

DATA DICTIONARY

IN ABAP (OR) SAP

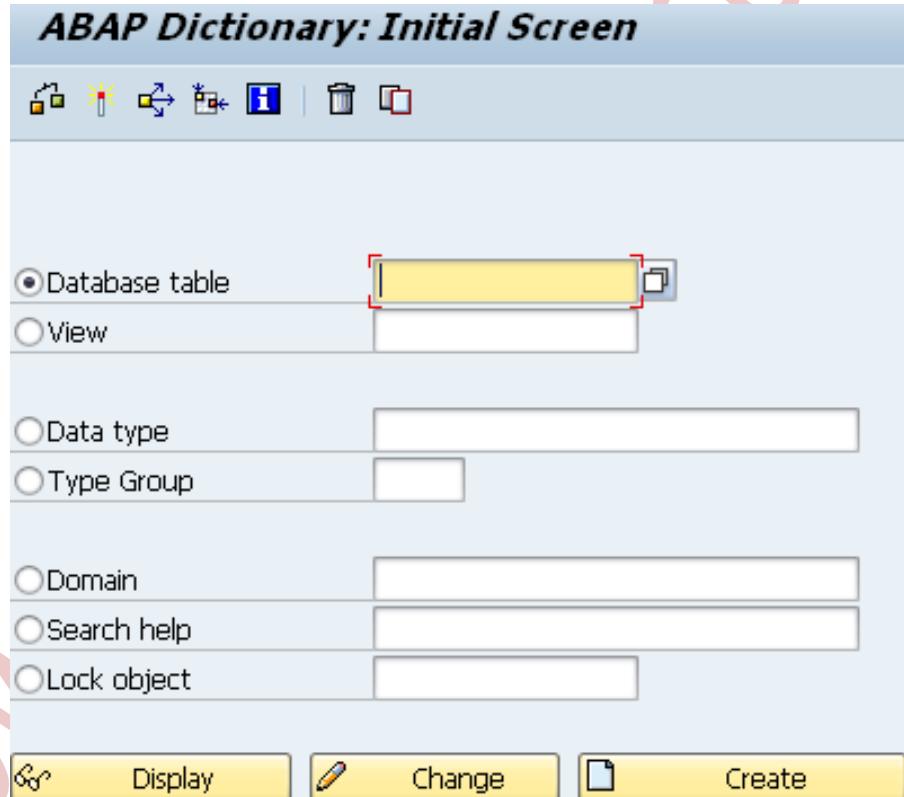
DATA DICTIONARY: (CONTAINS GLOBAL DEFINITIONS)

A DATA DICTIONARY IS GLOBAL REPOSITORY OF INFORMATION IN ABAP. IT CONTAINS ALL THE GLOBAL DEFINITIONS THAT CAN BE USED IN ANY APPLICATION OF SAP.

COMPONENTS OF DATA DICTIONARY (WHICH CONTAINS GLOBAL DEFINITIONS)

TRANSACTION CODE: **SE11**

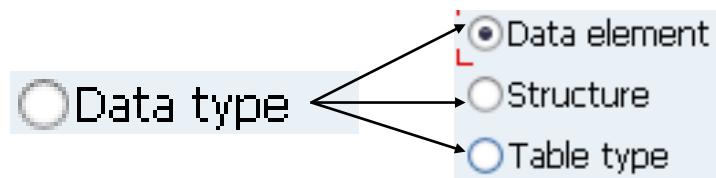
THE FOLLOWING OBJECTS
ARE MAINTAINED.



(OR)

DATA DICTIONARY: DICTIONARY IS A PLACE WHERE WE HAVE VOCABULARY

IT IS A CENTRAL REPOSITORY WHERE WE MAINTAIN (CREATE, CHANGE, DELETE) THE OBJECTS WHICH ARE RELATED TO DATABASE.



DATA DICTIONARY IS AN INTEGRAL COMPONENT OF APPLICATION LAYER

NOTE:

- TABLE IS ALWAYS CLIENT INDEPENDENT. IF A TABLE IS CREATED IN ONE CLIENT, THE TABLE IS AUTOMATICALLY GENERATED IN ALL THE CLIENTS WITHIN THE INSTANCE.
- DATA WHAT WE MAINTAIN IN TABLE IS CLIENT-DEPENDENT, MEANING, IF THE DATA IS MAINTAINED IN ONE CLIENT THEN THE DATA IS PRESENT ONLY IN THAT CLIENT.
- DATA (IS OF TWO TYPES) CAN BE MAINTAINED EITHER IN CLIENT-DEPENDENT AND CLIENT INDEPENDENT.
- THE CLIENT DEPENDENT DATA IS SAVED IN A TABLE WHICH HAS THE FIRST FIELD AS 'MANDT' (CLIENT ID).
- THE CLIENT INDEPENDENT DATA IS SAVED IN A TABLE WHICH DOES NOT HAVE FIRST FIELD AS 'MANDT'
- APPLICATION DATA IS A CLIENT DEPENDENT DATA. HENCE THE FIRST FIELD IN APPLICATION DATA BASE TABLE SHOULD BE 'MANDT' AND PART OF PRIMARY KEY.
- WHILE MAINTAINING THE DATA IN DATABASE TABLES THE VALUE FOR 'MANDT' WILL BE AUTOMATICALLY POPULATED BY THE SYSTEM.
- TABLE HAS DEFINITION PART -> IT IS WHERE DEFINITIONS ARE CREATED.
- TABLE HAS MEMORY PART -> IT IS WHERE DATA IS STORED IN ROWS AND COLUMNS
- TABLE IS GROUP OF RELATED ENTITY CALLED AS FIELDS. THIS IS CALLED AS DEFINITION. NO DATA COMES IN TO PICTURE.
- PROPERTIES FOR FIELD IS DOMAIN AND DATAELEMENT

DATA DICTIONARY DEFINITION:

DATA DICTIONARY IS A VIRTUAL DATABASE, BECAUSE FOR ALL THE DEFINITIONS IN DATA DICTIONARY THE MEMORY IS ALLOCATED IN THE DATABASE.

(OR)

ABAP DICTIONARY CENTRALLY DESCRIBES AND MANAGES ALL THE DATA DEFINITIONS USED IN THE SYSTEM. THE ABAP DICTIONARY IS COMPLETELY INTEGRATED IN THE ABAP WORK BENCH. ALL THE OTHER COMPONENTS OF THE WORKBENCH CAN ACTIVELY ACCESS THE DEFINITIONS STORED IN ABAP DICTIONARY.

THE ABAP DICTIONARY SUPPORTS THE DEFINITION OF USER-DEFINED TYPES (DATA ELEMENTS, STRUCTURE AND TABLE TYPES). YOU CAN ALSO DEFINE THE STRUCTURE OF DATABASE OBJECTS (TABLES, INDEXES AND VIEWS) IN ABAP DICTIONARY. THESE OBJECTS CAN BE AUTOMATICALLY CREATED IN THE DATABASE WITH THESE DEFINITIONS.

NOTE:

- DATA DICTIONARY WILL BE THERE IN APPLICATION LAYER.
 - DEFINITIONS FOR TABLE (APPLICATION LAYER)
- DEFINITION OF THE TABLE IS DEFINED IN DATA DICTIONARY OF APPLICATION LAYER.
- THE DB TABLE HAS TWO MEANINGS.

FROM DATA DICTIONARY POINT OF VIEW, THE DB TABLE IS A GROUP OF RELATED ENTITIES CALLED AS FIELDS.

DEFINITION EX:

STABLE
SNO
SNAME
SPLACE
SBRANCH

FROM DATABASE POINT OF VIEW, DB TABLE IS A STORAGE OF DATA IN ROWS AND COLUMNS

EG:

SNO	SNAME	SPLACE	SBRANCH

DATA DICTIONARY IS CALLED AS VIRTUAL DATABASE.

TABLE IS CREATED IN DATA DICTIONARY OF APPLICATION LAYER AND MEMORY IS ALLOCATED FOR IT IN DATABASE LAYER.

DATABASE TABLE:

- ✓ FROM DATA DICTIONARY POINT OF VIEW A TABLE IS A GROUP OF RELATED ENTITIES CALLED AS FIELDS. EACH FIELD IN THE TABLE DEFINITION CONTAINS CERTAIN PROPERTIES LIKE FIELD NAME, FIELD DESCRIPTION, DATA TYPE, FIELD LENGTH ETC.,
 - ✓ FROM DATA BASE POINT OF VIEW, A DATABASE TABLE IS THE DATA ARRANGED OR SAVED IN ROWS AND COLUMNS.

DEFINITION:

- A DATABASE TABLE IS A COLLECTION OF ROWS AND COLUMNS TO STORE BUSINESS DATA.
 - EACH ROW IS CALLED AS RECORD.
 - EACH COLUMN IS CALLED AS A FIELD.

THE PROPERTIES OF A FIELD IN A TABLE ARE CATEGORISED AS:

- ## ➤ DATA ELEMENT AND DOMAIN

HOW TO DEFINE A TABLE?

