|  |  |  |
| --- | --- | --- |
| C:\Users\Admin\Pictures\JATA KPM.png  **BAHAGIAN PENDIDIKAN TEKNIK DAN VOKASIONAL**  **KEMENTERIAN PENDIDIKAN MALAYSIA**  **ARAS 5 & 6, BLOK E14, KOMPLEKS E,**  **PUSAT PENTADBIRAN KERAJAAN #PERSEKUTUAN**  **KERTAS PENERANGAN**  ***(INFORMATION SHEET)*** | | |
| **KOD DAN NAMA PROGRAM** | IT-010-3:2016 APPLICATION DEVELOPMENT | |
| **TAHAP DAN SEMESTER** | 3 (SEMESTER 1) | |
| **KOD DAN TAJUK KURSUS** | KPD1043 PRINCIPLES AND PRACTICE IN DATABASE | |
| **NO.DAN TAJUK KOMPETENSI** | K1 INTRODUCTION TO LATEST DATABASE PROGRAMMING  **K2 STRUCTURE DATA MODELLING AND DATABASE DESIGN**  K3 CREATE AND MANAGE TABLE USING DATA DEFINITION LANGUAGE  K4 COMMIT PROTOTYPE SOURCE CODE | |
| **NO. KOD KSKV** | KPD 1043 / KP(2/4) | Muka Surat : 1 Drp : 23 |
| **NO. KOD NOSS** | IT-010-3:2016-C01/ P(2/4) |

**TAJUK :**

**2.0 STRUCTURE DATA MODELLING AND DATABASE DESIGN**

**TUJUAN:**

Diakhir sesi pembelajaran, pelajar akan dapat:

1. Struktur pangkalan data bagi Domain, *relation*, *tuples* dan atribut
2. Pengenalan kepada *Primary key* dan *Foreign key*
3. Pengenalan kepada Pernormalan
4. Proses evaluasi dan pembetulan struktur data bagi mengurangkan kelewahan data.
5. Pengenalan kepada *Functional Dependency Diagram*
6. Membentuk jadual normal dengan menggunakan 1NF,2NF dan 3NF

**PENERANGAN/**INFORMATION**:**

**2.1 STRUKTUR DATA DALAM PANGKALAN DATA**

1. **Data Dan Maklumat**

**Data**

Data adalah fakta asas (mentah) yang belum diproses dan bermakna. Data wujud dalam pelbagai bentuk seperti nombor, teks sebagai bit atau bait yang disimpan dalam bentuk memori elektronik.

**Maklumat**

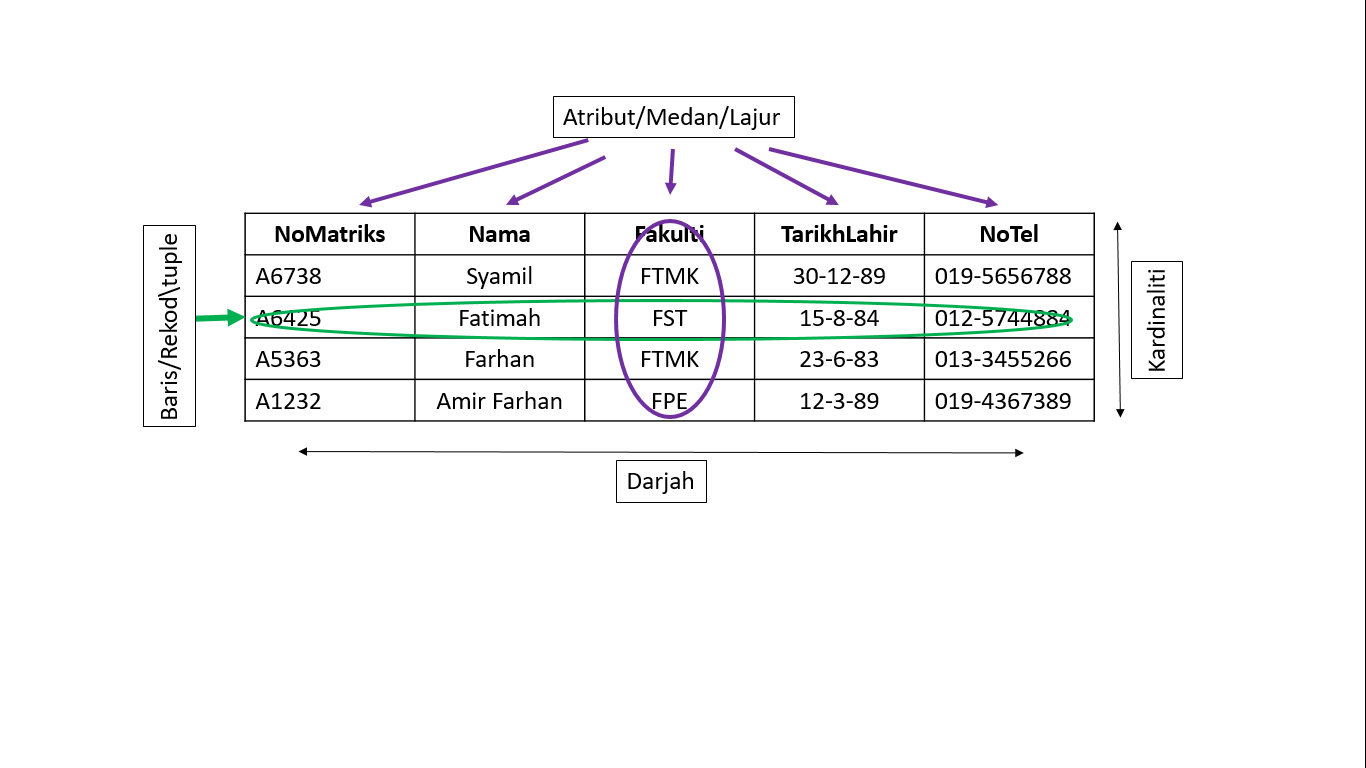
Maklumat atau informasi merupakan hasil daripada pemprosesan, pengumpulan dan penganalisan data yang dapat menambah pengetahuan kepada penerima maklumat. Secara ringkas, maklumat merupakan konteks apabila data digunakan.

