|  |  |  |
| --- | --- | --- |
| 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 : 13 |
| **NO. KOD NOSS** | IT-010-3:2016-C01/ P(2/4) |

**TAJUK :**

**1.0 INTRODUCTION TO LATEST DATABASE PROGRAMMING**

**TUJUAN:**

Diakhir sesi pembelajaran, pelajar akan dapat:

1. Sejarah Teknologi Pangkalan Data
2. Pengenalan kepada Pemprogramman Pangkalan Data
3. Konsep *Relational Database Management System*
4. Konsep Entity-Relationship Model
5. Jenis arahan dalam *Query Language*
6. Pengenalan kepada Senibina Pangkalan Data *ANSI-SPARC*

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

**1.1 SEJARAH TEKNOLOGI PANGKALAN DATA**

Definisi:

Pangkalan data atau disebut juga ***database***, terdiri dari dua perkataan iaitu ***data*** dan ***base***. *Database* ditakrifkan sebuah koleksi atau kumpulan data-data yang saling berhubungan (*relation*), disusun menurut aturan tertentu secara logik, sehingga menghasilkan informasi.

Pengenalan:

Pangkalan data secara ringkasnya boleh digambarkan sebagai satu tempat **pengumpulan daan penyimpanan data atau maklumat yang tersusun dalam bentuk elektronik** supaya data atau maklumat ini dengan mudahnya dapat **diakses, diuruskan dan dikemaskini.**

Bagi pengguna berinteraksi dengan data yang telah disimpan ini, pengguna memerlukan **perisian** yang dipanggil **Sistem Pengurusan Pangkalan Data** atau lebih dikenali sebagai ***Database Management System*** (DBMS).

DBMS adalah satu perisian komputer yang **mempunyai antara muka** bagi **menghubungka**n pengguna kepada pangkalan data.

DBMS bertanggungjawab untuk **mengekalkan integriti** dan **keselamatan data** yang **disimpan** dan **memulihkan** maklumat sekiranya **sistem lumpuh**.

Berikut adalah generasi bagi pangkalan data:

|  |
| --- |
| **Generasi Pertama** |
| Kemunculan sistem DBMS dimulakan oleh IBM pada tahun 1968 yang dikenali sebagai *Information Management System* (IMS) yang merupakan perisian pembangunan untuk program Apollo iaitu projek pendaratan kapal angkasa Apollo di bulan dan merupakan perisian berasaskan struktur hierarki. Namun pada pertengahan tahun 1960-an, terdapat banyak sistem pangkalan data berbentuk komersial.  Charles Bachman pula telah menubuhkan kumpulan yang dipanggil *Database Task Group* dalam kumpulan CODASYL iaitu satu kumpulan yang bertanggungjawab mengwujud dan menyeragamkan bahasa pengaturcaraan COBOL. Kemudian pada tahun 1971, kumpulan ini telah mewujudkan satu piawaian yang dipanggil *Codasyl Approach* yang telah dijadikan panduan kepada produk komersial pangkalan data pada waktu itu. |

|  |
| --- |
| **Generasi Kedua** |
| Pembangunan teknologi pangkalan data pada generasi ini telah menjadi satu penanda aras atau panduan kepada perisian-perisian pangkalan data sehingga ke hari ini. Nama Edgar F. Codd menjadi terkenal pada ketika ini kerana telah mencipta model hubungan bagi pengurusan pangkalan data. Beliau melihat kelemahan pada Codasyl Approach telah menyebabkan beliau menulis kertas kerja untuk membentangkan satu pendekatan baru kepada pembinaan pangkalan data.  Selain itu, penghasilan bahasa pertanyaan struktur iaitu SQL juga menjadi perkara penting kepada pembangunan teknologi pangkalan data pada generasi ini. SQL telah dibangunkan oleh IBM pada awal 1970-an dan merupakan antara bahasa yang petama dikomersialkan bagi model pangkalan data hubungan ciptaan Edgar F. Codd |

|  |
| --- |
| **Generasi Terkini** |
| Pembangunan teknologi pangkalan data seterusnya berkembang seiring dengan peningkatan penggunanaan pengaturcaraan berorientasikan objek. Selain itu kemunculan pangkalan data NoSQL dan XML dilihat menjadi pemangkin kepada pembangunan teknologi pangkalan data ini. Pangkalan data XML adalah sejenis pangkalan data berorientasikan struktur dokumen yang membolehkan pertanyaan berdasarkan kepada sifat dokumen XML manakala pangkalan data NoSQL pula menyediakan mekanisma penyimpanan dan mendapatkan semula data yang lebih laju. |

**1.2 SEJARAH PEMPROGRAMAN PANGKALAN DATA**

Bahasa Pertanyaan (Query Languange) adalah bahasa komputer yang digunakan untuk melakukan pertanyaan keapda pangkalan data. Berikut adalah antara bahasa pertanyaan yang biasa digunakan untuk pangkalan data.

