# Db2(DBTKOL1.TS009) TKOPL060

Database models

fms- datasets file management system

hierarchical -ims(information management system)

Network -idms(set) intigratedata management system

Relational - db2 v1r1 1 st version released by 1983 flexibility model

With respect to object relationship

fms - every object is independent objects it means object relation is declaration not possible

hirechical dbms root (parent) child or parent child every object must be relationship parent child

d

a

emps

empd

emppd

first level should be always parent object next level should be child middle level should be either parent or child based on dependency

network

emppd emps

empd in network every object acts as parent and child every object relation parent and child

RDBMS

In this object relationship is optional based on the relationship acting as fms and heretical and network.

RDBMS model

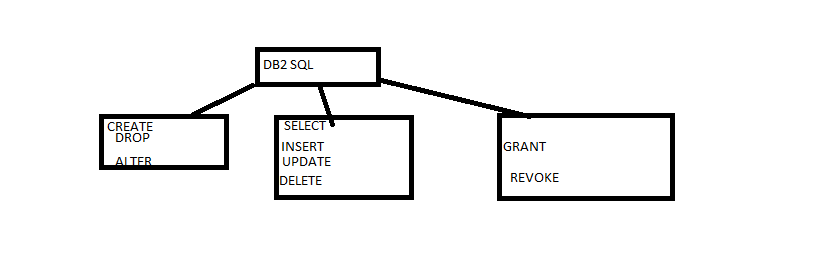
The rdbms model organize data in logical mathematical set in tabular from.

Here each data fields is considered as column and each record considered as row.

Structure of data as bytes

Db2 an abbreviation for ibm database 2 is ibms rdbms for mvs operating system as well as os/400 ,unix ,win etc,

Db2 is a sub system any os (mvs) that allows user to build, access and maintain relational databases, using the well –known relational language sql.



DDL DML DCL DDL

RELATE TO OBJECT LEVEL

ITS used to create table and drop table and remove the objects alter means change the existing objects

DML

DML operation is with respect to record level

Insert adding record, selecting records, updating records, deleting records

DCL

Dcl operation with respect to resource level the resource can be either user or object or record or component etc.

Grant to used permission to other user, revoke is used to remove the access to resource from user.

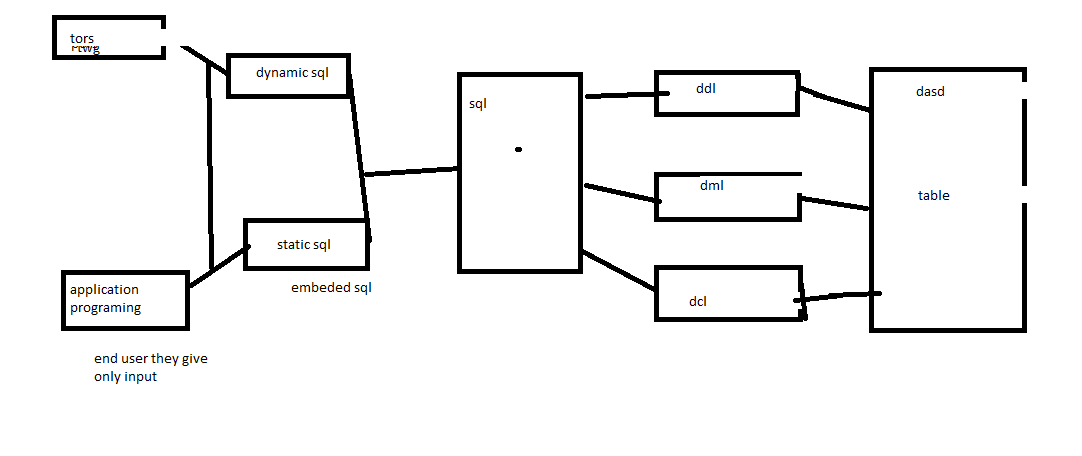
Commit and roll back means undo ,commit it will excute permanently

Begin transaction

Commit or rollback

\*\* code

Save point



Db2 inbuilt spufi (sql processing using file input) or else third party query manangement facility(qmf),caf,

The programing either dynamic or static (emebeded sql) application programing.

