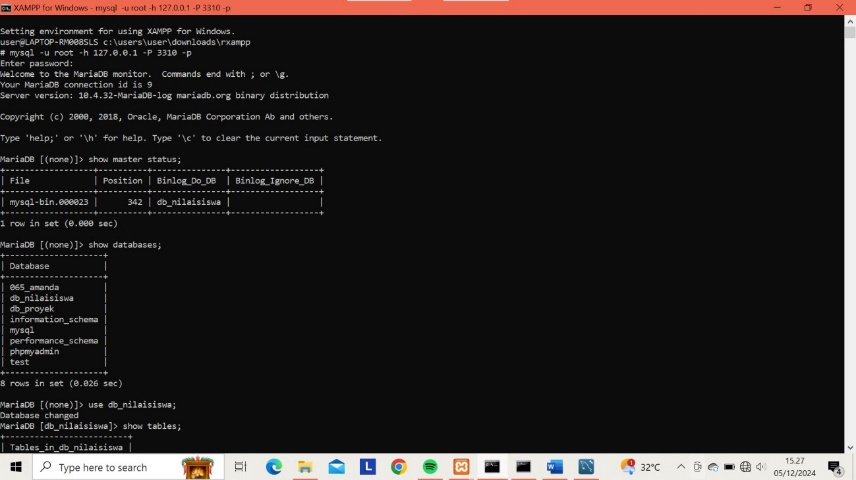
1. Buka XAMPP klik config > my ini > ketik binlog\_do\_db=db\_nilaisiswa



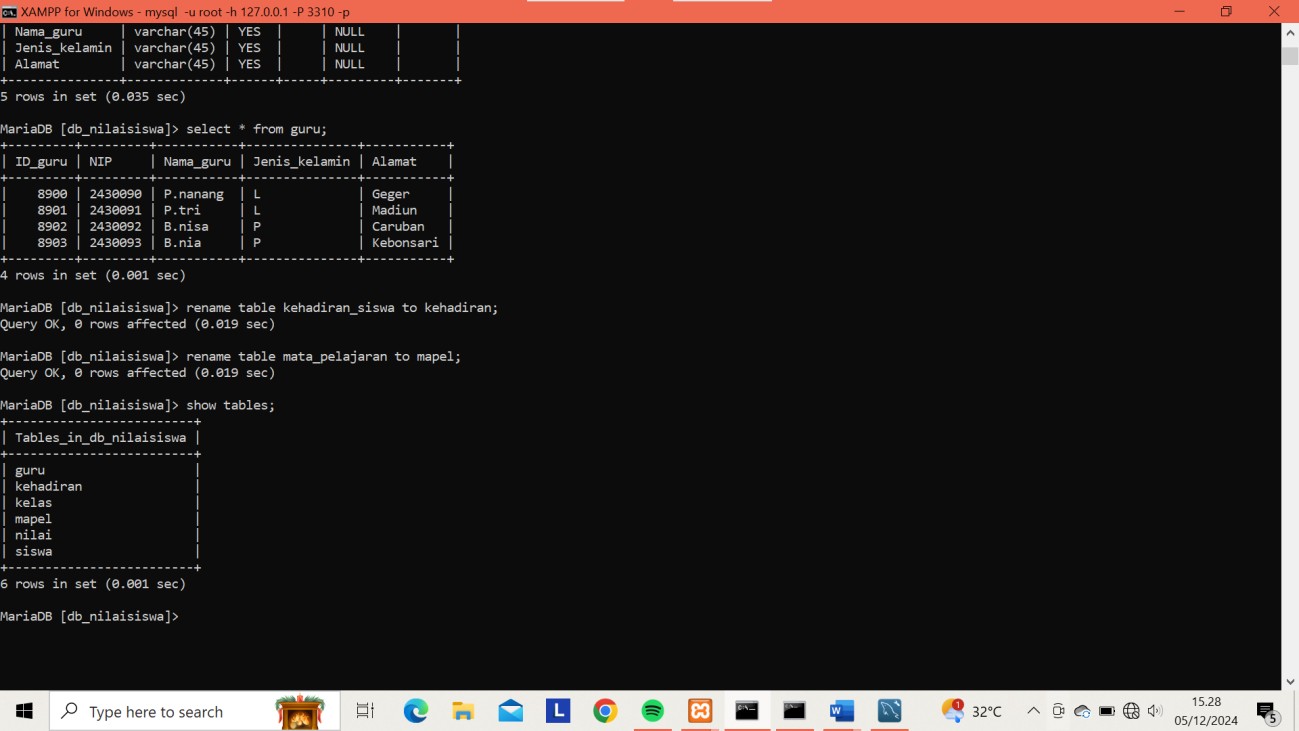
1. Klik start pada MySQL pastikan sudah running



1. Ubah nama tabel jika terdiri dari 2 kata, pastikan buat nama tabel yang singkat agar terkoneksi dengan user pertama



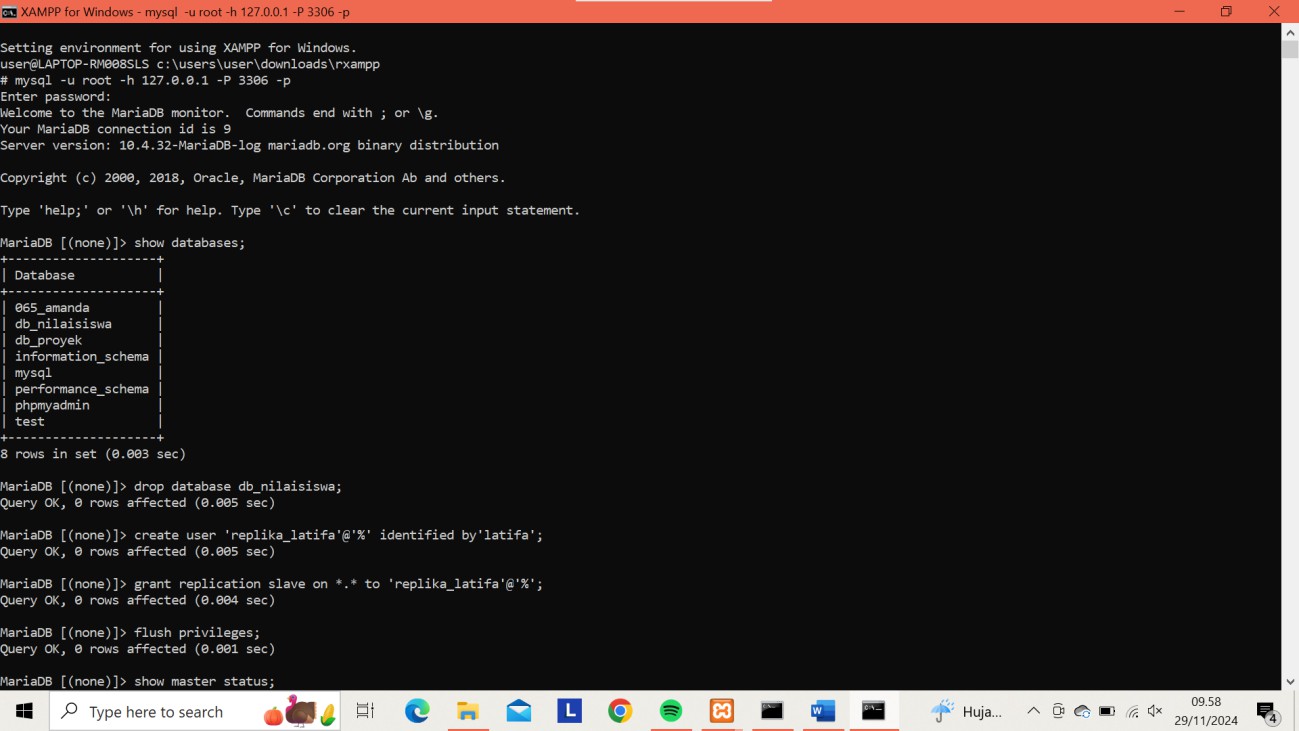
1. Show tabel untuk memastikan nama tabel sudah berubah



