

हेल्लो दोस्तों,

मेरा नाम युगल जोशी है सबसे पहले तो आप सब का इस बुक को खरीदने के लिए धन्यवाद .

यह ebook मैंने hindi भाषी छात्रों के लिए बनाई है. इस बुक में आपको महत्वपूर्ण सवालों के जवाब मिलेंगे जो अक्सर exam में पूछे जाते हैं. आशा करता हूँ कि यह बुक आपके लिए लाभकारी होगी.

अगर आपको इस बुक के लिए कोई सुझाव है तो हमें बताएं.

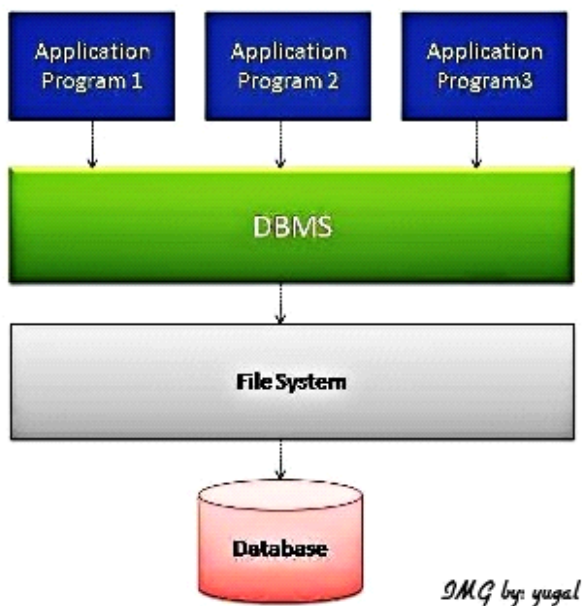
मेरा फ़ोन no.639597626 है आप मुझे whatsapp या call कर सकते हैं. या आप ehindistudy.com में comment करके भी बता सकते हैं. धन्यवाद.

copyright all rights reserved. ehindistudy.com

इस पुस्तक का कोई भी अंश लेखक की लिखित अनुमति के बिना किसी भी रूप में प्रकाशित नहीं किया जा सकता है.

लेखक :- युगल जोशी.

प्रश्न:- DBMS क्या है?



DBMS का पूरा नाम डेटाबेस मैनेजमेंट सिस्टम है। DBMS, प्रोग्राम्स का पूरा कलेक्शन होता है जो कि users को, डेटाबेस को create और maintain करने के लिए योग्य बनाता है।

तो हम कह सकते हैं कि Dbms एक general purpose software

system है जो कि हमें निम्नलिखित सुविधाएँ उपलब्ध कराता है:

1: **DEFINING**– डेटाबेस में स्टोर data के लिए data types, structures और constraints को specify करता है।

2: **CONSTRUCTING**– डेटा को किसी स्टोर मीडियम में स्टोर करने की प्रक्रिया को DBMS के द्वारा नियंत्रित किया जाता है।

3: **MANIPULATING**– इसमें डेटाबेस में उपस्थित डेटा को retrieve तथा update किया जाता है और रिपोर्ट्स को generate किया जाता है।

प्रश्न2:- dbms के लाभ तथा हानियाँ क्या हैं?

dbms के लाभ:- इसके लाभ निम्नलिखित हैं।

: **1.No data redundancy and inconsistency:** एक ही तरह के data का बहुत सारी जगह नकल (duplication) को हम Data redundancy कहते हैं। और एक तरह के data का बहुत जगह होना data inconsistency का कारण बनती है। जिससे storage और मूल्य (cost) बढ़ता है। लेकिन dbms में हमें इससे छुटकारा मिलता है।

2. Restricting Unauthorized Access: DBMS में DBA (database administrator) security और unauthorized subsystem का प्रयोग account restrictions को specify करने के लिए करता है।

3. Data Integrity and Security: DBMS में सुरक्षा और integrity (अखण्डता) का पूरा ध्यान रखा जाता है। डेटाबेस में किसी भी प्रकार की value को insert करने से पहले उसे कुछ conditions को satisfy करना आवश्यक होता है।

डेटाबेस में यूजर को सभी डेटा को access करने की अनुमति नहीं होती है जिससे डेटा integrity बढ़ती है।

4.Simple Access: DBMS में डेटाबेस को आसानी से access किया जा सकता है। आसानी से access करने के लिए application programming interface(API) का प्रयोग किया जाता है।

disadvantage of DBMS (dbms की हानियाँ):-

DBMS के disadvantage निम्नलिखित हैं:-

- 1:-DBMS सॉफ्टवेर तथा हार्डवेयर का मूल्य बहुत ही अधिक होता है.
- 2:-DBMS की functionality बहुत ही जटिल होती है.
- 3:-इसका आकार बहुत अधिक होता है.
- 4:-इसके सभी components को लगातार अपडेट करना पड़ता है जो कि बहुत ही कठिन होता है.
- 5:-यदि electric failure या डेटाबेस corrupt हो जायें तो जो डेटाबेस में महत्वपूर्ण डेटा स्टोर होता है वह lost(समाप्त) हो सकता है.

ehindistudy.com

प्रश्न3:- डेटाबेस क्या है?

डेटाबेस(database) एक collection of information(सूचना का समूह) होता है जिसे इस प्रकार organise(व्यवस्थित) किया जाता है कि

जिसमें information(सूचना) आसानी से access(प्रवेश),
manage(प्रबन्ध), और update की जा सकें।
Database में हम quickly(तेज़ी से) और आसानी से desired data
को select कर सकते हैं।

पारंपरिक Database तीन चीज़ों द्वारा organised किया जाता है।

1. fields
2. records
3. files

field एक single piece of information होता है। record एक
complete set of fields होता है और file जो होती है वह
collection of record होती है।

for Example: dictionary एक डेटाबेस के अनुरूप ही तो है। जो एक
फ़ाइल की तरह होती है, जिसमें रिकॉर्ड की सूची में शामिल fields होते हैं।

प्रश्न4:- डेटा independence को समझाइये?

data independence अगले लेवल में स्कीमा definition को बिना
प्रभावित किये, एक लेवल में स्कीमा definition को modify करने की
ability को data independence कहते हैं।

data independence दो प्रकार का होता है।

1. Logical data independence
2. Physical data independence

logical data independence

बिना external schema(view level) को बदले logical
schema(conceptual level) को बदलना logical data
independence कहलाती है।

उदाहरण के लिए- एक्सटर्नल स्कीमा में बिना बदलाव किये,
conceptual schema के लिए नयी entities, attributes को
जोड़ना या मिटाना संभव है।

physical data independence

logical schema में बिना बदलाव किये physical स्कीमा में बदलाव
करना physical data independence कहलाता है।

प्रश्न5:- Dbms की विशेषतायें लिखिए?

characteristics of DBMS

:Dbms की कुछ विशेषतायें निम्नलिखित हैं।

1. DBMS की सबसे बड़ी विशेषता ये है कि इसमें Data redundancy को control किया जा सकता है।
2. DBMS में डेटा को share कर सकते हैं।
3. DBMS में सुरक्षा(security) का पूरा ख्याल रखा जाता है।
4. DBMS में प्रोसेसिंग की गति अच्छी है।
5. DBMS की एक और विशेषता ये है कि इसमें डेटा independent होता है।

प्रश्न6:- DBMS users का वर्गीकरण कीजिए?

classification of DBMS users:

DBMS में बहुत से users होते हैं जिनका अपना एक particular काम होता है जैसे कुछ यूज़र्स डेटाबेस को manage, और construct करते

हैं, कुछ जो होते हैं वो डेटाबेस को define करते हैं तो हम डेटाबेस यूज़र्स को निम्नलिखित भागों में बाँट सकते हैं।

1. Database administrator

DBA वह व्यक्ति या समुह होता है जो डेटाबेस में किसी भी प्रकार का बदलाव करते हैं जैसे कि update, delete, और create। ये वह यूज़र्स होते हैं जो डेटाबेस से बहुत ज्यादा familiar होते हैं। इनका डेटाबेस पर पूर्णतया नियंत्रण होता है। DATABASE में किसी भी प्रकार के action के लिए DBA जिम्मेदार होता है।

2. Database designers

: डेटाबेस designers डेटाबेस में स्टोर डेटा को identify करते हैं और इस डेटा को represent करने के लिए सही structure का चुनाव करते हैं। डेटाबेस designers को डेटाबेस users की आवश्यकताओं को जानने के लिए उनके साथ communicate करना पड़ता है।

3. End users

: end users वे यूज़र्स होते हैं जो डेटाबेस को access करते हैं तथा end users के आधार पर ही हम डेटाबेस में कोई update तथा report तैयार करते हैं। end users को डेटाबेस की designing, access mechanism तथा working के बारे में पता नहीं होता है वो सिर्फ सिस्टम का प्रयोग task को पूरा करने में करते हैं।

4. Application programmers

: ये वे यूज़र्स होते हैं जो डेटाबेस के लिए एप्लीकेशन programs को लिखते हैं। ये प्रोग्राम सामान्यतया c, cobol, Fortran तथा अन्य जनरल पर्पस प्रोग्रामिंग लैंग्वेज में लिखे जाते हैं और ये अप्लिकेशन

प्रोग्राम्स भिन्न भिन्न प्रकार के कार्य को करने के लिए प्रयोग किये जाते हैं।

प्रश्न 7- डेटाबेस schema (स्कीमा) क्या है?

schema in hindi:- स्कीमा, database के लिए एक organisation या structure है जो कि पूरे डेटाबेस के logical view को represent करता है। यह दर्शाता है कि डेटाबेस में data किस प्रकार organise है तथा उनके मध्य relationship कैसी है। database designers, स्कीमा को design करते हैं जिससे की प्रोग्रामर आसानी से डेटाबेस को समझ सकें और डेटाबेस को प्रयोग करने योग्य बनायें।

सामान्यतया schema, डेटाबेस डिक्शनरी में स्टोर रहती हैं।

database schema का मुख्य purpose प्रत्येक टेबल के विभिन्न tables और fields को identify करना है और schema यह भी describe करता है कि tables के मध्य relationship कैसी है। यह सिस्टम में constraints को identify करने में मदद करता है।

subschema, स्कीमा की subset होती है और subschema भी स्कीमा की तरह समान property, inherit करती है।

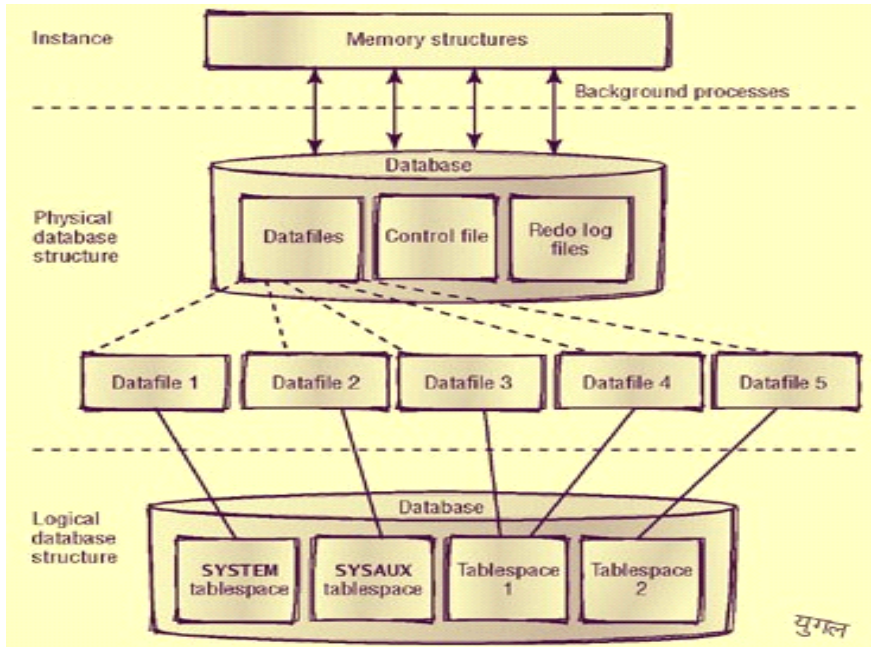
schema को दो भागों में विभाजित किया जा सकता है।

1. Physical schema
2. Logical schema

1. **Physical schema**— physical schema यह परिभाषित करता है कि डेटा को dbms में किस प्रकार represent किया जाता है तथा डेटा को किस प्रकार स्टोर किया जाता है?