Db2 catlog

Db2 directory

Db2 log

Etc

User db types

1. Sysadm system admin
2. Dbadm, database admin
3. Db ctrc database controller
4. Db maint database maintaince
5. Sysopr system operator

Sysadmin

Dbadm sysopr

Dbctrc

Db main

Components of DB2

FROM high-level point of view db2 has four major components

1. System services

2. Locking services

3. Database services

4. Distributed services

Each above divided into sub components

System services

Handles all system –wide tasks.

Support system operation, operator communication, logging and similar functions

Control connections to other mvs subsystem (cics,ims/dc,ims/db,and tso)

Handles system start up, shut down and operator communication.

Manages the system log (both active and archive logs).

Locking services

Provides necessary control for managing concurrent access the data, using irlm.

IRLM: IMS resource lock manager

Database services

Supports the definition, retrieval and update of db2 data (user and system data), using a series of six sub components.

1. Pre-compiler

2. Bind

3. run time supervise.

4. Database manager

5. Buffer manager

6. Utilities.

Distributed data facility(ddf)

Provides db2’s distributed database support

Note: of the above four components only the database services component is directly relevant to the user

Db2 objects are:

1. Storage groups

2. Database

3. Tables space

4. Table

5. Index

6. Synonym.

7. View

8. Buffer poll

9. Db2 catalog

Storage groups: its uniquely named collections of dasd volumes all of same device type that is issued by db2 to allocate space for system.

Within each storage group spaces and partitions are stored using vsam LDS.

Database: is named collection of logically related tables their associated indexes, view, synonym and the various spaces that contain those tables.

Is the unit of start and stop a database can be made available or unavailable for processing by the operator.

Table space: is the physical space is used to hold one or more tables

Is the unit for recovery and reorganization.

Type:

1. Simple

2. Partioned

3. Segmented

Simple table space:

Can hold or more tables

Only one table per table space is recommended,

Partitioned table space:

The entire table space is divided into partitions and each partition is stored in vsam file (lds)

Holds only one table, the entire table contained across multiple partitions.

Partitions can be can be recovered or reorganized individually

Segmented table space.

The pages of a table space are organized into segments and each segment contain one table.

Event it can have more than one table.

Each segment consists of the same number of pages

­­­page:

Is a block of physical storage (unit of I/O transaction between the secondary and primary storage in a physical I/O operation.)

Available page sizes are 4k, 8k, 16k, 32k, 64k, default is 4k.

Table:

Is a collection of rows that have a set of columns with data present as values at the intersection of a row and columns

One table for each entity.

One column for each element of the data needed.

One row for every occurrence of the entity.

Every table and every column with to be named.

Table name unique for owner

INDEX

Is used to locate the rows that contain a given value.

Gain efficient and faster direct access to data.

Is based on one or more columns of a table.

Has an entry for every value in the columns, with a RID FOR EACH ROW that holds that value.

Multiple indexes can be defined on a table.

Index space:

Required for storing an index.

Each index exactly contains one index.

The index space is automatically created by the system. When the index is created.

Consists of addresses of each record stored in the table.

View:

A predefined selection of data in base tables.

The view definition is stored in the db2 CatLog, and is called whenever the view is referred to.

Is a virtual table that does not physically exist. But is processed as a table.

Data represented by a view is actually stored in the underlying base tables.

It can be created from one or more base tables or other views or combinations of view and tables.

Synonym:

Is an alternate name for table or view.

Can be used only by the user who created it.

Alias: is similar to synonym, but can be used even by other users.

DB2CATALOG:

IS A DB2 DATABASE PRE-DEFINED TABLE.