- TO DEFINE A TABLE WE NEED TO DEFINE THE FIELDS FIRST.
 - TO DEFINE A FIELD WE NEED TO SPECIFY

1. WHAT IS THE DATA TYPE? }
2. WHAT IS THE LENGTH? }

CALLED AS DOMAIN

3. WHAT IS THE DESCRIPTION? } CALLED AS DATA ELEMENT

EG: KUNNR IS A FIELD



- IN SAP THE TABLES ARE CREATED USING THE DATA ELEMENTS AND DOMAIN CONCEPTS.
 - IN SAP, A FIELD IS DEFINED USING DATA ELEMENTS, DOMAIN

FIELD = DATA ELEMENT + DOMAIN

DATA ELEMENT: THE DATA ELEMENT GIVES THE MEANING OF A FIELD, NOTHING BUT DESCRIPTION OF A FIELD IN A TABLE.

(OR)

IT IS AN OBJECT WHICH SPECIFIES THE SEMANTIC INFORMATION SUCH AS FIELD DESCRIPTION, FIELD LABEL FOR A FIELD.

THE DATA ELEMENT IS INTERLINKED WITH DOMAIN.

DOMAIN: THE DOMAIN GIVES YOU THE TECHNICAL ATTRIBUTES OF A FIELD IN A TABLE.

(OR)

IT IS AN OBJECT WHICH SPECIFIES THE TECHNICAL INFORMATION SUCH AS DATA TYPE, LENGTH FOR A FIELD. AND ALSO DATA TYPE, LENGTH, FIXED VALUES, VALUE TABLE.

DOMAIN IS ASSIGNED TO DATA ELEMENT.

DATA ELEMENT IS ASSIGNED TO FIELD.

WHAT IS DIFFERENCE BETWEEN SYNTAX AND SEMANTIC?

SYNTAX: STRUCTURE

SEMANTIC: IS MEANING OF A STRUCTURE

KEYFIELD: IT IS A SINGLE FIELD OR A GROUP OF FIELDS WHICH ARE USED TO IDENTIFY A SINGLE RECORD UNIQUELY

STEPS TO CREATE A TABLE

- ☞ GO TO SE11.
- ☞ SELECT THE DATABASE TABLE.
- ☞ GIVE NAME AS EG: ZCUST_TABLE
- ☞ CLICK ON CREATE.
- ☞ GIVE DESCRIPTION.
- ☞ GIVE DELIVERY CLASS AS A
- ☞ SPECIFY DISPLAY / MAINTENANCE ALLOWED.
- ☞ CLICK ON FIELD TAB.
- ☞ SPECIFY THE FIELDS AS BELOW

FIELD NAME	KEY	DATA ELEMENT NAME
ZCUSTNO	<input checked="" type="checkbox"/>	ZCUSTNO

CREATING DATA ELEMENT AND DOMAIN

- ☞ DOUBLE CLICKON DATA ELEMENT: ZCUSTNO
- ☞ CLICK ON SAVE
- ☞ CLICK ON YES TO CREATE DATA ELEMENT.
- ☞ GIVE DESCRIPTION AS CUSTOMER NUMBER.
- ☞ GIVE DOMAIN NAME AS ZCUSTNO.

- ☞ DOUBLE CLICK ON ZCUSTNO.
- ☞ CLICK ON SAVE
- ☞ CLICK ON YES
- ☞ SPECIFY DATA TYPE AS **CHAR** AND LENGTH AS **10**.
- ☞ SAVE & ACTIVATE DOMAIN.
- ☞ CLICK BACK.
- ☞ SAVE & ACTIVATE DATA ELEMENT.
- ☞ CLICK ON BACK.
- ☞ SIMILARLY CREATE REMAINING FIELDS WITH DATA ELEMENT AND DOMAIN AS BELOW

FIELD NAME	KEY	DATA ELEMENT NAME
ZCUSTNO	<input checked="" type="checkbox"/>	ZCUSTNO
ZCUSTNAME		ZCUSTNAME
LAND1		ZLAND1

- ☞ FINALLY CLICK ON **TECHNICAL SETTINGS** BUTTON
- ☞ SPECIFY
 - DATA CLASS AS **APPLO**.
 - SIZE CATEGORY AS **0**.
- ☞ SAVE IT.
- ☞ CLICK ON BACK.
- ☞ SAVE AND ACTIVATE TABLE.

CREATING RECORDS INTO THE TABLE:

- ☞ CLICK ON UTILITIES FROM MENU BAR.
- ☞ CLICK ON TABLE CONTENT -> CREATE ENTRIES.
- ☞ SPECIFY CUSTNO, NAME, COUNTRY.
- ☞ CLICK ON SAVE.
- ☞ SIMILARLY CREATE THE RECORDS.

DISPLAY THE TABLE DATA.

- ☞ CLICK ON UTILITIES -> TABLE CONTENTS -> DISPLAY
- ☞ CLICK ON EXECUTE BUTTON.
- ☞ ALL THE RECORDS WILL BE DISPLAYED.

PROPERTIES OF THE TABLE

PROPERTIES OF THE TABLE:

DELIVERY CLASS

IT SPECIFIES THE TYPE OF DATA TO BE STORED IN A DATABASE TABLE.

THE FOLLOWING OPTIONS ARE AVAILABLE.

DELIVERY CLASS	SHORT TEXT
A	APPLICATION TABLE (MASTER & TRANSACTION DATA)
C	CUSTOMIZING TABLE, MAINTENANCE ONLY BY CUSTOMER, NOT SAP IMPORT.
L	TABLE FOR STORING TEMPORARY DATA, DELIVERY EMPTY
G, E, S, W	USED TO STORE SYSTEM DATA. MOSTLY USED BY SAP WE DONT CREATE ANY TABLE OF THIS TYPE.

WHAT IS DELIVERY CLASS?

DELIVERY CLASS SPECIFIES THE TYPE OF DATA THAT IS BEING STORED IN THIS DATA BASE TABLE. ALWAYS THE VALUE SHOULD BE 'A'. A - STANDS FOR APPLICATION DATA WHICH HOLDS EITHER MASTER OR TRANSACTION DATA.

DATA BROWSER / TABLE VIEW MAINTENANCE

DISPLAY / MAINTENANCE:

- ☞ DISPLAY MEANS DISPLAYING THE TABLE DATA.
- ☞ MAINTENANCE MEANS CREATION, MODIFYING, DELETING THE TABLE DATA.
- ☞ THIS OPTION WILL SPECIFY WHETHER THE TABLE DATA CAN BE DISPLAYED OR MAINTAINED.
- ☞ WE HAVE 3 OPTIONS AVAILABLE.

DISPLAY / MAINTENANCE ALLOWED:

- ☞ IF THIS OPTION IS SELECTED WE CAN DISPLAY THE DATA AND WE CAN MAINTAIN THE DATA.

DISPLAY / MAINTENANCE NOT ALLOWED:

- ☞ IF THIS OPTION IS SELECTED, WE CANNOT DISPLAY THE DATA AND WE CANNOT MAINTAIN THE DATA.

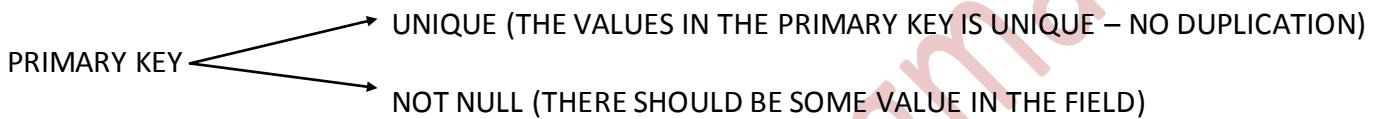
DISPLAY / MAINTENANCE ALLOWED WITH RESTRICTIONS:

- ☞ IF THIS OPTION IS SELECTED, ONLY A GROUP OF PEOPLE WILL HAVE THE AUTHORIZATION TO DISPLAY AND MAINTAIN THE DATA.
- ☞ THE AUTHORIZATION IS GIVE BY BASIS CONSULTANTS.

CAN WE CREATE A TABLE WITH OUT PRIMARY KEY? NO

PRIMARY KEY: (IT IS A SINGLE FIELD OR A GROUP OF FIELDS WHICH ARE USED TO IDENTIFY A SINGLE RECORD UNIQUELY.)

- ❖ IT IS A COMBINATION OF VALUES IN PRIMARY FIELDS. THE VALUE IN THE PRIMARY SHOULD BE UNIQUE AND NOT NULL.
- ❖ THE PROPERTIES UNIQUE AND NOT NULL ARE APPLICABLE TO THE PRIMARY KEY AS A WHOLE BUT NOT TO PRIMAR FIELDS INDEPENDENTLY.
- ✓ PROPERTIES OF PRIMARY KEY AS WHOLE ARE UNIQUE AND NOT NULL.
- ✓ PROPERTIES OF PRIMARY FIELD IS NOT UNIQUE AND NOT NULL



PRIMARY VALUE OF MANDT IS 800.

