NAMA : Dimas Syauqi Syafa

KELAS : LB20

NIM : 25020044005

**1. Jelaskan konsep tentang data, informasi, database, dan DBMS. Jelaskan pula manfaat database bagi perusahaan dan kehidupan seharai-hari. Berikan contoh untuk memudahkan pemahaman mengenai konsep tersebut!**

**Data :** data merupakan deskripsi mengenai sesuatu dan kejadian yang terjadi, data sendiri adalah fakta yang menggambarkan sesuatu peristiwa dan kesatuan nyata.

**Informasi :** informasi merupakan sebuah kumpulan data mengenasi suatau kejadian yang telah, sedang, bahkan yang akan dating.

**Database :** database adalah sebuah kumpulan data yang disimpan data bentuk yang sistematis dan system tersebut dirancang untuk memnuhi kebutuhan client atau user atau pemesan atau penerima.

**DBMS :** DBMS adalah sebuah hardware yang mempunyai sekaligus menangani smua akses terhadap database.

**Manfaat :** dalam mengolah data menjadi lebih mudah, penyimpanan informasi yang di butuhkan perushaan juga lebih mudah di akses oleh tim internal perushaan, lebih aman jika terjadi bencana alam karena jika menggunakan dokumen paper akan mengalami kerusakan lebih banyakj dari pada menggunakana database

**2. Jelaskan konsep Super Key, Candidate Key, Primary Key, Alternative Key dan Foreign Key. Berikan contoh kelima key tersebut dengan membuat dua tabel database, kemudian tentukan key nya beserta alasannya. Untuk topik dari tabel silakan tentukan sendiri!**

**Contoh database :**

CREATE TABLE `user` (

ID\_User int NOT NULL,

KTP\_User varchar(30) NOT NULL,

Nama\_User varchar(30) NOT NULL,

TmptLahir\_User varchar(30),

TglLahir\_User date,

Email\_User varchar(30),

PRIMARY KEY(ID\_User)

)

CREATE TABLE `Rental\_PS` (

ID\_Penyewaan int NOT NULL,

Kode\_PS varchar(30),

KTP\_User varchar(30),

Tanggal\_Peminjaman date,

PRIMARY KEY(ID\_Penyewaan)

)

**- Super Key :** super key sendiri merupakan kumpulan atribut yang mengidentifikasi data (baris) dalam sebuah table di database secara unik. Super key merupakan superset dari candidate key. Dalam satu relasi jumlah super key harusnya lebih banyak dari pada jumlah candidate key. (**ID\_User, KTP\_User, TmptLahir\_User**)