Is also called as system catalog.

Contains information about every db2 object that is maintained by it.

Contains a set of 30db2 tables stored in the dsndb06 database.

Whenever any db2 object is created, altered, or deleted, an entry is made in the db2 CataLog.

A user can retrieve information from a catalog.

However user cannot update catalog tables

Some tables in catalog:

Systables –contain information about the table’s views available.

Systablespace- contains one row of information for each table space available.

Sysviews –one row of information each view available.

Syscolumns- information about each columns of table and views including those of the catalog.

Every catalog table qualify by the owner name owner name is sysibm ex-sysibm.systables

DATA types

1. Numeric data types. – Small Integer, Integer, bigint, decimal (p,q),float(p)

2 .character data types. Char (2), varchar (n)

3. Graphic data types. Graphic (n), vargraphic (n), (needed for certain language whose character sets exceed256 characters/symbols.

4 .data/time data type. Date, time timestamp.

Small integer:

Data stored in binary format in half word

Two byte binary integer, 15 bits for data and 1 bit for sign range -32768 to +32768

Fractional values if inserted are truncated

Useful for columns that will always be whole number and used in arithmetic operations.

Integer:

Data stored in binary format in full word

Two byte binary integer, 31 bits for data and 1 bit for sign range -2147483648 to 2147483648

Fractional values if inserted are truncated

Useful for columns that will always be whole number and used in arithmetic operations

Bigint :

Data stored in binary format in two word

Either byte integer, with a precision of 19 digits

Decimal

Data stored in packed format.

P the total #of digits before and after the decimal point excluding the decimal point and the sign

Q the #of digits after the decimal points

Id there are no digits after the decimal point, q would be equal to zero

Float :

Data stored in floating point format ,p integer

P<22--------------------------- pb/w 22&53 both inclusive

Single precision stored in full word double precision stored in two words (8 bytes)

(4 bytes)

Used exclusively for scientific applaications where extreme precision is required.

Char (n):

Data stored in fixed length character format, string of 8 –bit character.

N take max value of 255.

Varchar (n)

Data stored in variable length character format, string of 8-bit characters.

The maximum number of characters is size of page.

2-byte length fields precedes the actual data.

Depends on page size

Graphic (n):

Data stored in fixed length character format , string of 16 – bit characters.

N takes max value of 128.

Vargraphic(n).

Data stored in variable length character format, string of 16-bit characters.

The maximum number of characters is half of page size.

DATE

Represented as sequence of eight digit unsigned packed decimal, occupying four bytes

YYYYMMDD

00010101 TO 99991231

Time:

Represented as a sequence of six digit unsigned packed, occurs three bytes

Range 000000 to 240000

Time stamp:

Is a combination of date and time ,accurate to the nearest microsecond.

Represented as asequence of 20 digit unsigned packed decimal digits (yyyymmddhhmmssnnnnnn)

Occupying ten bytes.

Range

Syntax

Create table [user id.] table name column name data type [not null],column name data type [not null],

[primary key

Db2 sql/ddl statements -4

If atable space name is not specified ,then the table space is implicitlry/automatically careated in the database specified.

If a database is not specified , the default database dsdndb04 is used.

Example.

Create table tab\_001 (ecode smallint not null,ename char(30) in dbtkol1.ts001;

Here DBTKOL1 IS THE database name, and ts001 is the tablespace name.

Syntax :

Insert into table name values (col1,col2);

Example:

Insert into employee values(111,ann);

SQL CODE IS o means success or fail sql code

Less than zero indicates error

Greater than zero means warning or exception.

Sql code is 100 means no more records

Update :

Update table –name set colname =new ;[where some condtion]

Example:

update employee set ecode=2222 where ecode=’e104’;

delete:

delete from table name where some condtion

ex:

delete from employee where esal=5000 and dept=’trng’; both condtion

where esal=5000 or dept=’trng’ both will remove

Primary key and foreign key used to define the relationship between the objects

Primary key: indicates its parent table the number of primary key is 1 and it’s to be unique and not null.