PRIMARY VALUE OF EMPNO IS ABCD1234

PRIMAY KEY VALUE IS 800ABC1234

PRIMAY KEY VALUE: THE COMBINATION OF VALUES IN PRIMARY FIELDS

PRIMARY KEY VALUE	F1 PRIMARY KEY	F2 PRIMARY KEY	F3 PRIMARY KEY	F4	F5	F6
AAA	A	A	A	B	C	D
AAC	A	A	C	B	O	R
A	A	-	-	P	Q	R
AAC (INVALID RECORD)	A	A	C	B	C	D
NULL (INVALID RECORD)	-	-	-	M	N	O

ALL THE PRIMARY FIELDS OR PRIMARY KEY IN A DATABASE TABLE SHOULD BE FIRST FIELD OF YOUR TABLE.

NOTE: THE ABOVE ARE DEFINTIONS OF TABLE

NOTE: MEMORY PART OF THE TABLE AS FOLLOWS:

POINTS:

- RACKS IN LIBRARY
- TABLE SPACE IN DATABASE

PROPERTIES OF THE TABLE:

TECHNICAL SETTINGS

- DATA CLASS
- SIZE CATEGORY
- BUFFERING
- LOGGING

DATA CLASS: (IT SPECIFIES THE PHYSICAL AREA OF A TABLE IN THE DATA BASE SERVER)

SPECIFIES WHERE EXACTLY THE MEMORY HAS TO BE ALLOCATED FOR THE DB TABLE IN THE DATABASE.

Tables in the ABAP Dictionary

OPTIONS:

There are the following data classes:

APPL0 (master data):

Data that is seldom changed.

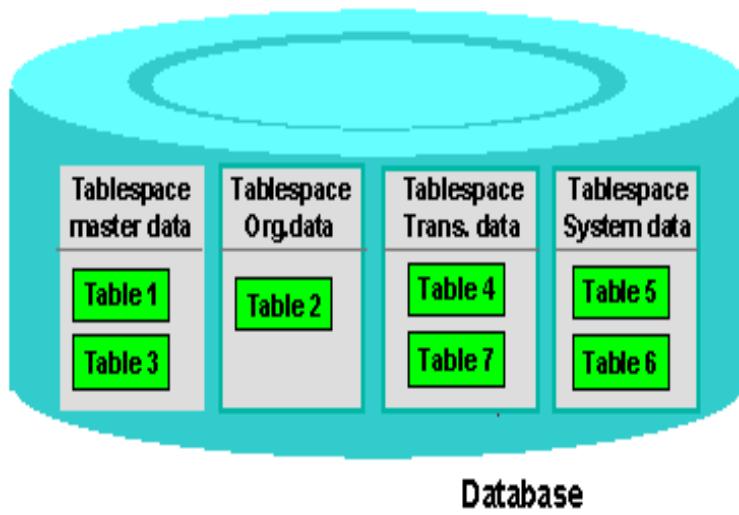
An example of master data is the data contained in an address file, such as the name, address and telephone number.

Master data	Organizational data	Transaction data	System data
Table 1 Table 3	Table 2	Table 4 Table 7	Table 5 Table 6

APPL1 (transaction data):

Data that is frequently changed.

An example of transaction data is the Goods in a warehouse which change after each purchase order.



☞ THE FOLLOWING OPTIONS ARE AVAILABLE.

DATA CLASS	DESCRIPTION
APPL 0	MASTER DATA, TRANSPARENT TABLES
APPL1	TRANSACTION DATA, TRANSPARENT TABLES

☞ DEPENDINGON THE TABLE TYPE I.E, WHETHER IT IS A MASTER TABLE OR TRANSACTION TABLE, WE HAVE TO SELECT EITHER **APPL0, APPL1**.

☞ THE REMAINING OPTIONS ARE USED BY BW CONSULTANTS.

SIZE CATEGORY:

IT SPECIFIES THE MAXIMUM NUMBER OF RECORDS THAT CAN BE STORED IN A TABLE.

(OR)

SPECIFY THE AMOUNT OF MEMORY THAT HAS TO BE ALLOCATED FOR THE TABLE IN THE DATABASE.

THE SIZE CATEGORY SPECIFIES THE AMOUNT OF MEMORY THAT HAS TO BE EXTENDED FOR A TABLE WHEN THE PREVIOUSLY ALLOCATED MEMORY IS FULL.

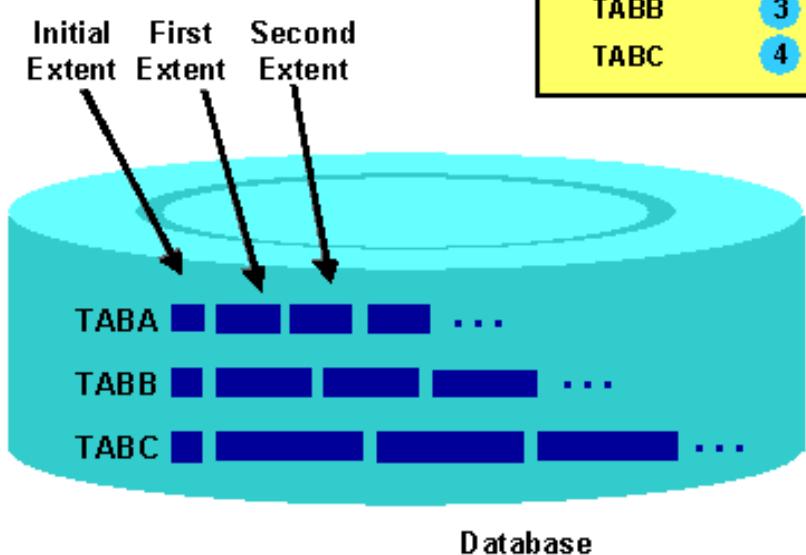
INITIAL MEMORY OF TABLE IS 8KB.

- WE ALWAYS SELECT THE SIZE CATEGORY AS '0' ONLY BECAUSE THE MEMORY WILL BE CREATED FOR SMALL AMOUNT OF DATA.
- IF THE MEMORY IS NOT SUFFICIENT THE SYSTEM WILL INCREASE THE MEMORY AUTOMATICALLY.

SzCat	Number of data records of table expected
0	0 to 240
1	240 to 970
2	970 to 3.900
3	3.900 to 15.000
4	15.000 to 62.000
5	62.000 to 120.000
6	120.000 to 250.000
7	250.000 to 500.000
8	500.000 to 1.000.000

Technical settings

Size category	
TABA	1
TABB	3
TABC	4



BUFFERING:

BUFFER: IS A TEMPORARY MEMORY.

BUFFERING: LOADING INTO MEMORY.

MECHANISM OF LOADING DATA FROM DB TABLE.

BUFFERING IS REQUIRED FOR MASTER DATA.

BUFFER IS ADVANTAGE FOR MASTER DATA.

SELECT LESS SIZE CATEGORY.

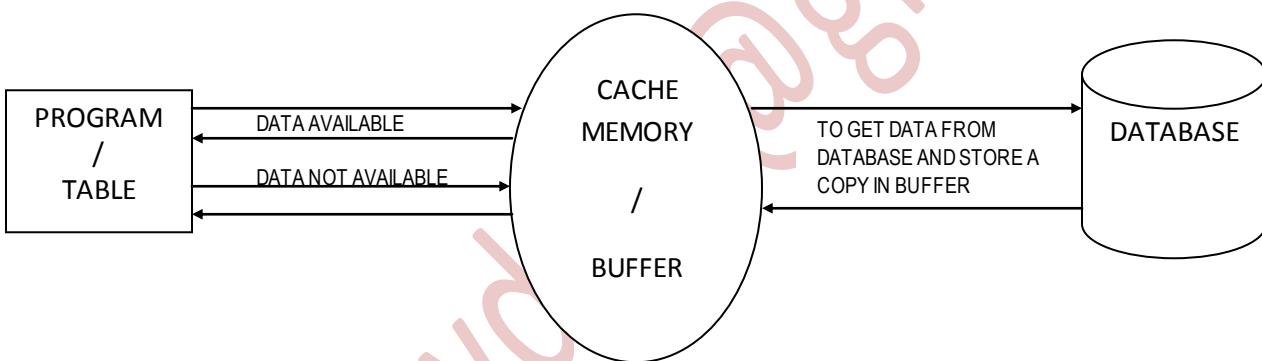
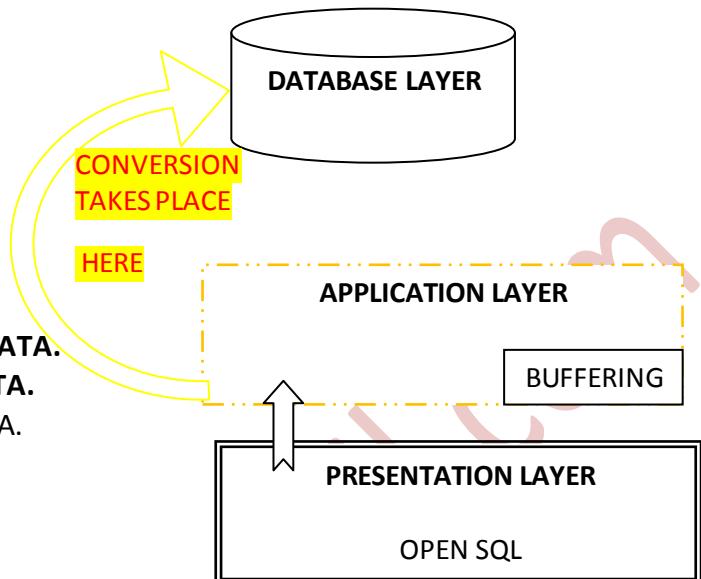
BUFFERING IS NOT REQUIRED FOR TRANSACTION DATA.

