

INTRODUCTION TO DB2

TOPICS COVERED

- DB2 Introduction
- NORMALIZATION
- DDL

DB2 is an abbreviation for ‘IBM DATABASE 2”

DB2 is a subsystem of the z/OS Operating System

It is a database relational management system (DBRM)
– RDBMS (Relational Database Management System)

Id	name	loc	salary
Numeric	char	char	decimal
1001	tommy	chennai	90000

It is a system that allows any number of z/OS users to access any number of relational databases by means of the well-known relational language SQL

What is Data ?

Data is a collection of Facts/Information.

What is Database ?

It is a self describing collection of Integrated data.
(Data + Application Program).

What is DBMS?

It is a system (hardware + software) that creates and manipulates the collection of inter-related data.

What is RDBMS?

It is a tool for organizing and manipulating data that is stored in a TABLE like structure.

E.g.: DB2, Oracle, Sybase etc.

WHAT IS DATA AND INFORMATION

Information is nothing but refined data and is the backbone of an organization.

Consists of

- Data
- Images
- Documents
- Even voice

NORMALIZATION

- It is a technique that allows the designer to detect redundancies in a given table.
- It is used to check the data and/or design your tables.

NORMALIZATION

- After you define entities and decide on attributes for the entities, you normalize entities to avoid redundancy.
- Normalization helps you avoid redundancies and inconsistencies in your data.
- It is a paper and pen work, not codes.

NORMALIZATION

**FIRST NORMAL FORM
SECOND NORMAL FORM
THIRD NORMAL FORM**

NORMALIZATION

- Norm I-Eliminate Repeating Groups.
- Norm II-Eliminate all columns that depend only on part of the key.
- Norm III - Eliminate columns not dependent on the key at all.

NORMALIZATION

- The rules for normal form are cumulative.
- In other words, for an entity to satisfy the rules of second normal form, it also must satisfy the rules of first normal form.
- An entity that satisfies the rules of third normal form also satisfies the rules of first and second normal forms.

NORMALIZATION

UN-NORMALIZED TABLE

Empno	Name	Work Dept	Dept name	Skill 1	Skill 2	Skill 3	Skill n
030	SAM	C01	MKET	141			
250	TOM	D21	PURCH	002	011	067	
270	TIM	E11	HR	415	447		
300	SID	D21	PURCH	011	032		

NORMALIZATION I - NORMAL FORM

Eliminate Repeating Groups.

EMPNO	NAME	WORK DEPT	DEPTN AME			
				EMPNO	SKILLNO	SKILLDESC
				030	141	RESEARCH
030	SAM	C01	MKET	250	002	BIO PREP
250	TOM	D21	PURCH	250	011	NEGOTIATION
270	TIM	E11	HR	250	067	PRODUCT SPEC
300	SID	D21	PURCH	270	415	BENEFITS ANALYSIS
				270	447	TRAINING
				300	011	NEGO TATION
				300	032	INVENTORY CONTROL

EMP TABLE

SKILL SET TABLE
REPEATED GROUPS

NORMALIZATION

II - NORMAL FORM

Eliminate all columns that depend only on part of the key.

EMP NO	SKILL NO	SKILLDESC
030	141	RESEARCH
250	002	BIO PREP
250	011	NEGOTATION
250	067	PRODUCT SPEC
270	415	BENEFITS ANALYSIS
270	447	TESTING
300	011	NEGOTATION
300	032	INVENTORY CONTROL

Skill table

Skill description depends on skill no. and not the empno, which is the primary key

Skill description table

NORMALIZATION

II - NORMAL FORM

Eliminate columns that is not dependent on the key at all.

EMPNO PK	NAME	WORKDEPT	DEPTNAME
030	SAM	C01	MARKET
250	TOM	D21	PURCH
270	TIM	E11	HR
300	SID	D21	PURCH

EMP table

DEPTNAME does not depend on EMPNO(PK)

DEPT table

DENORMALIZATION

Denormalization is the opposite of normalization process.

It is the process of putting one fact in many places to enhance the speed of data retrieval at the expense of data modification.

This is not necessarily a bad decision, but should be undertaken only when a completely normalized design will not perform optimally.

DENORMALIZATION

Issues to be considered before denormalizing:

1. Can the system achieve acceptable performance without denormalizing?
2. Will denormalization render the database design unusable for ad hoc queries (that is, specialized expertise required to code queries against the denormalized design)?
3. Will the performance of the system still be unacceptable after denormalizing?
4. Will the system be less reliable due to denormalization?