Maklumat merupakan data yang diproses

1. **Istilah Terminologi Model Hubungan Pangkalan Data**

|  |  |
| --- | --- |
| **ISTILAH** | **MAKSUD** |
| Domain | Domain merupakan set nilai yang dibenarkan untuk satu atau lebih attribut |
| Hubungan (*relation*) | Jadual yang mempunyai lajur dan baris |
| Tuple | Baris bagi hubungan / jadual |
| Atribut | Maklumat atau ciri-ciri sesuatu entity yang menerangkan mengenai entity tersebut |
| Kardinaliti | Bilangan baris bagi jadual |
| Darjah | Bilangan lajur bagi jadual |

Contoh Hubungan:



**JADUAL 1** HUBUNGAN PELAJAR

Hubungan mempunyai nama yang unik seperti PELAJAR atau HUBUNGAN KURSUS, setiap sel iaitu pertindihan diantara baris dan lajur mengandungi satu nilai sahaja. Setiap attribut mempunyai nama yang unik, nilai attribut mestilah datang dari domain yang sama. Setiap tuple juga adalah unik, oleh itu hubungan mesti mempunyai kunci utama untuk menjamin keunikan setiap tuple, susunan attribute dan susunan tuple tidak datang.

1. **Istilah Alternatif Yang Digunakan Dalam Terminologi Model Hubungan Pangkalan Data**

|  |  |  |
| --- | --- | --- |
| **ISTILAH FORMAL** | **ALTERNATIF 1** | **ALTERNATIF 2** |
| Hubungan | Jadual | Fail |
| Atribut | Lajur | Medan |

|  |  |
| --- | --- |
| **ISTILAH DALAM BAHASA MELAYU** | **ISTILAH DALAM BAHASA INGGERIS** |
| Hubungan | Relation |
| Lajur | Table |
| Jadual | Column |
| Medan | Field |
| Baris | Row |
| Darjah | Degree |

**2.2 PENGENALAN KEPADA PRIMARY KEY DAN FOREIGN KEY**

1. **KUNCI UTAMA *(PRIMARY KEY)* DAN KUNCI ASING *(FOREIGN KEY)***

**2.1 KUNCI UTAMA *(PRIMARY KEY)***

Primary key atau kunci utama adalah suatu aturan yang berguna untuk memastikan bahawa setiap baris data pada jadual **bersifat unik** (berbeza antara baris satu dengan baris lainnya).

**2.2 KUNCI ASING (*FOREIGN KEY)***

   Foreign Key adalah satu set atribut atau set atribut sebagai **key penghubung kedua tabel** dan **melengkapi satu *relationship*** (hubungan) terhadap primary key yang menunjukan induknya. Jika sebuah *primary key* terhubung ke table/entity lain, maka keberadaan primary key pada entity tersebut di sebut sebagai foreign key.

**b) MENGENALPASTI KUNCI UTAMA *(PRIMARY KEY)* DAN (*FOREIGN KEY)***

1. **KUNCI UTAMA *(PRIMARY KEY)***Primary key pada pangkalan data MySQL digunakan pada kolum – kolum sebagai perbezaan,Sebagai contoh , perhatikan pada jadual dibawah ini.

|  |  |
| --- | --- |
| kod\_mtp | nama\_matapelajaran |
| TI10188 | Paket pemrograman 1 |
| TI66779 | Algo & Struktur Data 1 |
| TI45612 | Kalkulus |

**JADUAL 2** *PRIMAY KEY*

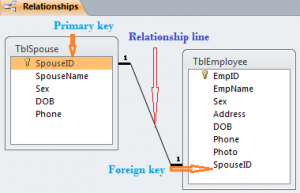
Dari contoh diatas dapat dilihat bahawa setiap data dapat dibezakan berdasarkan kolum kod\_mtp, ini berarti nilai pada kod\_mtk tidak boleh sama dengan yang lain (bersifat unik).

Peraturan dalam primary key adalah :

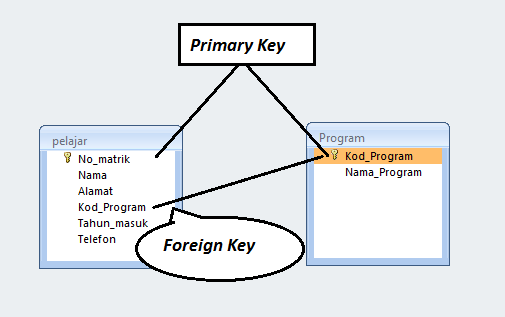
* Suatu jadual atau table hanya dapat **memiliki satu** primary key
* Nilai primary key **besifat unik** dan **tidak boleh NULL** (kosong , tidak diketahui, tidak dapat ditentukan)
* **Unik** dan **tidak boleh ganda**

**2. KUNCI ASING *(FOREIGN KEY)***

Foreign Key adalah kolum atau field pada suatu tabel yang berfungsi sebagai kunci tamu dari tabel lain. Foreign Key sangat berguna bila kita bekerja dengan banyak table atau jadual yang saling berelasi satu sama lain. Contoh dari Foreign key adalah sebagai berikut :



**RAJAH 1** *RELATIONSHIPS BETWEEN PRIMARY KEY AND FOREIGN KEY*

****

**RAJAH 2** *RELATIONSHIPS BETWEEN PRIMARY KEY AND FOREIGN KEY*

**2.3 PENGENALAN KEPADA PERNORMALAN**

1. **MAKSUD PENORMALAN**

* Merupakan satu teknik untuk memastikan dan menilai bahawa model data logikal yang terhasil adalah tepat dan mempunyai anomaly paling minimum
* Merupakan kaedah formal untuk mengenal pasti hubungan berdasarkan kepada kunci utama dan keberkesanan fungsian diantara atribut.

**2. TUJUAN**

* Menghasilkan skema hubungan dengan pengumpulan atribut yang paling optimum.
* Mengurangkan data lewah.
* Mengatasi masalah anomali pengemaskinian.
* Bertujuan untuk menerangkan hubungan diantara atribut-atribut dalam satu hubungan.