BUFFER IS DIS-ADVANTAGE FOR TRANSACTION DATA.

IT IS GREAT DISADVANTAGE FOR TRANSACTION DATA.

TRANSACTION DATA REQUIRE MORE MEMORY

SELECT HIGH SIZE CATEGORY.



- ☞ IT IS TEMPORARY MEMORY TO STORE THE DATA. IT IS ALSO CALLED AS CACHE MEMORY. THE FUNCTIONALITIES OF BUFFER.
- ☞ THE PROGRAM / TABLE CHECK WHETHER THE DATA IS AVAILABLE IN BUFFER MEMORY.
- ☞ IF IT IS AVAILABLE THE DATA FROM CACHE MEMORY IS GIVEN BACK TO THE PROGRAM OR TABLE, THEREBY INCREASING THE PERFORMANCE OF A PROGRAM / TABLE.
- ☞ IF IT IS NOT AVAILABLE THE CACHE MEMORY WILL FORWARD THE REQUEST TO THE DATABASE. THE DATABASE WILL RESPOND WITH THE DATA TO THE CACHE MEMORY. THE CACHE MEMORY WILL RETURN THE DATA TO THE PROGRAM / TABLE BY STORING A COPY IN THE CACHE MEMORY.

BUFFERING PERMISSIONS: (OR) BUFFERING OPTIONS

BUFFERING NOT ALLOWED : DATA IS NOT STORED IN BUFFER. BY DEFAULT THIS IS SELECTED.

BUFFERING ALLOWED BUT SWITCHED OFF

BUFFERING SWITCHED ON

BUFFERING TYPES

SINGLE RECORDS BUFFERING: ONLY THOSE RECORDS THAT WE READ FROM DATABASE TABLE ARE INTO BUFFER.

GENERIC AREA BUFFERED: WHERE ALL THE RECORDS THAT SATISFY THE GENERIC KEY WILL BE CREATED AS A BUFFER.

GENERIC KEY: NUMBER OF KEY FIELDS THAT WE ARE CONSIDERING FOR BUFFERING.

GENERIC KEY = 1 THAT MEANS, CONSIDERING ONLY THE FIRST FIELD. WHICH IS A
BUFFERING WILL BE CREATED FOR THE VALUE OF A
CREATED FOR FOUR RECORDS

GENERIC KEY = 2 THAT MEANS, CONSIDERING ONLY THE FIRST FIELD & SECOND FIELD. WHICH IS AA
BUFFERING WILL BE CREATED FOR THE VALUE OF AA.
CREATED FOR 3 RECORDS

GENERIC KEY = 3 THAT MEANS, CONSIDERING ONLY THE FIRST FIELD, SECOND FIELD & THIRD FIELD. WHICH IS AAA
BUFFERING WILL BE CREATED FOR THE VALUE OF AAA.
CREATED FOR 1 RECORDS

GKV 3	GENERIC KEY VALUES 2	GENERIC KEY VALUES 1	P.K. VALUES	F1 (P.K)	F2 (P.K)	F3 (P.K.)	F4	F5	F6
AAA	VALUES=AA	A	AAA	A	A	A	B	C	D
AAB	AA	A	AAB	A	A	B	C	D	E
APQ	AP	A	APQ	A	P	Q	R	S	T
AAC	AA	A	AAC	A	A	C	O	H	K

1 RECORD 3 RECORD 4 RECORDS = SAME VALUES A, AA, AAA

FULLY BUFFERED: ALL RECORDS WILL BE FETCHED IN TO THE BUFFER. THE MOVEMENT YOU TOUCH THE TABLE, ALL THE RECORDS WILL BE FETCHED FROM DB TABLE TO BUFFER.

WHAT IS THE RELATION BETWEEN FIELD, DOMAIN AND DATA ELEMENT?

DOMAIN IS REFERRED / ASSIGNED TO DATA ELEMENT.

DATA ELEMENT IS ASSIGNED TO FIELD.

CLIENT NUMBER:

A NUMBER WHICH IS USED TO PROVIDE SECURITY FOR THE DATA AT DATABASE LEVEL IS CALLED AS CLIENT NUMBER.

CLIENT-DEPENDENT TABLE:

- A TABLE WHICH STORES THE DATA IN A SPECIFIED CLIENT IS CALLED CLIENT DEPENDENT TABLE.
- THE DATA IS NOT ACCESSABLE FROM OTHER CLIENTS.
- TECHNICALLY IF THE FIRST FIELD IS MANDT, THEN IT IS CALLED AS CLIENT DEPENDENT TABLE.

MANDT:

- IT IS A FIELD WHICH IS USED TO STORE THE CLIENT NUMBER.
- THE DATA ELEMENT AND DOMAIN IS ALSO MANDT.

CLIENT-INDEPENDENT TABLE:

A TABLE WHICH STORES THE DATA IN ALL THE CLIENTS IS CALLED CLIENT INDEPENDENT TABLE.
TECHNICALLY IF THE FIRST FIELD ISNOT MANDT, THEN IT IS CALLED AS CLIENT INDEPENDENT TABLE.

NOTE:

IN THE REAL TIME 95% OF TABLES WHAT WE CREATE ARE CLIENT DEPENDENT TABLES.

ADVANGES OF DATA ELEMENTS AND DOMAINS:

1. THE SAME DATA ELEMENT AND DOMAINS CAN BE RE-USSED IN MULTIPLE TABLES FOR DEFINING THE FIELDS.
2. THE DATA ELEMENTS AND DOMAINS ARE USED IN CREATING FOREIGN KEY RELATIONSHIPS.
3. THE SAME DATAELEMENTS AND DOMAINS ARE USED IN CREATING SEARCH HELPS.
4. THE SAME DATA ELEMENTS AND DOMAINS ARE REUSED IN CROSS APPLICATIONS FOR ALE, IDOC'S.

PROPERTIES OF DOMAIN

SIGN:

- IT IS A CHECKBOX WHICH IS USED TO STORE THE SIGN OF A NUMBER.
- IF THIS OPTION IS SELECTED WE CAN STORE NEGATIVE SIGN FOR A NUMBER.
- IF IT IS NOT SELECTED WE CANNOT STORE THE SIGN AND BY DEFAULT IT IS POSITIVE NUMBER
- EG: DOMAIN WERTV8 STORES NEGATIVE NUMBER.

LOWERCASE:

- IT IS A CHECKBOX WHICH IS USED TO STORE THE CHARACTERS IN A MIXTURE OF CAPITAL AND SMALL LETTERS.
- IF IT NOT SELECTED BY DEFAULT ALL THE CHARACTERS ARE STORED IN CAPTIAL LETTERS
- EG: DOMAIN NAME1.

FIXED VALUES:

- THIS OPTION IS USED TO SET THE FIXED VALUES OR CONSTANTS SO THAT DOMAIN CAN ACCEPT ONLY THOSE VALUES. IF WE ENTER OTHER VALUES, THE SYSTEM THROUGHS AN ERROR.
- EG: DOMAIN GENDER.

CONVERSION ROUTINE:

- IT IS BASICALLY FUNCTION MODULE WHICH IS USED TO CONVERT A VALUE FROM EXTERNAL FORMAT TO INTERNAL FORMAT AND VICEVERSA.
- EG: CONVERSION_EXIT_ALPHA_INPUT IS A FUNCTION MODULE FOR DOMAIN KUNNR.

QUESTIONS

CAN WE CREATE A FIELD WITH OUT DATA ELEMENT AND DOMAIN?

ANSWER IS YES, BY CLICKING ON PRE-DEFINED TYPE.

WHAT IS THE PURPOSE OF DELIVERY CLASS?

WHILE UPDATING ONLY APPLICATION DATA WILL BE UPDATED.

CUSTOMIZING DATA -> IT WILL ASK WHETHER TO UPDATE.

TEMPORARY DATA -> WILL NOT BE UPDATED.

TABLE MAINTENANCE GENERATOR (TMG)

NOTE:

TMG WILL BE CREATED IN FORM OF FUNCTIONS.

TMG IS STANDARD SAP PROGRAM.

- IT IS USED TO MAINTAIN THE MASS / BULK AMOUNT OF DATA I.E., CREATE, CHANGE, DELETE BULK DATA.
- IT IS ALSO USED FOR VALIDATING (USING EVENTS) THE TABLE FIELDS BY WRITING ABAP CODE INSIDE THE FORM ROUTINES (SUB PROGRAMS).
- THE FORM ROUTINES ARE AVAILABLE IN TABLE EVENTS.