**5.2. KONFIGURASI KEAMANAN DATABASE MENGGUNAKAN DCL**

**a. grant replication slave on \*.\* to ‘replica\_latifa’@’%’;**

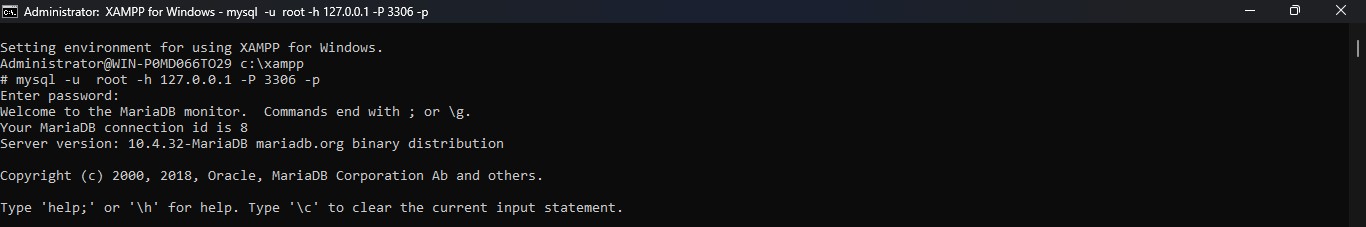
**flush privileges;**



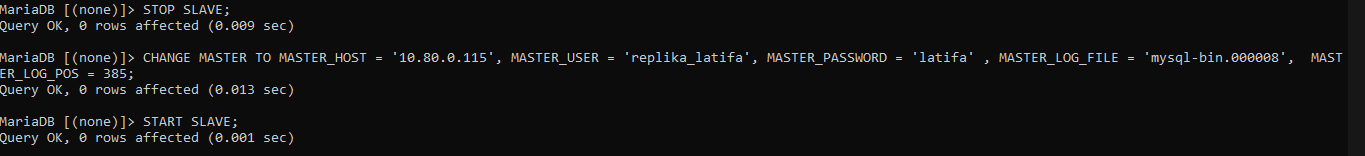
**5.3. IMPLEMENTASI REPLIKA DATABASE**

**SEBAGAI ANGGOTA 3 (SLAVE)**

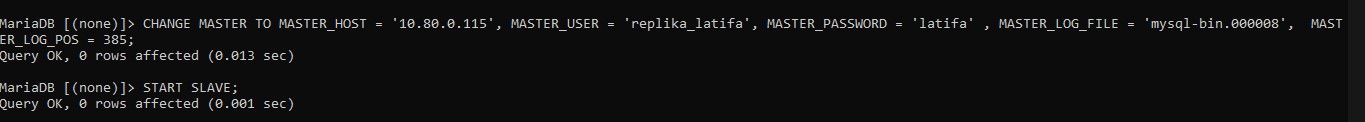
1. **Connec ke MariaDB**



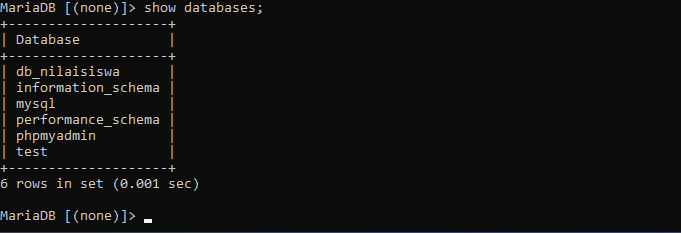
1. **Stop slave**



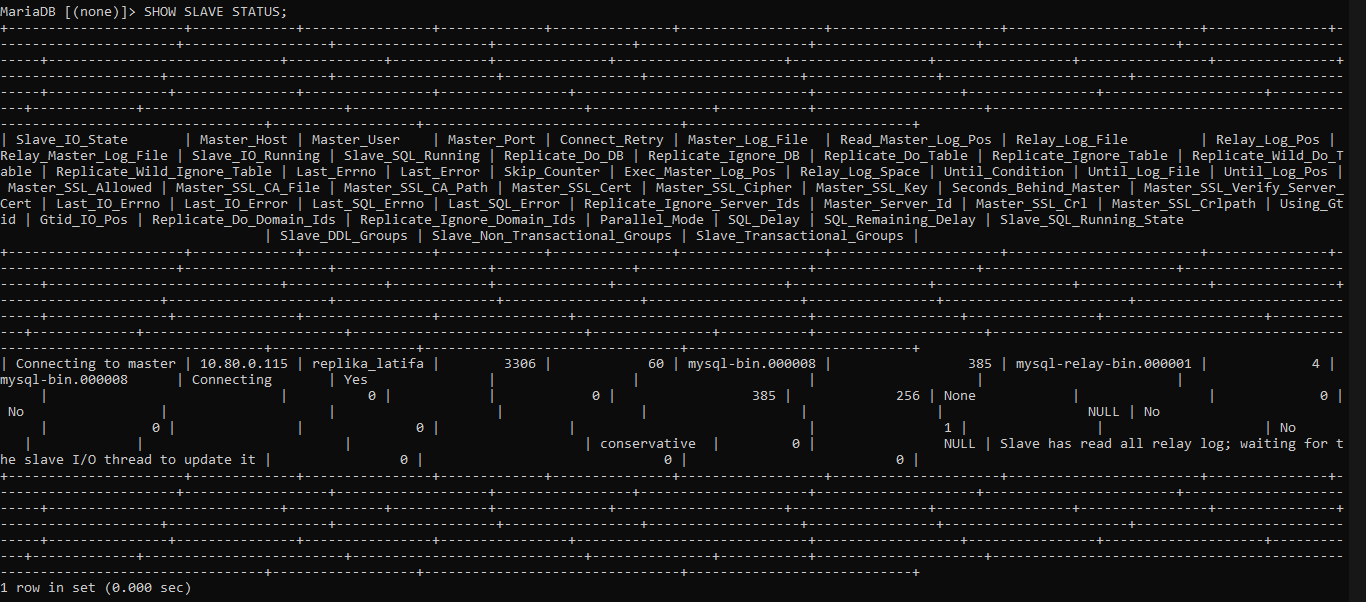
1. **Atur slave untuk mereplikasi master**



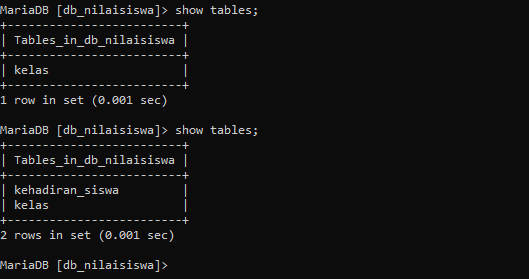
1. **Start slave**
2. **Show databases**