Foreign key indicates child the numbers of foreign key is more than one. The value will be unique or non-unique either null or not null the foreign key always refer by parent table.

Primary key(col1)or primary key(col1,col2,col3..) composite primary key.

Foreign key(keyname)

Create employee (ecode int(5),ename varchar(20),eadd varchar(20), deptname varchar(20),depaid int(5)),Primary key(ecode) in dbtkol01.ts006;

Create dept (ecode int(5),deptname varchar(20),depaid int(5)),

Foreign key (ecode int(5)) reference by employee

Whenever delete record from parent table check the corresponding record in child table if it is available remove record automatically from table

On delete cascade

On delete restriction

When ever if you want remove the record first delete the data I child and after that we can delete in parent. Other wise you get error.

ON DELETE SETNULL

WHENEVER DELETE RECORD FROM PARENT TABLE THE DB2 SYSTEM CHECK THE CORRSPODING ENTRY IN CHILD TAB;LE IF IT IS AVAILABLE the foreign key value set to null

Unique key the col of unique is not null.

Syntax for creating index :

Create index index name on table name (col name [desc],col name-2,);

This creates an unique index ‘indxsale’ on the base table sales based on the ‘salesnum’ fileds.

Create unique index indxsale on sales(salesnum);

View:

Create view view name[(view column-names,..0]

As select table column-names….from table name where condition.

Example:

Create view sales-v1 as select salesnum, prodcutname, quantity, tot amount from sales;

Create view sales-v2 as select salesnum, product name,quantity,tota-amt from sales where quantity >30;

Synonym:

Create synonym synonym name for userid.tablename/userid.view-name

Example: create synonym empsyn for educ08.emptab;

Alter :

Modifies the specifications associated with a table.

Allow to add a new column to table , and add or remove a primary or foreign key.

Syntax for adding new col:

Alter table table name add col name data type [not null with default]

Ex:

Alter table employee add empp char(5);

If no default is specified, the new column will have null values for all rows already in the database table sales

Drop:

Srop db2-object object-name;

Drop table sales;

Drop index indsal;

Drop vie view ame;

Select [all |distanact] scalar expression from data source names[where condtion] group by columns having condtion

Order by by column

All select al the record default is all select all the rows including duplicate rows

Select all rows elimate duplicate rows is distinct

Scalar expression can be either column names or any arithmetic expression or \*

\*indicates dall columns

Data source name indicate may can be table or view or synonym or alias

From must

Select c1 as c1new from t1; alias column name changing

Select all from t1;

Select c1 from t1;

Select \* from t1;

Select all c1 from t1;

Select c1+10 from t1;

Group by means certain columns how many data same in the column like a,b,b,a,c

Where condition check before group by column after having condition after order by condition.

Select c1 from t1 order by desc;

Select sal distinct from sal; means remove duplicates sal

Select fname,ename,cname,ssn from employee wheressn=5;

Select s# from s where city=’chennai’ and status>20;

While where condition same database content case sentsitive;

Select s# ,ststus from s where city =’chennai’

Order by ststus desc;

Between:

Selection values which are in particular range ,inclusiving fromstaring and ending.

Select \* from employee where dn= 5 and salary between 16 and 20;

We can use also not in range salary not between 16 and 20;

Retrevial using in:

Selection values where the column value corresponds to any of the given list of values.

Ex:

Select s#,pname,color,weight from p where weight in (12,16,,17);

Select s#,pname,color,weight from p where weight Not in (12,16,17);

Like:

Retrieval of rows from table which match the corresponding set of characters given In the like option.

‘%’symbol represents any number of any type of characters at that location.

‘-‘ symbol represents exactly one character occurring at that location.