STEPS TO CREATE TMG:

- ✓ SM30 IS THE TRANSACTION CODE FOR TABLE MAINTENANCE GENERATOR.
- ✓ GO TO SE11, GIVE THE TABLE NAME.
- ✓ CLICK ON CHANGE.
- ✓ CLICK ON UTILITIES -> TABLE MAINTENANCE GENERATOR.
- ✓ GIVE AUTHORIZATION GROUPS & NC& (WITH OUT AUTHORIZATION GROUP)
- ✓ GIVE THE FUNCTION GROUP AS TABLE NAME
- ✓ SELECT ONE STEP AS MAINTENANCE TYPE.
- ✓ CLICK ON FIND SCREEN NUMBER BUTTON.
- ✓ PRESS ENTER ON THE POP-UP.
- ✓ CLICK ON CREATE ICON.
- ✓ THE TMG WILL BE CREATED.
- ✓ CLICK ON SAVE & CLICK ON BACK.
- ✓ GO TO SE80.
- ✓ SELECT FUNCTION GROUP FROM THE LIST.
- ✓ GIVE THE FUNCTION GROUP NAME AS SAME OF TABLE NAME.
- ✓ PRESS ENTER.
- ✓ RIGHT CLICK ON THE ROOT FOLDER OF (TABLE NAME) AND ACTIVATE.

TESTING TMG:

- ✓ GO TO SM30 GIVE THE TABLE NAME.
- ✓ CLICK ON MAINTAIN BUTTON.
- ✓ CLICK ON NEW ENTRIES BUTTON.
- ✓ ENTER DATA AND SAVE IT.
- ✓ THE DATA SHOULD BE SAVED IN THE TABLE.

NOTE:

- ONCE TMG IS CREATED, THE TABLE SHOULD NOT BE CHANGED.
- SUPPOSE IF THE TABLE IS CHANGED LIKE ADDING EXTRA FIELD, THE TMG WILL NOT WORK.
- SO THE EXISTING TMG MUST BE DELETED AND SHOULD BE RE-CREATED AGAIN.

EXAMPLE ON TMG

BUSINESS REQUIREMENT: VALIDATE THE CUSTOMER NAME FIELD IN THE TABLE (ZCUSTOM) TO CHECK IF THE NAME = 'SPACE'. IF IT IS SPACE THEN RAISE AN ERROR MESSAGE.

QUESTIONS

WHY, WE ARE GOING FOR TABLE MAINTENANCE GENERATOR (TMG)?

BECAUSE, IN PRODUCTION SYSTEM END USER'S ARE NOT AUTHORIZED FOR SE11.

CAN WE CREATE A TABLE FIELD WITH OUT A DATA ELEMENT?

YES, WE CAN CREATE BY USING PRE-DEFINED DATA TYPES. IF WE DO LIKE THIS WE CANNOT CREATE LABELS.

CAN WE CREATE A DATA ELEMENT WITH OUT DOMAIN?

YES, WE CAN CREATE BY USING PRE-DEFINED DATA TYPE (RADIO BUTTON AT DATA ELEMENT LEVEL) BUT WE CANNOT DEFINE VALUE RANGES IN DOMAIN.

IN HOW MANY WAYS WE CAN CREATE TABLE?

WE CAN CREATE TABLE IN TWO WAYS

TOP-TO-BOTTOM APPROACH:

TABLE -> FIELD -> DATA ELEMENT -> DOMAIN

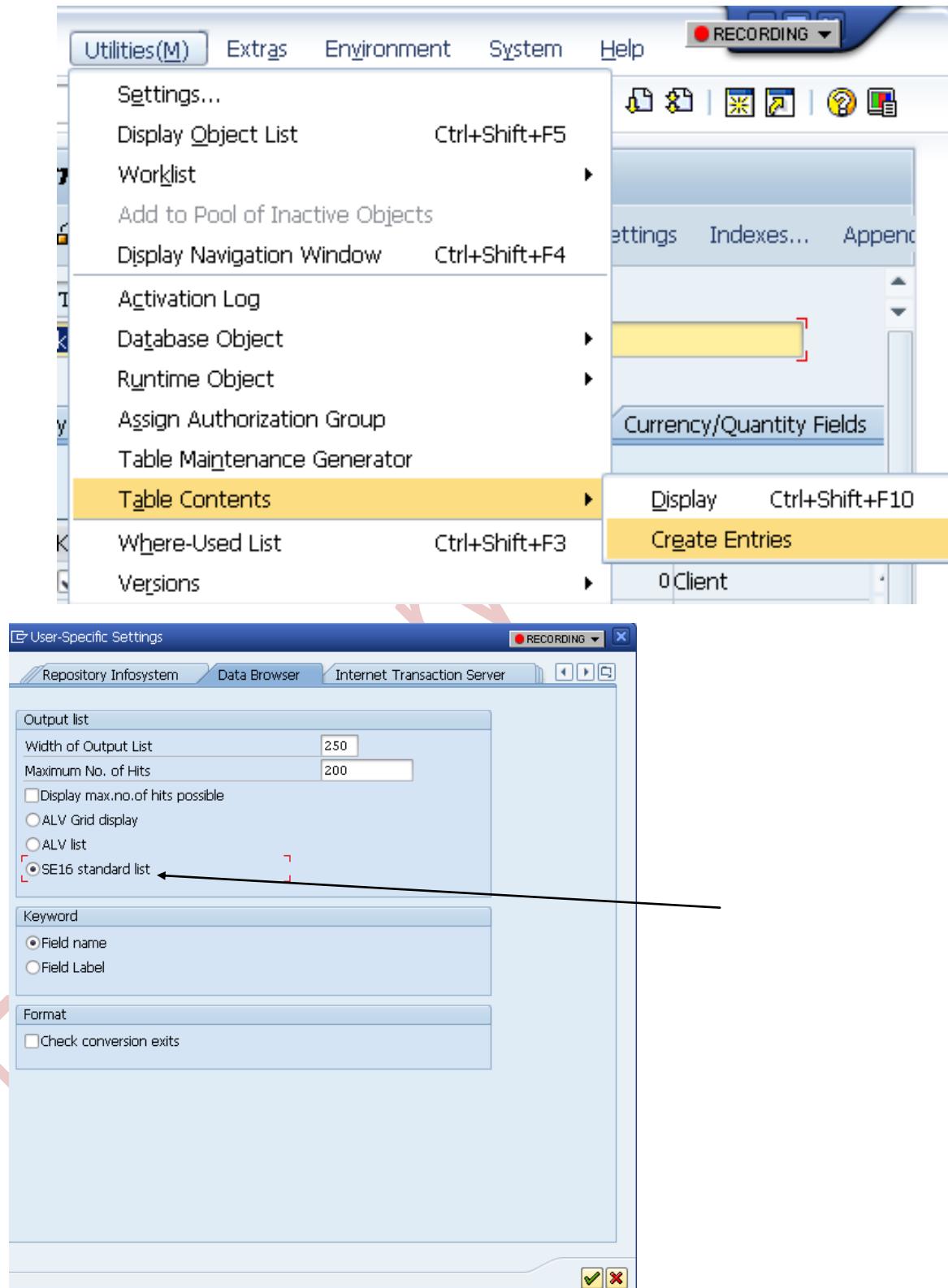
BOTTOM-TO-TOP APPROACH:

TABLE <- FIELD <- DATA ELEMENT <- DOMAIN



TABLE MAINTENANCE GENERATOR (TMG)

ONE OF THE WAYS TO MAINTAIN DATA



SECOND WAY:

- ☞ IN PRODUCTION SERVER END USERS WILL MAINTAIN DATA.
- ☞ END USER WILL NOT BE GIVE AUTHORIZATION FOR SE11 TRANSACTION.
- ☞ THEY GO AND MODIFY THE TABLE IF THEY HAVE AUTHORIZATION.

NOTE:

SO GO FOR TMG.

TMG IS A MECHANISM WHICH IS USED FOR MAINTAINING DATA **NOT FROM SE11**.

DEFINITION:

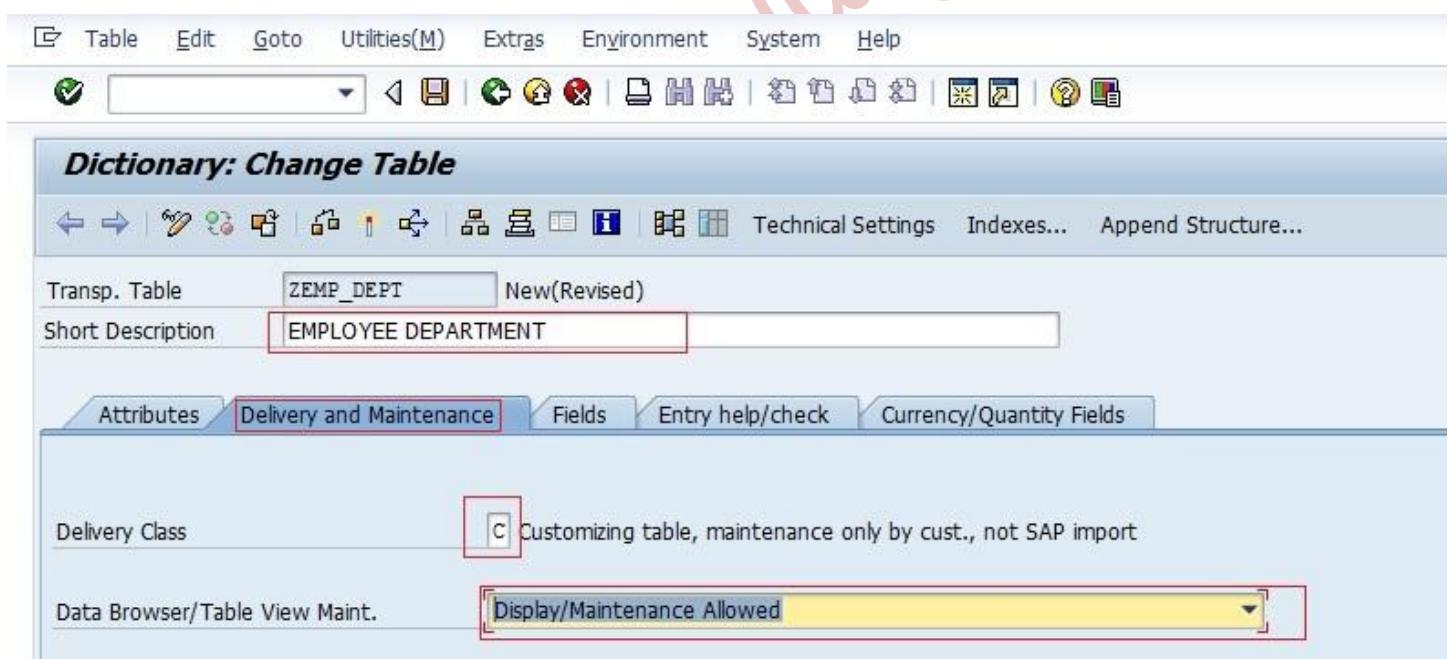
TMG IS A MECHANISM THAT IS USED TO MAINTAIN THE DATA IN A DATABASE TABLE IN THE PRODUCTION SERVER.