2. **Logical schema**— यह स्कीमा ऐसे सभी logical

constraints को परिभाषित करती हैं जो कि डेटा को स्टोर करने में लागू किये जाते हैं। एक लॉजिकल स्कीमा डेटा का conceptual model होती है।



प्रश्न:- instances किसे कहते हैं?

instances:- किसी विशेष समय पर डेटाबेस में डेटा के समुह(collection) का स्टोर होना instance कहलाता है। instance term का प्रयोग सम्पूर्ण database environment को describe करने के लिए किया जाता है जिसमें कि DBMS software, tables तथा अन्य functionality सम्मिलित होती है।

हम इसे इस प्रकार भी कहते हैं कि ऐसी जगह जहाँ डेटा को manageable तरीके से संग्रहित किया जाता है। डेटाबेस में instances स्थिति के अनुसार बदलते रहते हैं।

प्रश्न:- dbms में interface (इंटरफ़ेस) क्या होते हैं?

interfaces:-

database में बहुत सारे interfaces होते हैं जो निम्न हैं।

1. **form-based interface:-** इस प्रकार के interface डेटाबेस में form पर आधारित होते हैं। डेटाबेस में entry के लिए यूजर को form में उपस्थित fields को भरना पड़ता है। form based interface का प्रयोग बहुत किया जाता है और यह interface, DBMS के साथ interact करने के लिए बहुत उपयोगी भी है। इस प्रकार के interface को आसानी से यूज किया जा सकता है तथा इसका लाभ ये है कि इसके लिए डेटाबेस language (SQL) की knowledge की आवश्यकता नहीं होती है।

2. **Menu-based user interface:-** यह interface, menus पर आधारित होता है। menu-based interface यूजर को options की एक लिस्ट उपलब्ध कराता है जिसमें से यूजर को उन options में से एक choose करना पड़ता है। यह एक easy interface है।

3. **Natural language interface:-** DBMS में user अपनी आम बोल चाल की भाषा का प्रयोग कर सकता है। Natural language interface की अपनी खुद की एक schema होती है जो कि conceptual schema की तरह होती है।

4. **Graphical user interface:-** GUI आमतौर पर user को diagrammatic रूप में स्कीमा को प्रदर्शित करता है। उसके बाद यूजर diagram को manipulate करके एक query को specify करता है। बहु तसारी स्थितियों में GUI, Menu और form दोनों का

उपयोग करते हैं।

5. **Interfaces for DBA**:- बहु तसारेँ database system ऐसे कमांड्स को सम्मिलित करते हैं जो कि केवल database administrator द्वारा प्रयोग किये जाते हैं। किसी account को create करने, स्कीमा को बदलने और storage structure को फिर से organise करने तथा account authority प्रदान करने सम्बंधित commands इसके अंतर्गत आते हैं।

ehindistudy.com

प्रश्न:- डेटाबेस लैंग्वेजेज(language) क्या है?

Database languages:-

किसी system में डेटाबेस को create और maintain करने के लिए database languages का प्रयोग किया जाता है। डेटाबेस में निम्नलिखित languages का प्रयोग किया जाता है।

DDL:- DDL का पूरानाम data definition language है। जो कि conceptual schema को define करने के लिए यूज़ किया जाता है तथा यह इस बात की जानकारी भी देता है कि physical devices में इस schema को कैसे implement किया जाता है।

SQL में जो सबसे महत्वपूर्ण DDL statements हैं, वो निम्न हैं:

- 1.CREATE:- डेटाबेस में objects को create करने के लिये।
- 2.ALTER:- डेटाबेस के structure में करने के लिए।
- 3.DROP:- database में से objects को delete करने के लिए।
- 4.COMMENT:- data dictionary में comments को add करने के लिए।
- 5.RENAME:- object का rename करने के लिए।

DML:- DML का पूरा नाम data manipulation language है। और वह language जो डेटाबेस में डेटा को manipulate करने के काम आती है वह लैंग्वेज DML(data manipulation language) कहलाती है।

इसके कुछ उदाहरण निम्नलिखित हैं:-

- 1.SELECT:- एक डेटाबेस में से डेटा को retrieve करना।
- 2.INSERT:- table में डेटा को insert करना।
- 3.UPDATE:- table में मौजूदा डेटा को update करना।
- 4.DELETE:- table में से सभी records को delete करना।
- 5.CALL:- java subprogram को कॉल करने के लिए।
- 6.LOCK TABLE:- concurrency को नियंत्रित करने के लिए।

DCL:- DCL का पूरा नाम data control language कहते हैं। डेटाबेस

में संग्रहित डेटा के access को control करने के लिए DCL का प्रयोग किया जाता है।

इसके कुछ उदाहरण निम्नलिखित हैं।

1. GRANT:- database के लिए users को privilege प्रदान करने के लिए।

2. REVOKE:- GRANT command द्वारा दिए गए विशेषाधिकार को वापस लेने के लिए REVOKE command का प्रयोग किया जाता है।

प्रश्न:- डेटाबेस का वर्गीकरण कीजिये?

classification of DBMS:-

DBMS को विभिन्न आधार पर वर्गीकृत कर सकते हैं। जो कि निम्नलिखित हैं।

1.Data models के आधार पर:-

a. Traditional models:- Relational, Network तथा Hierarchical मॉडल्स इसके अंतर्गत आते हैं।

b. Emerging models:- object-oriented तथा object-relational मॉडल्स इसके अंतर्गत आते हैं।

2. Number of users के आधार पर:-

a. single user और multiple यूज़र्स

b. centralized और distributed यूज़र्स

3. number of sites के आधार पर

4. cost के आधार पर

5. Access path के प्रकार के आधार पर

6. purpose के आधार पर

प्रश्न:- dbms architecture (आर्किटेक्चर) क्या है?

DBMS architecture in hindi:-

हम यहाँ three schema architecture को describe करते हैं।

इसमें तीन Levels or schema होती है। जो निम्नलिखित हैं:-

1. PHYSICAL LEVEL or INTERNAL LEVEL

* यह लेवल यह describe करता है कि DBMS database में डेटा कैसे स्टोर होता है?

* e.g.- index, B-tree, hashing

* यह abstraction का सबसे निचला लेवल (lowest level) है।

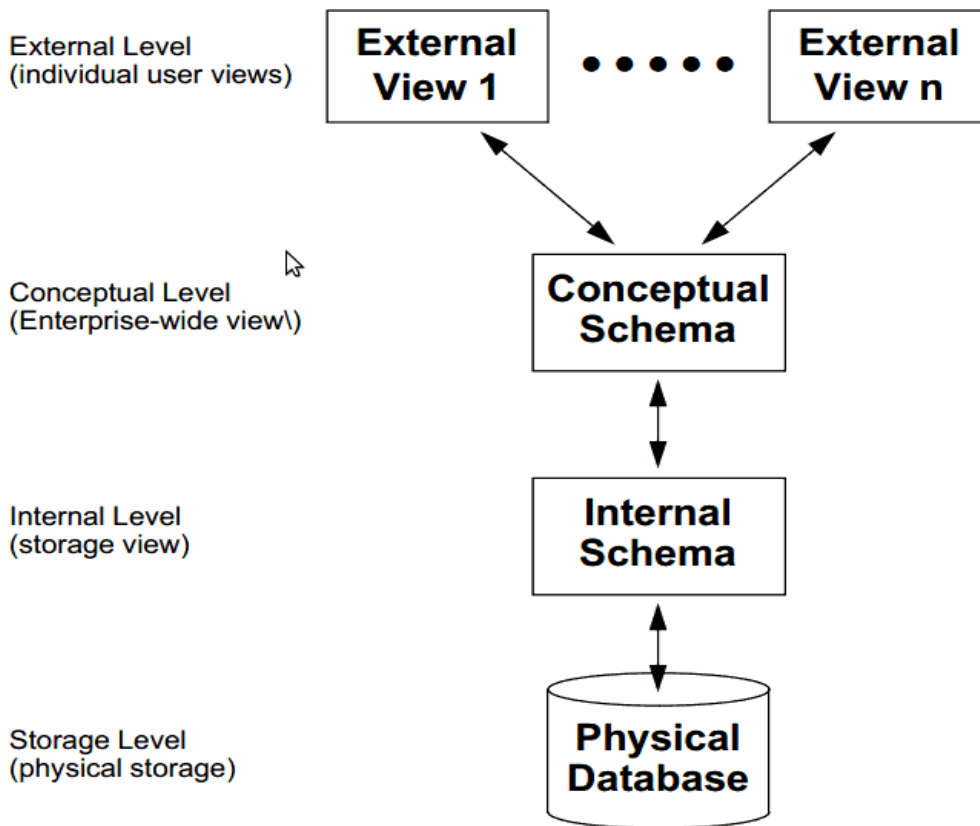
2. CONCEPTUAL or LOGICAL LEVEL:-

- * conceptual लेवल abstraction का अगला ऊँचा लेवल (next highest level) है।
- * यह लेवल describe करता है कि DBMS database में क्या डेटा स्टोर रहता है और उनके मध्य relationship क्या है?
- * यह Database administrator लेवल है।

3. EXTERNAL or VIEW LEVEL:-

ज्यादातर डेटाबेस user पूरे डेटाबेस का प्रयोग नहीं करते हैं लेकिन उसका कुछ भाग ही read करते हैं। इसलिए वे उस part के view level को provide कराते हैं।

- * यह abstraction का सबसे highest level है।
- * किसी विशिष्ट ग्रुप के users के लिए डेटाबेस के एक भाग(part) को describe करता है।
- * डेटाबेस के बहु तसारेँ विभिन्न views हो सकते हैं।



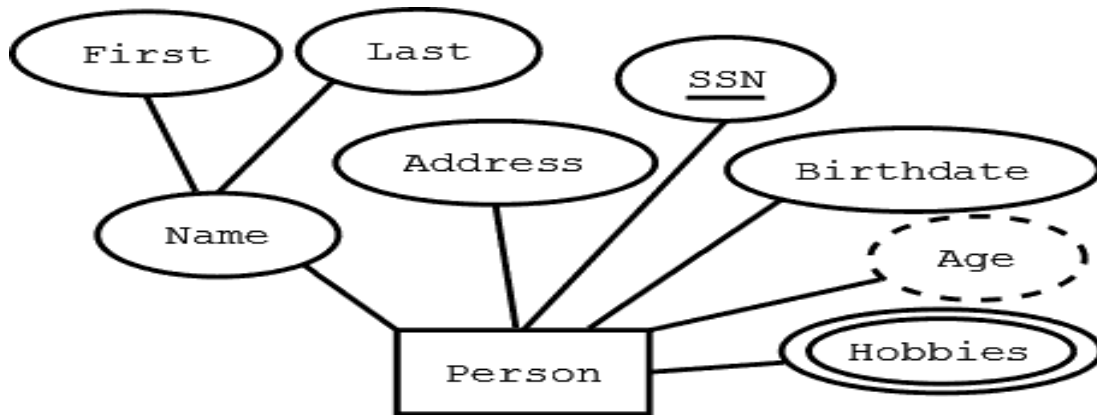
प्रश्न:- E-R model क्या है विस्तार से समझाइये?

E-R MODEL:-

E-R model को 1976 में peter ने प्रस्तावित किया था। E-R मॉडल का उपयोग real-world की conceptual schema को प्रस्तुत करने के लिए किया जाता है। E-R model डेटाबेस के conceptual view को define करता है।

E-R model को E-R diagram भी कहते हैं क्योंकि यह entities को graphical रूप से represent करता है और entities के मध्य

relationship को दर्शाता है।



Components of E-R MODEL:-

E-R MODEL दो मुख्य components पर आधारित है; पहला entities और दूसरा उनके मध्य relationships.