**3. LANGKAH UTAMA DALAM PEMETAAN KE MODEL LOGIKAL**

* Penyingkiran hubungan M:N
* Penyingkiran hubungan kompleks
* Penyingkiran hubungan rekursif
* Penyingkiran hubungan yang mempunyai atribut
* Penyingkiran atribut multi nilai
* Penyemakan semula hubungan 1:1
* Penyingkiran hubungan berulang

**4. KEPERLUAN PROSES PENORMALAN**

**(a) Pertindihan Maklumat dan Anomali Penyuntingan**

Masalah yang timbul dari penggunaan jadual yang mempunyai pertindihan

maklumat adalah masalah anomali penyuntingan. Anomali penyuntingan yang

dimaksudkan adalah penyelitan (*add*), pemadaman (*delete*) dan perubahan ke atas

rekod-rekod (*edit*).

**(b) Anomali Penyelitan (*Insert Anomaly*)**

Terdapat dua jenis anomali penyelitan yang utama, diterangkan dengan

menggunakan jadual Staff\_Branch dalam Jadual 3.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Staff\_ No. | SName | SAddress | Position | Salary | Branch\_ No. | BAddress | Tel. No. |
| SL21 | John White | 19 Taylor Street,  London | Manager | 30000 | B5 | 23 Deer Road, London | 0171-886-1212 |
| SG37 | Ann Beech | 81 George Street,  Glasgow | Senior  Assistant | 12000 | B3 | 163 main Street, Glasgow | 01241-339-2178 |
| SG14 | David Ford | 63 Ashby Street,  Glasgow | Deputy | 18000 | B3 | 163 main Street, Glasgow | 01241-339-2178 |
| SA9 | Mary Howe | 2 Elm Place,  Aberdeen | Assistant | 9000 | B7 | 10 Argylt Street, Glasgow | 01224--67111 |
| SG5 | Susan Brand | 5 Gt Western Road,  Glasgow | Manager | 24000 | B3 | 163 main Street, Glasgow | 01241-339-2178 |
| SL41 | Julie Lee | 28 Malvern Street,  Kilburn | Assistant | 9000 | B5 | 23 Deer Road, London | 0171-886-1212 |

Jadual 3

Jadual Staff\_Branch

JADUAL STAFF

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Staf\_No. | SName | SAddress | Position | Salary | Branch\_No. |
| SL21 | John White | 19 Taylor Street,  London | Manager | 30000 | B5 |
| SG37 | Ann Beech | 81 George Street,  Glasgow | Senior  Assistant | 12000 | B3 |
| SG14 | David Ford | 63 Ashby Street,  Glasgow | Deputy | 18000 | B3 |
| SA9 | Mary Howe | 2 Elm Place,  Aberdeen | Assistant | 9000 | B7 |
| SG5 | Susan Brand | 5 Gt Western Road,  Glasgow | Manager | 24000 | B3 |
| SL41 | Julie Lee | 28 Malvern Street,  Kilburn | Assistant | 9000 | B5 |

JADUAL BRANCH

|  |  |  |
| --- | --- | --- |
| Branch\_No | BAddress | Tel. No |
| B5 | 23 Deer Road, London | 0171-886-1212 |
| B7 | 10 Argylt Street, Glasgow | 01224-67111 |
| B3 | 163 main Street, Glasgow | 01241-339-2178 |

Jadual 4 (Jadual\_Staf dan Jadual\_Branch)

* Apabila anda ingin menyelitkan maklumat staf baru ke dalam jadual Staff\_Branch, anda perlu memasukkan maklumat lanjut mengenai cawangan tersebut supaya maklumat cawangan tersebut konsisten dengan rekod lain yang merujuk pada cawangan yang sama. Sebagai contoh, sekiranya ingin memasukkan maklumat tentang staf baru di cawangan B7, anda mesti memasukkan maklumat seperti Branch\_No, BAddress dan Tel\_No untuk cawangan B7 walaupun maklumat tersebut telah dimasukkan untuk beberapa staf di cawangan tersebut. Jadual yang ditunjukkan dalam Jadual 4 tidak mempunyai masalah data yang tidak konsisten kerana bagi setiap staf untuk merujuk kepada cawangan cuma dimasukkan Branch\_No dalam jadual Staff dan maklumat lanjut tentang setiap cawangan cuma dimasukkan dalam satu baris iaitu di dalam jadual Branch.
* Untuk memasukkan maklumat lanjut tentang cawangan yang belum mempunyai staf ke dalam jadual Staff\_Branch akan menimbulkan masalah. Bila kita cuba meninggalkan maklumat tentang Staff\_No tidak dibolehkan kerana menyalahi integriti entiti di mana nilai bagi kekunci primer tidak boleh ditinggalkan (*null*). Rekabentuk jadual pada Jadual 4 boleh mengelak masalah ini kerana maklumat mengenai cawangan dan staf diletakkan dalam jadual berasingan.

**(c) Anomali Pemadaman (*Delete Anomaly*)**

Jika kita ingin memadamkan mana-mana rekod staf dalam jadual Staff\_Branch, maklumat cawangan juga terpadam. Sebagai contoh, jika rekod bagi staf SA9 (Mary Howe) dihapuskan, maklumat berkaitan dengan cawangan B7 juga terpadam. Masalah akan timbul kerana Cuma rekod tersebut yang mempunyai maklumat mengenai cawangan B7. Rekabentuk pada Jadual 4 dapat mengelakkan maklumat mengenai cawangan terpadamkerana maklumat mengenai cawangan diletakkan berasingan dari maklumat mengenai staf. Walaupun rekod bagi staf SA9 dihapuskan tetapi maklumat lanjut mengenai B7 masih kekal dalam jadual Branch.

**(d) Anomali Perubahan (*Update Anomaly*)**

Dalam jadual Staff\_Branch jika nilai dalam mana-mana atribut bagi cawangantertentu hendak diubah, contohnya nombor telefon bagi cawangan atribut No\_Tel bagi semua rekod staf yang bertugas di cawangan tersebut mesti diubah supaya pangkalan data adalah konsisten.