QUESTION: DO WE CREATE TMG FOR ALL THE CUSTOM TABLES THAT WE CREATE?

ANSWER: NO.

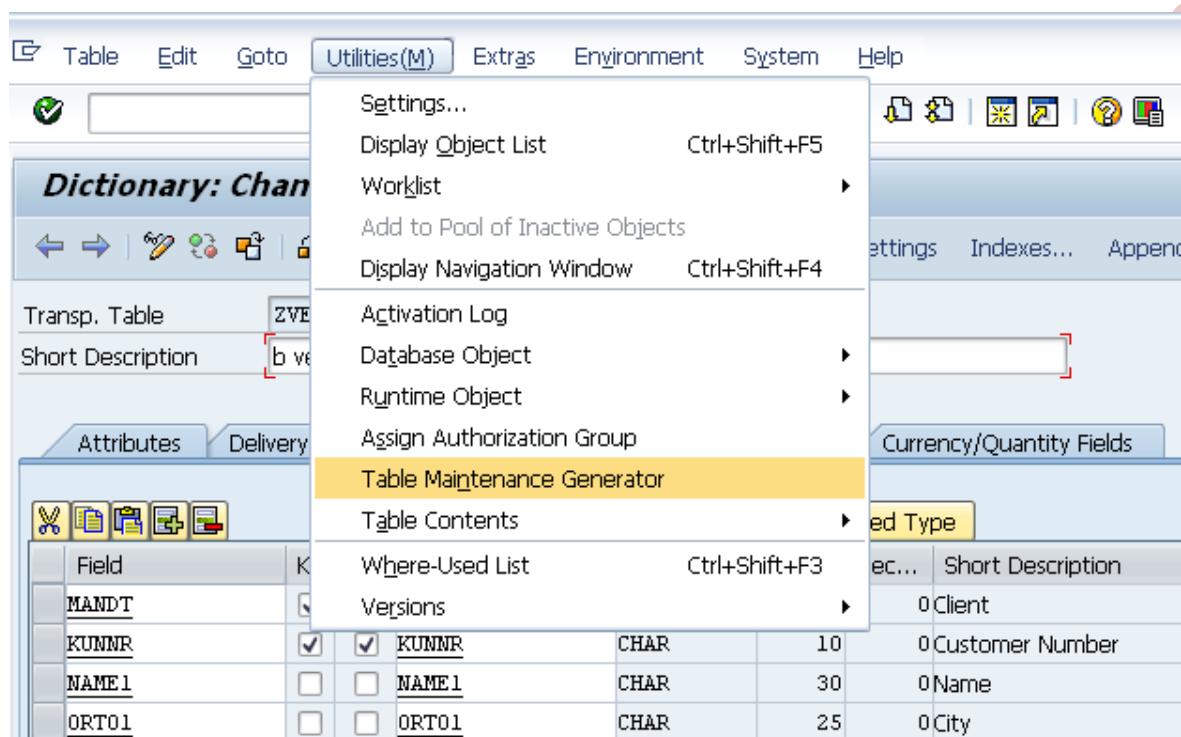
QUESTION: THEN FOR WHAT & WHICH TABLE WE CREATE TMG.

ANSWER: FOR WHICH WE MAINTAIN DATA AND USE IN APPLICATION, ONLY FOR THEM WE CREATE TMG.



STEPS FOR TMG

UTILITIES-> TABLE MAINTENANCE GENERATOR



Generate Table Maintenance Dialog: Generation Environment

This dialog box contains several configuration sections:

- Find Scr. Number(s)**: A checkbox with a red arrow pointing to it.
- Table/View**: Set to "ZVEJAY".
- Technical Dialog Details**:
 - Authorization Group: &NC& (w/o auth. group)
 - Authorization object: S_TABU_DI...
 - Function group: zvejay
 - Package: (empty)
- Maintenance Screens**:
 - Maintenance type: one step (radio button selected)
 - Maint. Screen No.:
 - Overview screen (radio button selected)
 - Single screen
- Dialog Data Transport Details**:
 - Recording routine: no, or user, recording routine (radio button selected)
 - Compare Flag: Automatically Adjustable

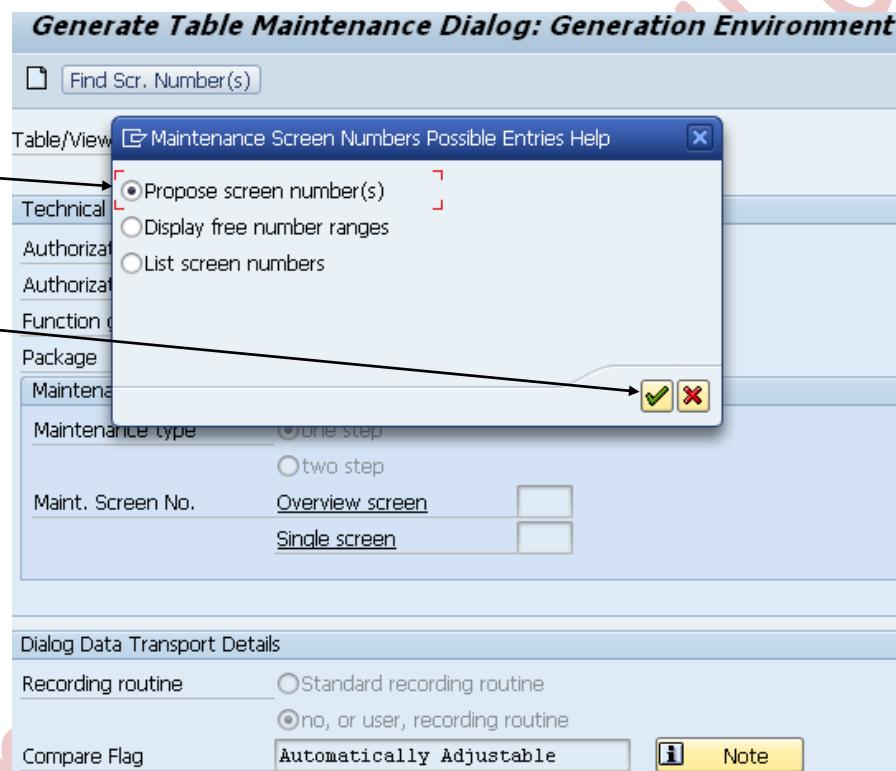
ENTER AUTHORIZATION GROUP (FOR E.G. OSBC)

YOU CAN SELECT EITHER ONE STEP OR TWO STEP. ASSIGN SCREEN NUMBERS (TO ASSIGN SCREEN NUMBERS CLICK ON THE BUTTON 'FIND SCR NO'. IT WILL PROPOSE SCREEN NO'S)

THEN CREATE. SAVE

SINGLE STEP: ONLY OVERVIEW SCREEN IS CREATED I.E. THE TABLE MAINTENANCE PROGRAM WILL HAVE ONLY ONE SCREEN WHERE YOU CAN ADD, DELETE OR EDIT RECORDS.

TWO STEP: TWO SCREENS NAMELY THE OVERVIEW SCREEN AND SINGLE SCREEN ARE CREATED. THE USER CAN SEE THE KEY FIELDS IN THE FIRST SCREEN AND CAN FURTHER GO ON TO EDIT FURTHER DETAILS.



Generate Table Maintenance Dialog: Generation Environment

Find Scr. Number(s)

Table/View

Technical Dialog Details

Authorization Group w/o auth. group
 w/ auth. group

Authorization object S_TABU_DI...

Function group ZVEJAY

Package \$TMP Temporary Objects (never transported!)