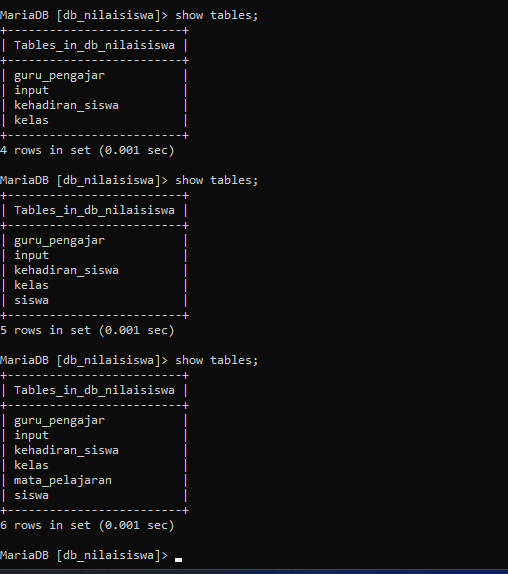


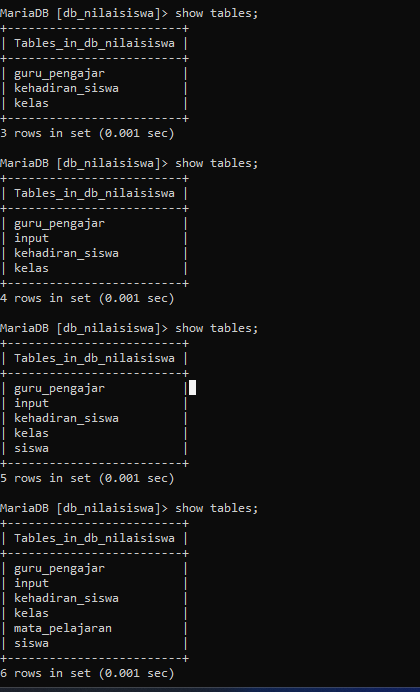
1. **Menampilkan slave**



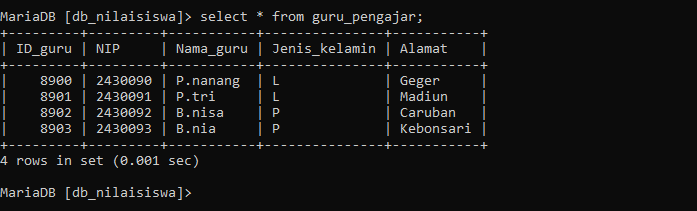
1. **Show tables**



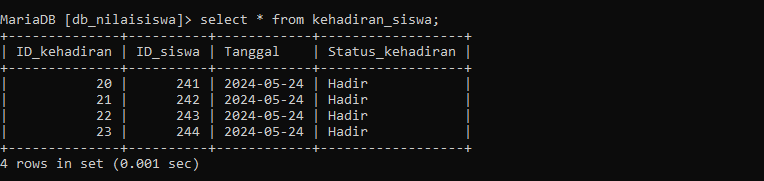




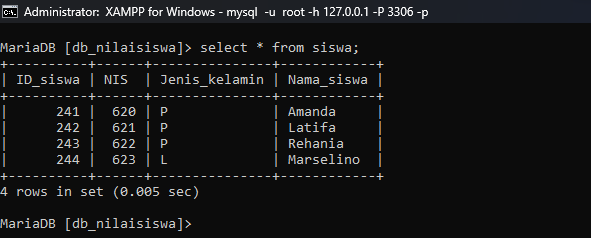
1. Menampilkan isi tabel guru pengajar

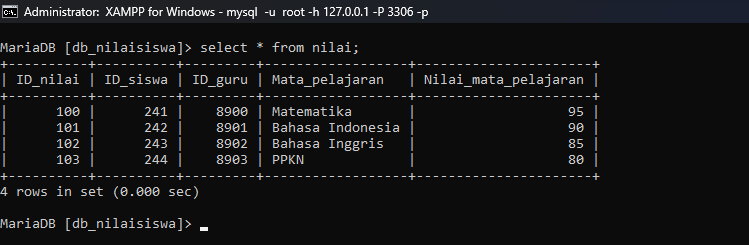


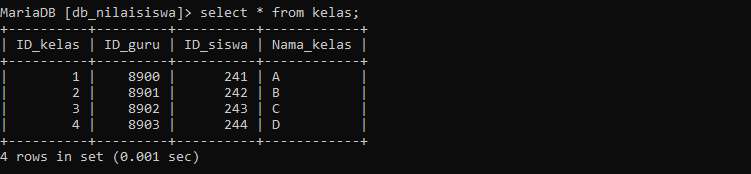
1. Menampilkan isi tabel kehadiran siswa



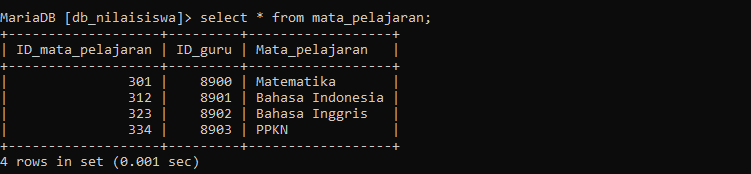
1. Menampilkan isi tabel siswa



1. Menampilkan isi tabel nilai
2. Menampikan isi tabel kelas

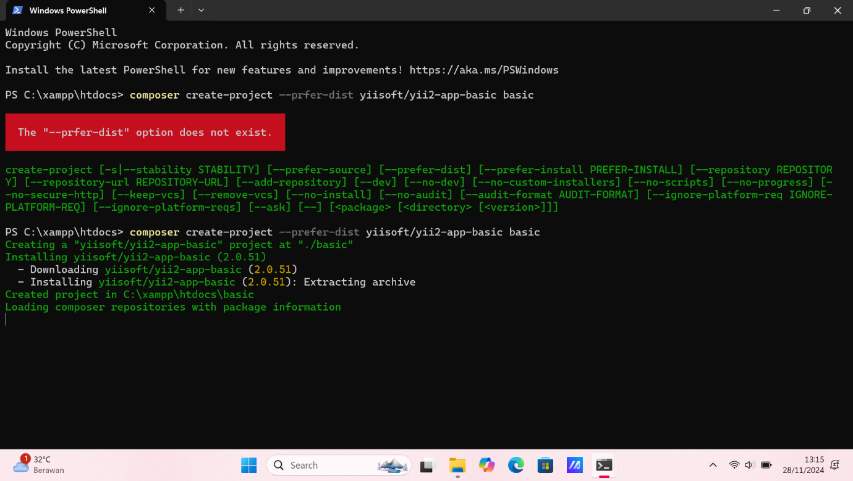


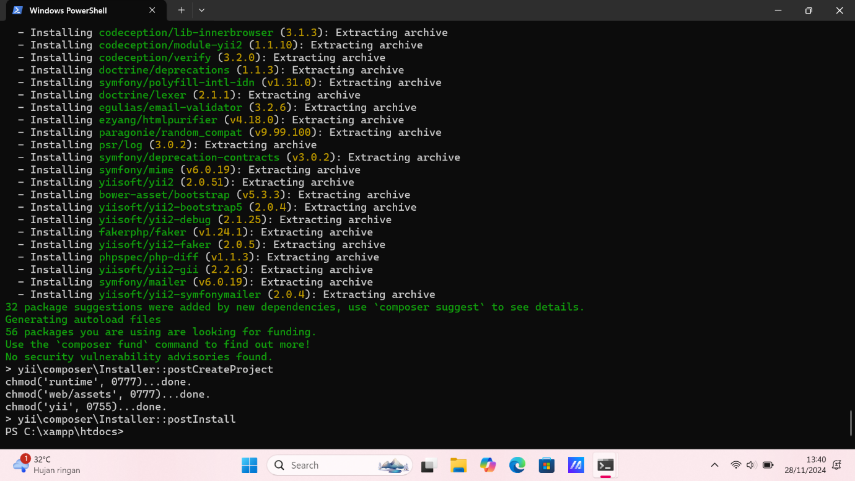
1. Menampilkan isi tabel mata pelajaran



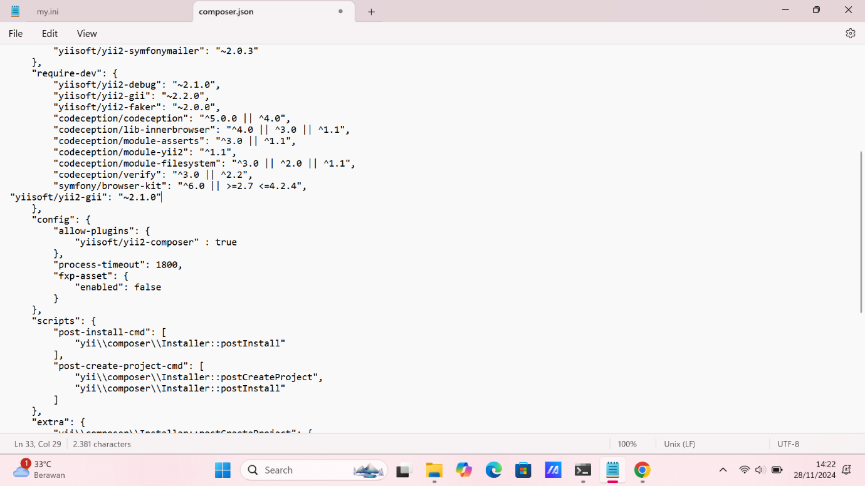
* **SEBAGAI ANGGOTA 1(WEB)**

1. **Install composer**
2. **Ketik composer create-project –prefer-dist yiisoft/yii2-app-basic basic pada terminal**