**1.3 KONSEP *RELATIONAL DATABASE MANAGEMENT SYSTEM* (RDBMS)**

Secara asasnya, terdapat beberapa jenis model pangkalan data:

1. Model Hierarki
2. Model Rangkaian
3. Model Hubungan

Konsep RDBMS:

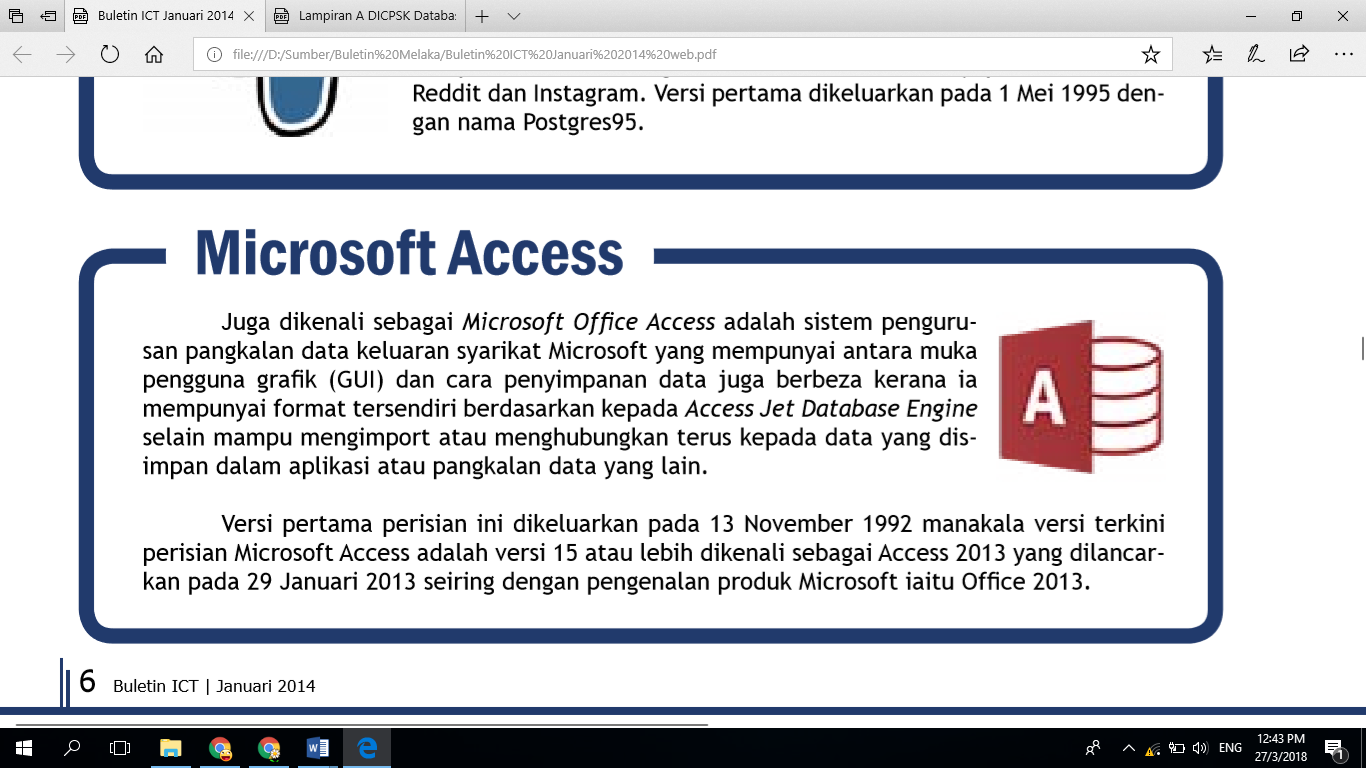
Satu konsep penting berkaitan pangkalan data yang perlu diketahui adalah Sistem Pengurusan Pangkalan Data Hubungan atau lebih dikenali sebagai *Relational Database Management System* (RDBMS). RDMS adalah sistem pengurusan pangkalan data yang berasaskan kepada **model hubungan** yang dicipta dan diperkenalkan oleh Edgar F. Codd. Kebanyakkan perisian pangkalan data pada masa kini adalah berasaskan kepada model pangkalan data hubungan.