If the answer to any of these questions is "yes," you should not denormalize your tables because the benefit will not be much.

DB2 DATABASES ARE RELATIONAL

A relational database is a database that is perceived by its users as a collection of tables (related as well as unrelated).

Empid	Ename	Dno fk	Projectid fk
01	John	20	P01
02	Peter	10	P02
03	Walter	30	P04
04	tommy	10	p01

ON Delete rules.

Defined in the child table.

Will be activated when deletion happens in the Parent table.

1. cascade-delete those depending employees.
2. Set null – the depending column
3. Restrict – deletion IS STOPPED from the parent table, if there are depending records in the child table.

pid -pk	client	pm
P01	icici	Tom
P02	hdfc	Jim
p04	hsbc	Jerry

Did -pk	Dname	HOD
10	Mkt	Tom
20	Sales	Jim
30	Dev	Jerry
40	Accounts	Harris

DB2 DATABASES ARE RELATIONAL

Empid	Ename	Dno fk	Projectid fk
01	John	20	P01
02	Peter	10	P02
03	Walter	30	P04
04	tommy	10	p01

pid -pk	client	Ploc fk
P01	icici	india
P02	hdfc	us
p04	hsbc	uk

Ploc pk	state	phone
india	Tamil nadu	4565432132
Us	Texas	457878789
uk	Welsh	78787878

WHAT IS RELATION?

“ Relation” is just a mathematical term for a table.

Relational systems are based on what is called the relational model of data

The principles of the relational model introduced by Dr. E F Codd – IBM researcher.

All the RDBMS follows CODD's Relational rules
12 rules

CODD'S 12 RULES.

- Information rule
- Guaranteed Access rule
- Systematic treatment of NULLs rule
- Active on-line catalog based on relational model
- Comprehensive data sub-language rule
- View Updating rule

CODD'S 12 RULES.

(CONT ...)

- High-level Insert, Update and Delete
- Physical data independence
- Logical data independence
- Integrity independence
- Distribution Independence
- Non-subversion rule

OTHER DATABASES

Apart from Relational Data Base, We have

- Hierarchical Data Base and
- Network Database

HIERARCHICAL

- Top down structure resembling an upside-down tree
- Parent child relationship
- First logical database model
- Available on most of the Mainframe computers
- Example – IMS

NETWORK

- Does not distinguish between parent and child. Any record type can be associated with any number of arbitrary record types
- Example - IDMS

DATA OBJECTS

Stogroup, tablespace, database

Sec mem volumes. Alwyn1,alwyn2,alwyn3,alwyn4

Storage Group

It's a set of volume on single DASD → hdd

Create stogroup stg1 using alwyn1,alwyn2.

Partitioned storage group is table space.

Temp buffer → bufferpools.

bp01,bp02

Create database shrdb4 using stg1,bp01,bp02;

Database

It contains a stogroup & bufferpool

Create tabelemp (in char(03)) in shrdb4.shrts4;

Tablespace

Logical address space on secondary storage
to hold one or more tables

A ‘SPACE’ is basically an extendable collection of pages
with each page of size 4K or 32K bytes. (K=1024)

Three Type of Tablespace –

1. Simple: Can contain more than one table. Usually only one is preferred.
2. Partitioned :
 - (a)The table space is divided into equal partitions.
 - (b)The entire table space can have only one table.
 - (c)Utilities can be run on one partition.
3. Segmented : The table space is divided into equal segments.
Each Segment can have only one table.

RDBMS Terminologies

- Data Structure
- Data Integrity

RDBMS Terminologies – Data Structures

Table – Relation

Record – Tuple/Row

Attribute – Field/Column

Domain : A range of values (pool) of same column.

Entity : some object about which we wish to store information. Employee is the entity for a table in which information about him is stored.

NULL : represents an unknown /empty value.

Atomic Value : smallest unit of data ; the individual data value.

RDBMS Terminology : Data structures

Degree - The number of attribute in a relation is called the degree of the relation.
A relation of degree two is called binary.

Cardinality - The number of tuples or rows in a relation is called the cardinality of the relation.