ENTITY:- एक entity कोई भी person, place और real word object हो सकता है। जैसे Customer id, customer name और Customer city आदि attributes है जिन्हें customer entity में define किया गया है। E-R diagram में entity को rectangles के द्वारा प्रदर्शित किया जाता है।

Entity sets:- एक entity set एक ही प्रकार के entities का एक समूह है जोे समान properties(गुण) share करता है।

types of entities:- DBMS में entity निम्नलिखित प्रकार की होती है।

1. Weak entity:- weak entity एक ऐसी entity होती है जो अपने attributes के द्वारा uniquely identify नहीं हो पाती है। तो हम कह सकते हैं कि इसमें primary key नहीं होती है।

2. strong entity:- वह entity जिसके पास primary key होती है strong entity कहलाती है।

Relationship:- E-R diagram में relationship, entities के मध्य relation को दर्शाता है। Relationship को diamonds के द्वारा प्रदर्शित किया जाता है।

उदाहरण के लिए:- teacher teaches_at school और soldier enrolls in a military. जहाँ teaches_at और enrolls, relationship कहलाते हैं।

DEGREE OF RELATIONSHIP:- relationship तीन प्रकार की होती है।

1:- Binary:- अगर relationship type की degree दो होती है तो उसे binary कहते हैं।
[binary= degree 2.]

2. Ternary:- अगर relationship type की degree तीन होती है तो उसे ternary कहते हैं।
[Ternary= degree 3.]

3.N-ary:- अगर relationship type की degree n होती है तो उसे n-ary कहते हैं।
[n-ary=degree n.]

Attributes:- एक attribute, entity की प्रॉपर्टी होती है जो की अन्य entities से different होती है और वो entity के बारे में सूचना provide करती है। वे एट्रिब्यूट्स जो entity को identify करते हैं उन्हें key attributes कहते हैं और वो attributes जो entity को describe करते हैं उन्हें non-key attributes कहते हैं। एक attribute type, entity type की प्रॉपर्टी होती है। attribute को ellipse के द्वारा represent किया जाता है। उदाहरण के लिए- student एक entity है और उसका subject name, subject code तथा gender उसके attributes हैं।

types of attributes:-

DBMS में attributes निम्नलिखित प्रकार के होते हैं:

1.Simple and composite attribute:- जो simple attribute होते हैं वो subparts में divide नहीं होते हैं जबकि composite attribute subparts में विभाजित हो जाते हैं। उदाहरण के लिए:- name attribute, first name तथा last name में विभाजित हो जाता है। जहाँ name एक composite

attribute है और first name तथा last name, simple attributes हैं।

2. Single valued and multivalued attributes:- वह attribute जिसके पास किसी विशेष entity के लिए केवल एक ही value होती है single value attribute कहलाती है। उदहारण के लिए- किसी व्यक्ति की age एक सिंगल वैल्यू एट्रिब्यूट है। वह attribute जिसके पास एक entity के लिए बहु तसारी values होती है multivalued attribute कहलाती है। उदहारण के लिए:- किसी car के लिए colors तथा employee का फ़ोन नंबर।

3. Stored and derived attribute:- इस प्रकार के attribute की value को दूसरे related attribute की value से derived किया जाता है। उदहारण के लिए:- किसी person की age को उसके birth date से derived किया जाता है। तो age एक derived attribute है तथा birth date एक stored attribute है।

प्रश्न:- entities के मध्य relationship को समझाइये?

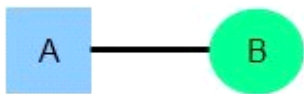
Relationship among entities:-

entities के मध्य तीन प्रकार की relationship होती है।

1. One-to-one(1:1) relationship:-

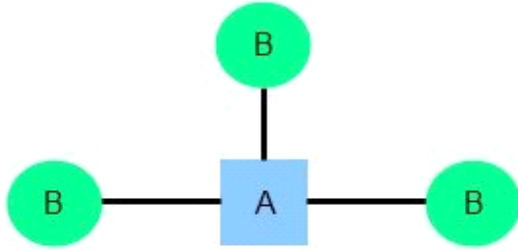
यह तब होती है जब entity A का एक instance, entity B के एक instance के साथ associated होता है।

उदहारण के लिए:- किसी office में employees का अपना अलग-अलग office होता है। प्रत्येक employee के लिए एक unique office होता है और प्रत्येक office के लिए एक unique employee होता है।



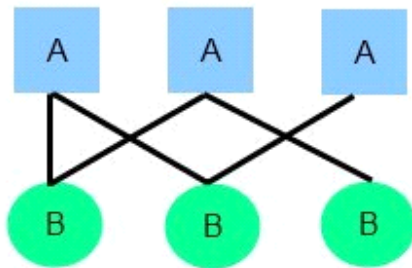
2. One-to-many(1:N):- Relationship तब होती है जब entity A का एक instance, entity B के बहु तसारे instance के साथ associated होता है।

उदहारण के लिए:- एक Employee एक department में कार्य करता है; एक department के पास कई employee हो सकते हैं।



Many-to-many(M:N):- Relationship तब होती है जब entity A के बहु तसारे instance, Entity B के बहु तसारे instance के साथ associated होता है।

उदाहरण के लिए:- बहु तसारे employees, बहु तसारे projects में कार्य कर सकते है।



प्रश्न:- domain किसे कहते है?

Domain:-

एक attribute के लिए permit की जाने वाली सभी unique values के सेट को हम domain कहते है।

उदाहरण के लिए:- gender के लिए एक field डोमेन [male,female,others] हो सकते हैं। जहां ये तीनों values ही permitted entries है।

प्रश्न:- relation क्या है?

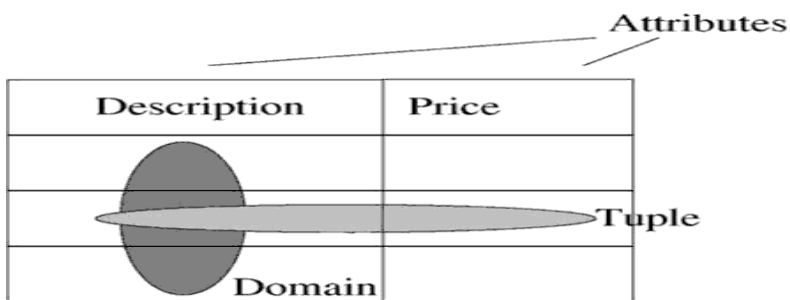
Relation:-

एक relation, rows or tuples का collection होता है।

प्रश्न:- tuple क्या है?

Tuple:-

एक Tuple, columns or attributes का collection होता है। tuple एक instance के लिए table की attributes के बारें में information का collection होता है। एक tuple को हम row कह सकते है यदि row, unique हो तो।



प्रश्न:- join किसे कहते है?

join:-

दो या दो से अधिक tables में से डेटा को प्राप्त करना join कहलाता है। प्रत्येक table में डेटा को insert करने के लिए कुछ business rules होते हैं। ये business rules, data constraints कहलाते हैं।

ehindistudy.com

प्रश्न:- dbms की सभी keys को परिभाषित कीजिए?

All DBMS keys:-

Rdbms में निम्नलिखित प्रकार की keys होती हैं।

1:- primary key:- किसी relational table की primary key टेबल के प्रत्येक record को uniquely identify करता है।

primary key दो प्रकार की होती है।

1. simple primary key
2. composite primary key

1. Simple primary key:- simple primary key केवल एक field से मिलकर बना होता है।

Customer ID	Forename	Surname
1	Simon	Jones
2	Emma	Price
3	Laura	Jones
4	Jonathan	Hale
5	Emma	Smith

Simple primary key

2. composite primary key:- composite primary key एक से ज्यादा fields से मिलकर बनी होती है।

Suit	Value	No. times played
Hearts	Ace	5
Diamonds	Three	2
Hearts	Jack	3
Clubs	Three	5
Spades	Five	1

One is not enough to identify, but both combined make a unique value

primary key को define करना:-

1. primary key अद्वितीय(unique) होती है।
2. किसी भी टेबल में केवल एक ही primary key हो सकती है।
3. ये single या multi column हो सकती है, multi column primary key को हम composite primary key कहते हैं।
4. composite primary key में अधिकतम 16 column हो सकते हैं।

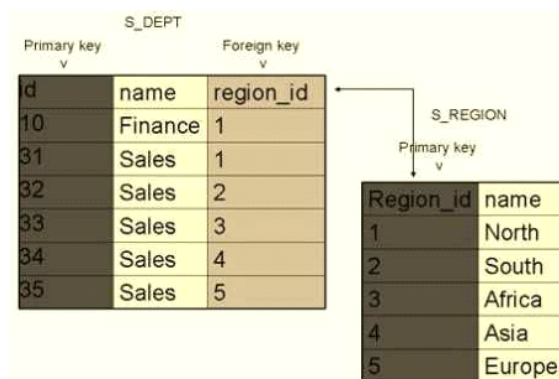
5. यह null values को contain नहीं करती है।

2:-foreign key:- रिलेशनल डेटाबेस टेबल में एक foreign key columns का एक समूह होता है जो कि दो tables में data के मध्य link उपलब्ध कराता है।

कहीं कहीं पर foreign key को referencing key भी कहा जाता है।

जब किसी एक primary key को किसी दूसरे टेबल में प्राइमरी key के रूप में प्रयोग करते हैं तो उसे foreign key कहते हैं।

foreign key, data में integrity को maintain करने के लिए method उपलब्ध कराता है।



3.Composite key:-

जब कोई primary key बहु तसारीं attributes से मिलकर बनी होती है तो उसे हम composite key कहते हैं।

रिलेशनल डेटाबेस टेबल में composite key दो या दो अधिक

columns का समूह होता है जो कि table में प्रत्येक row को uniquely identify करता है।

Composite Key

ITEM

Supplier_ID	Item_ID	Item_Name	Quantity
S ₁	I ₁	AC	5
S ₁	I ₂	Inverter	8
S ₂	I ₂	Inverter	4
S ₂	I ₃	UPS	15
S ₂	I ₄	Generator	5
S ₃	I ₃	UPS	10

Table: 5.5

4. Artificial key:-

artificial key तब permit होती है जब

- (a)- किसी भी attributes के पास primary key की सभी properties नहीं होती है।
- (b)- primary key बहु तबड़ी तथा complex होती है।

5- Super key:-

RDBMS में एक super key, columns का एक combination होता है जो कि row को uniquely identify करता है।

6:- candidate key:-

RDBMS में candidate key वह key होती है जो कि एक table में primary key की candidate की तरह कार्य करती है। हम simple words में कह सकते हैं कि candidate key primary key की

जरूरतों को पूर्ण करती है।

प्रश्न:- डेटा इंटीग्रिटी (data integrity) को समझाइये?

data integrity:-

database management system में डेटा का accurate तथा सम्पूर्ण होना data integrity कहलाती है तथा डेटा में duplicate ness नहीं होनी चाहिए इसका मतलब यह है कि डेटा consistence होना चाहिए जिससे डेटा की हानि ना हो। जिससे डेटा की integrity बनी रहती है।

डेटा integrity दो प्रकार की होती है।

1. Entity integrity
2. Referential integrity

1. Entity integrity:-

entity integrity का नियम यह है कि प्रत्येक टेबल की अपनी एक primary key होती है और प्रत्येक primary key यूनिक होती है और null नहीं होती है।

2. Referential integrity:-

referential integrity का नियम यह है कि किसी associated table में foreign key की value, primary key की value से

match करती है। referential integrity का नियम यह सुनिश्चित करता है कि tables के मध्य रिलेशनशिप consistence होती है।

प्रश्न:- dbms constraints के प्रकार लिखिए?

Types of dbms constraints:- प्रत्येक table में डेटा को insert करने के लिए कुछ business rules होते हैं। ये business rules, data constraints कहलाते हैं।

DBMS में constraints निम्नलिखित प्रकार के होते हैं।

1:-NOT NULL constraints:- इस constraints का प्रयोग तब किया जाता है जब किसी column को खाली ना रखना हो। NOT NULL constraints एक column की value को null होने से रोकता है। जब किसी column में इस constraints का प्रयोग किया जाता है तो उस कॉलम को null वैल्यू provide नहीं की जा सकती।

2:-Primary key constraints:- यह constraints डेटाबेस टेबल में प्रत्येक row (record) को विशिष्ट रूप से identify करता है। primary key कभी भी null वैल्यू contain नहीं कर सकती

है। यह हमेशा एक यूनिक वैल्यू contain करती है।

3:- Foreign key constraints: Foreign key constraints का प्रयोग दो tables में data के मध्य लिंक(link) provide करने के लिए करते हैं। इस constraints का प्रयोग referential integrity को लागू करने के लिए किया जाता है।

4:- Check constraints:-Check constraints का प्रयोग column में स्थापित की गयी value को limit करने में किया जाता है। इस constraints का प्रयोग domain integrity को लागू करने के लिए किया जाता है।

5:- UNIQUE constraints:-unique constraints, column की वैल्यू की uniqueness को दर्शाता है। Unique constraints यह ensure करता है कि column या field केवल unique values contain करती है इसका अर्थ यह है कि column की सभी values अलग अलग होंगी।

6:- DEFAULT Constraints:-इस constraints का प्रयोग column को default वैल्यू provide करने के लिए किया जाता है। जब किसी column को कोई भी value नहीं provide की जाती है तब Default Constraints का प्रयोग किया जाता है।

प्रश्न:- relational algebra (रिलेशनल अलजेब्रा) को विस्तार से समझाइये?