Secara keseluruhannya, rekabentuk jadual-jadual pada Jadual lebih berkesan berbanding dengan Jadual kerana ia dapat mengurangkan pertindihan data dan mengelakkan masalah yang timbul semasa anomali penyuntingan.

**2.4 MENGHURAIKAN PROSES UNTUK PENILAIAN DAN PEMBETULAN STRUKTUR *TABLE* UNTUK MEMINIMAKAN PROSES PERTINDIHAN DATA**

* **Penyingkiran hubungan M:N**

Pecahkan kepada 2 hubungan 1:M.

Cara mudah dengan menterjemahkan hubungan kepada entiti komposit.

* **Penyingkiran hubungan kompleks**

Ubahsuai hubungan kompleks menjadi 2 hubungan yang mudah iaitu hubungan binari.

Caranya dengan mewujudkan satu hubungan baru yang mungkin bagi hubungan sedia ada.

* **Penyingkiran hubungan rekursif**

Wujudkan kedua-dua entiti.

Sekiranya hubungan M:N, maka perlu penstrukturan semula mengikut langkah (1)

* **Penyingkiran hubungan yang mempunyai atribut**

Hubungan tersebut diterbitkan menjadi satu entiti komposit.

Atribut bagi hubungan akan menjadi atribut bagi entiti komposit tersebut.

* **Penyingkiran atribut multi nilai**

Huraikan atribut berbilang tersebut menjadi satu entiti baru.

Hubungan baru akan wujud – hubungan 1: M

* **Penyemakan semula hubungan 1:1**

Sekiranya kedua-dua entiti mewakili data yang sama, maka salah satu entiti boleh dihapuskan.

Atribut bagi entiti tersebut diserapkan ke entiti utama

* **Penyingkiran hubungan berulang**
  1. **PENGENALAN KEPADA *FUNCTIONAL DEPENDENCY DIAGRAM***

**1.** **KESANDARAN FUNGSIAN**

Menerangkan perhubungan diantara atribut-atribut di dalam sesebuah jadual.

**2. RAJAH DAN SENARAI KESANDARAN FUNGSIAN**

Contoh : Jika A dan B merupakan antara atribut-atribut di dalam jadual R, B dikatakan kesandaran fungsian ke atas A (ditandakan A → B) jika setiap nilai A di dalam jadual B dikaitkan dengan hanya satu nilai B di dalam R. Dengan kata lain, jika kita mengetahui nilai A, maka hanya satu nilai B di dalam jadual R yang dikaitkan dengan nilai A dalam satu-satu masa. Jadi jika terdapat dua baris yang mempunyai nilai yang sama bagi A, mereka juga mempunyai nilai yang sama bagi B. Tetapi bagi satu nilai B yang sama mungkin mempunyai nilai yang berbeza bagi A.

Kesandaran antara atribut A dan B diwakili oleh gambaran berikut:

A B

B adalah bersandar fungsian ke atas A

Rajah 3

**Penentu (*determinant*)**

Atribut atau sekumpulan atribut di sebelah kiri dari anak panah adalah penentu.

Contoh :

Berdasarkan jadual di dalam Jadual 4 **(Rujuk ms.8)**, untuk Staff\_No tertentu seperti SL21, kita boleh menentukan jawatan staf tersebut iaitu Manager. Jadi atribut Jawatan (*Position*) adalah bersandar secara fungsian ke atas Staff\_No sepertimana ditunjukkan di dalam Gambarajah 4(a). Sebaliknya gambaran Gambarajah 4(b) adalah tidak benar kerana Staff\_No tidak bersandar fungsian ke atas Jawatan. Setiap staf memegang satu jawatan tetapi beberapa staf mungkin memegang jawatan yang sama.

Position

Staff\_No

*Position* bersandar fungsian ke atas *Staff\_No*

Staf bernombor SL21 Manager

Rajah 4(a)

Position

Staff\_No

*Staff\_No* tidak bersandar fungsian ke atas *Position*

Staf bernombor SL21

Manager

Staf bernombor SG5

Rajah 4(b)

(a) Jawatan adalah kesandaran fungsian ke atas Staff\_No (Staff\_No → Position)

(b) Staff\_No tidak kesandaran fungsian ke atas jawatan (Position → Staff\_No)

Perhubungan (*relationship*) di antara Staff\_No dan Position adalah 1:1 di mana

setiap staf cuma memegang satu jawatan sahaja. Manakala perhubungan di antara

Position dan Staff\_No adalah 1:M di mana lebih dari satu staf dikaitkan dengan satu

jawatan. Staff\_No adalah penentu bagi kesandaran fungsian.

Berikut disenaraikan kesandaran fungsian bagi jadual Staf\_Branch :

Staff\_No → SName

Staff\_No → Saddress

Staff\_No → Position

Staff\_No → Salary

Staff\_No → Branch\_No

Staff\_No → BAddress

Staff\_No → Tel\_No

Branch\_No → BAddress

Branch\_No → Tel\_No

BAddress → Branch\_No

Tel\_No → Branch\_No

**3. MENGELUARKAN DATA FUNGSI DENGAN MENGGUNAKAN DATA CONTOH**

Terdapat 11 kesandaran fungsian di mana Staff\_No, Branch\_No, BAddress dan

Tel\_No sebagai penentu. Kesandaran fungsian juga boleh dipaparkan dalam bentuk

berikut :

Staff\_No → Sname, Saddress, Position, Salary, Branch\_No, BAddress,Tel\_No

Branch\_No → BAddress, Tel\_No

BAddress → Branch\_No

Tel\_No →Branch\_No

Bagi kekunci primer, semua atribut akan bersandar fungsian ke atasnya. Untuk mengenalpasti kekunci calon bagi jadual Staff\_Branch, mesti ketahui atribut (atau sekumpulan atribut untuk memperkenalkan setiap baris dalam jadual tersebut. Jika jadual tersebut mempunyai lebih dari satu kekunci calon, kenalpasti kekunci calon yang berfungsi sebagai kekunci primer. Semua atribut yang bukan merupakan kekunci primer mesti bersandar fungsian kepada kekunci tersebut.