Maintenance Screens

Maintenance type one step
 two step

Maint. Screen No. Overview screen Single screen

Dialog Data Transport Details

Recording routine Standard recording routine
 no, or user, recording routine

Compare Flag Automatically Adjustable

CLICK ON CREATE

NOW DO THIS

Utilities(M) Extras Environment System Help

Dictionary: Chan

Transp. Table
Short Description

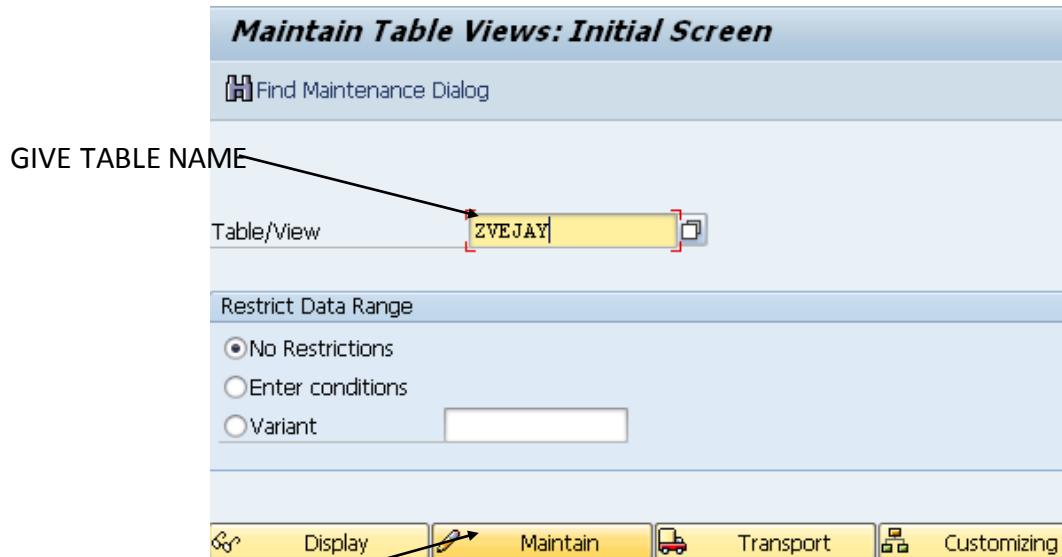
Attributes Delivery

Field	K	Where-Used List	Ctrl+Shift+F3
MANDT		Versions	
KUNNR	<input checked="" type="checkbox"/>	KUNNR	CHAR
NAME1	<input type="checkbox"/>	NAME1	CHAR
ORT01	<input type="checkbox"/>	ORT01	CHAR

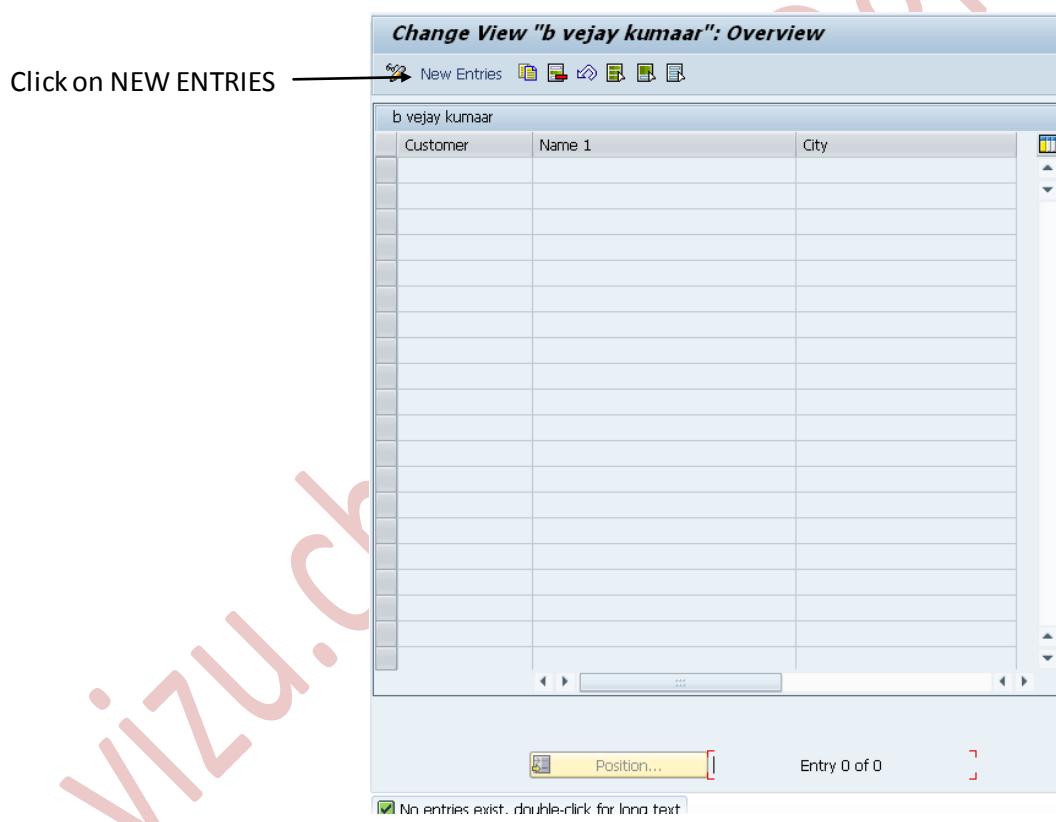
Table Contents

- Display Ctrl+Shift+F10
- Create Entries
- 0 Client
- 0 Customer Number
- 0 Name
- 0 City

TCODE :::: **SM30**



CLICK ON MAINTAIN



WE COPY AND PASTE

OR

WE CAN MANUALLY CREATE ENTRIES

Untitled - Notepad		
File	Edit	Format
View	Help	
23	vejay	chennai
02	kumaar	bangalore
1980	b	hyderabad



New Entries: Overview of Added Entries



b vejay kumaar

	Customer	Name 1	City
	23	vejay	chennai
	02	kumaar	bangalore
	1980	b	hyderabad

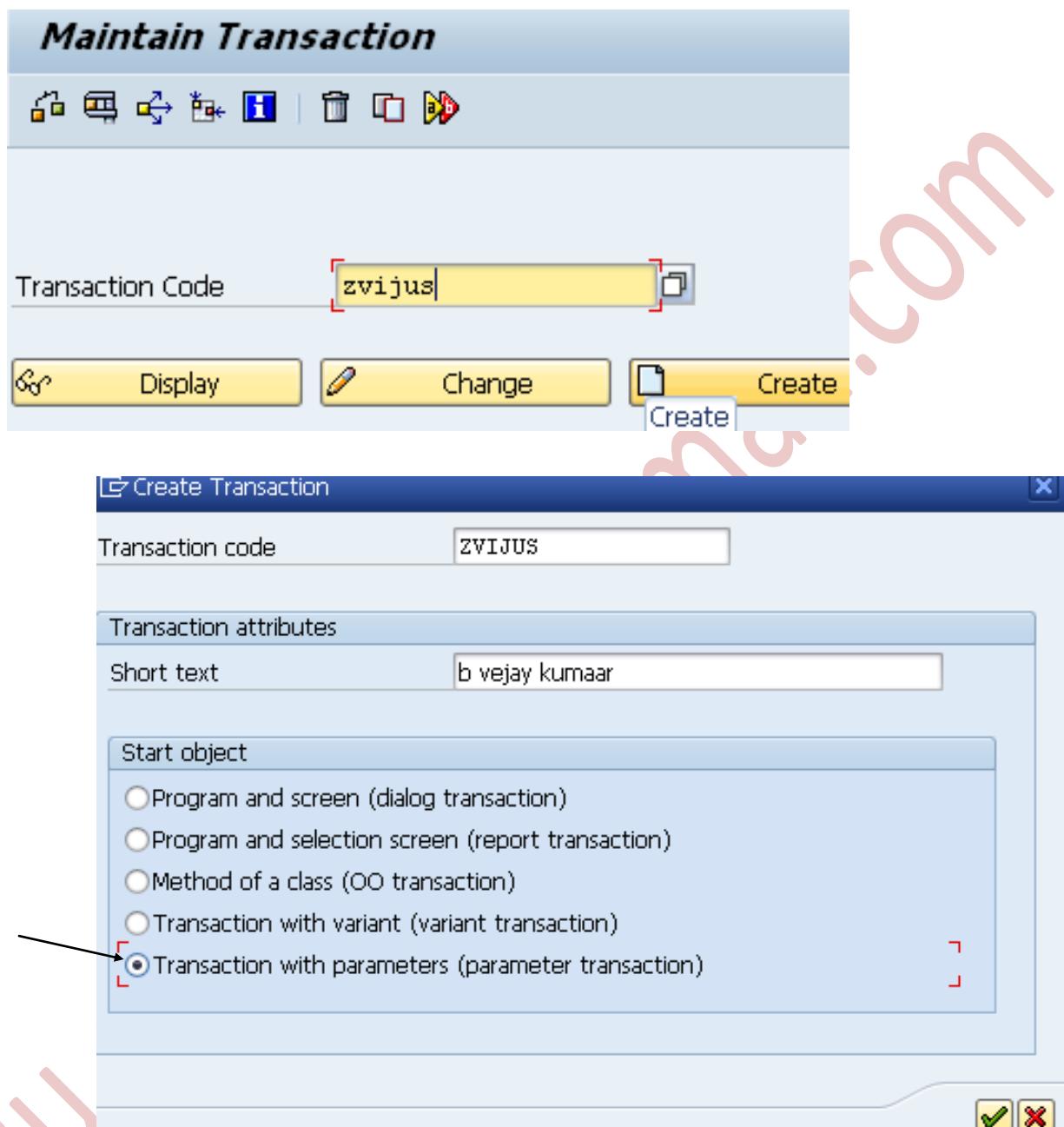
NOW TO GO TO SE11 AND EXECUTE THE TABLE.