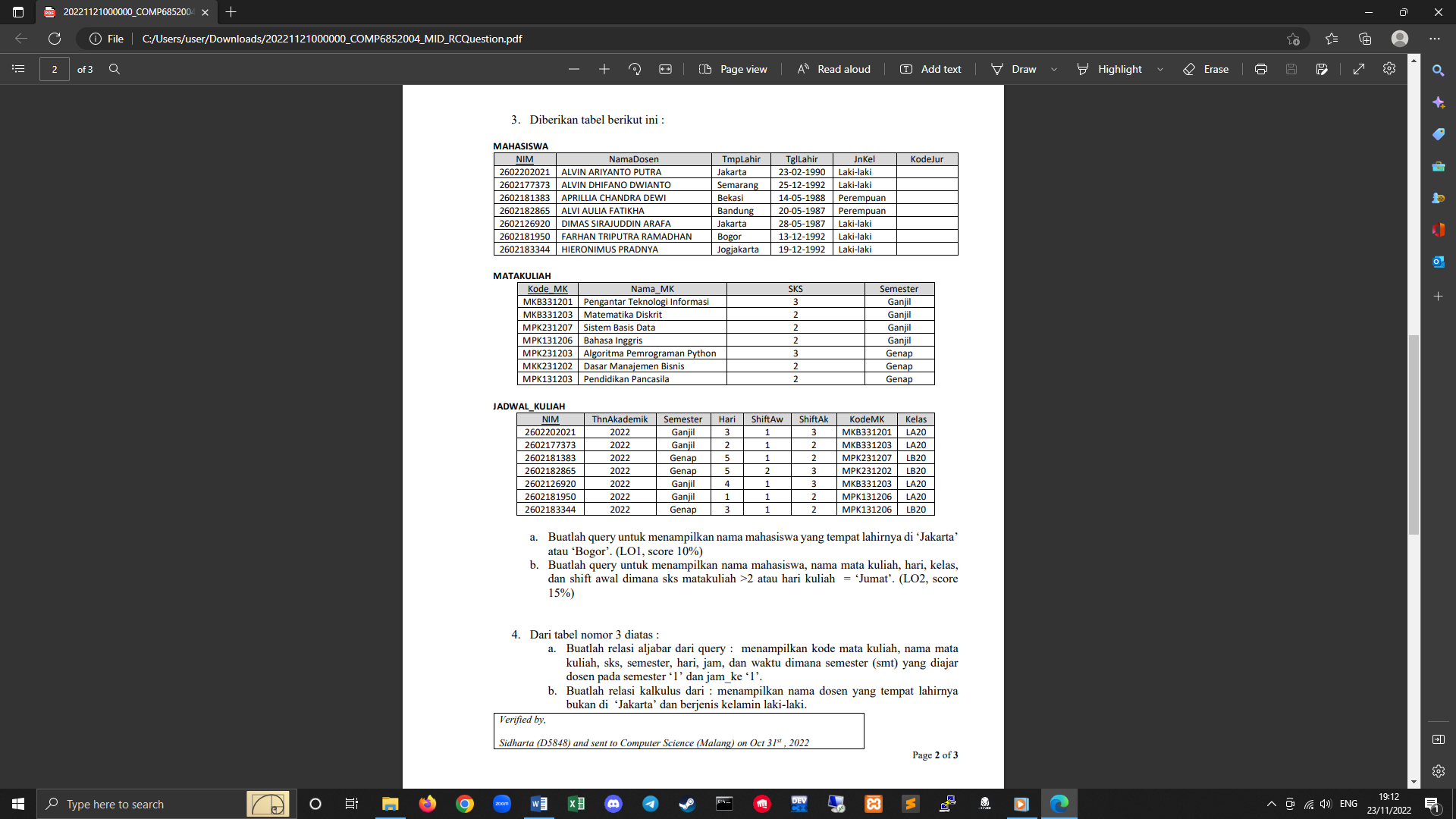
**- Candidate Key :** candidatekey merupakan sekumpulan attribute yang unik mengidentifikasi tupel dalam table di database, candidate key adalah super key tanpa atribut berulang. Jika pada super key ada beberapa gabungan field yang bias dijadikan key, maka di candidate key ini kita melihat paling sedikit dari gabungan field tersebut, contoh ada gabungan 3 field, ada gabungan lagi tetapi 2 field, nah yang akan diambil sebagai candidate key yaitu gabugan 2 field yang cocok karena gabungan tersebut lebih sedikit. (**ID\_User, KTP\_User, TmptLahir\_User, EMAIL**)

**- Primary Key :** primary key itu seperti ‘tanda pengenal yang unik’ dimana setiap table / data yang ada di table database memiliki primary key yang berbeda, konsep primary key ini harus berisi nilai yang unik agar semua data tidak memiliki kesamaan pada primary keynya, kolom primary key ini juga tidak boleh berisi NULL, dan setiap table hanya memilki 1 primary key. (**NIM**)

**- Alternative Key :** alternate / alternative key merupakan candidate key yang tidak dipakai atau dipilih sebagai primary key, setiap candidate key yang misalnya tidak menjadi primary key disebut alternative key, konsep ini sebenarnya mirip seperti primary key yaitu berjenis kunci yang unik maksudnya primary key yang kita buat pada table adalah auto increment sehingga tidak dimungkingkan terdapat nilai sama, dikarenakan tidak diperbolehkan ada 2 primary key maka dapat di tambah dengan alternative key ini. (**KTP\_User, Email, ID\_User**)

**- Foreign Key :** sama seperti primary key, foregin key juga merupakan tanda pengenal yang unik, letak perbedaannya yaitu, foreign key untuk menghubungkan dua table atau lebih dalam sebuah database, konsep untuk menentukan foreign key kita harus tau dahulu mana sebuah objek dan mana sebuah subjek dalam hubungannya. (**KTP\_user**)