Sejak tahun 1980-an, RDBMS menjadi pilihan utama untuk urusan penyimpanan maklumat di dalam pangkalan data bagi kegunaan rekod kewangan, maklumat logistik, data peribadi dan banyak lagi. Pangkalan data hubungan sering kali menggantikan legasi pangkalan data berasaskan hierarki dan pangkalan data berasaskan rangkaian kerana ianya mudah untuk difahami dan digunakan. Walaubagaimanapun, konsep pangkalan data hubungan ini mendapat persaingan daripada konsep pangkalan data berasaskan objek

Sistem Pangkalan Data yang Popular:

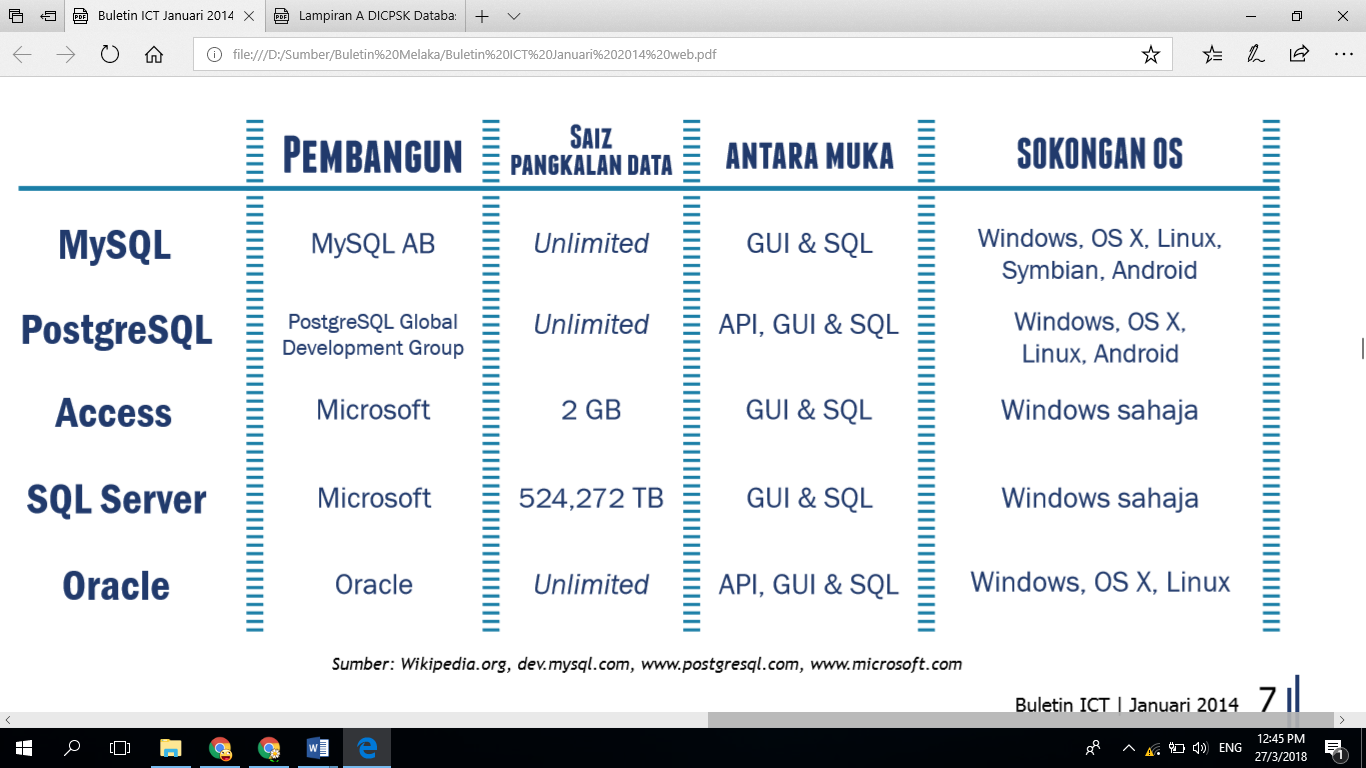












**1.4 KONSEP ENTITI-RELATIONSHIP DIARAGM MODEL (ERD)**

1. **MAKSUD MODEL HUBUNGAN ENTITI**
   * Model perhubungan entiti atau *entity–relationship model* merupakan model data berupa notasi [grafik](https://ms.wikipedia.org/wiki/Grafik) dalam pemodelan [data](https://ms.wikipedia.org/wiki/Data) konseptual yang menggambarkan hubungan antara penyimpan.
   * Model data merupakan sekumpulan cara, peralatan untuk menggambarkan data-data yang hubungannya satu sama lain, semantiknya, serta batasan ketekalan. Model data terdiri dari model hubungan entiti dan model relasional
2. **KEGUNAAN MODEL HUBUNGAN ENTITI**