Relational algebra:-

Relational algebra, operations का एक समूह होता है जिसका प्रयोग relations से data को manipulate करने के लिए किया जाता है। Relational algebra एक procedural language है। इसमें operators का प्रयोग queries को perform करने के लिए किया जाता है और यह DBMS में प्रयुक्त होने वाली intermediate language है।

Relational Algebra में निम्नलिखित operators होते हैं:-

1. Union(U)
2. Difference(-)
3. Selection (σ)
4. Projection (Π)
5. Cartesian Product(X)
6. Intersection(\cap)
7. Rename(ρ)
8. Join

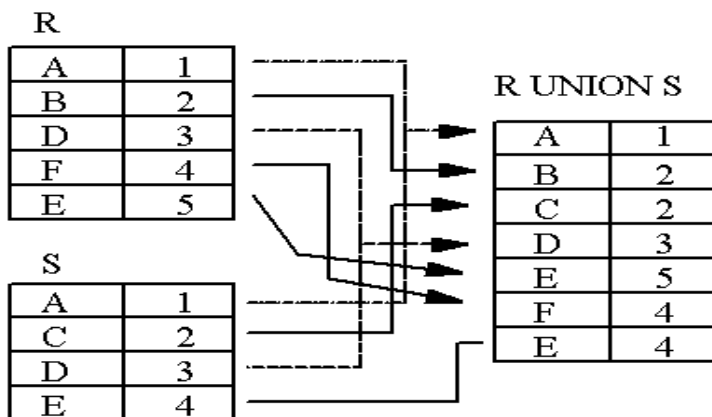
1. Union(U):- union operator को U symbol द्वारा दर्शाया जाता है।

इसका syntax निम्नलिखित है।

$[R=P \cup Q]$

जहां P और Q दो इनपुटरिलेशन है और R आउटपुटरिलेशन है।

यूनियन ऑपरेशन में दो input relation होते है जो कि union compatible हों। इस operation का output relation उन tuples(rows) को contain करते है जो कि relation 1 और relation 2 में हो अथवा दोनों में हो और जो duplicate tuples होते है उन tuples को eliminate कर दिया जाता है।



2. Difference(-):- Difference operator को – symbol

द्वारा दर्शाया जाता है। इसका syntax निम्नलिखित है:-

$[R=P-Q]$

Difference operation, relation 1 में स्थित common tuples को हटा देता है। हमारे जो output आएगा वह relation 1 में स्थित tuples होंगे जो कि relation 2 में न हो। हम इसे निम्न

चित्र के द्वारा आसानी से समझ सकते हैं।

R		R DIFFERENCE S	
A	1	B	2
B	2	F	4
D	3	E	5
F	4		
E	5		

S		S DIFFERENCE R	
A	1	C	2
C	2	E	4
D	3		
E	4		

3. Selection (σ):- इस operator को σ (सिग्मा) symbol द्वारा दर्शाया जाता है। इस operator का प्रयोग tuples को select करने के लिए किया जाता है जो कि दी हुई condition को satisfy करती है। यह operation एक unary operation हैं जो सिर्फ एक ही relation या table में define होता है।

4. Projection (Π):- इस operation को Π (pi) चिन्ह द्वारा दर्शाया जाता है। यह एक unary ऑपरेशन है अर्थात इसमें सिर्फ एक relation होता है।

Projection operator का प्रयोग एक relation में से attributes के subset(उपसमुच्चय) को select करने के लिया जाता है और जिन attributes को सेलेक्ट नहीं किया जाता है उनको eliminate कर दिया जाता है।

5. Cartesian Product(X):- इस ऑपरेटर को X चिन्ह द्वारा

दर्शाया जाता है। इसका syntax निम्न है:-

$$[R=P \times Q].$$

इस ऑपरेटर का प्रयोग दो विभिन्न relations की सूचनाओंको एक relation में सम्मिलित करने के लिए किया जाता है।

R		R CROSS S			
A	1	A	1	A	1
B	2	A	1	C	2
D	3	A	1	D	3
F	4	A	1	E	4
E	5	B	2	A	1
		B	2	C	2
		B	2	D	3
		B	2	E	4
		D	3	A	1
		D	3	C	2
		D	3	D	3
		D	3	E	4

6. Intersection(\cap):- इस ऑपरेटर को \cap चिन्ह द्वारा दर्शाया जाता है। इस ऑपरेटर का प्रयोग दो relations में से common tuples को select करने के लिए किया जाता है। इसका syntax निम्न है:-

$$[R=P \cap Q].$$

R		R INTERSECTION S	
A	1	A	1
B	2		
D	3	D	3
F	4		
E	5		

7. Rename(ρ):- Rename operator को ρ (rho) चिन्ह से

दर्शाया जाता है। इस ऑपरेटर का प्रयोग relation को दुबारा नाम देने के लिए किया जाता है।

9. join:- यह ऑपरेटर दो relations को एक नए relations में combine करता है।

join operator निम्नलिखित प्रकार के होते हैं:-

1. Theta(θ) Join
2. Left Outer Join
3. Right Outer Join
4. Full Outer Join
5. Natural Join ()

ehindistudy.com

प्रश्न:- डेटाबेस normalization को समझाइये?

database Normalization:-

DBMS में, Normalization डेटा को organise(व्यवस्थित) करने की प्रक्रिया है। सामान्यतया यह two step process है जो निम्न है:-

पहला step:- पहले step में, यह रिलेशनल टेबल में से redundant data(वह डेटा जो एक से अधिक बार स्टोर हुआ हो) को eliminate करता है।

दूसरा step:- दूसरे step में, यह ensure करता है कि table में

केवल उससे सम्बंधित डेटा ही store हो।

Normalization का मुख्य उद्देश्य ऐसे रिलेशनल टेबल के समूह को create करना है जिसमें कि redundant डेटा न हो तथा जो निरंतर और सही तरीके से modified हो सकें।

Types of Normal Forms

: -DBMS में normal forms पाँच प्रकार के होते हैं। लेकिन हम यहां पर तीन प्रकार के नार्मल फॉर्म के बारे में discuss करेंगे जिनको कि E.F CODD ने प्रस्तावित किया है।

1. First normal form(1NF): - एक टेबल तब first normal form में होता है जब वह repeating groups(बार-बार एक ही डेटा) को contain नहीं करता।

वह रिलेशन या टेबल जो repeating groups को contain किये रहती है वह रिलेशन या टेबल un-normalized कहलाता है।

2. Second normal form(2NF): - एक टेबल या रिलेशन तब 2nd normal form में होता है जब वह 1st normal form की सभी requirements को पूरी करता हों और सभी non key attributes पूरी तरह से primary key पर निर्भर हों।

3. Third normal form(3NF): - कोई टेबल या रिलेशन तब 3rd

normal form में होता है जब वह 2nd normal form की सभी जरूरतों को पूरी करता हो तथा उनमें transitive function dependency नहीं होनी चाहिए।

” Transitive functional dependency का अर्थ है यदि X functional dependent है Y पर तथा Y functional dependent है Z पर तो Z transitively dependent होगा X और Y पर।”

Fourth normal form(4NF):-

एक relation या table तब 4NF में होती है जब वह निम्नलिखित condition को satisfy करते हैं:-

” एक relation या टेबल तब 4NF में होती है यदि वह 3 normal form(3NF) में हो तथा उसके पास कोई multivalued dependencies ना हो।”

Multivalued dependency क्या होती है?

“Multivalued dependency तब होती है जब एक table में एक से ज्यादा independent(स्वतंत्र) multivalued attributes होते हैं।” multivalued dependency को \twoheadrightarrow चिन्ह से प्रदर्शित

किया जाता है।

उदाहरण के लिए:- कोई मोबाइल कंपनी प्रत्येक model के दो color(white व grey) के मोबाइल बनाती है।

Mobile_model	Manuf_year	Color
C_07	2013	White
C_07	2013	Grey
C_11	2014	White
C_11	2014	Grey

अब यहां पर manu_year और color एक दूसरे से independent है तथा वे mobile_model पर dependent है। तो हम इस प्रकार की dependencies को निम्न प्रकार से प्रदर्शित करते है।

mobile_model->>manu_year
mobile_model->>color

Fifth normal form(5NF):-

एक relation या table तब 5NF में होती है जब वह निम्नलिखित condition को satisfy करती है:-

“एक टेबल या रिलेशन तब 5NF में होती है जब वह 4NF में हो तथा table में कोई non-loss decomposition ना हो।”

BOYCE CODD NORMAL FORM:-

boyce codd normal form, 3NF का उच्चतर version है इसे कभी कभी 3.5NF भी कहा जाता है। वैसे तो वह table या relation जो 3NF में होती है वह BCNF में भी होती है परन्तु BCNF के लिए एक विशेष condition होती है जो निम्न है:-

“एक table या relation तब BCNF में होती है यदि प्रत्येक determinants एक super key होती है।”

Domain-key normal form(DKNF):-

यह normal form अन्य normal forms से अलग है। DKNF दो पदों domain constraints तथा key constraints पर आधारित है। इस normal form को 1981 में Fagin ने प्रस्तावित किया था। DK/NF में anomalies(विसंगतियाँ) को insert, delete तथा modify नहीं किया जाता है।

relation या table तब DKNF में होती है जब वह निम्नलिखित condition को satisfy करती है:-

एक table तथा relation तब DKNF में होती है यदि relation या table का प्रत्येक constraint, domain तथा key की definitions का परिणाम(result) होता है.

प्रश्न:- functional dependency क्या है?

Functional dependency:-

Functional dependency एक table में दो attributes के मध्य constraints के समूहको कहते है।

Functional dependency तब होती है जब table(relation) का एक attribute दूसरे attribute को uniquely identify करता है।

Functional dependency को एक तीर(->) चिन्ह द्वारा प्रदर्शित किया जाता है। यदि $X \rightarrow Y$, तो हम कह सकते है कि " X dependent है Y पर"।

प्रश्न:- non loss decomposition किसे कहते है?

Non loss decomposition:-

जब कभी table या relation में data normalization किया जाता है यदि उसमें सूचना(information) का loss(हानि) नहीं होता है तब उसे non loss decomposition कहते है।

कभी-कभी हम इसे lossless decomposition भी कहते है।

आसान भाषा में कहें तो loss less decomposition ऐसी

process है जिसमें duplicate data को हटा दिया जाता है और यह भी देखा जाता है कि इसमें original data की हानि ना हो।

ehindistudy.com

प्रश्न:- डेटा डिक्शनरी (data dictionary) क्या है?

Data dictionary in hindi:-

DBMS में, Data dictionary एक फ़ाइल या फाइलों का समूह होती है जो कि डेटाबेस के मेटाडेटा (metadata) को store करती है।

data dictionary डेटाबेस के वास्तविक डेटा को contain नहीं करती है, बल्कि यह सिर्फ डेटा को manage करने के लिए बहीखातों के रूप में information (जैसे-टेबल का नाम तथा विवरण आदि) को स्टोर करता है।

data dictionary के बिना dbms डेटाबेस से डेटा को access नहीं कर सकता है।

data dictionary को metadata भी कहते हैं और वह डेटा जो डेटा के बारे में सूचना provide करती है metadata कहलाती है।

Data dictionary का प्रयोग डेटाबेस ऑपरेशन, डेटा integrity

तथा accuracy को नियंत्रित करने के लिए किया जाता है।

data dictionary दो प्रकार की होती है:-

1:-Active data dictionary

2:-Passive data dictionary

1:-Active data dictionary:-वह data dictionary जो कि हर समय अपने आप ही DBMS के द्वारा update हो जाती है, Active data dictionary कहलाती है।

2:-Passive data dictionary:-Passive data dictionary भी active data dictionary के समान होती है परन्तु इसमें यह automatically DBMS के द्वारा update नहीं होती है।

प्रश्न:- aggression, generalization, specialization तथा association को समझाइये?