**3. Diberikan tabel berikut ini :**

****

1. **Buatlah query untuk menampilkan nama mahasiswa yang tempat lahirnya di ‘Jakarta’ atau ‘Bogor’.**

SELECT NamaMahasiswa FROM `MAHASISWA` WHERE TmpLahir='Jakarta' OR TmpLahir ='Bogor'

1. **Buatlah query untuk menampilkan nama mahasiswa, nama mata kuliah, hari, kelas, dan shift awal dimana sks matakuliah >2 atau hari kuliah = ‘Jumat’.**

SELECT ms.NamaDosen, mk.Nama\_MK, j.Hari, j.Kelas, j.ShiftAw FROM MAHASISWA ms, MATAKULIAH mk, JADWAL\_KULIAH j WHERE mk.SKS > 2 OR j.Hari=5

**4. Dari tabel nomor 3 diatas :**

1. **Buatlah relasi aljabar dari query : menampilkan kode mata kuliah, nama mata kuliah, sks, semester, hari, jam, dan waktu dimana semester (smt) yang diajar dosen pada semester ‘1’ dan jam\_ke ‘1’.**

**Query :**

SELECT mk.Kode\_MK, mk.Nama\_MK, mk.SKS, mk.Semester, j.Hari, j.ShiftAw FROM MATAKULIAH mk, JADWAL\_KULIAH j WHERE mk.Semester=’Ganjil’ AND j.ShiftAw=1

**Relasi Aljabar :**

π (mk.Kode\_MK, mk.Nama\_MK, mk.SKS, mk.Semester, j.Hari, j.ShiftAw)(σ(Semester=’Ganjil’^ShiftAw=1))(j x mk)

1. **Buatlah relasi kalkulus dari : menampilkan nama dosen yang tempat lahirnya bukan di ‘Jakarta’ dan berjenis kelamin laki-laki.**