* Dapat menyelesaikan perwakilan data organisasi yang kompleks
* Perwakilan data yang menggambarkan perjalanan keseluruhan DBMS
* untuk membina model data konseptual, memodelkan struktur data dan hubungan antara data dan perlaksanaan [pangkalan data](https://ms.wikipedia.org/wiki/Pangkalan_data) secara logik mahupun secara fizikal dengan **[DBMS](https://ms.wikipedia.org/wiki/Sistem_pengurusan_pangkalan_data)** (Database Management system).
* Dengan model perhubungan entiti ini kita dapat menguji model dengan mengabaikan proses yang perlu dilakukan.
* Model perhubungan entiti dapat membantu dalam menjawab persoalan tentang data yang diperlukan dan bagaimana data tersebut saling berhubungan.

1. **KOMPONEN *ERD***
   * + Entiti*(Entity)*
     + Atribut *(Attribute)*
     + Hubungan *(Relationship)*
     + Kekangan Kardinaliti(*Cardinality Constrain*)
     + *Primary Key/Foreign key*

**SIMBOL-SIMBOL DALAM GAMBARAJAH HUBUNGAN ENTITI (ERD)**

|  |  |
| --- | --- |
| .**SIMBOL NOTASI** | **KETERANGAN** |
| entiti | Entiti ialah suatu objek atau konsep dalam dunia nyata yang boleh dikenalpasti kewujudannya melalui sifat-sifat tertentu |
| Relation/  hubungan | Hubungan ialah merupakan set yang berada di antara entiti. |
|  | Atribut adalah sifat yang memperihalkan sesuatu entiti |
|  | Garis/line merupakan satu garisan yang menghubungkan *attributes*kepada set entiti masing-masing dan daripada set entiti kepada set hubungan |
|  | Primary key ialah kunci yang mempunyai nilai yang unik |
|  | Foreign key merupakan kekunci utama bagi jadual yang pertama |

1. **ENTITI**

* **Entiti** ialah suatu objek atau konsep dalam dunia nyata yang boleh dikenalpasti kewujudannya melalui sifat-sifat tertentu (unik).
* Dalam konteks pangkalan data, entiti adalah ‘objek’ yang menjadi sumber data

Contoh entiti fizikal – Pelajar, Pensyarah, Buku.

Contoh entiti konsep – Peperiksaan, Pengalaman

* Kumpulan entiti yang sama dipanggil **set entiti**.
* Entiti akan diwakili secara grafik dengan simbol segi empat tepat dengan nama bagi entiti tersebut diletakkan di dalam segi empat tepat tersebut. Disebabkan nama bagi entiti hanya terhad di dalam segi empat itu sahaja maka seboleh-bolehnya ia perlu diringkaskan.

1. **SIMBOL ENTITI**

* Objek segiempat melambangkan entiti

NAMA ENTITI

**Contoh perwakilan entiti dalam ERD**

STAF

PELAJAR



1. **JENIS-JENIS ENTITI**
   * Terdapat dua jenis set entiti
     + Entity kuat
     + Entity lemah
   * Set entiti kuat – kewujudannya tidak bersandar atas kewujudan entiti lain. Cth : Pelajar

Symbol entity kuat

PELAJAR

* + Set entiti lemah – kewujudannya bersandar kepada kewujudan satu set entiti kuat. Cth : Waris

**Symbol entity lemah**

PELAJAR

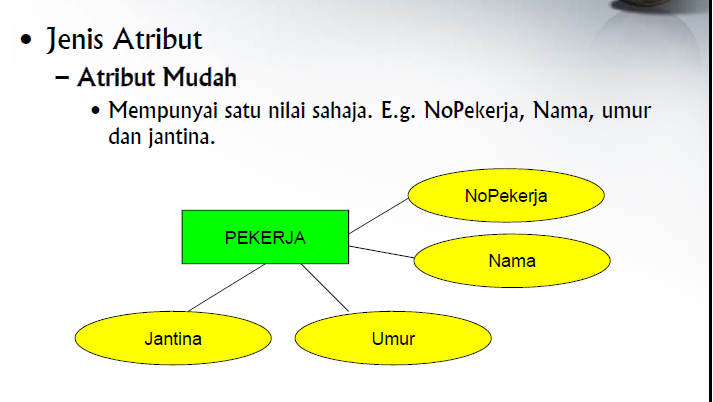
1. **MAKSUD ATRIBUTE**

* Atribut adalah sifat yang memperihalkan sesuatu entiti
* Setiap ahli dalam satu set entiti mempunyai atribut yang sama tapi nilainya adalah unik untuk dirinya sendiri. Cth : Nombor K/Pengenalan
* Set nilai yang dibenarkan bagi sesuatu atribut dipanggil domain. Cth : [(Nama, Halim), NoPel, 1111)] – set entiti Pelajar bernama Halim dengan domain 20 aksara, NoPel dengan domain integer 10 digit.