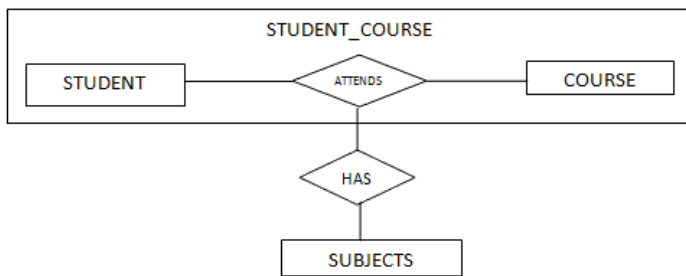
Aggregation in hindi:-

Aggregation में “HAS-A” relationship होती है। यह Association का एक special प्रकार होता है।

हम Aggregation को निम्न प्रकार से समझ सकते हैं:-

“Aggregation एक ऐसी प्रक्रिया है जिसमें information को

summary form में इकट्ठा(gather) तथा व्यक्त(express) किया जाता है। यह प्रक्रिया तब होती है जब दो entity के मध्य relation को single entity के रूप में treat किया जाता है।” Aggregation, relationships के मध्य relations को allow करके abstract entities को प्रदर्शित करता है।



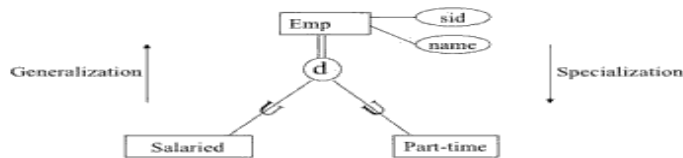
उदहारण:- यहां student तथा course के मध्य relation को एक entity के रूप में दिखाया गया है।

Generalization:- Generalization एक ऐसी प्रक्रिया है जिसमें दो या अधिक classes से समान विशेषताओं(characteristics) को extract(निकलना) किया जाता है और उनको “generalized” superclass में जोड़ दिया जाता है।

Generalization को त्रिकोण(triangle) द्वारा दर्शाया जाता है। यह IS-A relationship प्रदर्शित करती है। यह एक Bottom-up

approach है।

GENERALIZATION AND SPECIALIZATION



Specialization:- Specialization की प्रक्रिया

Generalization से विपरीत है।

इस प्रक्रिया में आप केवल एक superclass से एक से ज्यादा subclass बना सकते हो।

इसमें भी generalization की तरह IS-A relationship होती है।

यह top-down प्रक्रिया है।

association in hindi:-

Association दो या ज्यादा objects के मध्य relationship होती है। जिसमें प्रत्येक object की अपनी खुद की life-cycle(जीवन-चक्र) होता है और उनका कोई owner नहीं होता है।

Association को एक line(रेखा) के द्वारा प्रदर्शित किया जाता है।



उदहारण के लिए:- student तथा teacher की एक दूसरे के मध्य relationship होती है। एक student बहु तसारेँ teacher के साथ associate हो सकता है और एक टीचर भी बहु तसारेँ student के साथ associate हो सकता है। लेकिन दोनो objects के मध्य कोई ownership नहीं होती है और दोनों की अपनी life cycle होती है।

प्रश्न:- डेटा वेयरहाउसिंग (data warehousing) क्या है?

Data warehousing in hindi:- Data warehousing डेटा को एक स्थान में store करने का माध्यम होता है। इस माध्यम में डेटा को स्टोर करने के लिए कई प्रकार के hardware का प्रयोग किया जाता है। data को servers के समूह या कंप्यूटर पर store करना data warehousing कहलाता है। data warehouse में नये और historical data को store किया जाता है।

data warehousing का कार्य सिर्फ डेटा को स्टोर करना ही नहीं

होता बल्कि यह डेटा की security भी provide कराता है। Data warehouse कई प्रकार के resources से प्राप्त जानकारी को एक स्थान में एकत्रित करता है जिसे repository कहते हैं। इसमें सारी information एक ही schema में स्टोर की जाती है। एक बार collect हु आडेटा लम्बे समय तक store रहता है और वह लम्बे समय तक access किया जा सकता है।

data warehouse के निम्नलिखित विशेषतायें हैं:-

1:-Subject oriented:- Data warehouse का प्रयोग किसी एक विशेष subject को analyze करने के लिए किया जाता है अर्थात यह particular subject पर आधारित होता है।

2:-Integrated:-data warehouse बहु तसारे डेटा के स्रोतों से डेटा को एकत्रित(integrates) करता है।

3:-Time variant:- अर्थात इसमें नया और historical डेटा को रखा जाता है। किसी भी समय का डेटा data warehouse में रखा जा सकता है।

4:- Non-volatile:-एक बार जब डेटा data warehouse में स्टोर कर दिया जाता है उस डेटा को change नहीं किया जा सकता है मतलब historical data को बदला नहीं जा सकता है।

प्रश्न:- डेटा माइनिंग (data mining) क्या है?

Data mining in hindi:-

data mining को data या knowledge discovery भी कहते हैं। data mining, बहु तबड़े डेटा के समूह में से small डेटा को search करने की प्रक्रिया है। इस प्रक्रिया में परम्परागत statistics, artificial intelligence तथा computer graphics का प्रयोग किया जाता है। data mining में डेटा को analyze करने के लिए data mining tools का प्रयोग किया जाता है। ये tool बहु तही powerful होते हैं।

data mining के निम्नलिखित goals होते हैं:-

1:-Explanatory:- इसमें देखी गयी घटना या परिस्थिति को explain किया जाता है।

2:-Confirmatory:- इसमें संभावनाओं से मुक्त परिकल्पनाओं की confirmation की जाती है।

3:-Analyzatory:- इसमें नए डेटा को analyze किया जाता है जिससे positive feedback दी जा सके।

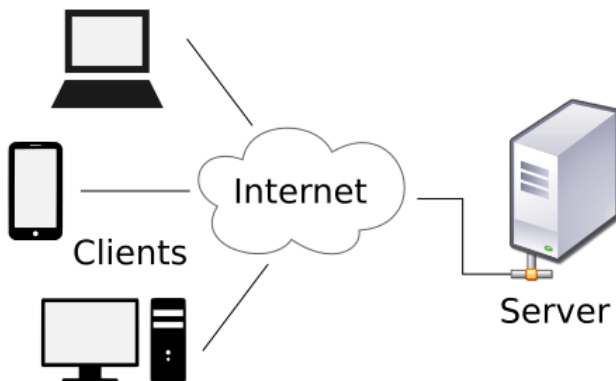
प्रश्न:- 2 tier आर्किटेक्चर तथा 3 tier आर्किटेक्चर क्या है?

2 tier & 3 tier architecture in hindi:-

हम निम्न तरीके से यह समझ सकते हैं:-

2 tier architecture in hindi:-

2 tier architecture जो होता है वह client-server architecture पर आधारित होता है। इसमें क्लाइंट तथा सर्वर के मध्य direct कम्युनिकेशन होता है तथा क्लाइंट तथा सर्वर के मध्य कोई तीसरा मध्यवर्ती नहीं होता है।



जैसा की चित्र में दिखाया गया है, 2 tier architecture में दो tier होती है:-

1:-data tier

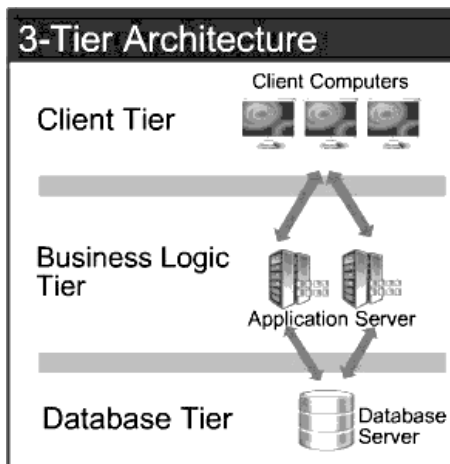
2:-client tier.

इस architecture में क्लाइंट, सर्वर को कोई task परफॉर्म करने के लिए request करता है तथा सर्वर उस task को पूरा करके

क्लाइंट की request को पूरा करता है।

3 tier architecture in hindi:-

जिस तरह 2-tier architecture क्लाइंट सर्वर आर्किटेक्चर होता है उसी तरह 3 tier architecture जो होता है वो client, server तथा database architecture होता है।



जैसा कि चित्र में दिखाया गया है 3 tier architecture में 3 tier होते हैं:-

- 1:-Client tier
- 2:-Businesses logic tier
- 3:-database tier

इस architecture में क्लाइंट, सर्वर को request भेजता है और सर्वर इस request को डेटाबेस को भेज देता है। फिर डेटाबेस request को पूरा करके सर्वर को भेजता है तथा सर्वर इसको क्लाइंट को वापस भेज देता है।

प्रश्न:- OLTP तथा OLAP को समझाइये?

OLTP in hindi:-

OLTP का पूरा नाम on-line transaction processing है। यह original अर्थात operational डेटा के साथ work करता है और OLTP प्रत्येक दिन के transaction को process करता है।

उदाहरण के लिए:-बैंक से पैसे निकालने में जो भी प्रोसेस होती है

जैसे:- atm card insert करना, पिन कोड डालना इत्यादि आदि

OLTP के अंदर आता है।

इसका प्रयोग तेज गति से डेटा को store करने के लिए किया जाता

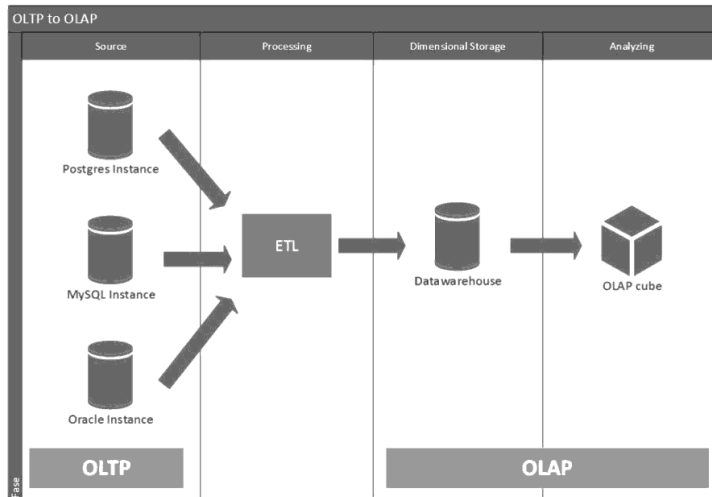
है। इसमें volatile डेटा तथा वर्तमान डेटा को execute किया

जाता है तथा इसमें historical डेटा नहीं होता है तथा OLTP में

detailed डेटा स्टोर होता है।

OLTP डेटाबेस में डेटा normalized होता है और इसका मतलब

यह है कि इसमें डेटा redundant(फ़ालतू) नहीं होता है



OLAP in hindi:-

OLAP का पूरानाम Online analytical processing है।

OLAP जो है वह historical डेटा के साथ work करता है। OLTP में डेटा normalized नहीं होता है अर्थात डेटा redundant होता है।

OLAP का प्रयोग डेटा को analysis करने के लिए किया जाता है तथा डेटा multi dimensional स्कीमा में स्टोर रहता है। इसमें डेटा non-volatile होता है।

उदाहरण के लिए:- रेलवे रिजर्वेशन के पिछले 5 सालों के रिकॉर्ड को analyze करें तो हमें पता चलेगा कि रेल का time क्या था तथा किस प्रकार के लोगों ने travel किया था आदि।

प्रश्न:- dbms transactions को समझाइये?