Getb all parts whose name begin with the letter ‘c’

Select p#,pname,color from person where like ‘c%’;

Select pname from person where like ‘%abc%’;

Handling null values:

Handling ‘null’ in sql is done is done by using the where caluse along with the ‘is null’ operator.

Null cannot

Sub queries:

A sub query involves embedding a query (select) inside another query.

The sub query excutes first and main query’s execution depends on the execution of the sub query.

Sub query used to establish link between object in the run time select sname from s where s# in (select s# from sp where p#=’p2’);

Select sname from s where s# in (select s# from sp where p# in (select p# from p where color=’red’));

Correlated subquery

Example;

Select sname from s where ‘p2’ in (select p# from sp where s3=s.s#)

Slelect sname from s where ‘p2’ in (select p# from sp where sp.s# =s.s#);

Join queries:

Is is used to select data from multiple tables based on column

Join queries can be classified into two types as:

Inner join

outer join.

Left outer join.

Right outer join

Full outer join.

Inner join :

An inner join selects only matching rows from both the tables specified in the join.

Ex:

Select emp.\* ,dep.\* from emp,dep where emp.deptno.=dep.dnum;

Select emp.\*,dep.\* from emp inner join dep on epm.deptno=dep.dnum;

Select \* from emp,dep;

Slect \* from emp inner join dep;

Lef join or leftouter join:

A left outer join selects all rows In the table specified in the left side and only matching rows from the table specified in the right side specified in the join.

Select emp.\*,dept.\* \* from emp left outer join. Dep on emp.deptno =dep.dnum;

Right outer join or right join:

An right outer join selects all rows in the right side table and only matching rows from the table specified in left side specified in the join.

Select emp.\*,dept.\* from emp right outer join dep on emp.deptno=dep.num;

Fullouter join:

Full outer join selects all rows in the table matched or un matched join.

Select emp.\*,dept.\* from emp full outer join dep on emp.deptno=dep.num;

Union:

It is used to select data from multiple tables based on row ex:

Select p# from p where weight >16 union (no dupliactes) or union all (allow duplicates) select p# from sp where s#=’s2’;

The number column selction must be equal

Group by:

Max,min,count,sum ,avg aggregate functions

Select p# ,sum(qty) from sp group by p#

Having :

Same as where but after group by

Select p# from sp group by p#

Having count (\*)>

Embedded sql:

An sql application program written in a host language with embedded sql statements.

Db2 supports the following host language.

Pl/i

Cobol

Fortran

Assembly language.

Delimiting sql statements:

Coding sql statements in a cobol application requires beginning and ending delimiters.

The delimiters are exec sql and end-exec.

Format:

Exec sql (any sql statement)

End-exec.

COBOL verb:

It is used execute other than COBOL instructions inside a COBOL.

Steps to be followed:

1.declaring sql communication area(sqlca).

2.declaration of the table to used with in the application program.

3.host variable declaration.

4.coding sql statements to manipulatate db2 data.

Sql communication area.

Db2 program communication is accomplished through the sql communication area.

Sqlca provides the fields set by db2 after the sql statement execution.

The return code placed in the field sqlcode.

Its indicates the success, failure or exception encountered.

Heart code method and include method.

Declaring sqlca:

Code from vamsi

An sqlca must be declared in the working storage section of every db2/sql cobol program.

Declration of sqlca can be done in two ways:

1:coding the cobol description of sqlca directly.

2. coding sql statement:

Exec sql

Include sqlca

End-exec.

Identification division.

Program-id. Lab41.

Data division.

Working-storage-section.

Exec sql

include sqlca

end-exec.

Declaring db2 tables:

Db2 tables to be used in a cobol program must be declared in the working –storage section.

Two ways of declaring tables.

1.coding the table declaration directly ,using the declare table statement.

Exec sql

Declare enptable table

(ecode int not null, ename char(10)

End-exec.