**JENIS-JENIS ATRIBUT**

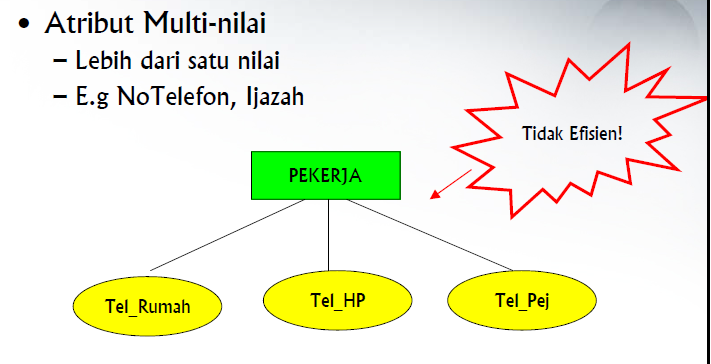
1. **Attribute ringkas/mudah**

* Atribut Ringkas/Mudah- Ialah attribute yang mengandungi komponen tunggal dengan kewujudannya sendiri
* Contoh : setiap orang hanya mempunyai satu nama sahaja



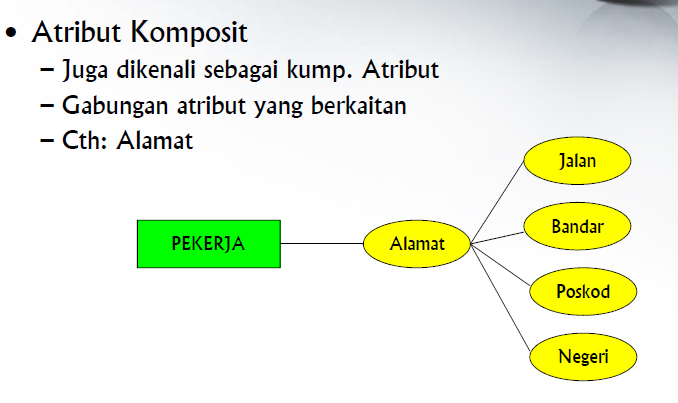
1. **Attribute Multi Nilai**

* Atribut Multi Nilai- Ialah attribute yang mempunyai banyak nilai
* Contoh : seorang pekerja mungkin mempunyai dua nombor telefon iaitu no tel rumah dan no telefon bimbit



1. **Attribute Komposit**

* **Atribut Komposit**- Ialah attribute yang mengandungi banyak komponen yang setiap satunya dengan kewujudan tersendiri
* Contoh : attribute alamat boleh dibahagikan kepada attribute bandar, jalan dan negeri



1. **Attribute terbitan**

* **Atribut terbitan**- Ialah attribute yang terbit daripada nilai attribute yang lain
* Contoh : umur boleh diketahui daripada tarikh lahir, jadi attribute umur adalah atribut



1. **PERHUBUNGAN DALAM PANGKALAN DATA**

Perhubungan menghubungkan pelbagai entity dalam sesuatu gambar rajah ER. Ia diwakili oleh bentuk intan (*diamond*).

Perhubungan biasanya berbentuk kata kerja dan ditulis dalam bentuk intan tersebut. Garisan digunakan sebagai penghubung anta set entiti dan set perhubungan.

Contoh:

DOKTOR

PESAKIT

RAWAT

PELAJAR

BUKU

PINJAM

Rawat dan Pinjam merupakan penghubung bagi entiti yang dihubungkan denganya.

Hubungan yang paling asas bagi membina pangkalan data hubungan adalah terdapat di dalam Model Hubungan Entiti (ERD) dan hubungan yang digunakan adalah lebih kepada hubungan Binari.   
  
**Terdapat 3 jenis hubungan binary** dalam pangkalan data iaitu  
  
**a. Hubungan 1:1 (one to one)**  
  
Dimana dilihat dari segi satu jenis entiti tunggal berhubungan dengan satu entiti tunggal yang lain.