DBMS Transaction in hindi:-

Transaction, operations का समूह होता है जो कि केवल एक लॉजिकल ऑपरेशन की तरह treat होता है।

उदाहरण के लिये:-जब हम अपने बैंक अकाउंट से पैसे निकालते हैं या बैंक अकाउंट में पैसे जमा करवाते हैं।

ACID Properties:-Transaction की चार properties होती हैं

जिसे हम ACID Properties कहते हैं।

1:-A-Atomicity

2:-C-Consistency

3:-I-Isolation

4:-D-Durability

1:-Atomicity:-जब Transaction एक ही स्टेप में पूर्ण हो जाता है तब transaction atomic होता है।

2:-Consistency:-जब transaction होता है तो डेटाबेस एक state से दूसरे स्टेट में consistence होता है।

3:-Isolation:-जब दो या उससे अधिक transaction एक साथ execute होते हैं तो एक transaction दूसरे transaction को प्रभावित नहीं करता है।

4:-Durability:-जब एक transaction पूरी तरह से complete हो जाता है तो जो परिवर्तन होते हैं वह permanently सिस्टम में रहते हैं।

अर्थात् transaction durable होना चाहिए।

प्रश्न:- concurrency control क्या है?

Concurrency control in hindi:-

जब DBMS में बहु तसारे transactions एक साथ execute होते हैं तो Concurrency control एक ऐसी प्रक्रिया है जो यह सुनिश्चित करती है कि DBMS में डेटा सही तरीके से update हुआ है या नहीं।

हमें concurrency control की जरूरत इसलिए पड़ती है क्योंकि जो DBMS में डेटा होता है वह सही ढंग से update नहीं होता है तथा डेटा अपूर्ण तथा inconsistent होता है।

इसलिए DBMS में इन सभी परेशानियों को हल करने के लिए Concurrency control का प्रयोग किया जाता है।

प्रश्न:- dblc क्या है विस्तार से समझाइये?

DBLC in hindi:-

DBLC का पूरा नाम data base life cycle है। DBLC में 6 stages/phases होते हैं। monitoring, maintenance तथा modification ये तीनों DBLC का हिस्सा होते हैं इसलिए ये life cycle लगातार चलती रहती है और कभी समाप्त नहीं होती है।

इसमें निम्न stages होती है:-

1:- database initial study

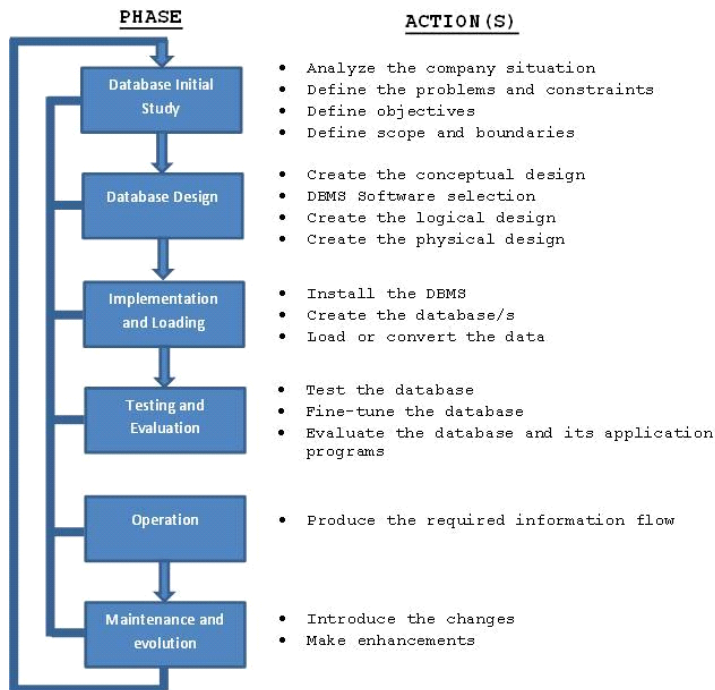
2:-database design

3:-implementation and loading

4:-testing and evaluation

5:-operation

6:-maintenance, monitoring and modification.



1:-database initial study:- यह डेटाबेस का सबसे पहला और महत्वपूर्ण phase है। इस phase में designer कंपनी के भीतर मौजूदा सिस्टम के ऑपरेशन की जांच करता है और यह निर्धारित करता है कि क्यों और कैसे system विफल(fail) हुआ है। database initial study का उद्देश्य निम्न है:-

- *कंपनी की स्थिति का विश्लेषण करना।
- *समस्याओं और बाधाओं को परिभाषित करना।
- *उद्देश्यों को परिभाषित करना।

2:-database design:- यह phase DBLC का सबसे critical phase होता है। इस phase में logical और physical दोनों प्रकार का design किया जाता है। logical design में conceptual model का निर्माण किया जाता है। एक conceptual model आमतौर पर E-R मॉडल होता है जो कि डेटाबेस के fields, tables और primary key को दर्शाता है और यह भी देखता है कि tables एक दूसरे से किस प्रकार सम्बंधित है। physical design का सिर्फ एक goal होता है :”डेटाबेस की क्षमता को बढ़ाना।” अर्थात डेटाबेस की performance को बढ़ाना।

3:-Implementation and loading:- इस phase में e-r diagram में बनाई गयी tables को SQL statement में convert किया जाता है। इस stage में तीन steps होते हैं:-

- *DBMS को install किया जाता है।
- *डेटाबेस को create किया जाता है।

*डेटा को load और convert किया जाता है।

4:-testing and evaluation:- testing डेटाबेस की quality को ensure करता है। जिससे risk को कम किया जाता है। testing के first स्टेप में डेटाबेस को टेस्ट किया जाता है जिससे कि data की integrity और security को maintain किया जा सके। database testing में हमें password security, data encryption आदि को check करने की आवश्यकता होती है। जब डेटाबेस और एप्लीकेशन प्रोग्राम्स को create कर दिया जाता है तथा उसकी testing भी पूर्ण कर दी जाती है तब उनको evaluate(मूल्यांकित) किया जाता है।

5:-Operation:- जब डेटाबेस का evaluation कर दिया जाता है तब हम operational phase में आते हैं। इस phase का मुख्य उद्देश्य यह ensure करना है कि डेटाबेस तथा एप्लीकेशन प्रोग्राम्स fully functional हों तथा ये अपने पूर्ण जीवन काल अपना job अच्छी तरह करें।

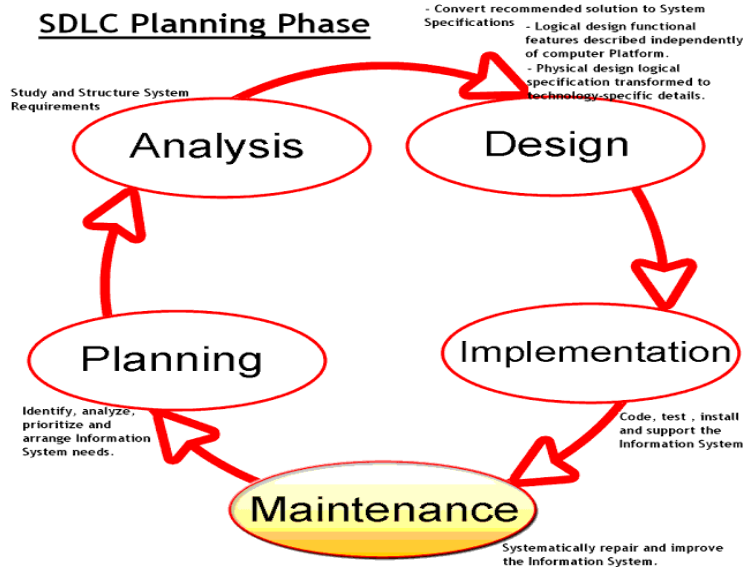
6:-maintenance, monitoring and modification:- यह dblc का अंतिम phase है इस phase में database को maintain, monitor तथा modify किया जाता है। जब सिस्टम

बनके तैयार हो जाता है तथा यूजर उसका प्रयोग करना शुरू कर देते हैं तब जो problems उसमें आती हैं उनको time-to-time हल करना पड़ता है। user requirements के हिसाब से डेटाबेस को modify करना पड़ता है। इसमें डेटाबेस का backup लेना भी इसमें शामिल है।

प्रश्न:- sdlc को समझाइये?

SDLC:-

SDLC का पूरा नाम system design life cycle है इसे system development life cycle भी कहते हैं। SDLC एक इनफार्मेशन सिस्टम(सूचना प्रणाली) के life cycle(जीवन चक्र) की व्याख्या करता है। डेटाबेस डिज़ाइन SDLC का एक मूलभूत घटक है। किसी भी सिस्टम को develop करने में जितने steps या process आते हैं उन सभी steps को मिलाके SDLC कहते हैं।



किसी इनफार्मेशन सिस्टम को develop करने में पाँच

phases/steps होते हैं। जो निम्न हैं:-

- 1.Planning
- 2.Analysis
- 3.Detailed system design
- 4.Implementation
- 5.Maintenance

1:-Planning:- planning phase में हम सिस्टम को हम क्या achieve कराना चाहते हैं या उसके goals क्या हैं?, discuss करते हैं। planning phase में सिस्टम की quality requirements तथा risk identification भी आते हैं। इस phase में हम यह देखते हैं कि जो वर्तमान सिस्टम है वह अपना काम सही तरीके से कर रहा है या नहीं। अगर वह सही से अपना

job कर रहा है तो उसे बदलने की कोई जरूरत नहीं है। इसमें हम cost को भी evaluate करते हैं। सिस्टम को design करने में कितने cost की जरूरत पड़ेगी यह सब planning phase में आता है।

2:- Analysis:- planning phase में problems को define किया जाता है तथा analysis phase में उन problems को अधिक details के साथ examine किया जाता है। analysis phase में यूजर की requirements को देखा जाता है, कि End users की क्या क्या जरूरतें हैं। इस phase में सिस्टम के हार्डवेयर और सॉफ्टवेयर को अच्छी तरह से study किया जाता है। इसमें end users तथा designers दोनों मिलके problem areas को हल करते हैं।

3:-Detail system design:- detail system design phase में डिज़ाइनर सिस्टम के प्रोसेस के डिज़ाइन को पूरा करता है। system में सम्पूर्ण technical specifications को include किया जाता है जिससे सिस्टम और भी ज्यादा interactive तथा efficient बन जाएँ।

4:-Implementation:- इस फेज में, हार्डवेयर, DBMS

सॉफ्टवेयर तथा एप्लीकेशन प्रोग्राम्स को install किया जाता है तथा डेटाबेस डिज़ाइन को implement किया जाता है। इससे पहले की डेटाबेस डिज़ाइन को implement किया जाए सिस्टम को टेस्टिंग, कोडिंग, तथा debugging प्रोसेस से होकर गुजरना पड़ता है। SDLC में यह सबसे लम्बे समय तक चलने वाला phase है।

5:-Maintenance:- यह SDLC का सबसे अंतिम phase है।

जब सिस्टम बनके तैयार हो जाता है तथा यूजर उसका प्रयोग करना शुरू कर देते हैं तब जो problems उसमें आती हैं उनको time-to-time हल करना पड़ता है। तैयार सिस्टम को समय अनुसार उसका खयाल रखना तथा उसे maintain रखना ही maintenance कहलाता है। SDLC में तीन प्रकार के maintenance होते हैं:-

1. corrective maintenance
2. adaptive maintenance
3. perfective maintenance

ehindistudy.com

प्रश्न:- डेटा मॉडल्स का वर्गीकरण (classification) कीजिए?