**2.6 MEMBENTUK JADUAL NORMAL DENGAN MENGGUNAKAN 1NF,2NF DAN 3NF**

**1. PROSES PENORMALAN**

Penormalan merupakan satu teknik untuk menganalisa jadual-jadual berdasarkan

kekunci primer dan kesandaran fungsian. Ia melibatkan satu siri peraturan yang

boleh diuji pada jadual tertentu secara individu supaya pangkalan data boleh

dinormalkan pada mana-mana peringkat.

1NF

2NF

3NF

BCNF

4NF

5NF

Higher normal

forms

Rajah 5

Gambarajah perhubungan di antara bentuk-bentuk penormalan

**2. BENTUK TIDAK NORMAL (UNF)**

Jadual yang mempunyai satu atau lebih kumpulan yang berulang. Kumpulan

berulang bermaksud satu sekumpulan atribut dalam jadual yang mempunyai

pelbagai nilai bagi satu kekunci atribut bagi jadual tersebut.

**3. BENTUK NORMAL PERTAMA (1NF)**

Jadual di mana pertembungan antara setiap baris dan lajur mempunyai hanya satu

nilai. Bagi menghasilkan jadual 1NF dari UNF, kenalpasti dan keluarkan kumpulan

berulang daripada jadual UNF. Terdapat **dua** pendekatan digunakan untuk

mengeluarkan kumpulan berulang.

* **Pendekatan pertama :**

Keluarkan kumpulan berulang secara memasukkan data bersesuaian ke

dalam lajur dan baris kosong yang mengandungi data berulang. Dengan kata

lain, memenuhi ruangan kosong dengan data yang tidak berulang.

* **Pendekatan kedua :**

Kita pilih satu atribut atau sekumpulan atribut sebagai kekunci primer untuk

jadual tidak normal (UNF) dan keluarkan kumpulan berulang dengan

memasukkan data berulang bersama-sama kekunci primer ke dalam jadual

berasingan. Kekunci primer dikenalpasti di dalam setiap jadual baharu ini.

UNF boleh mempunyai lebih dari satu kumpulan berulang atau mempunyai

kumpulan berulang di dalam kumpulan yang lain. Proses diulang sehingga

jadual tidak mempunyai kumpulan berulang. Satu set jadual di dalam 1NF

jika tidak mempunyai kumpulan berulang.

Contoh :

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Customer  No | CName | Property  No | PAddress | Rent  Start | Rent  Finish | Rent | Owner  No | OName |
| CR76 | John Kay | PG4  PG16 | 6 Lawrence Street, Glasgow  5 Novar Drive, Glasgow | 1-Jul-93  1-Sep-95 | 31-Augt-95  1-Sep-96 | 350  450 | C040  C093 | Tina Murphy  Tony Shaw |
| CR56 | Aline Stewart | PG4  PG36  PG16 | 6 Lawrence Street, Glasgow  2 Manor Road, Glasgow  5 Novar Drive, Glasgow | 1-Sep-92  10-Oct-93  1-Jan-95 | 10-June-93  1-Dec-94  10-Aug-95 | 350  375  450 | C040  C093  C093 | Tina Murphy  Tony Shaw  Tony Shaw |

Jadual 5

Jadual Customer\_Rental yang tidak normal

Jadual 1 di atas menunjukkan maklumat hartanah yang disewa oleh dua pelanggan bernama John Kay dan Aline Stewart. Bagi menghasilkan 1NF, langkah-langkah yang perlu dibuat :

* Mula-mula kenalpasti kekunci primer bagi jadual di atas iaitu Customer\_No.
* Kemudian kenalpasti kumpulan berulang iaitu maklumat lanjut tentang hartanah yang disewa yang diulang bagi setiap pelanggan.

Struktur bagi kumpulan berulang adalah :

Kumpulan berulang = (Property\_No, PAddress, RentStart, RentFinish, Rent, Owner\_No, OName)

Bagi memindahkan bentuk jadual tidak normal kepada 1NF, pastikan Cuma ada satu nilai bagi setiap pertembungan di antara baris dan lajur. Berdasarkan pendekatan pertama, keluarkan kumpulan berulang (maklumat lanjut tentang hartanah disewa) iaitu dengan memasukkan maklumat pelanggan ke dalam setiap baris supaya setiap baris mempunyai satu nilai sahaja. Hasil jadual dalam bentuk normal pertama ditunjukkan di dalam Jadual 2. Kekunci primer adalah kekunci komposit mengandungi (Customer\_No, Property\_No).

Jadual Customer\_Rental didefinisi seperti berikut :

Customer\_Rental (Customer\_No, Property\_No, Cname, PAddress, RentStart, RentFinish, Rent, Owner\_No, OName)

Jadual Customer\_Rental sekarang berada dalam 1NF kerana mengandungi Cuma satu nilai pada setiap pertembungan baris dan lajur. Tetapi bentuk jadual ini masih mempunyai masalah bagi anomali penyuntingan.

Mengikut pendekatan kedua, keluarkan kumpulan berulang dengan memasukkan data berulang bersama-sama kekunci atribut (Customer\_No) di dalam jadual berasingan sebagaimana ditunjukkan di dalam Jadual 3. Kita perlu kenalpasti kekunci primer bagi setiap jadual.

Hasil format jadual di dalam 1NF adalah :

Customer (Customer\_No, CName)

Prop\_Rental\_Owner (Customer\_No, Property\_No, PAddres, RentStart, RentFinish, Rent,

Owner\_No, OName)

Kedua-dua jadual Customer dan Prop\_Rental\_Owner berada di dalam 1NF kerana mempunyai satu nilai bagi setiap sel. Tetapi jadual-jadual ini masih mempunyai masalah pertindihan data.