Rajah 1.1 : Contoh Hubungan 1:1

Berdasarkan kepada contoh di atas, terdapat dua entiti yang saling berhubungan iaitu entiti PENSYARAH dan entiti PERJAWATAN. Dalam hal ini **satu** PENYARAH hanya mempunyai **satu** maklumat PERJAWATAN sahaja. Selain itu, di sini dilihat entiti PERJAWATAN merupakan satu entiti yang lemah dan ia amat bergantung kepada entiti PENSYARAH. Ini bermaksud maklumat PERJAWATAN hanya wujud berdasarkan kepada kewujudan PENSYARAH dan ia akan menjadi **satu** maklumat PERJAWATAN hanya untuk **seorang** PENSYARAH sahaja.

**b. Hubungan 1: N (one to many)**



Rajah 1.2 : Contoh Hubungan 1:N

Dalam membangunkan sistem pangkalan data, hubungan yang sering saya temui adalah hubungan satu kepada banyak atau hubungan one to many ( 1:N). Bagi Rajah 1.2, ia akan dilihat sebagai **satu** JABATAN akan mempunyai lebih dari seorang (**ramai)**PENSYARAH manakala **satu** PENSYARAH akan hanya bekerja untuk **satu**JABATAN sahaja.  
  
**c. Hubungan M:N (many to many)**



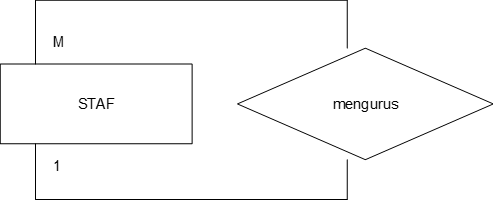
Rajah 1.3 : Contoh Hubungan M:N

**JENIS-JENIS DARJAH HUBUNGAN**

* Darjah Perhubungan merujuk kepada bilangan entiti yang terlibat dalam sesuatu perhubungan.
* Tiga kategori darjah perhubungan iaitu:
  + 1. *Unari (satu entiti terlibat)*
    2. *Binari(dua entiti terlibat)*
    3. *Ternari(tiga entiti terlibat)*

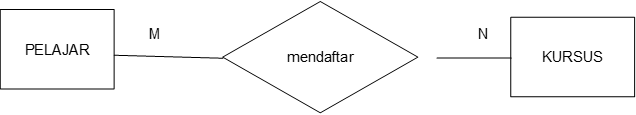
1. **PERHUBUNGAN UNARI**

* Perhubungan jenis ini adakalanya dikenali sebagai perhubungan rekursif, manakala entiti tersebut dikenali sebagai entiti rekursif.
* Contoh di bawah menunjukkan, **satu pengurus** mengurus **ramai subordinat** yang mana pengurus dan subordinat merupakan dari jenis entiti yang sama iaitu STAF



1. **PERHUBUNGAN BINARI**

* Perhubungan jenis ini merupakan perhubungan di antara dua jenis entiti. Perhubungan yang paling biasa terjadi.
* Contoh di bawah menunjukkan perhubungan M:N, satu pelajar boleh mendaftar banyak kursus dan satu kursus boleh diambil oleh lebih daripada satu orang pelajar.

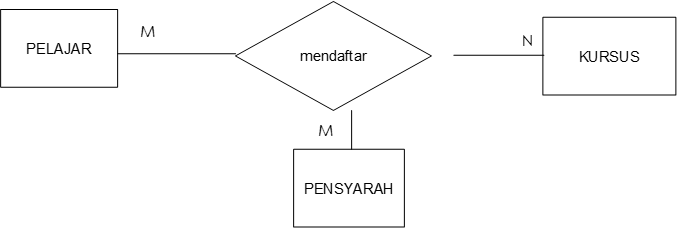
****

1. **PERHUBUNGAN TENARI**

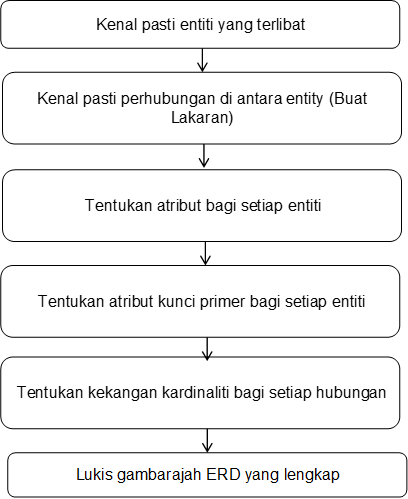
* Perhubungan jenis ini merupakan perhubungan yang melibatkan tiga jenis entiti secara serentak.

Contoh dibawah menunjukkan:

1. Satu pelajar boleh mendaftar banyak kursus dan satu kursus boleh di ambil oleh lebih daripada satu orang pelajar.
2. Satu pelajar boleh mendaftar dengan seorang pensyarah dan seorang pensyarah boleh menerima satu atau lebih pelajar
3. Satu pensyarah boleh mendaftar banyak kursus dan satu kursus boleh di ajar oleh ramai pensyarah.