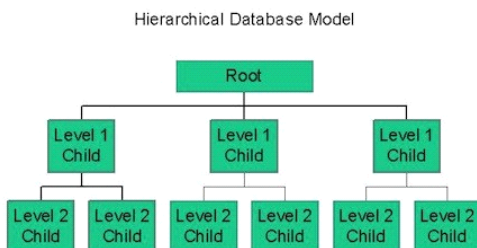
Classification of data models in hindi:-

data models निम्नलिखित प्रकार के होते हैं।

Hierarchical model:-

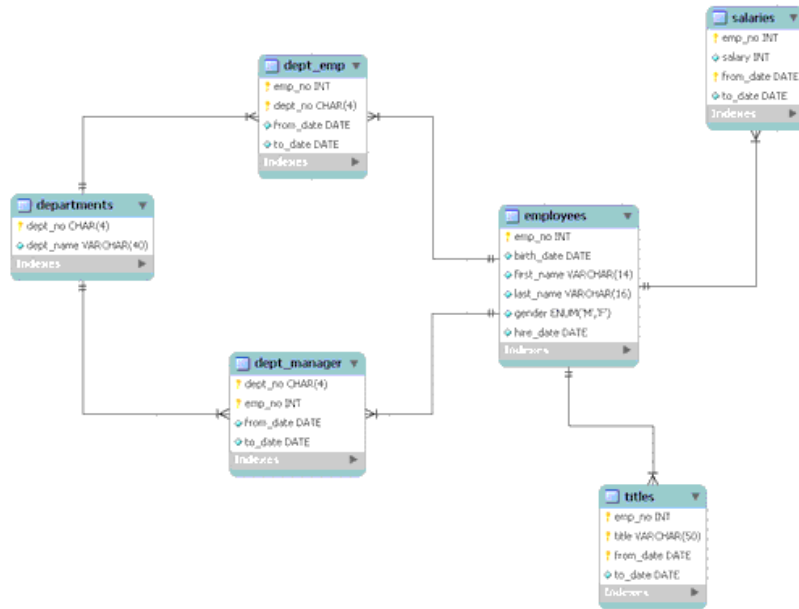
इस मॉडल में parent-child रिलेशनशिप होती है। इस model में प्रत्येक entity के पास केवल एक parent होता है और बहु तसारे children होते है। इस मॉडल में केवल एक entity होती है जिसे हम root कहते है। इस model में डेटा को tree like structure में organised किया जाता है। इसमें डेटा को records की तरह store किया जाता है जो कि एक दूसरे से जुड़े रहते है।

Hierarchical database model



RELATIONAL MODEL:-

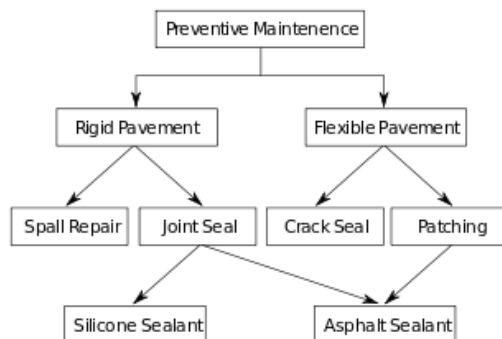
इस मॉडल में, data को relations अर्थात tables में स्टोर किया जाता है तथा प्रत्येक relation में rows तथा columns होते है। relational model टेबल्स का एक समूह होता है जिसमें डेटा तथा रिलेशनशिप को specify किया जाता है। relational model को 1969 में E.F Codd द्वारा प्रस्तावित किया था।



NETWORK MODEL:-

नेटवर्क मॉडल में entities को graph में organise किया जाता हैं। और इनमें से कुछ entity अनेक paths में से access कर सकती हैं। तो हम कह सकते है कि इस मॉडल में डेटा को नेटवर्क के रूप में स्टोर और access करते है।

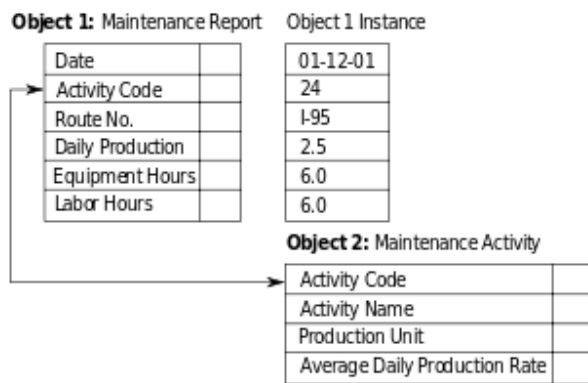
Network Model



OBJECT ORIENTED MODEL:-

object oriented model में information या data को object के रूप में प्रदर्शित किया जाता है तथा ये ऑब्जेक्ट्स instance variable में value को स्टोर किये रहते हैं। इस model में object oriented programming छमताओं का प्रयोग किया जाता है।

Object-Oriented Model



प्रश्न:- data models को समझाइये?

डेटा मॉडल्स यह describe करता है कि डेटा एक दूसरे से किस प्रकार जुड़े रहते हैं एवं उनके मध्य में relationship कैसी है। data model हमें यह भी define करता है कि किस प्रकार डेटाबेस का logical structure बनाया गया है। data model डेटा को संगठित तथा स्टोर करता है।

data model में logical design और physical design दो भाग शामिल होते हैं। हम कह सकते हैं कि मॉडल्स जो होते हैं वो

design की आधारशिला होते हैं।

जिस प्रकार इंजीनियर मकान को बनाने से पहले उसका मॉडल तैयार करते हैं उसी प्रकार डेटाबेस डिज़ाइनर, डेटाबेस डिज़ाइन को improve करने के लिए data model तैयार करते हैं। डेटा मॉडल का मुख्य उद्देश्य concept को communicate और specify करने के लिए किया जाता है।

Following figure shows role of data models

:-



प्रश्न:- SQL क्या है?

SQL को हम निम्नलिखित बिंदुओं के आधार पर आसानी से समझ सकते हैं:-

- 1- SQL का पूरा नाम structured query language है।
- 2- SQL का वास्तविक नाम SEQUEL है जिसे हम Structured English query Language कहते हैं।
- 3- SQL के द्वारा हम डेटाबेस को access कर सकते हैं।
- 4- SQL के द्वारा हम डेटाबेस में नए records को insert कर सकते हैं।
- 5- SQL के द्वारा हम डेटाबेस से डेटा को delete कर सकते हैं।
- 6- SQL के द्वारा हम डेटाबेस में स्थित data को update कर सकते हैं।
- 7- यह एक ansi(American national standard Institute) language है।
- 8- यह Oracle, MS-Access, DB2 तथा MS SQL Server के

साथ प्रयोग में लाया जाता है।

प्रश्न:- SQL data type क्या है?

SQL Data types:-

SQL में data types निम्नलिखित प्रकार के होते हैं तो आइये हम इन्हें विस्तार से जानते हैं।

* Exact numeric types:- SQL में निम्नलिखित प्रकार के exact numeric data types होते हैं।

1. Int:- int की सीमा(range) -2,147,483,648 से लेकर +2,147,483,648 तक होती है। इसको 4 bytes की storage की आवश्यकता होती है।

2. Bigint:- Bigint की रेंज -9,223,372,036,854,775,808 से लेकर +9,223,372,036,854,775,808 होती है। इसको 8 bytes की storage की आवश्यकता होती है।

3. Smallint:- इसकी रेंज -32,768 से लेकर +32,768 तक होती है। इसको 2 bytes की storage की आवश्यकता होती है।

4. Tinyint:- इसकी रेंज 0 से लेकर 255 तक होती है। इसको 1

bytes के storage की आवश्यकता होती है।

5. Bit:- bit डेटा टाइप की रेंज 0 से लेकर 1 तक होती है। इसको 1 bytes की storage की आवश्यकता होती है।

6. Money:- इस डेटा टाइप की रेंज -922,337,203,685,477.5808 से लेकर +922,337,203,685,477.5808 तक होती है। इसको 8 bytes की स्टोरेज की आवश्यकता होती है।

7. Smallmoney:- इस data type की range -214,748.3648 से लेकर +214,748.3648 तक होती है। इसको 4 bytes की आवश्यकता होती है।

8. Decimal:- इसकी रेंज $-10^{38}+1$ से लेकर $+10^{38}+1$ तक होती है।

9. Numeric:- इसकी रेंज भी $-10^{38}+1$ से लेकर $+10^{38}+1$ तक होती है।

*Datetime type:- SQL में निम्नलिखित प्रकार के datetime data types होते हैं।

1. Date:- यह डेटा टाइप साल, महीने, तथा दिन की वैल्यू को

स्टोर करता है। इसको 3 bytes की स्टोरेज की आवश्यकता होती है।

2. time:- यह डेटा टाइप घंटा, मिनट, तथा सेकण्ड की वैल्यू को स्टोर करता है।

3. Smalldatetime:- इस डेटा टाइप को 8 bytes की स्टोरेज की आवश्यकता होती है।

*Character data type:- SQL में निम्नलिखित प्रकार के character डेटा टाइप होते हैं।

1. Char
2. Varchar
3. Varchar(max)
4. Text

*Unicode character type:- Unicode character निम्नलिखित प्रकार के होते हैं।

1. nchar
2. nvarchar
3. nvarchar(max)
4. ntext

*Binary data type:- SQL में बाइनरी डेटा टाइप निम्नलिखित प्रकार के होते हैं।

1. Binary
2. Varbinary

- 3. Varbinary(max)
- 4. Image

*Boolean data type:- Boolean data type की सिर्फ दो वैल्यू होती है true या false.

प्रश्न:- SQL में table कैसे create करते हैं?

Create Table in SQL:-

SQL में हम TABLE को निम्नलिखित तरीके से create करते हैं।

SYNTAX:-

```
CREATE TABLE table_name  
(  
column_name1 data_type,  
column_name2 data_type,  
_____  
_____  
)
```

अब हम निम्न उदाहरण के द्वारा इसको आसानी से समझ सकते हैं:-

इस उदाहरण में हमने Student के नाम से table बनाई है जिसमें चार columns हैं। column के नाम Last Name, First Name,

Address तथा Roll number है।

```
CREATE TABLE Student  
(  
Last Name varchar,  
First Name varchar,  
Address varchar,  
Roll number int,  
)
```

यदि हमने column के अधिकतम साइज को specify करना हो

तो-

```
CREATE TABLE Student  
(  
Last Name varchar(30),  
First Name varchar,  
Address varchar,  
Roll number int(5),  
)
```

ehindistudy.com

प्रश्न:- DROP TABLE AND ALTER TABLE

DROP TABLE AND ALTER TABLE:-

DROP TABLE:- Table को ड्रॉप करने के लिए निम्नलिखित

syntax का प्रयोग किया जाता है;

SYNTAX:-

[DROP TABLE table_name]

ALTER TABLE:- Alter table स्टेटमेंट का प्रयोग टेबल में column को add और drop करने के लिए किया जाता है।

SYNTAX:-

[ALTER TABLE table_name
ADD to column_name datatype
ALTER TABLE table_name
DROP COLUMN column_name]

प्रश्न:- SQL built in function लिखिए?

SQL BUILT-IN-FUNCTIONS:-

SQL को built-in-functions और भी ज्यादा powerful बनाते हैं। Built-in-functions से हम मुश्किल tasks को भी आसानी से हल कर सकते हैं। इसलिए हम यहां built-in-functions के बारे में चर्चा करेंगे। जो निम्न हैं:-

1. SUM:- SUM फंक्शन का प्रयोग न्यूमेरिक कॉलम के total sum को select करने के लिए किया जाता है।

इसका syntax निम्न है:-

【*SELECT SUM (column) FROM table*】

उदाहरण के लिए:-

table name:- students

LAST NAME	FIRST NAME	AGE
Joshi	Yugal	21
Bhatt	Kamal	23
Sharma	Piya	20

यह टेबल students नाम से है। इस table की कॉलम के सभी ages को जोड़ दिया जाता है।

【SELECT SUM (age) FROM students】

इस उदहारण का answer= 64

2-AVG:- AVG फंक्शन, कॉलम की औसत वैल्यू को return करता है। इसका syntax निम्न है:-

【SELECT AVG (column) FROM table】

उदहारण:- ऊपर के चित्र की table में:-

【SELECT AVG (age) FROM students】

इस उदहारण का answer=21.33

3-MAX:- MAX फंक्शन कॉलम में सबसे उच्च(highest) वैल्यू को return करता है। इसका syntax निम्न है:-

【SELECT MAX (column) FROM table】

उदहारण:- ऊपर की टेबल से

【SELECT AVG (age) FROM students】

इस उदहारण का answer= 23

4- MIN:- MIN फंक्शन का प्रयोग column की निम्न(lowest) वैल्यू को return करने के लिए किया जाता है।

इसका syntax निम्न है:-

【 SELECT MIN (column) FROM table 】

उदहारण:- ऊपर की टेबल से

【 SELECT MIN (age) FROM students 】

इस उदहारण का answer= 20

5- COUNT:- COUNT फंक्शन का प्रयोग एक डेटाबेस टेबल में rows की संख्या को गिनने(count) के लिये किया जाता है।

इसका syntax निम्न है:-

【 SELECT COUNT (column) FROM table 】

6-SQRT:- इस फंक्शन का प्रयोग दिए हु एनंबर के वर्ग मूल(square root) को ज्ञात करने के लिए किया जाता है।