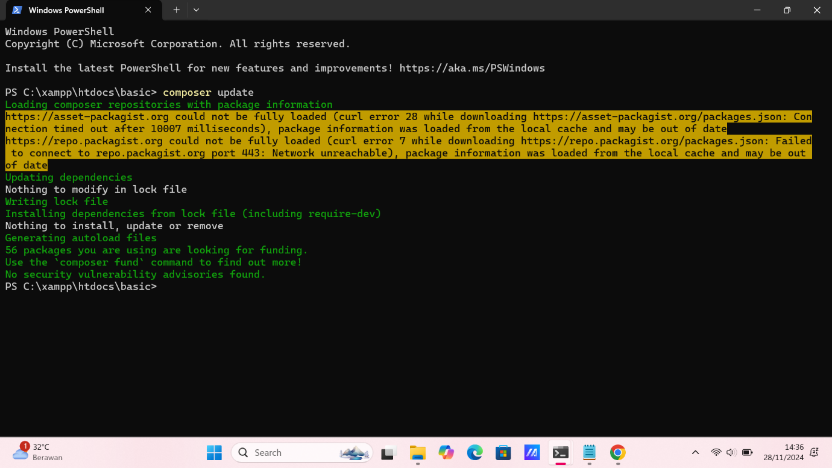
****

****

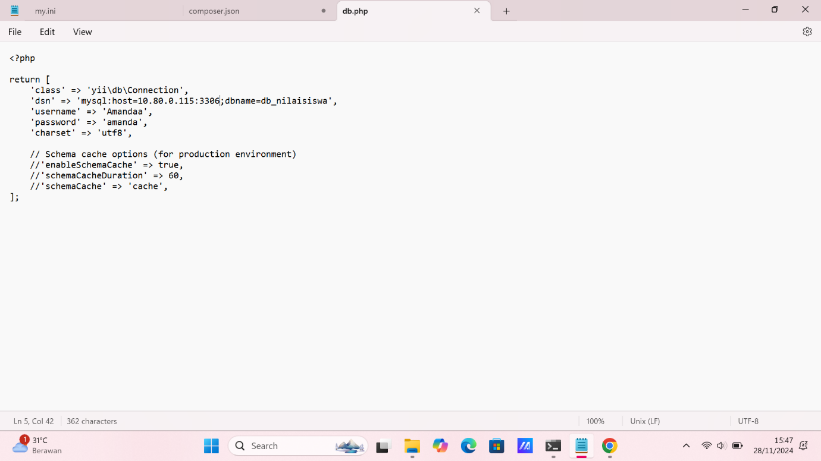
1. **Buka composer.json pada notepad lalu ketik “yiisoft/yii2-gii”: “~2.0.0”**

****

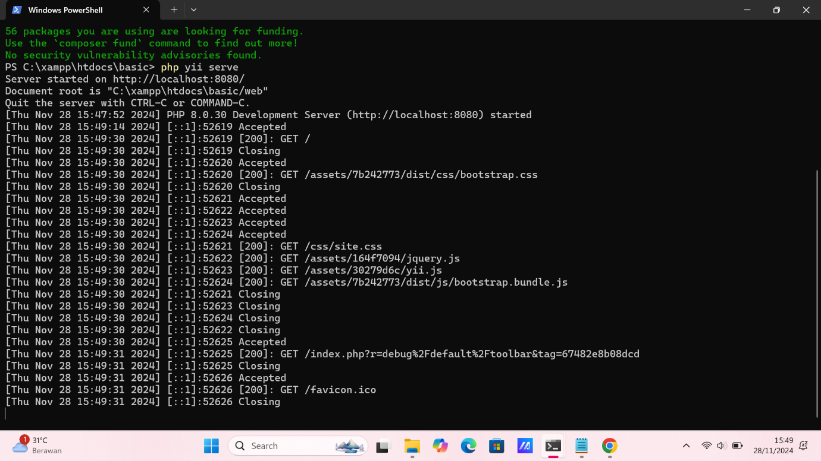
1. **Lalu buka terminal ketik composer update**

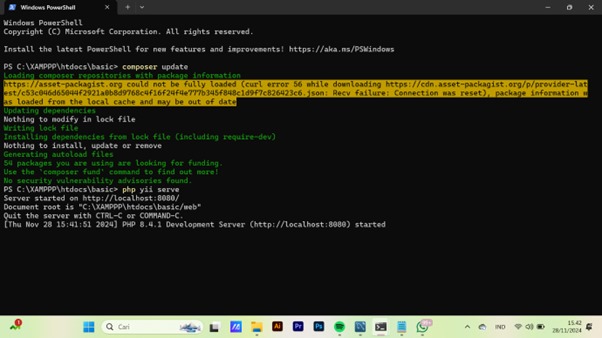
****

1. **Buka db.php pada notepad lalu isikan host, portnya berapa, nama databasenya, username dan password yang telah dibuat oleh anggota 2 yang berperan sebagai pembuat user.**

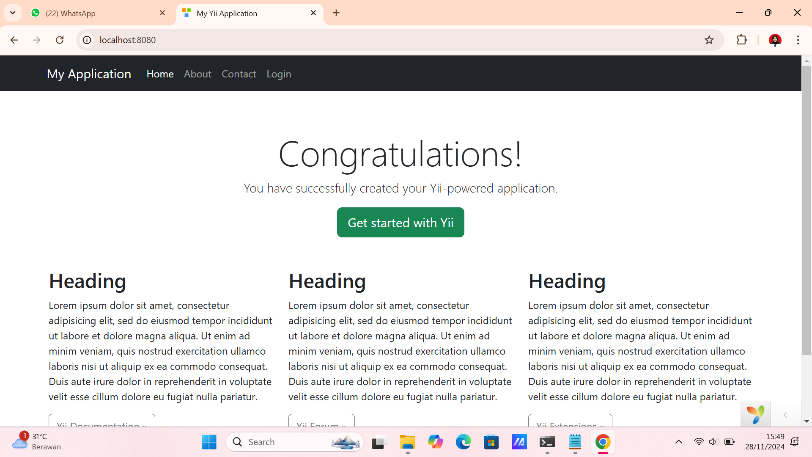
****

1. **Lalu buka terminal ketik php yii serve.**

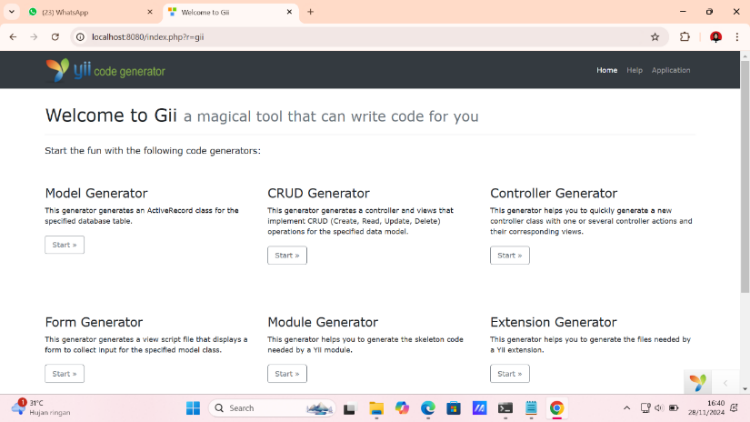
****

****

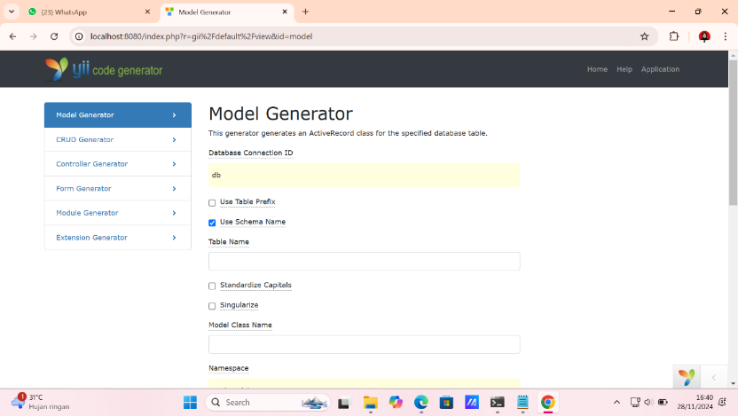
1. **Lalu salin localhost yang ada diterminal, tempelkan pada browser.**