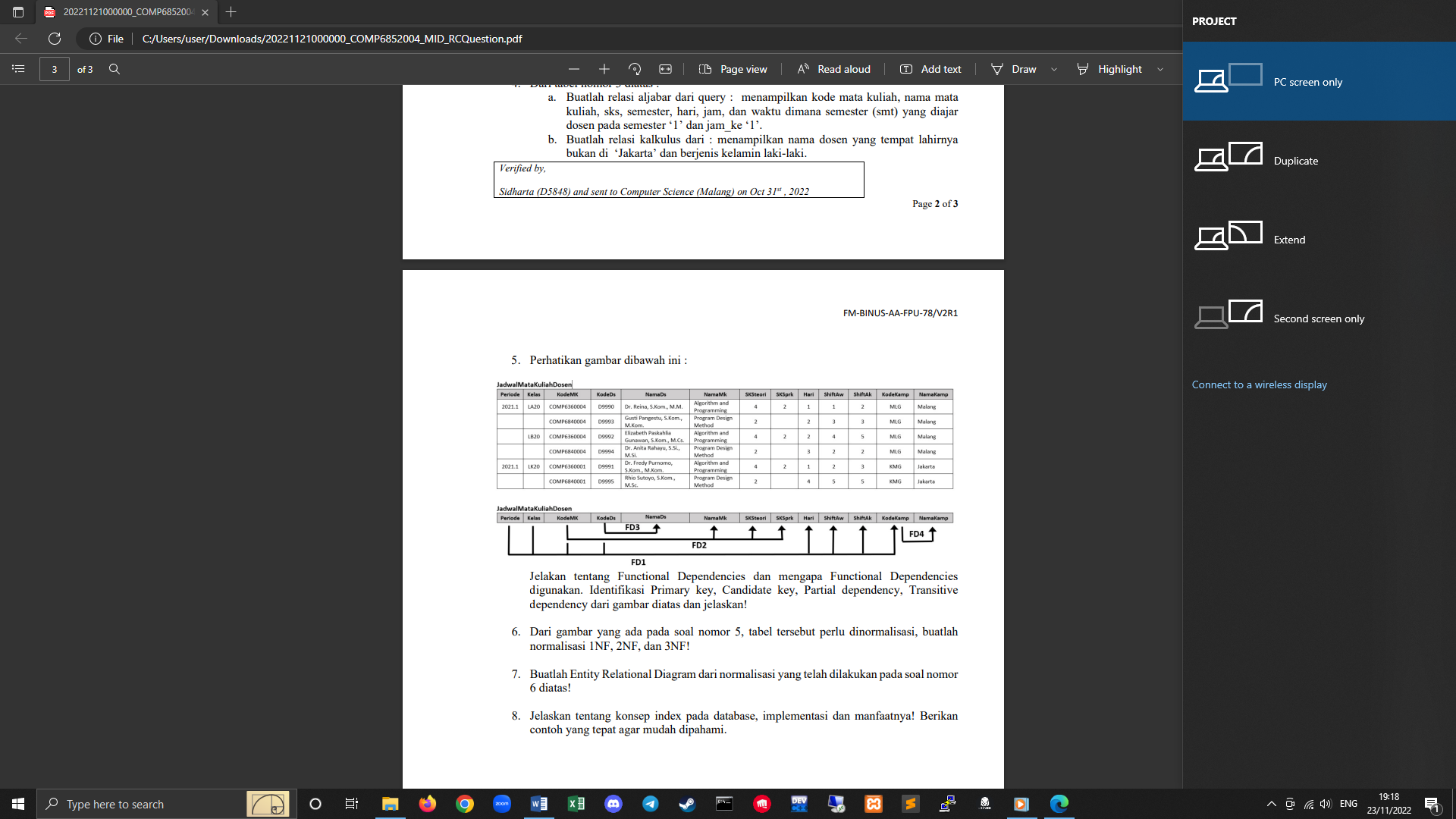
**Query :**

SELECT NamaDosen FROM MAHASISWA WHERE TmpLahir != ‘Jakarta’ AND JnKel=’Laki-laki’

**Relasi Kelkulus :**

{NamaDosen(MAHASISWA (NamaDosen,) AND TmpLahir ≠’Jakarta’ AND JnKel=’Laki-laki’)}

**5. Perhatikan gambar dibawah ini :**

****

**Jelakan tentang Functional Dependencies dan mengapa Functional Dependencies digunakan. Identifikasi Primary key, Candidate key, Partial dependency, Transitive dependency dari gambar diatas dan jelaskan!**

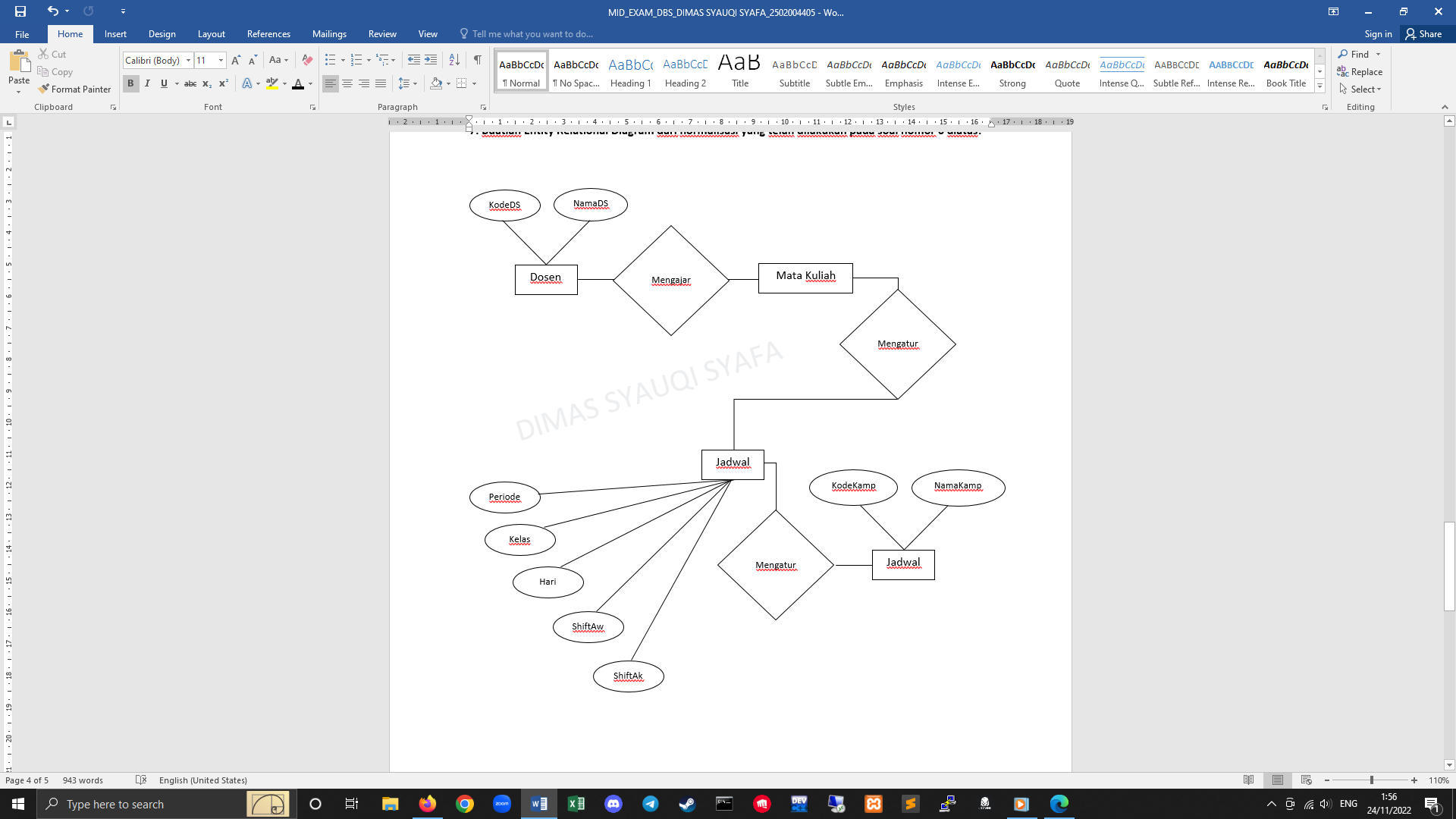
**6. Dari gambar yang ada pada soal nomor 5, tabel tersebut perlu dinormalisasi, buatlah normalisasi 1NF, 2NF, dan 3NF!**