The Cardinality of the relation keeps changing with time as more and more tuples get added or deleted, but the degree does not.
(Substantiate the above statement)

RDBMS Terminology : Data Integrity

- Entity Constraints
 - Referential constraints
 - NULL Constraints

ENTITY CONSTRAINTS

- Constraints(Conditions/Rules) can be written for a column while creating a table.
- The value that does not satisfy the constraints will not be accepted while inserting a record in to a table.
 - CHECK
 - UNIQUE

ENTITY CONSTRAINTS

■ CHECK

- any condition can be written along with the attribute definition.

EX.

```
CREATE TABLE EMP  
  ( ENO NUM NOT NULL,  
    AGE NUM CHECK (AGE > 30))  
IN DB.TS
```

While inserting a record with age =20, the system will not allow the record to be inserted because the check condition says that age must be greater than 30

ENTITY CONSTRAINTS

■ UNIQUE

- This is written along with table creation.
- It just ensures that INSERT operation is successful only if that particular attribute has a UNIQUE value.

```
- CREATE TABLE EMP  
  (ENO ...,  
   TEL NUM UNIQUE)  
IN DB.TS
```

The above SQL statement will never allow duplicate values to be inserted for TEL column.

RDBMS TERMINOLOGY : REFERENTIAL CONSTRAINS

Candidate key :
set of attributes which uniquely identify the row

Primary Key :
is an attribute of user's choice from the candidate key. It provides a way to identify each row in the table.

Alternate key :
Remaining candidate keys that were not chosen as primary key.

RDBMS TERMINOLOGY : REFERENTIAL INTEGRITY

Foreign key

is column(s) in a table whose values have matching values in primary key columns in another table, providing a way to establish relationship between tables.

Referential Constraint

The limiting of a set of foreign key values (of a child table) to a set of primary key values (of its corresponding parent table).

Referential Integrity

The automatic enforcement of referential constraints.

EMPLOYEE

Emp No (PK)	Emp Name	Dept NO (FK)	Salary
001	John	01	85654
002	Peter	02	56322
003	Walter	01	85654

DEPT			Address			
Dept No(PK)	Dept Name	Location(FK)	Location (PK)	Address 1	Address 2	Address 3
01	Sales	Chennai	Chennai	Infoville	Perungudi	600034
02	Account	Delhi	Delhi	Techno	New delhi	110236
03	Mkt	Pune	Mumbai	Mofin	Thane	300326
04	Training	Mumbai	Pune	Surya plaza	Navipeth	311042

RDBMS TERMINOLOGY : REFERENTIAL INTEGRITY

- The table which has a FOREIGN KEY definition is the CHILD/ DERIVED table. Eg. EMPLOYEE, DEPT.
- The corresponding tables in which these FOREIGN KEY attributes are declared as PRIMARY KEY are called PARENT/BASE table. Eg. DEPT is the parent table for EMPLOYEE table and LOCATION table is the parent table of DEPT table.
- DEPT table is both, a child table as well as a parent table for two different tables.
- LOCATION table is BASE/PARENT only.
- EMPLOYEE table is CHILD/DERIVED table only.

RDBMS TERMINOLOGY : REFERENTIAL INTEGRITY

Every foreign key value must have a matching primary key value.

An insert into a foreign key table cannot violate referential integrity.

Deletion of a primary key value cannot violate referential integrity.

RDBMS TERMINOLOGY : REFERENTIAL INTEGRITY

The insertion of any given foreign key value or an update to that value (in child table) is allowed only if the matching value exists in the primary key (of the corresponding parent table).

Changes in primary key values are allowed only for those values that do not have matching foreign key values.

ON DELETE RULES

1. SET NULL
2. CASCADE
3. RESTRICT

ON DELETE RULES

Deletion of a primary key (in parent table) value when corresponding foreign key (in child table) value exists

Will ::

1. set the corresponding foreign key values to null if the constraint has been specified as SET NULL.
2. cause deletion of the corresponding rows containing foreign key if the constraint has been specified as CASCADE.
3. be barred if the foreign key constraint has been specified as RESTRICT.(deletion operation will not be allowed)

RDBMS TERMINOLOGY : NULL CONSTRAINTS

Missing or Unknown information can be represented in RDBMS through a NULL value.

NULL is neither zero nor spaces but just that the value is unknown.