JADUAL CUSTOMER\_RENTAL

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Customer  No | CName | Property  No | PAddress | Rent  Start | Rent  Finish | Rent | Owner  No | OName |
| CR76 | John Kay | PG4 | 6 Lawrence Street, Glasgow | 1-Jul-93 | 31-Augt-95 | 350 | C040 | Tina Murphy |
| CR76 | John Kay | PG16 | 5 Novar Drive, Glasgow | 1-Sep-95 | 1-Sep-96 | 450 | C093 | Tony Shaw |
| CR56 | Aline Stewart | PG4 | 6 Lawrence Street, Glasgow | 1-Sep-92 | 10-June-93 | 350 | C040 | Tina Murphy |
| CR56 | Aline Stewart | PG36 | 2 Manor Road, Glasgow | 10-Oct-93 | 1-Dec-94 | 375 | C093 | Tony Shaw |
| CR56 | Aline Stewart | PG16 | 5 Novar Drive, Glasgow | 1-Jan-95 | 10-Aug-95 | 450 | C093 | Tony Shaw |

Jadual 6 - Bentuk Normal Pertama (1NF) bagi jadual Customer\_Rental

(Mengikut pendekatan pertama)

JADUAL CUSTOMER

|  |  |
| --- | --- |
| Customer\_No | CName |
| CR76 | John Kay |
| CR56 | Aline Stewart |

JADUAL PROP\_RENTAL\_OWNER

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Customer  No | Property  No | PAddress | Rent  Start | Rent  Finish | Rent | Owner  No | OName |
| CR76 | PG4 | 6 Lawrence Street, Glasgow | 1-Jul-93 | 31-Augt-95 | 350 | C040 | Tina Murphy |
| CR76 | PG16 | 5 Novar Drive, Glasgow | 1-Sep-95 | 1-Sep-96 | 450 | C093 | Tony Shaw |
| CR56 | PG4 | 6 Lawrence Street, Glasgow | 1-Sep-92 | 10-June-93 | 350 | C040 | Tina MurphY |
| CR56 | PG36 | 2 Manor Road, Glasgow | 10-Oct-93 | 1-Dec-94 | 375 | C093 | Tony Shaw |
| CR56 | PG16 | 5 Novar Drive, Glasgow | 1-Jan-95 | 10-Aug-95 | 450 | C093 | Tony Shaw |

Jadual 7

Bentuk Normal Pertama (1NF) bagi jadual Customer dan Prop\_Rental\_Owner

(Mengikut pendekatan kedua)

**4. BENTUK NORMAL KEDUA (2NF)**

Bentuk Normal Kedua adalah berdasarkan konsep kesandaran fungsian penuh (*full*

*functional* *dependencies*).

* **Kesandaran fungsian penuh**

Jika A dan B merupakan antara atribut-atribut bagi sesebuah jadual, B adalah

bersandar fungsian penuh atas A jika B bersandar fungsian ke atas A tetapi

bukan ke atas mana-mana subset A. Dengan kata lain, satu kesandaran

fungsian A → B adalah kesandaran fungsian penuh jika mana-mana atribut

dari A jika dikeluarkan akan menyebabkan kesandaran tidak wujud lagi.

Satu kesandaran fungsian A → B adalah separa kesandaran jika mana-

mana atribut boleh dikeluarkan dari A dan kesandaran masih lagi wujud.

Contoh :

Staff\_No, SName → Branch\_No

Setiap nilai bagi (Staff\_No, SName) dikaitkan dengan hanya satu nilai bagi

Branch\_No. Walaubagaimanapun ini bukannya kesandaran fungsian penuh

kerana Branch\_No juga bersandar fungsian ke atas subset (Staff\_No,

SName). Dengan kata lain, Branch\_No adalah kesandaran fungsian penuh

hanya ke atas Staff\_No.

* **Definisi Bentuk Normal Kedua**

Digunakan hanya kepada jadual yang **mempunyai kekunci komposit** iaitu

jadual yang mempunyai kekunci primernya terdiri dari dua atau lebih atribut.

Ia merupakan jadual yang terletak dalam 1NF dan setiap atribut bukan

kekunci primer adalah bersandar fungsian penuh ke atas kekunci primer.

Penormalan 1NF kepada 2NF melibatkan pengeluaran separa kesandaran dengan meletakkan atribut kesandaran fungsian ke dalam jadual baru bersama dengan penentunya.

Contoh :

Jadual Customer\_Rental mempunyai kesandaran fungsian seperti berikut :

Customer\_No, Property\_No → RentStart, RentFinish

Customer\_No → CName

Property\_No → Paddress, Rent, Owner\_No, OName

Owner\_No → OName

CName

Kesandaran Transitif

OName

Owner\_No

Rent

PAddress

RentFinish

RentStart

Property\_No

Customer\_No

Kekunci Primer

Rajah 6

Kesandaran fungsian bagi jadual Customer\_Rental

Didapati atribut pelanggan (CName) adalah kesandaran separa ke atas kekunci primer iaitu hanya ke atas atribut Customer\_No. Atribut hartanah (PAddress, Rent, Owner\_No, OName) adalah separa kesandaran ke atas kekunci primer iaitu hanya ke atas atribut Property\_No. Atribut hartanah disewakan (RentStart, RentFinish) adalah kesandaran penuh ke atas kekunci primer iaitu Customer\_No dan Property\_No.

Juga didapati di dalam Gambarajah 1, wujudnya kesandaran transitif ke atas kekunci primer. Walaupun kesandaran transitif boleh mempengaruhi anomali penyuntingan, ia akan dikeluarkan di dalam 3NF.

Kewujudan separa kesandaran di dalam jadual Customer\_Rental menunjukkan jadual tersebut bukan di dalam bentuk normal yang kedua. Bagi memindahkan jadual Customer\_Rental kepada 2NF maka kita perlu mencipta jadual baru supaya atribut bukan kekunci utama dikeluarkan dengan salinan sebahagian kekunci primer di mana ianya berkesandaran penuh. Ini akan menghasilkan tiga jadual berasingan iaitu jadual Customer, Rental dan Property\_Owner sebagaimana ditunjukkan di dalam Jadual 8.

Ketiga-tiga jadual ini berada di dalam bentuk normal kedua kerana setiap atribut bukan kekunci utama adalah kesandaran penuh ke atas kekunci primer bagi setiap jadual.