****

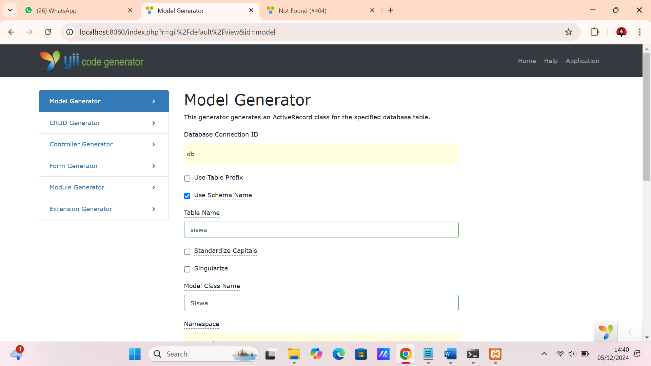
1. **Lalu tambahkan /index.php?r=gii. Lalu klik start pada model generator.**

****

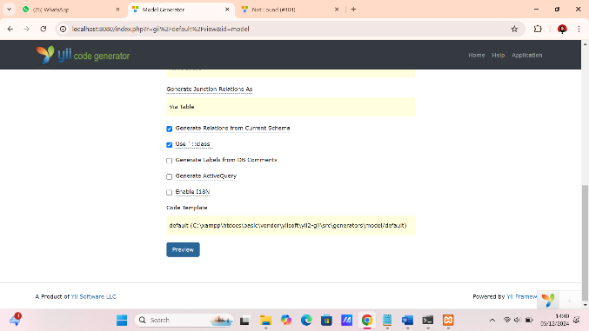
1. **Tampilannya akan seperti ini**

****

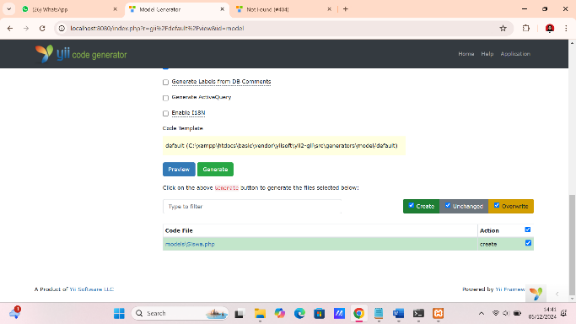
1. **Membuat Tabel Siswa**
2. Membuat Model Generator dengan mengisi table name : siswa dan model class name : Siswa



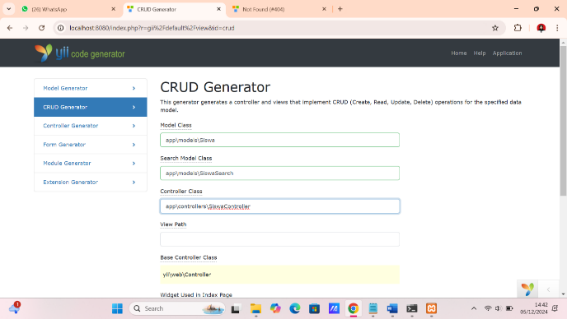
1. Klik “Preview”



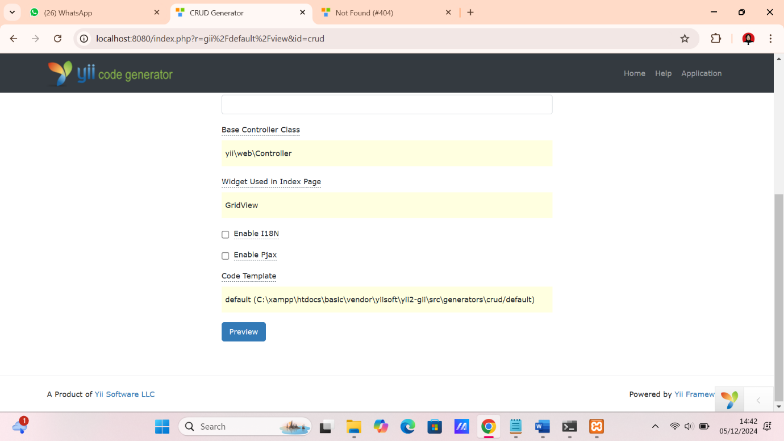
1. Lalu scroll ke bawah klik “Generate”



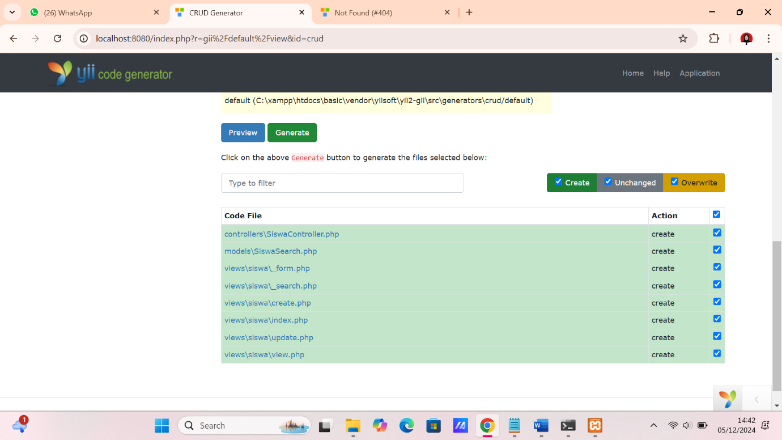
1. Membuat CRUD Generator dengan Model Class : app\models\Siswa , Search Model Class : app\models\SiswaSearch , Controller Class : app\controllers\SiswaController



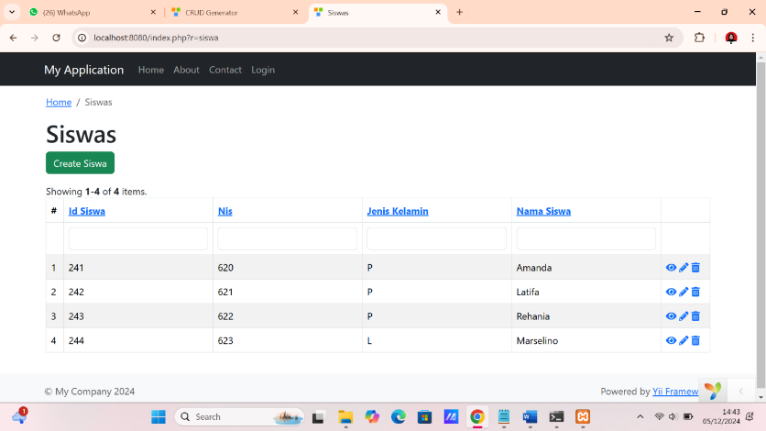
1. Klik “Preview”



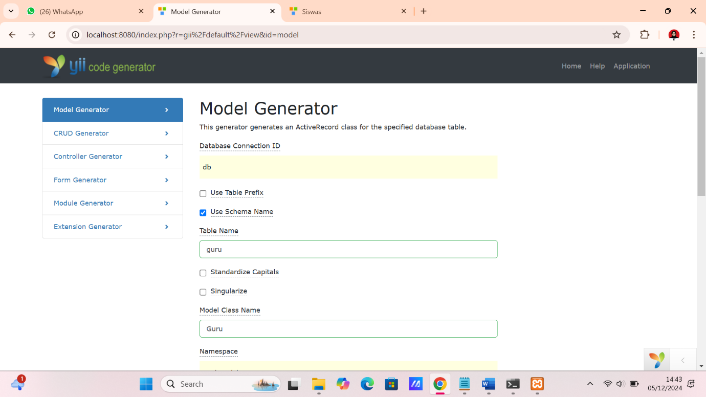
1. Lalu klik “Generate”