7-RAND:- इस फंक्शन का प्रयोग random numbers को generate करने के लिए किया जाता है।

8-Numeric Function:- इन फंक्शनस का प्रयोग सामान्यतया गणितीय manipulation करने के लिए किया जाता है। कुछ numeric functions निम्न है:-

ASIN, ABS, FLOAT, INT, INTEGER, LN, COT, COS,

DOUBLE, LOG10

9-String functions:- string functions का प्रयोग string manipulation के लिये किया जाता है। कुछ string functions निम्न हैं:-

CHAR, CONCAT, GRAPHIC, DIGITS, LCASE, LEFT, LENGTH, LOCATE, LOWER, DIFFERENCE

प्रश्न:- SQL में indexes क्या हैं? लिखिए.

indexes:-

डेटा को तेजी तथा सुविधापूर्णतरीके से ढूँढने(find) के लिए index को टेबल में create करते हैं।

INDEX के द्वारा हम data को बिना पूरी table पढ़ें तेजी से ढूँढ सकते हैं।

डेटाबेस की index उस तरह समान होती है जिस तरह किसी किताब के पीछे index होती है। किताब में हम किसी topic को पढ़ने से पहले index को देखते हैं कि वह topic किस पेज पर है फिर उस पेज में जाते हैं।

Indexes को existing table में create करते हैं जिससे कि हम rows को तेजी से तथा सुविधापूर्णतरीके से locate कर सकते हैं।

जो users होते हैं वो indexes को देख नहीं सकते हैं वे सिर्फ speed up queries का प्रयोग करते हैं।

Indexes को किसी table की एक या उससे ज्यादा columns में create कर सकते हैं तथा प्रत्येक index को एक नाम दे सकते हैं।

Unique index:- यूनिक इंडेक्स उस इंडेक्स को कहते हैं जिसमें दो rows के पास समान index वैल्यू नहीं होती है।

Unique index को create करने के लिए निम्नलिखित syntax का प्रयोग किया जाता है:-

```
CREATE UNIQUE INDEX index_name  
ON table_name (column_name)
```

column_name उस column को specify करती है जिसे हमने indexed करना हो।

Simple index:- यदि UNIQUE keyword का प्रयोग नहीं किया जाता है तो duplicate वैल्यू को permit किया जा सकता है।

इसका syntax निम्न है:-

```
CREATE INDEX index_name  
ON table_name (column_name)
```

composite index:- दो या उससे अधिक table में जब index को create करते हैं तो वह composite index कहलाता है।

इसका syntax निम्न है:-

```
CREATE INDEX index_name  
ON table_name (column1,column2);
```

implicit index:- implicit index वह indexes होती है जो डेटाबेस server द्वारा अपने आप create हो जाती है जब ऑब्जेक्ट(object) create होता है।

DROP INDEX:- Index को drop करने के लिए निम्नलिखित syntax का प्रयोग किया जाता है:-
Drop index_name index_name

प्रश्न:- sql view को समझाइये?

SQL VIEW:-

VIEW एक virtual टेबल होती है लेकिन यह किसी वास्तविक टेबल की तरह rows और columns को contain करती है।

जब एक टेबल create होती है और उसमें डेटा को insert किया जाता है। प्रत्येक user को इस डेटा को access करने के लिए permission की आवश्यकता होती है परन्तु हर किसी को इसकी permission नहीं दी जाती है जिससे डेटा की security बनी रहती है तथा data को दुबारा भी नहीं लिखा जाता है जिससे डेटा

redundant भी नहीं होता है।

View कोई भी record नहीं रखता यह केवल टेबल की definition रखता है तथा टेबल से डेटा fetch करके show करता है।

CREATE VIEW:- View को create करने के लिए निम्न

syntax का प्रयोग किया जाता है:-

```
CREATE VIEW view_name AS
  SELECT columns
  FROM tables
  WHERE conditions;
```

यहां पर हम view को कुछ भी नाम दे सकते हैं।

DROP VIEW:- view को drop करने के लिए निम्न syntax

का प्रयोग किया जाता है:-

```
DROP VIEW view_name;
```

UPDATE VIEW:- view को अपडेट करने के लिए निम्न

syntax का प्रयोग किया जाता है:-

```
CREATE OR REPLACE VIEW view_name AS
  SELECT columns
  FROM table
  WHERE conditions;
```

view का प्रयोग data manipulation के लिए भी किया जाता है

परन्तु view निम्न condition फॉलो करता हो:-

1:- view सिंगल टेबल से बनाया गया हो।

2:- यदि यूजर record insert करना चाहता हो तो view में primary key तथा सभी not null column भी include किये गए हो।

3:-view को update तथा delete करने के लिए view में से primary key तथा not null column को exclude(निकालना) किया गया हो।

ehindistudy.com

प्रश्न:- PL/SQL procedures को समझाइये?

PL/SQL PROCEDURE:-

Procedure एक stored प्रोग्राम होता है जिसमें से हम parameters pass कर सकते हैं। procedures का प्रयोग एक specific task को perform करने के लिए किया जाता है। यह दूसरे प्रोग्रामिंग लैंग्वेज के procedure की तरह ही similar होता है।

PROCEDURE फंक्शन(function) की तरह ही कार्य करता है परन्तु function में return value पहले से निर्धारित होती है

जबकि procedure में ऐसा नहीं किया जाता है। function को call करना तथा procedure को call करना अलग-अलग विधियां हैं। Function call करते समय उसकी वैल्यू या तो किसी variable में ली जाती है या output में print करायी जाती है। जबकि PROCEDURE को direct call किया जाता है तथा आवश्यक parameters pass कर दिए जाते हैं। PROCEDURE भी FUNCTION की तरह oracle database में store रहते हैं जिसे आवश्यकता के अनुसार user call कर सकता है।

TYPES OF PARAMETERS IN PL/SQL PROCEDURE:-

1.IN :- IN पैरामीटर procedure को एक वैल्यू pass करने देता है। यह एक read-only पैरामीटर है। Procedure के अंदर, IN पैरामीटर Constant की तरह कार्य करता है।

2.OUT:- एक OUT पैरामीटर calling program को वैल्यू return करता है। Procedure के अंदर, OUT पैरामीटर variable की तरह कार्य करता है।

3.IN OUT:- IN OUT पैरामीटर procedure को एक initial वैल्यू pass करता है और caller को एक updated वैल्यू return

करता है।

प्रश्न:- SQL OBJECT क्या है?

SQL OBJECT:-

SQL OBJECT यूजरके द्वारा define किये जाते है। SQL द्वारा कई प्रकार के object बनाये जाते है जैसे:-table, view, synonym तथा series इत्यादि। Objects को बनाने के लिए कुछ rules होते है जो निम्न है:-

1:-Object का नाम अधिकतम 30 characters का हो सकता है।

2:-Object का नाम case sensitive नहीं होता है। इसे upper तथा lower case में लिखा जा सकता है। object का नाम alphabet से शुरू होता है।

3:-object के नाम में " " या ' ' का use नहीं किया जाता है।

4:-object के नाम में digit alphabets, \$, # इत्यादि का प्रयोग किया जा सकता है। Database का नाम 8 अक्षरों से अधिक नहीं होना चाहिए

प्रश्न:- SQL cursor को समझाइये?

SQL CURSOR IN HINDI:-

SQL स्टेटमेंट को प्रोसेस करने के लिए oracle एक मेमोरी एरिया create करता है, जिसे context एरिया भी कहा जाता है।

Cursor इस मेमोरी एरिया के लिए पॉइंटर होता है और cursor के द्वारा इस मेमोरी(context) एरिया को नियंत्रित किया जाता है।

cursor का size उसके डेटा के अनुसार flexible होता है। Oracle cursor को open करने के लिए main memory में कुछ space predefined रखता है इसलिए cursor की size limited होती है।

जब भी कोई query run होती है तो cursor कार्य करता है।

Cursor दो प्रकार की होता है:-

- 1:-Implicit cursor
- 2:-Explicit cursor

1:-Implicit cursor:- जब भी SQL स्टेटमेंट execute होता है, Oracle के द्वारा implicit cursor को automatically create कर दिया जाता है(जब स्टेटमेंट के लिए कोई explicit cursor नहीं होता है।)

“प्रोग्रामर्स implicit cursor तथा उसमें उपस्थित सूचना को नियंत्रित नहीं कर सकते हैं।”

Oracle द्वारा internal processing के लिए जो cursor open किया जाता है उसे implicit cursor कहते हैं ।

2:- Explicit cursor:- Explicit cursor यूजर-डिफाइंड होता है। Explicit cursor के द्वारा हम मेमोरी(context) एरिया में ज्यादा नियंत्रण कर सकते हैं।

जब किसी टेबल से कुछ रिकॉर्ड को PL/SQL code block में प्रयोग किया जाता है तब cursor का उपयोग किया जाता है। इस cursor को declare करबे के लिए SQL queries का प्रयोग किया जाता है।

इन दोनों प्रकार के cursor में चार common attributes होते हैं, जो निम्नलिखित हैं:-

1:-%isopen:-यदि cursor open है तो true अथवा false वैल्यू return करता है।

2:-%found:-यदि record सफलतापूर्वक fetch किये गए हैं तो true अथवा false वैल्यू return करता है।

3:-%notfound:-यदि record successfully fetch नहीं किये गए हो तो true अथवा false वैल्यू return करता है।

4:-%Rowcount:-यह process किये गए record की संख्या return करता है।

Explicit cursor management:- Explicit cursor management करने के लिए निम्न steps होते हैं:-

1:-SQL की select query द्वारा cursor एरिया select करते हैं।

2:-Cursor को open करते हैं।

3:-Cursor variable में Record को fetch करते हैं।

Explicit cursor को create करने के लिए निम्नलिखित syntax है:-

[CURSOR cursor_name IS select_statement]

प्रश्न:- PL/SQL TRIGGERS को समझाइये?

PL/SQL TRIGGERS IN HINDI:-

Trigger संग्रहित(stored) प्रोग्राम्स होते हैं जब कभी table में कोई action जैसे:-insertion, deletion, update किया जाता है

तो triggers अपने आप स्वयं execute हो जाते हैं।

Triggers को call या invoke नहीं किया जा सकता क्योंकि जब भी किसी DML स्टेटमेंट को perform किया जाता है तो trigger अपने आप execute हो जाता है।

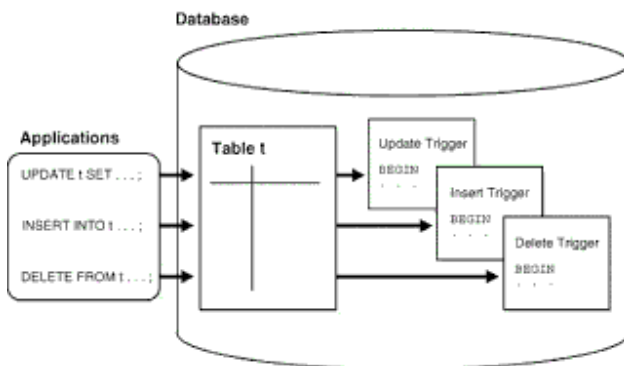


Fig:-PL/SQL TRIGGER IN HINDI

syntax of creating trigger:-

```
CREATE [OR REPLACE ] TRIGGER trigger_name
{BEFORE | AFTER | INSTEAD OF }
{INSERT [OR] | UPDATE [OR] | DELETE}
[OF col_name]
ON table_name
[REFERENCING OLD AS o NEW AS n]
[FOR EACH ROW]
WHEN (condition)
BEGIN
--- sql statements
END;
```

Types of triggers:- Trigger दो प्रकार के होते हैं:-

1:-Row level trigger:- जब प्रत्येक row में update,

deletion, तथा insertion होता है तो तब Row level trigger घटित होता है।

2:-Statement level trigger:-जब प्रत्येक SQL स्टेटमेंट execute होता है तब statement level trigger घटित होता है।

किताब खरीदने के लिए धन्यवाद तथा आपके लिए एक सुविधा:-

अगर आपको लगता है कि इसमें कोई टॉपिक या notes नहीं है तो आप मुझे बता सकते हैं. मैं आपको वह टॉपिक send करूंगा. आप

मुझे whatsapp तथा call करें.

मेरा मोबाइल नंबर 6395976262 है.

वेबसाइट:- www.ehindistudy.com