DB2 automatically moves the following values

Zero (0) for Numeric Field

Blanks (spaces) for Alphanumeric or Alpha

RDBMS TERMINOLOGY : NULL CONSTRAINTS

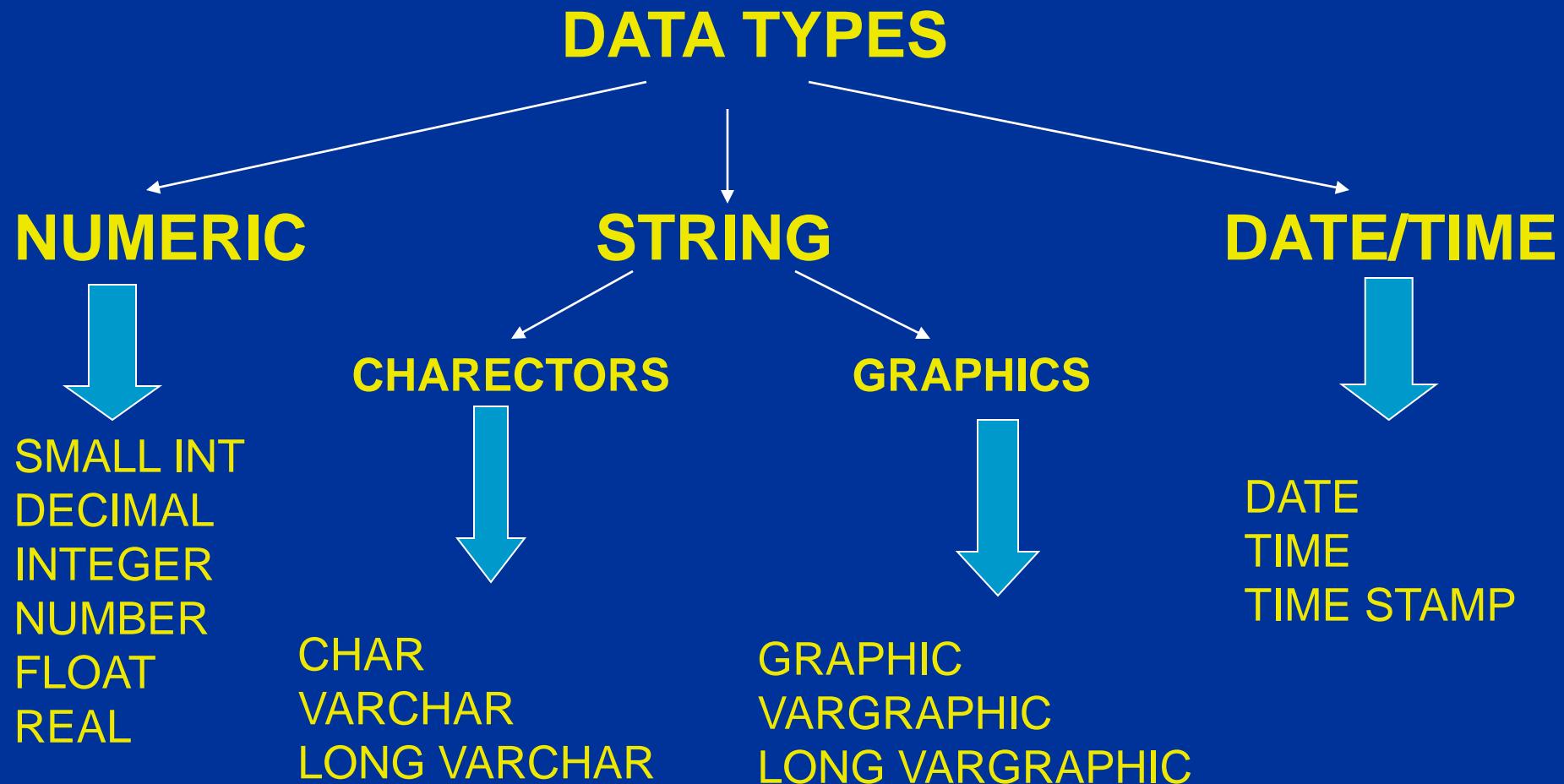
Allow NULL

NOT NULL

NOT NULL WITH DEFAULT

NOT NULL WITH DEFAULT “ XYZ”

DB2 - DATA TYPES



DB2 – DATA TYPES NUMERIC

- **Char (n)** $1 < n < 254$ bytes
- **Varchar(n)** variable length
- **Smallint** 2 bytes +- 32 K
- **Integer** 4 bytes +- 2 GB
- **Decimal(x,y)** max 31 digits
- **Date** 0001-01-01
- **Time** 00:00:00
- **Timestamp** 0001-01-01-00.00.000000

DB2 – DATA TYPES

STRING

- CHAR(n)
- CHARACTER(n) - Fixed –length character string. N is the number of characters.
- VARCHAR(n) - Variable-length character string. N is the maximum number of characters.
- GRAPHI(n) - Fixed – length graphic string. N is the number of characters.
- CARDGRAPHIC(n) - Variable-length graphic string. N is the maximum number of characters.