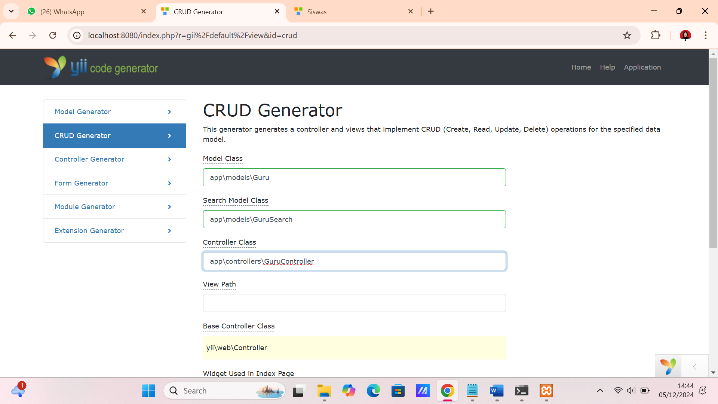
1. Cara melihat hasil tabelnya yaitu dengan cara menyalin localhost:8080/index.php?r=gii%2Fdefault%2Fview&id=crud. Setelah itu buka tab baru dan copypaste pada tab baru tersebut dan ubah menjadi localhost:8080/index.php?r=siswa



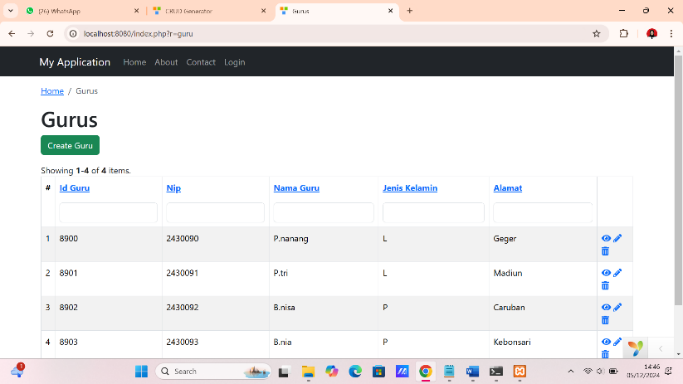
1. **Membuat Tabel Guru**
2. Membuat Model Generator dengan mengisi table name : guru dan model class name : Guru



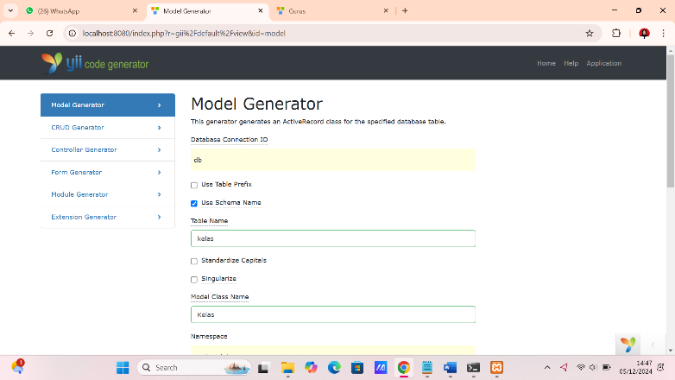
1. Klik “Preview”
2. Lalu scroll ke bawah klik “Generate”
3. Membuat CRUD Generator dengan Model Class : app\models\Guru , Search Model Class : app\models\GuruSearch , Controller Class : app\controllers\GuruController



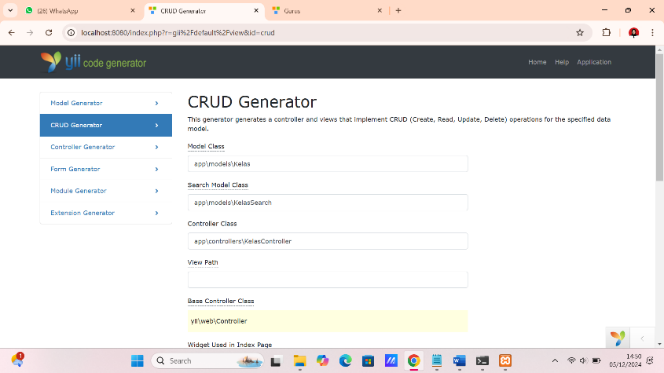
1. Klik “Preview”
2. Lalu klik “Generate”
3. Cara melihat hasil tabelnya yaitu dengan cara menyalin localhost:8080/index.php?r=gii%2Fdefault%2Fview&id=crud. Setelah itu buka tab baru dan copypaste pada tab baru tersebut dan ubah menjadi localhost:8080/index.php?r=guru



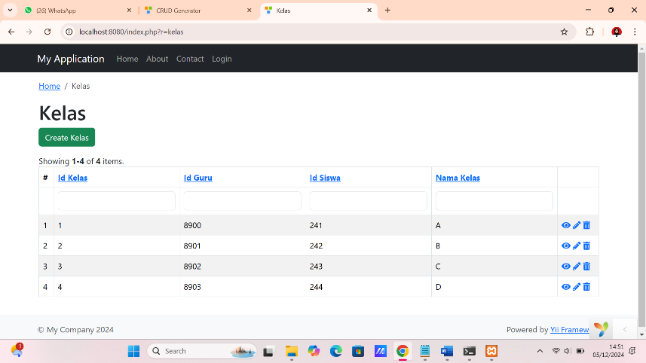
1. **Membuat Tabel Kelas**
2. Membuat Model Generator dengan mengisi table name : kelas dan model class name : Kelas

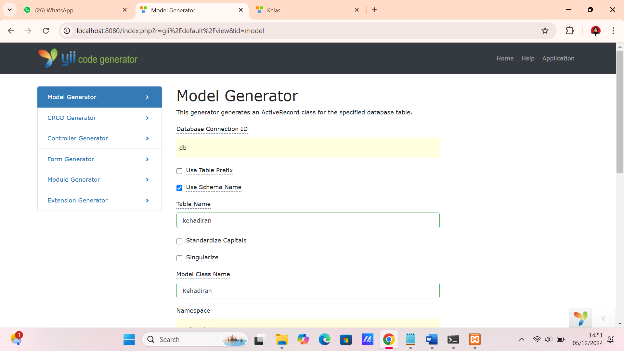


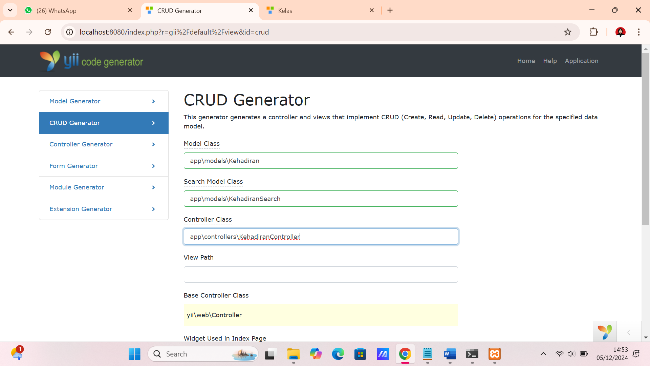
1. Klik “Preview”
2. Lalu scroll ke bawah klik “ Generate”
3. Membuat CRUD Generator dengan Model Class : app\models\Kelas , Search Model Class : app\models\KelasSearch , Controller Class : app\controllers\KelasController



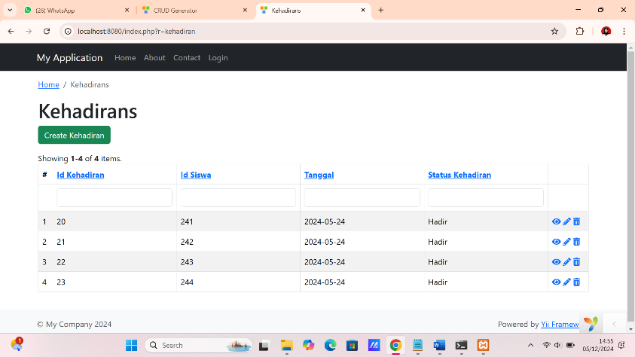
1. Klik “Preview”
2. Lalu klik “Generate”
3. Cara melihat hasil tabelnya yaitu dengan cara menyalin localhost:8080/index.php?r=gii%2Fdefault%2Fview&id=crud. Setelah itu buka tab baru dan copypaste pada tab baru tersebut dan ubah menjadi localhost:8080/index.php?r=kelas