**Normalisasi 1NF :**

**Normalisasi 2NF :**

**Normalisasi 3NF :**

**7. Buatlah Entity Relational Diagram dari normalisasi yang telah dilakukan pada soal nomor 6 diatas!**

****

**8. Jelaskan tentang konsep index pada database, implementasi dan manfaatnya! Berikan contoh yang tepat agar mudah dipahami.**

Index sendiri adalah sebuah object dalam sebuah database yang dapat mempercepat proses kita dalam pencarian atau sering disebut dengan ‘query’ data, jika database yang kita buat tanpa menggunakan index maka pemrosesan respon dari server akan lambat sekali. Ini dikarenakan resource cpu pada server banyak sekali digunakan untuk pencarian data atau saat mengakses query sql dengan metode scan table database. Fungsi index inilah yang dapat mempercepat atau dapat lebih optimal karena tidak banyak menghabiskan resource CPU, setiap index memiliki nilai kolom dan row id ke baris yang bernilai tersebut, row id tersbut secara langsung menunjuk ke baris data yang tepat sehingga dapat menghindari ‘full table scan’, tapi jika banyak index dalam table tidak berarti dapat lebih cepat proses query, malah semakin banyak akan menyababkan kelambatan proses perintah data manipulation language (DML).

Contoh serta Implementasi:

Misalnya saya memiliki database penyewaan bed cover dengan jumlah penyewa (user) di dalam database saya berjumlah 2 juta user, contoh query untuk menampilkan jumlah : SELECT COUNT(\*) FROM `user`, maka akan muncul jumlahnya 2 juta user.. jika kita melakukan query seperti biasa misalnya ingin mengambil data pada table user di baris ‘X’ dimana baris X terdapat data user berupa KTP user yang bernilai ‘123778124’ : SELECT \* FROM `user` WHERE ktp=’123778124’, jika baris tersebut berada pada urutan ke 1 juta maka akan lama pencarian tersebut karena database akan scan table secara urut, Lalu kita coba membuat index pada table user dengan query : CREATE INDEX user\_ktp\_idx ON user(ktp), kemudian GO dan index sukses terbuat, lalu kita coba melalkukan query yang sama sebelum membuat index : SELECT \* FROM `user` WHERE ktp=’123778124’, dan hasilnya pasti akan lebih cepat dari pada sebelum memakai index.