****

1. **LANGKAH-LANGKAH MEMBINA GAMBARAJAH ER**



1. **CARA MELUKIS GAMBARAJAH ERD**

Situasi : Pelajar meminjam buku yang ditulis oleh pengarang

1. Tentukan entity

* Pelajar
* Buku
* Pengarang

1. Kenalpasti hubungan:

* Pelajar membaca buku
* Buku dituli oleh penulis

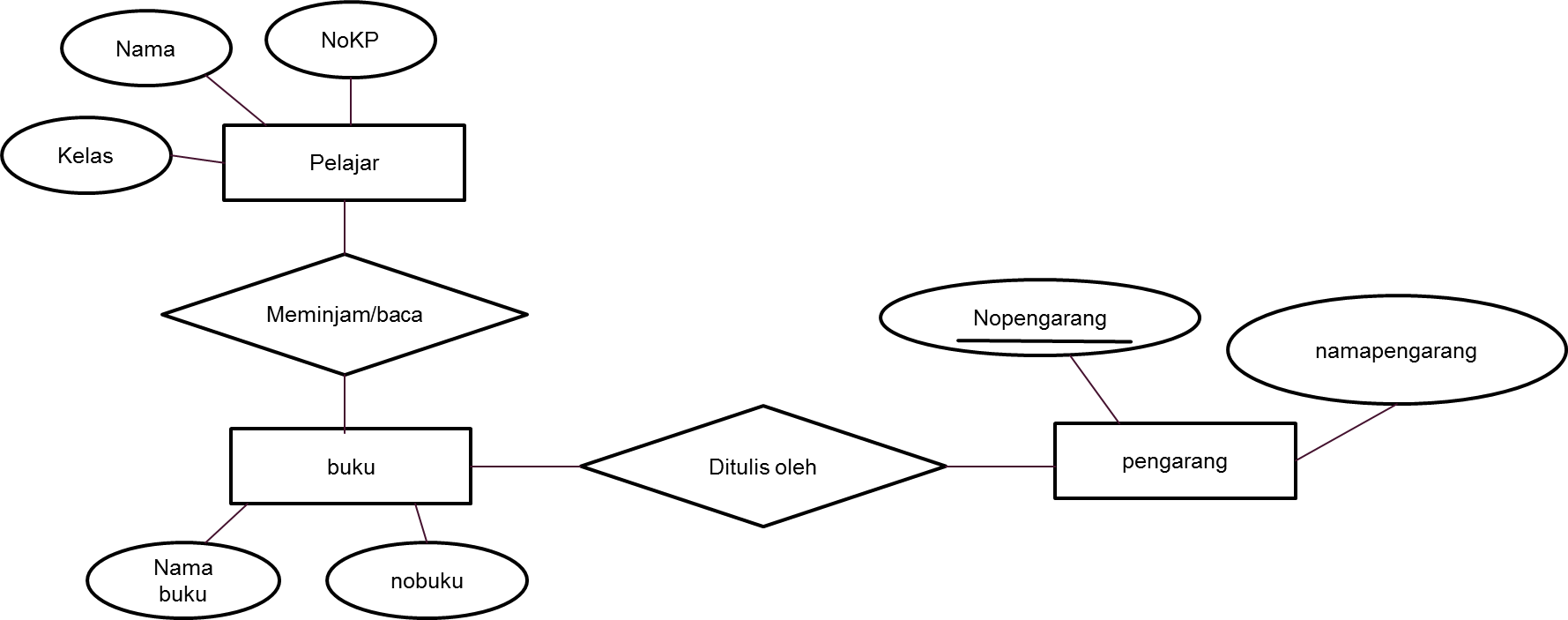
1. Tentukan atribut bagi setiap entiti :

* Pelajar – nama, noKP, kelas
* Buku – namabuku, nobuku,
* Penulis – nama, nopenulis

1. Tentukan kunci primer bagi setiap entiti

* Pelajar – NoKP
* Buku – nobuku
* Pengarang – nopengarang

1. Tentukan kekangan kardinaliti bagi dua entiti seterusnya lukis gambarajah ERD lengkap



1

1

M

**Rajah ERD :Perhubungan Pelajar membaca buku yang ditulis oleh Pengarang**

**1.5 JENIS ARAHAN SQL.**

1.5.1 Data Definition Language (DDL).

- digunakan untuk mentafsirkan pangkalan data / jadual.