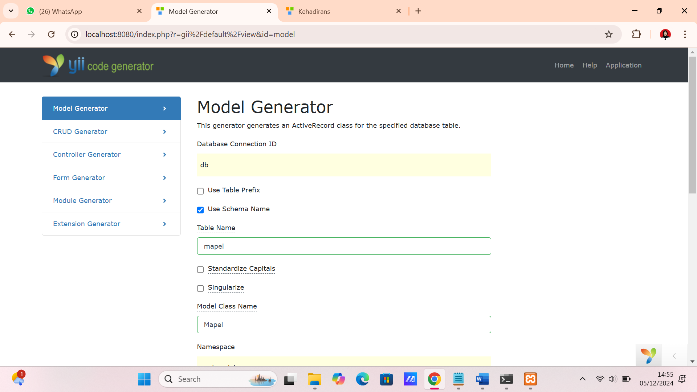
1. **Membuat Tabel Kehadiran**
2. Membuat Model Generator dengan mengisi table name : kehadiran dan model class name : Kehadiran
3. Klik “Preview”
4. Lalu scroll ke bawah klik “ Generate”
5. Membuat CRUD Generator dengan Model Class : app\models\Kehadiran , Search Model Class : app\models\Kehadiran , Controller Class : app\controllers\KehadiranController



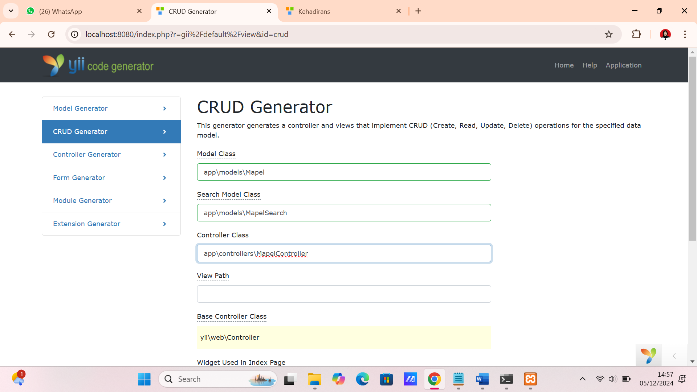
1. Klik “Preview”
2. Lalu klik “Generate”
3. Cara melihat hasil tabelnya yaitu dengan cara menyalin localhost:8080/index.php?r=gii%2Fdefault%2Fview&id=crud. Setelah itu buka tab baru dan copypaste pada tab baru tersebut dan ubah menjadi localhost:8080/index.php?r=kehadiran



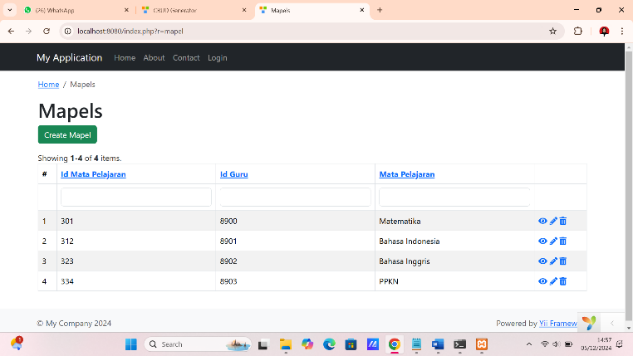
1. **Membuat Tabel Mapel**
2. Membuat Model Generator dengan mengisi table name : mapel dan model class name : Mapel



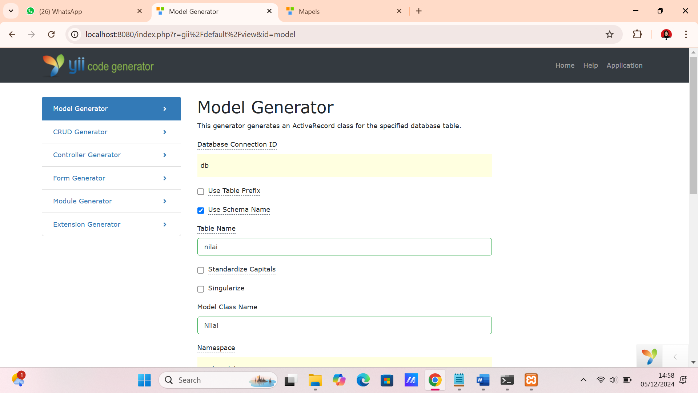
1. Klik “Preview”
2. Lalu scroll ke bawah klik “Generate”
3. Membuat CRUD Generator dengan Model Class : app\models\Mapel , Search Model Class : app\models\Mapel , Controller Class : app\controllers\MapelController



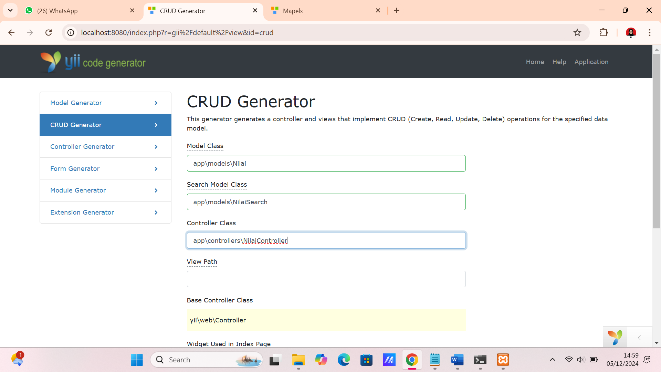
1. Klik “Preview”
2. Lalu klik “Generate”
3. Cara melihat hasil tabelnya yaitu dengan cara menyalin localhost:8080/index.php?r=gii%2Fdefault%2Fview&id=crud. Setelah itu buka tab baru dan copypaste pada tab baru tersebut dan ubah menjadi localhost:8080/index.php?r=mapel



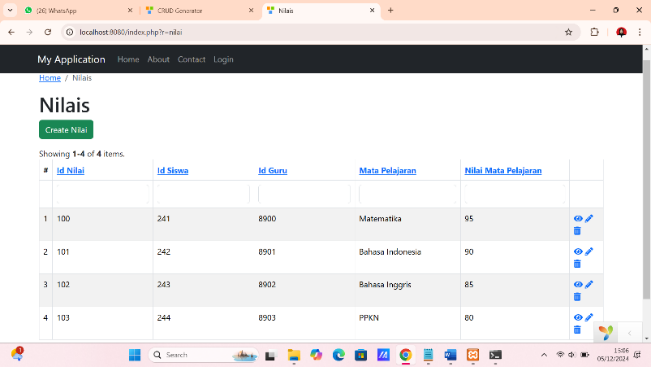
1. **Membuat Tabel Nilai**
2. Membuat Model Generator dengan mengisi table name : nilai dan model class name : Nilai



1. Klik “Preview”
2. Lalu scroll ke bawah klik “Generate”
3. Membuat CRUD Generator dengan Model Class : app\models\Nilai , Search Model Class : app\models\Nilai , Controller Class : app\controllers\NilaiController