DB2 – DATA TYPES

DATE/TIME

- DATE – Designates a point in time according to the Gregorian calendar.
- TIME -- Designates a time of day according to a 24 hours clock. HH:MM:SS: mS:mS.
- TIME STAMP – Designates TIME and DATE

STRUCTURED QUERY LANGUAGE

Introduction

Characteristics

Components and

Data Types

SQL INTRODUCTION (STRUCTURED QUERY LANGUAGE)

- Consists of set of facilities for defining, accessing and managing relational databases.
- SQL can be invoked either interactively(spufi,qmf), or by embedding them in application programs(cobol).
- Very flexible

ADVANTAGES OF SQL

- Provides greater degree of abstraction than procedural languages.
- Enables end-users to deal with multiple number of databases at the same time.
- Specifies what is required and not how it should be done.

PURPOSE AND USE OF SQL

- Query
- Definition of data structures
- Controlling access to the data
- Delete/Modify occurrences of the data.

DB2 COMPONENTS

- DDL Data Definition Language
- DML Data Manipulation Language
- DCL Data Control Language
- TCL Transaction Control Language
- DQL DATA QUERY LANGUAGE
SELECT

DDL – DATA DEFINITION LANGUAGE

- Create – Create schema objects
- Alter – Alter Schema objects
- Drop – Delete schema objects.
- Rename – Rename the schema object.
- TRUNCATE – TO EMPTY THE TABLE.
- Schema Object: Tables, Views, Indexes, Triggers Etc.,

DML – DATA MANIPULATION LANGUAGE

- Insert – Add new rows into the table
- Update – Modify values in the existing rows
- Delete – Removing already existing rows
- Select – Retrieving records from table.

DCL – DATA CONTROL LANGUAGE

- Grant – Grant privileges and roles
- Revoke – Take away privileges.

TCL – TRANSACTION CONTROL LANGUAGE

- COMMIT – Saves all the transaction till that point where Commit command is given
- ROLLBACK – It is like UNDO command. It saves only till the previous commit

INTERACTING DB2 WITH SPUFI

- Now we will discuss about how to communicate with DB2 with SPUFI
- In ISPF panel choose option 8 for DB2 as follows

How to execute SQL with SPUFI

Step 1

Login to TSO region. Create a PDS for all the DB2 source codes and a member in it for Input data set. Create a PS for the outputs (output data set) of the SQL statements to be displayed.

Step 2

Select 8.1 option in ISPF main menu.

Step 3

Specify the Input dataset and Output Dataset for the SQL Query and result of the Query.

Step 4

Now Give the SQL query for executing.

Step 5

Type END;;; or press F3 for execution of the query.

Connect Close Exit Edit Print Screen Setup Help

----- ISPF/PDF PRIMARY OPTION MENU -----

OPTION ==> 8

- | | | |
|----------------|---|------------------------------------|
| 0 ISPF PARMS | - Specify terminal and user parameters | * PREFIX -MAPLE44
* SYSTEM ID - |
| 1 BROWSE | - Display source data or output listings | |
| 2 EDIT | - Create or change source data | |
| 3 UTILITIES | - Perform utility functions | |
| 4 FOREGROUND | - Invoke language processors in foreground | |
| 5 BATCH | - Submit job for language processing | |
| 6 COMMAND | - Enter TSO command or CLIST | |
| 7 DIALOG TEST | - Perform dialog testing | |
| 8 DB2 | - Perform DATABASE 2 interactive functions | |
| 9 IBM Products | - IBM program development products | |
| QMF QMF | - QMF QUERY MANAGEMENT FACILITY | |
| SD SDSF | - System Display And Search Facility | |
| C CHANGES | - Display summary of changes for this release | |
| T TUTORIAL | - Display information about ISPF/PDF | |
| X EXIT | - Terminate ISPF using log and list defaults | |

Select option
8 for DB2

Enter END command to terminate ISPF.