2.declaration:

Using dclgen(declaration generation) to generate the table declaration and then copying the same into the program in the working-storage section using the include statement.

Identification division.

Program-id. Lab41.

Data division.

Working-storage-section.

Exec sql

include sqlca end-exec.

Exec sql Include emdc1 End-exec.

Empdc1 is the member of partitioned data setgenerated by dclgen and contains the declaration of table table-name.

Indicater variable picture class must be s9(4)comp

Variable corresponding to table column is called host variable the host variable data type must be equivalent to column data type.the host variable used for I/o operation. The number of host varibles equal the number of column in table.

Host variables:

1.are preceeded by a colon when used in in sql statements to signify that they are host variables and not sql variables.

Exec sql

Insert into emp\_tab1 values(:hv-ecode,:hv-ename.:hv-eadd,:hv-esad)

End –exec.

Are just regular cobol data items when use in a non-sql

Cobol:

Move input-ecode to hv-ecode .

Accept hv-name.

Display hv-eaddr-text.

Example for update:

Move 12345 to hv-ecode

Exec sql

Update emp-tab2\_013

Set eaddr=:hv-eaddr

Where ecode=:hv-ecode

End-exec.

Exec sql

Update emp\_tab2\_013

Set eaddr=’new address’

Where ecode=11111

End-exec.

Update query

Exmplae delete:

Exec sql

Delete from empl\_tab2\_013

Where esal>150000

End-exec.

Exec sql

Delete from emp\_tab2\_013

From esal >:hv-esal

End-exec.

Example for select

Move ‘guru’ to hv-ename.

Exec sql

Select ecode,ename,eaddr

Into :hv-ecode,:hv-ename,:hv-easddr

From emp\_tab2\_013

Where ename=:hv-ename

End-exec.

Rules for select

Each evaluation of the select statement must returns only one row from the table emp\_tab2\_013.

Db2 places the retrived data directly into the host varibles.

If it is return code more than one row then, the select statement return error message-811 (sqlcode)

Note: the row should not have any null data

If it is return any column has null data then, the select statement return error message -305 (sql code)

Indicator variable : used for handling null values

Indiacter variable set to zero indicate not null without truncation

Indicator variable set =-1 indicate null

Indicator variable set= >1 indicate not null but truncation is poosible

Greater than zero indicates value indicates before trumcates for example:

Char(4)

raja

Hv-name picx(3)

Raj

Cursor facility:

The cursor facility allows a COBOL program to gain addressability to individual row occurrences of many –rows result table. (Read more than one columns)

Steps:

1. Declare cursor either in working –storage or procedure division.

2. Open the cursor

3. Fetch rows until the end of result table’s condition occurs.

4. Close the cursor.

Format:

Exec sql

Declare cursor-name cursor for

Select column names

From datasource(s)

[where conditions to be met(if any)]

[for update of column name’s]

End-exec.

Using cursor for updates:

Example:

Exec-sql

Update table-name

Set column=<new-value>

Where current of <cursor name>

End-exec.

Declare cursor statement example:

Exec sql

Declare c1 cursor for

Select ecode,ename,eaddr,

From emp\_tab\_3060

Where eaddr=:hv-eaddr

End-exec.

Open statement:

Syntax.

Exec sql

Open cursor name

End-exec.

Example:

Exec sql

Open c1

End-exec.

Open statement built result table and place point over the cursor

Fetch statement

Returns a row from the result table to the program’s data area

Format,

Exec sql

Fetch cursor-name

Into host-varible

End-exec.

Host variable correspond to the parameters in the associated cursor statement.

Example:

Exec sql

Fetch c1

Into :hv-eocde,:hv-ename,:hv-eaddr

End-exec.

Close statement:

Tell the db2 system that the accessing of the result table is completed.

Syntax

Exec sql

Close cursor name

End-exec.

Example:

Exec sql

Close c1

End-exec.

It is used to remove the table from main memory.