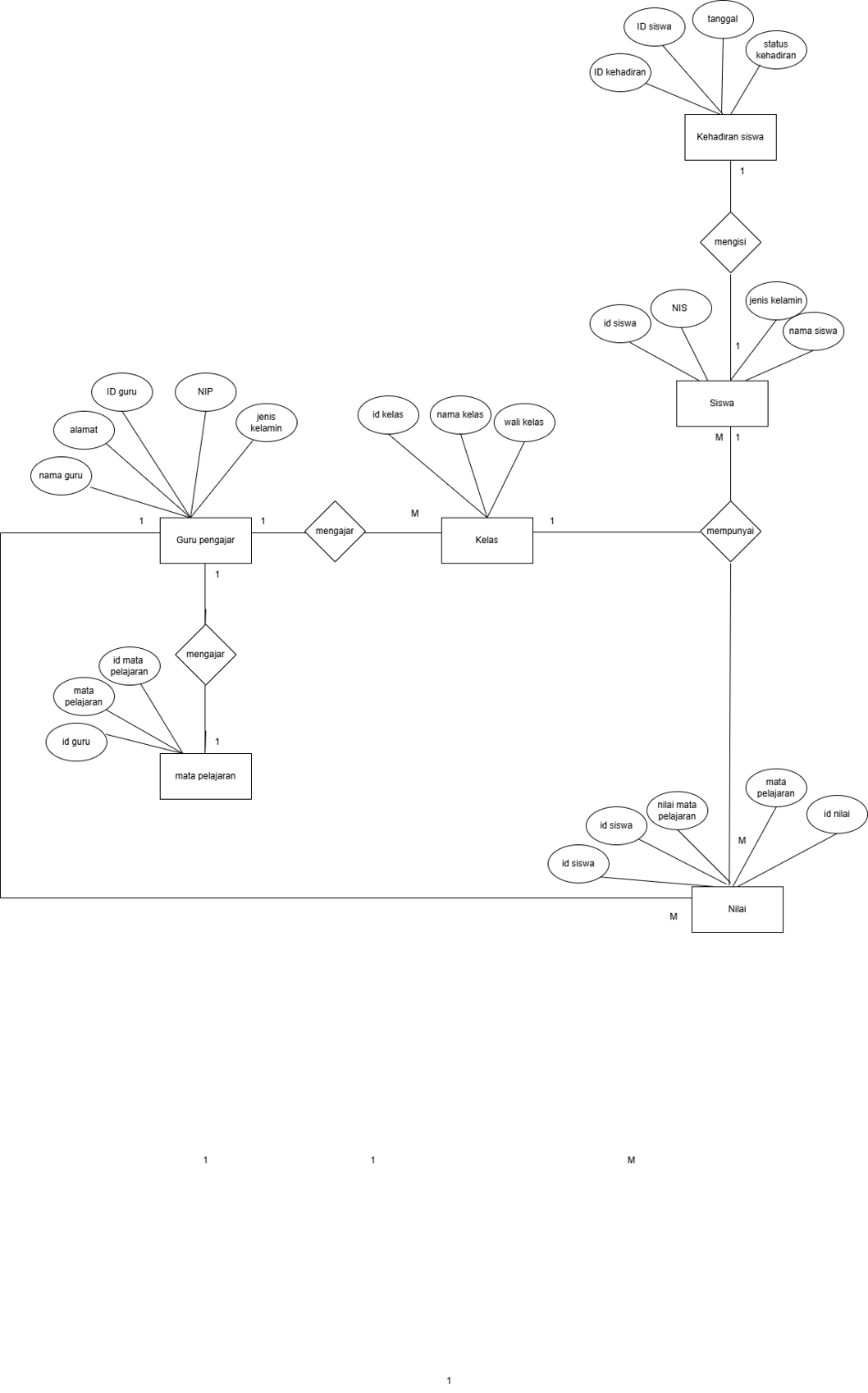


1. Klik “Preview”
2. Lalu klik “Generate”
3. Cara melihat hasil tabelnya yaitu dengan cara menyalin localhost:8080/index.php?r=gii%2Fdefault%2Fview&id=crud. Setelah itu buka tab baru dan copypaste pada tab baru tersebut dan ubah menjadi localhost:8080/index.php?r=nilai



**6.DOKUMENTASI**

**6.1. DOKUMENTASI DESAIN DATABASE**

****

**a. DESAIN BASIS DATA**

**- MODEL KONSEPTUAL**

* Entitas: Nilai, mata pelajaran, siswa, guru pengajar,kelas,kehadiran siswa
* Hubungan:

Nilai: siswa memiliki nilai mata pelajaran.

Guru pengajar: guru mengajar mata Pelajaran di setiap kelas

Kehadiran siswa : siswa memiliki absensi kehadiran

* **MODEL LOGIS**

Tabel :

* Siswa(ID Siswa,NIS, Jenis\_Kelamin, Nama)
* Mata Pelajaran(ID Mata Pelajaran, ID\_Guru, Mata\_Pelajaran)
* Nilai(ID Nilai, ID Siswa, ID\_Guru, mata\_Pelajaran,nilai\_mata Pelajaran)
* Guru Pengajaran(ID Guru, NIP, Nama Guru, Jenis\_Kelamin, Alamat)
* Kelas(ID Kelas, ID Guru, ID Kelas, Nama Kelas)
* Kehadiran Siswa(ID Kehadiran, ID Siswa, Tanggal, Status Kehadiran (Hadir, Sakit, Izin, Alpa))
* **MODEL FISIK**

Indeks: Indeks pada Kolom id\_siswa dan id\_pelajaran

Optimasi: Karena pencarian berdasarkan siswa dan mata pelajaran sering dilakukan, indeks pada kedua kolom ini dapat mempercepat proses pencarian nilai untuk siswa tertentu atau untuk mata pelajaran tertentu.

Indeks: Indeks Gabungan (id\_siswa)

Optimasi: Dapat digunakan untuk mempercepat query yang sering kali melibatkan filter berdasarkan siswa dan semester tertentu.

**6.2.MANUAL PENGGUNAAN SISTEM**

**- Tata Cara Penggunaan Xampp**

1.Menjalankan XAMPP dan MySQL

* Buka XAMPP Control Panel.
* Start Apache dan MySQL dengan mengklik tombol Start.

2. Membuka MySQL Shell

* Buka Command Prompt (CMD) atau Terminal.
* Login ke MySQL Shell dengan mengetikkan perintah berikut:

**# mysql -u root -h 127.0.0.1 -P 3306 -p**

* Tekan Enter. Jika tidak ada password, langsung tekan Enter lagi.
* Jika password telah diatur, masukkan password-nya.
* **Tata Cara Penggunaan WorkBeanch**
  1. **Instalasi MySQL Workbench**
* Unduh MySQL Workbench dari situs resmi MySQL
* Instal perangkat lunak sesuai instruksi yang muncul.
  1. **Membuka MySQL Workbench**
* Jalankan MySQL Workbench setelah proses instalasi selesai.
  1. **Membuat Koneksi ke Server Database**
* Klik tanda **+** di tab *"MySQL Connections"* untuk menambahkan koneksi baru.
* Isi detail berikut:
  + **Connection Name**: Nama koneksi yang Anda inginkan.
  + **Hostname**: Biasanya localhost jika server berjalan secara lokal.
  + **Port**: 3306 (default untuk MySQL).
  + **Username**: Nama pengguna database.
  + **Password**: Kata sandi pengguna.
* Klik **Test Connection** untuk memeriksa koneksi.
* Klik **OK** jika koneksi berhasil.
  1. **Membuat Database Baru**
* Buka koneksi yang telah dibuat.
* Di tab **Navigator**, klik kanan pada bagian **Schemas** > pilih **Create Schema**.
* Isi nama database, lalu klik **Apply**.
  1. **Membuat Tabel di Database**
* Pilih database yang sudah dibuat di **Schemas**.
* Klik kanan > pilih **Create Table**.
* Isi nama tabel dan tambahkan kolom-kolom dengan detail seperti:
  + **Name**: Nama kolom.
  + **Data Type**: Jenis data (misal: INT, VARCHAR, DATE, dsb.).
  + **PK** (Primary Key), **NN** (Not Null), dll jika diperlukan.
* Klik **Apply** untuk menyimpan tabel.
  1. **Menjalankan Query SQL**
* Klik tab **SQL Query** di toolbar utama.
* Ketik perintah SQL, misalnya: SELECT \* FROM nama\_tabel;
* Klik tombol **Execute** (ikon petir) untuk menjalankan query.
* **Tata Cara Penggunaan Composer**
  1. Install composer
  2. Ketik composer create-project –prefer-dist yiisoft/yii2-app-basic basic pada terminal
  3. Buka composer.json pada notepad lalu ketik “yiisoft/yii2-gii”: “~2.0.0”
  4. Lalu buka terminal ketik composer update
  5. Buka db.php pada notepad lalu isikan host, portnya berapa, nama databasenya, username dan password yang telah dibuat oleh anggota 2 yang berperan sebagai pembuat user.
  6. Lalu buka terminal ketik php yii serve