Bentuk jadual :

Customer (Customer\_No, CName)

Rental ((Customer\_No, PropertyNo, RentStart, RentFinish)

Property\_Owner (Property\_No, PAddres, Rent, Owner\_No, OName)

JADUAL CUSTOMER

|  |  |
| --- | --- |
| Customer\_No | CName |
| CR76 | John Kay |
| CR56 | Aline Stewart |

JADUAL RENTAL

|  |  |  |  |
| --- | --- | --- | --- |
| Customer\_No | Property\_No | RentStart | RentFinish |
| CR76 | PG4 | 1-Jul-93 | 31-Augt-95 |
| CR76 | PG16 | 1-Sep-95 | 1-Sep-96 |
| CR56 | PG4 | 1-Sep-92 | 10-June-93 |
| CR56 | PG36 | 10-Oct-93 | 1-Dec-94 |
| CR56 | PG16 | 1-Jan-95 | 10-Aug-95 |

JADUAL PROPERTY\_OWNER

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property\_No | PAddress | Rent | Owner\_No | OName |
| PG4 | 6 Lawrence Street, Glasgow | 350 | C040 | Tina Murphy |
| PG16 | 5 Novar Drive, Glasgow | 450 | C093 | Tony Shaw |
| PG4 | 6 Lawrence Street, Glasgow | 350 | C040 | Tina Murphy |
| PG36 | 2 Manor Road, Glasgow | 375 | C093 | Tony Shaw |
| PG16 | 5 Novar Drive, Glasgow | 450 | C093 | Tony Shaw |

Jadual 8

Jadual Bentuk Normal Kedua (2NF) yang diperolehi dari jadual Customer\_Rental

**5. BENTUK NORMAL KETIGA (3NF)**

Walaupun jadual-jadual yang berada di dalam 2NF dapat mengurangkan pertindihan data berbanding dengan jadual di dalam 1NF tetapi ia masih mempunyai masalah semasa anomali penyuntingan. Sebagai contoh, jika kita hendak mengubah nama pemilik seperti Tony Shaw (Owner\_No C093), kita terpaksa menyunting kedua-dua baris dalam jadual Property\_Owner seperti di dalam Jadual 8 supaya data adalah konsisten. Masalah anomali penyuntingan ini timbul disebabkan oleh kesandaran transitif (*transitive dependency*) yang boleh dikeluarkan semasa di dalam bentuk normal ketiga (3NF).

* **Kesandaran Transitif**

Satu keadaan di mana A, B dan C merupakan antara atribut-atribut di dalam

satu jadual di mana jika A → B dan B → C, maka C merupakan kesandaran

transitif ke atas A melalui B (dengan syarat A tidak kesandaran fungsian ke

atas B atau C).

Contoh :

Staff\_No → Branch\_No dan Branch\_No → BAddress

Jadi kesandaran transitif Staff\_No → BAddress wujud melalui atribut

Branch\_No dengan syarat Staff\_No tidak kesandaran fungsian ke atas

Branch\_No atau BAddress.

* **Definisi Bentuk Normal Ketiga (3NF)**

Satu jadual yang berada di dalam bentuk normal pertama dan kedua dan di

mana tiada atribut bukan kekunci utama yang kesandaran transitif ke atas

kekunci primer. Jika kesandaran transitif wujud, letakkan atribut kesandaran

transitif dari jadual tersebut ke dalam jadual baru bersama-sama dengan

penentu.

Contoh :

Bagi jadual Customer, Rental dan Property\_Owner, didapati kesandaran fungsian adalah seperti berikut :

Jadual Customer

Customer\_No → CName

Jadual Rental

Customer\_No, Property\_No → RentStart, RentFinish

Jadual Property\_Owner

Property\_No → Paddress, Rent, Owner\_No, OName

Owner\_No → OName

Jadual-jadual Customer dan Rental tidak mempunyai kesandaran transitif dan sudah berada dalam 3NF. Semua atribut bukan kekunci utama dalam jadual Property\_Owner adalah bersandaran fungsian kepada kekunci primer kecuali OName yang juga bersandar kepada Owner\_No. Ini merupakan contoh kesandaran transitif yang terhasil apabila atribut bukan kekunci utama (OName) yang bersandar kepada satu atribut yang juga bukan kekunci primer (Owner\_No).

Untuk menghasilkan jadual Property\_Owner yang berada di dalam 3NF, mesti mengeluarkan kesandaran transitif dengan menghasilkan dua jadual yang berasingan dipanggil Property\_for\_Rent dan Owner sebagaimana ditunjukkan di dalam Jadual 9.

Bentuk jadual yang baru :

Property\_for\_Rent (Property\_No, Paddress, Rent, Owner\_No)

Owner (Owner\_No, OName)

JADUAL PROPERTY FOR RENT

|  |  |  |  |
| --- | --- | --- | --- |
| Property\_No | PAddress | Rent | Owner\_No |
| PG4 | 6 Lawrence Street, Glasgow | 350 | C040 |
| PG16 | 5 Novar Drive, Glasgow | 450 | C093 |
| PG4 | 6 Lawrence Street, Glasgow | 350 | C040 |
| PG36 | 2 Manor Road, Glasgow | 375 | C093 |
| PG16 | 5 Novar Drive, Glasgow | 450 | C093 |

JADUAL OWNER

|  |  |
| --- | --- |
| Owner\_No | OName |
| C040 | Tina Murphy |
| C093 | Tony Shaw |

Jadual 9

Jadual 3NF yang dihasilkan dari jadual Property\_Owner

Jadual Customer\_Rental di dalam Jadual 6 telah dipindahkan kepada empat jadual berasingan dalam bentuk normal ketiga melalui proses penormalan.

Pemecahan jadual Customer\_Rental 1NF kepada jadual dalam 3NF.