F1=HELP
F7=UPF2=SPLIT
F8=DOWNF3=END
F9=SWAPF4=RETURN
F10=LEFTF5=RFIND
F11=RIGHTF6=RCHANGE
F12=RETRIEVE

14:03:44 IBM-3278-4

Clear

Erase EOF

New Line

PA1

PA2

PA3

Connect Close Exit Edit Print Screen Setup Help

DB2I PRIMARY OPTION MENU

CSTD - DSND

COMMAND ==> ■

Select one of the following

- | | |
|-----------------------|----------------------------------|
| 1 SPUFI | (Process SQL) |
| 2 DCLGEN | (Generate SQL and PL/I) |
| 3 PROGRAM PREPARATION | (Prepare a DB2 application) |
| 4 PRECOMPILE | (Invoke DB2 precompiler) |
| 5 BIND/REBIND/FREE | (BIND, REBIND, or FREE programs) |
| 6 RUN | (RUN an SQL program) |
| 7 DB2 COMMANDS | (Issue DB2 commands) |
| 8 UTILITIES | (Invoke DB2 utilities) |
| D DB2I DEFAULTS | (Set global parameters) |
| X EXIT | (Leave DB2I) |

Select option 1
for SPUFI
(or)
Type 8.1 option
directly in main
menu

PRESS:

END to exit

HELP for more information

F1=HELP
F7=UPF2=SPLIT
F8=DOWNF3=END
F9=SWAPF4=RETURN
F10=LEFTF5=RFIND
F11=RIGHT
INSERTF6=RCHANGE
F12=RETRIEVE
14:05:14 IBM-3278-4

Clear

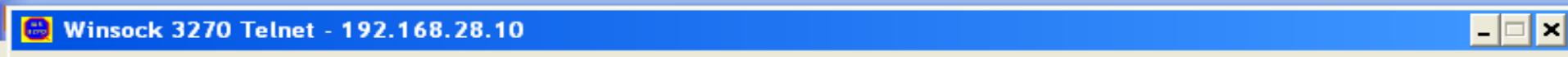
Erase EOF

New Line

PA1

PA2

PA3



Connect Close Exit Edit Print Screen Setup Help

SPUFI

SSID: DSN2

====>

Enter the input data set name: (Can be sequential or partitioned)

1 DATA SET NAME ... ==> 'MAPLE44.DB2.INPUT(UDFS)'

2 VOLUME SERIAL ... ==> (Enter if not cataloged)

3 DATA SET PASSWORD ==> (Enter if password protected)

Enter the output data set name: (Must be a sequential data set)

4 DATA SET NAME ... ==> MAPLE44.DB2.OUTPUT'

Specify processing options

5 CHANGE DEFAULTS === YES (Y/N - Display SPUFI defaults panel?)

6 EDIT INPUT YES (Y/N - Enter SQL statements?)

7 EXECUTE YES (Y/N - Execute SQL statements?)

8 AUTOCOMMIT YES (Y/N - Commit after successful run?)

9 BROWSE OUTPUT ... = YES (Y/N - Browse output data set?)

For remote SQL process:

10 CONNECT LOCATION >

PRESS: ENTER to pro

END to exit

HELP for more information

Type the Input
dataset name
(PDS Data Set)

Type the output dataset
name (PS dataset)

F1=HELP
F7=UP

F2=SPLIT
F8=DOWN

F3=END
F9=SWAP

F4=RETURN
F10=LEFT

F5=R FIND
F11=RIGHT

F6=RCHANGE
F12=RETRIEVE

14:06:59 IBM-3278-4

Clear

Erase EOF

New Line

PA1

PA2

PA3



Connect Close Exit Edit Print Screen Setup Help

CURRENT SPUFI DEFAULTS

SSID: DSN2

====> █

Enter the following to control your SPUFI session:

1 SQL TERMINATOR .. ===> ; (SQL Statement)
2 ISOLATION LEVEL .. ===> RR (RR=Repeatable)
3 MAX SELECT LINES .. ===> 250 (Maximum number)

Give the settings as per our requirement then press enter

Output data set characteristics:

4 RECORD LENGTH ... ===> 4092 (LRECL=Logical record length)
5 BLOCK SIZE ===> 4096 (Size of one block)
6 RECORD FORMAT ... ===> VB (RECFM=F, FB, FBA, V, VB, or VBA)
7 DEVICE TYPE ===> SYSDA (Must be DASD unit name)

Output format characteristics:

8 MAX NUMERIC FIELD ===> 33 (Maximum width for numeric fields)
9 MAX CHAR FIELD .. ===> 80 (Maximum width for character fields)
10 COLUMN HEADING .. ===> NAMES (NAMES, LABELS, ANY or BOTH)

PRESS: ENTER to process

END to exit

HELP for more information

F1=HELP
F7=UP

F2=SPLIT
F8=DOWN

F3=END
F9=SWAP

F4=RETURN
F10=LEFT

F5=RFIND
F11=RIGHT

F6=RCHANGE
F12=RETRIEVE

14:11:03 IBM-3278-4

Clear

Erase EOF

New Line

PA1

PA2

PA3

Connect Close Exit Edit Print Screen Setup Help

File Edit Edit_Settings Menu Utilities Compilers Test Help

EDIT MAPLE44.DB2.INPUT(MEM1) - 01.00

Columns 00001 00072

Command ==> end;;;

Scroll ==> PAGE

000100 SELECT * FROM EMPLOYEE;

000200 SELECT * FROM EMP123;

***** Top of Data *****

***** Bottom of Data *****

Give the queries to be
executed

Type END;; for
execution of
these queries

F1=Help
F7=UpF2=refss
F8=DownF3=Exit
F9=SwapF4=save;sub
F10=LeftF5=end
F11=RightF6=start sd
F12=Cancel

14:13:33 IBM-3278-4

Clear

Erase EOF

New Line

PA1

PA2

PA3

Connect Close Exit Edit Print Screen Setup Help

Menu Utilities Compilers Help

BROWSE MAPLE44.DB2.OUTPUT

Line 00000000 Col 001 080

Command ==> end;;■

Scroll ==> 0001

***** Top of Data *****

+-----+-----+-----+-----+-----+-----+-----+

SELECT * FROM EMPLOYEE; 00010000

+-----+-----+-----+-----+-----+-----+-----+

EMPNO FIRSTNAME MIDINIT LASTNAME WORKDEPT PHONENO HI

+-----+-----+-----+-----+-----+-----+-----+

E00101 KUMAR L RAMAN A00 3456 19

DSNE610I NUMBER OF ROWS DISPLAYED IS 1

DSNE616I STATEMENT EXECUTION WAS SUCCESSFUL, SQLCODE IS 100

+-----+-----+-----+-----+-----+-----+-----+

SELECT * FROM EMP123; 00020000

+-----+-----+-----+-----+-----+-----+-----+

ENO SAL

+-----+-----+-----+-----+-----+-----+-----+

00E001 1234.67

00E002 2345.67

00E003 3445.67

00E004 2345.67

00E005 7245.67

DSNE610I NUMBER OF ROWS DISPLAYED IS 5

DSNE616I STATEMENT EXECUTION WAS SUCCESSFUL, SQLCODE IS 100

+-----+-----+-----+-----+-----+-----+-----+

DSNE617I COMMIT PERFORMED, SQLCODE IS 0

DSNE616I STATEMENT EXECUTION WAS SUCCESSFUL, SQLCODE IS 0

+-----+-----+-----+-----+-----+-----+-----+

DSNE601I SQL STATEMENTS ASSUMED TO BE BETWEEN COLUMNS 1 AND 72

DSNE620I NUMBER OF SQL STATEMENTS PROCESSED IS 2

DSNE621I NUMBER OF INPUT RECORDS READ IS 2

DSNE622I NUMBER OF OUTPUT RECORDS WRITTEN IS 29

***** Bottom of Data *****

F1=Help
F7=UpF2=end
F8=DownF3=Exit
F9=SwapF4=ref
F10=LeftF5=autotype F6=start
F11=Right F12=Cancel
14:16:43 IBM-3278-4

Clear

Erase EOF

New Line

PA1

PA2

PA3

SQLCODE: the system will throw a Numeric code that reflects the status of the SQL statement that is executed.

The SQLCODE will be negative for ERRORS.

The SQLCODE will be positive for Warnings and information.

The SQLCODE will be zero for successful execution.