|  |  |
| --- | --- |
| **Statement** | **Description** |
| CREATE | Membina pangkalan data / jadual baru. |
| ALTER TABLE  (ADD COLUMN,  MODIFY COLUMN, DROP COLUMN) | Mengubah suai struktur pangkalan / atau jadual |
| DROP | Menghapuskan pangkalan data / jadual. |
| TRUNCATE | Menghapuskan semua rekod. |
| RENAME | Menukar nama pangkalan data / jadual. |
| COMMENT | Menulis komen. |

1.5.2 Data Manipulation Language (DML).

- membuat manipulasi (tambah, ubahsuai, hapus) data dalam jadual.

|  |  |
| --- | --- |
| **Statement** | **Description** |
| INSERT | Menambah data dalam jadual. |
| UPDATE | Mengemaskini. |
| DELETE | Menghapus baris rekod. |

1.5.3 Data Control Language (DCL).

- memberi akses kepada pangkalan data.

|  |  |
| --- | --- |
| **Statement** | **Description** |
| GRANT | Mengakses pangkalan data. |
| REVOKE | Menamatkan akses. |

1.5.4 Data Query Language (DQL).

- memilih data

|  |  |
| --- | --- |
| **Statement** | **Description** |
| SELECT | Memilih data dalam jadual. |

1.5.5 Transaction Control Commands (TCC).

- mengawal data.

|  |  |
| --- | --- |
| **Statement** | **Description** |
| COMMIT | Simpan secara kekal. |
| ROLLBACK | Memanggil semula data selepas COMMIT terakhir. |
| SAVEPOINT | Membuat point untuk ROLLBACK. |
| SET TRANSACTION | Memulakan transaksi. |

**1.6 PENGENALAN KEPADA SENIBINA PANGKALAN DATA – (ANSI-SPARC)**

Seni bina pangkalan data terdiri daripada 3 paras:

1. Paras luaran
2. Paras Konseptual
3. Paras Dalaman

**PANDANGAN**

**PENGGUNA**

**PANDANGAN**

**PENGGUNA**

**SKEMA KONSEPTUAL**

**SKEMA FIZIKAL**

**PARAS LUARAN**

**PARAS KONSEPTUAL**

**PARAS DALAMAN**

Seni Bina tiga paras ANSI/SPARC

Contoh rajah di sebuah organisasi:

**Sistem Permohonan Cuti**

**Sistem Gaji**

**DBMS**

**Storan Komputer**

**PARAS LUARAN**

**PARAS KONSEPTUAL**

**PARAS DALAMAN**

**Pengurusan Fail**

**Sistem Operasi**

Senario Paras Luaran:

* Mengunakan antara muka bagi melalukan aktiviti antara pengguna dengan pangkalan data.
* Had Capaian juga berbeza antara pengguna. Oleh itu, lain pengguna menggunakan antara muka yang mungkin berbeza
* Contoh: Sistem Permohonan Cuti di sesebuah syarikat terdapat tiga tahap

1. Seorang pengurus boleh mencapai keseluruhan maklumat permohonan dan pengesahan cuti staf di jabatannya
2. Ketua jabatan boleh mencapai maklumat permohonan cuti staf di jabatan bagi menyokong permohonan cuti atau sebaliknya
3. Seorang staf boleh mencapai dan menyemak status cuti yang dimohon oleh nya sahaja.

Paras Konseptual :

Menerangkan struktur data yang disimpan dan hubungan di antara data-data tersebut.

Paras ini mewakili elemen:

1. Model data hubungan, hierarki, rangkaian atau berorentasikan objek
2. Kekangan ke atas data
3. Maklumat semantik tentang data
4. Maklumat keselamatan dan integriti

Paras Dalaman:

Menerangkan bagaimana data disimpan dalam pangkalan data. Mengandungi implementasi fizikal pangkalan data untuk mencapai prestasi dan penggunaan ruang storan yang optimm,

Paras ini memberi penekanan kepada:

1. Peruntukan ruang storan untuk data dan indeks
2. Penerangan rekod untuk storan (dengan saiz untuk data item)
3. Tempat atau kedudukan rekod
4. Ketumpatan data atau muatan data dan teknik enkripsi data

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

1. Apakah rasionalnya kewujudan pangkalan data
2. Berikan lima contoh aplikasi pangkalan data digunakan
3. Lukiskan gambar rajah konsep pangkalan data
4. Bezakan ketiga-tiga paras dalam seni bina tiga paras ANSI-SPARC menggunakan gambar rajah
5. Nyatakan ciri-ciri Entiti Relationship Diagram (ERD)

**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
4. Buletin ICT Januari 2014 , Bahagian K-Ekonomi, Jabatan Ketua Menteri Melaka