1NF

Customer\_Rental

2NF

Property\_Owner

3NF

Owner

Customer

Rental

Property\_for\_Rent

Rajah 7

Pemecahan jadual Customer\_Rental yang di dalam 1NF kepada jadual di dalam 3NF

Hasil bentuk jadual-jadual yang berada di dalam 3NF :

Customer (Customer\_No, CName)

Rental (Customer\_No, Property\_No, RentStart, RentFinish)

Property\_for\_Rent (Property\_No, Paddress, Rent, Owner\_No)

Owner (Owner\_No, OName)

JADUAL CUSTOMER

|  |  |
| --- | --- |
| Customer\_No | CName |
| CR76 | John Kay |
| CR56 | Aline Stewart |

JADUAL RENTAL

|  |  |  |  |
| --- | --- | --- | --- |
| Customer\_No | Property\_No | RentStart | RentFinish |
| CR76 | PG4 | 1-Jul-93 | 31-Augt-95 |
| CR76 | PG16 | 1-Sep-95 | 1-Sep-96 |
| CR56 | PG4 | 1-Sep-92 | 10-June-93 |
| CR56 | PG36 | 10-Oct-93 | 1-Dec-94 |
| CR56 | PG16 | 1-Jan-95 | 10-Aug-95 |

JADUAL PROPERTY\_FOR\_RENT

|  |  |  |  |
| --- | --- | --- | --- |
| Property\_No | PAddress | Rent | Owner\_No |
| PG4 | 6 Lawrence Street, Glasgow | 350 | C040 |
| PG16 | 5 Novar Drive, Glasgow | 450 | C093 |
| PG4 | 6 Lawrence Street, Glasgow | 350 | C040 |
| PG36 | 2 Manor Road, Glasgow | 375 | C093 |
| PG16 | 5 Novar Drive, Glasgow | 450 | C093 |

JADUAL OWNER

|  |  |
| --- | --- |
| Owner\_No | OName |
| C040 | Tina Murphy |
| C093 | Tony Shaw |

Jadual 6

Kesimpulan bagi jadual 3NF yang diperolehi dari jadual Customer\_Rental

Secara keseluruhannya, proses penormalan ini boleh digambarkan seperti turutan berikut :

(1) Hubungan Tidak Normal (UNF)

(2) Singkirkan kumpulan yang berulang dalam jadual.

(3) Bentuk Normal Pertama (1NF)

(4) Singkirkan kebersandaran separa

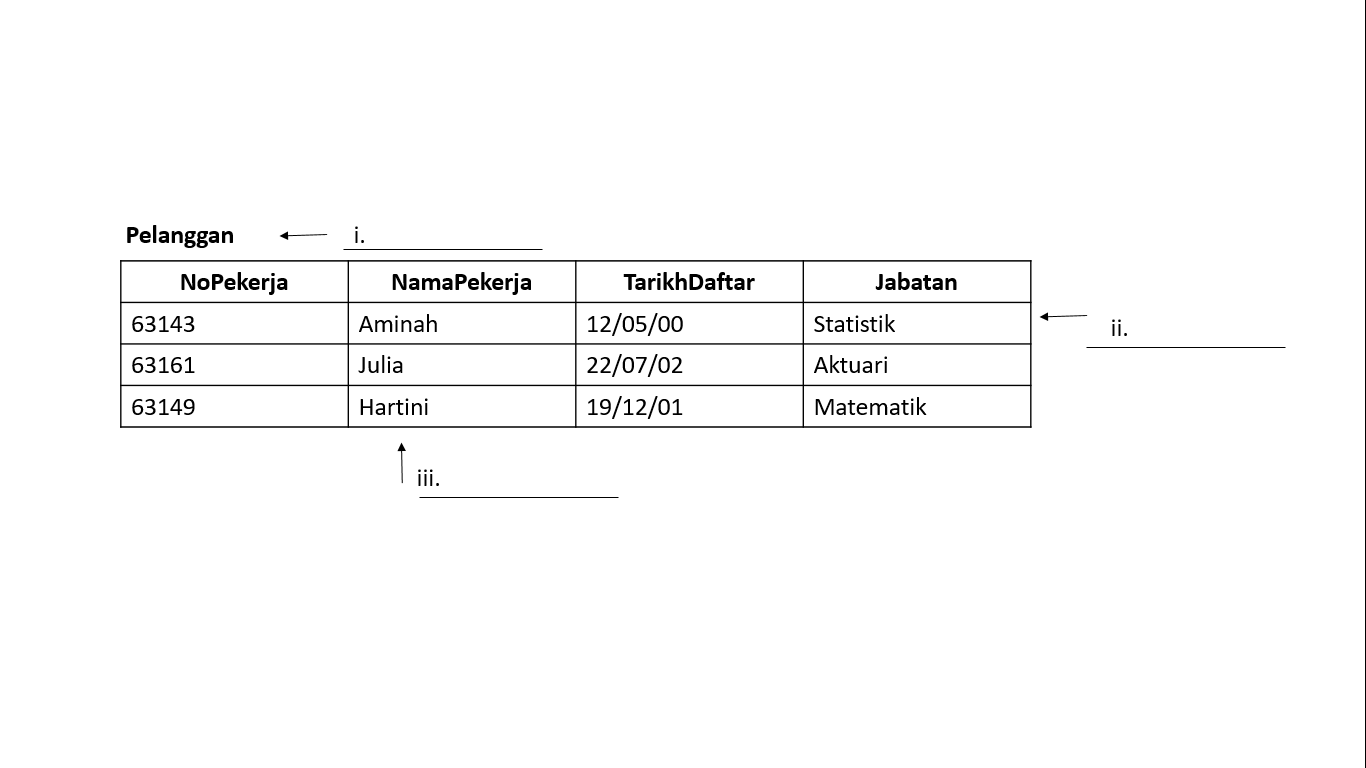
(5) Bentuk Normal Kedua (2NF)

(6) Singkirkan kebersandaran transitif

(7) Bentuk Normal Ketiga (3NF)

**SOALAN/***QUESTION* **:**

1. Berikan maksud bagi Domain, *relation*, *tuples* dan atribut.
2. Isikan tempat-tempat kosong berikut dengan istilah model hubungan yang bersesuaian:



**RUJUKAN/***REFERENCE* **:**

1. Cosmopoint Institut of information technology , 2001, data analysis & database design, Penerbitan cosmopoint
2. Abdullah Embong, 2000, sistem pangkalan data konsep asas, rekabentuk dan pelaksanaan, tradisi ilmu sdb.bhd
3. Modu Lakulu, Aslina Saad, 2007, Sistem Pangkalan Data, Universiti Pendidikan Sultan Idris