Data Browser: Table ZVEJAY Select Entries				3
Table:	ZVEJAY	Displayed Fields:	4 of 4	Fixed Columns: 2 List Width 0250
MANDT	KUNNR	NAME1	ORT01	
800	0000000002	kumaar	bangalore	
800	0000000023	vejay	chennai	
800	0000001980	b	hyderabad	

NOTE:

IF A TMG IS ALREADY CREATED FOR A TABLE, FOR ANY FURTHER CHANGES TO THE TABLE, THE EXISTING TMG HAS TO BE DELETED AND REGENERATED ONCE AGAIN TO REFLECT THE NEW CHANGES IN THE TMG.

CREATE TRANSACTION CODE: SE93



Create Parameter Transaction

Default values for

Transaction	SM30	←
<input checked="" type="checkbox"/> Skip initial screen		
Obsolete: Use default values for transaction		
Screen	<input type="text"/>	
From module pool <input type="text"/>		

Classification

<input type="checkbox"/> Inherit GUI attributes
Transaction classification
<input checked="" type="radio"/> Professional User Transaction
<input type="radio"/> Easy Web Transaction
<input type="checkbox"/> Pervasive enabled

GUI support

<input type="checkbox"/> SAPGUI for HTML
<input type="checkbox"/> SAPGUI for Java
<input type="checkbox"/> SAPGUI for Windows

Default Values

Name of screen field	Value
VIEWNAME	ZVEJAY
UPDATE	X

The object will be created in the original language English (EN)

If we select SHOW = X in default values, then table will only be displayed, we cannot update it.

NOTE:

PATH TO CREATE TMG:

UTILITIES -> TABLE MAINTENANCE GENERATOR IN CHANGE MODE.

- IF WE MAKE ANY CHANGES IN TABLE AFTER CREATING A TMG FOR A TABLE, FURTHER CHANGES TO THE TABLE WILL NOT BE REFLECTED CHANGES TO THE TABLE WILL NOT BE REFLECTED IN TMG, THE EXISTING TMG HAS TO BE DELETED & REGENERATED.
- FOR CREATING & DELETING WE DO ONLY IN SE11.
- IN CHANGE MODE OF THE TABLE WE CAN FIND DELETE BUTTON.
- EVEN SM30 IS ALSO NOT AUTHORIZED FOR END USERS, SO WE NEED TO CREATE A CUSTOM TRANSACTION CODE.
- ❖ A DATABASE TABLE IS A CLIENT INDEPENDENT COMPONENT WHERE AS THE DATA THAT WE MAINTAIN IN THE TABLE IS CLIENT DEPENDENT.
- ❖ THE DISADVANTAGE OF CREATING FIELDS WITH OUT DATA ELEMENT IS HERE WE CANNOT GET THE FIELD LABELS.

EVENTS IN TABLE MAINTENANCE GENERATOR

TOTALLY WE HAVE 39 EVENTS IN TABLE MAINTENANCE GENERATOR.

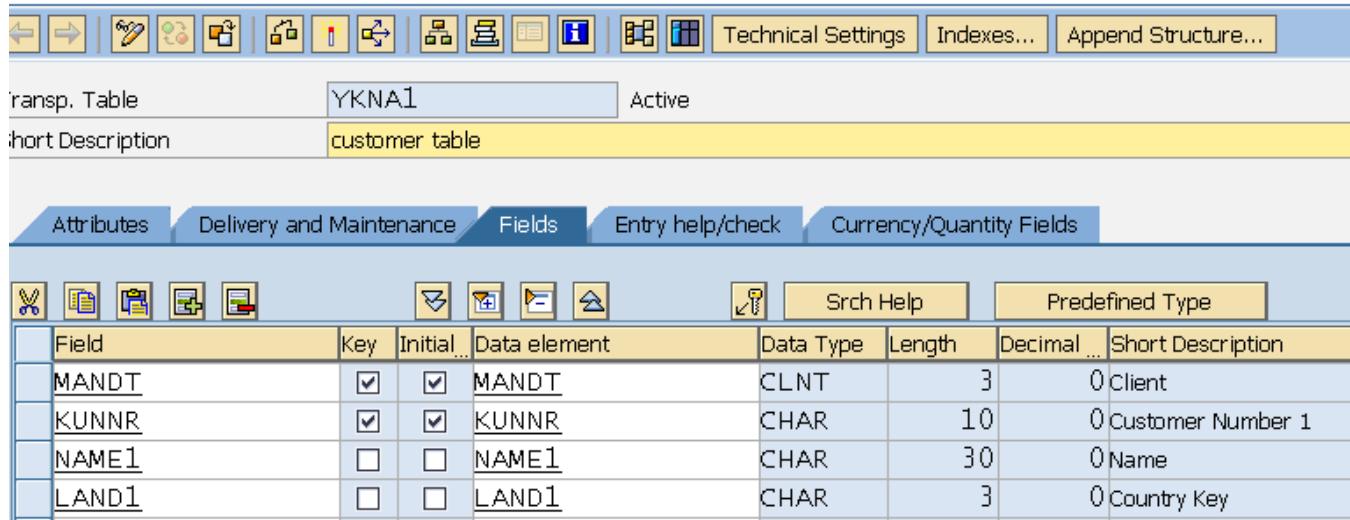
Events in the Table Maintenance Generator:

- + Events are used in TMG for Field Validation.
- + We need to create TMG and then we need to need create events.

Steps to create events:

- + Create Table Maintenance Generator for the table with the below fields.

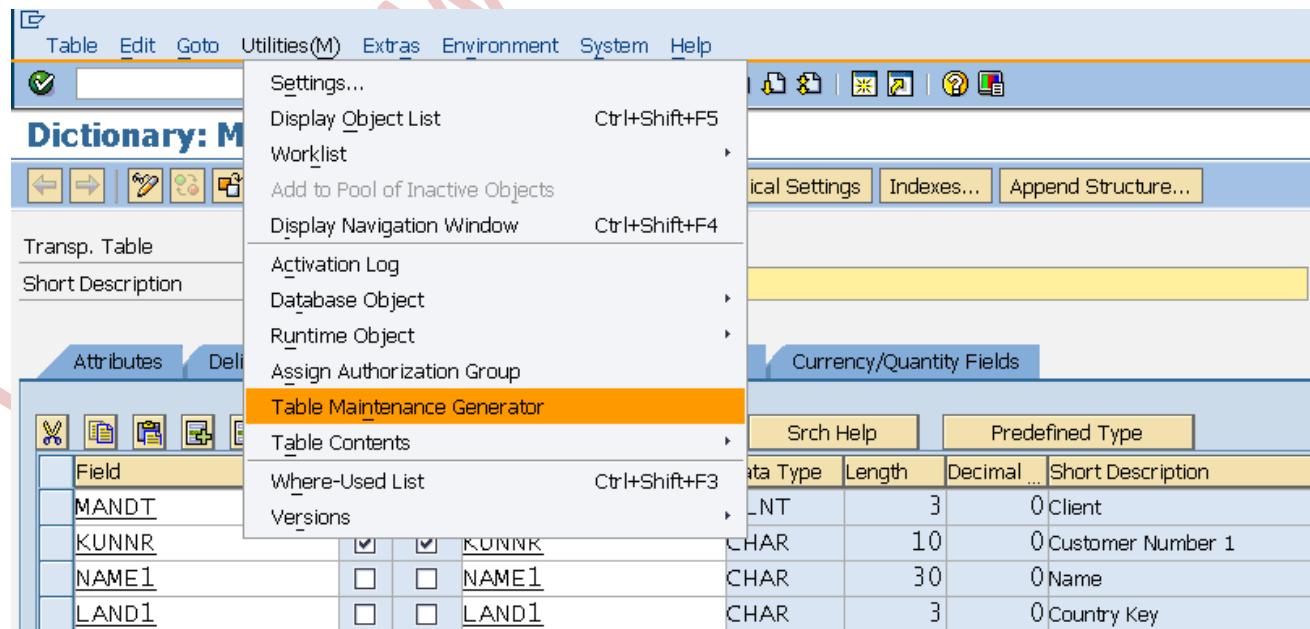
Dictionary: Maintain Table



The screenshot shows the SAP Dictionary: Maintain Table interface. At the top, there is a toolbar with various icons. Below it, the table name 'YKNA1' is set to 'Active'. A yellow-highlighted row contains the short description 'customer table'. The interface includes tabs for Attributes, Delivery and Maintenance, Fields (which is selected), Entry help/check, and Currency/Quantity Fields. The main area displays a table of fields:

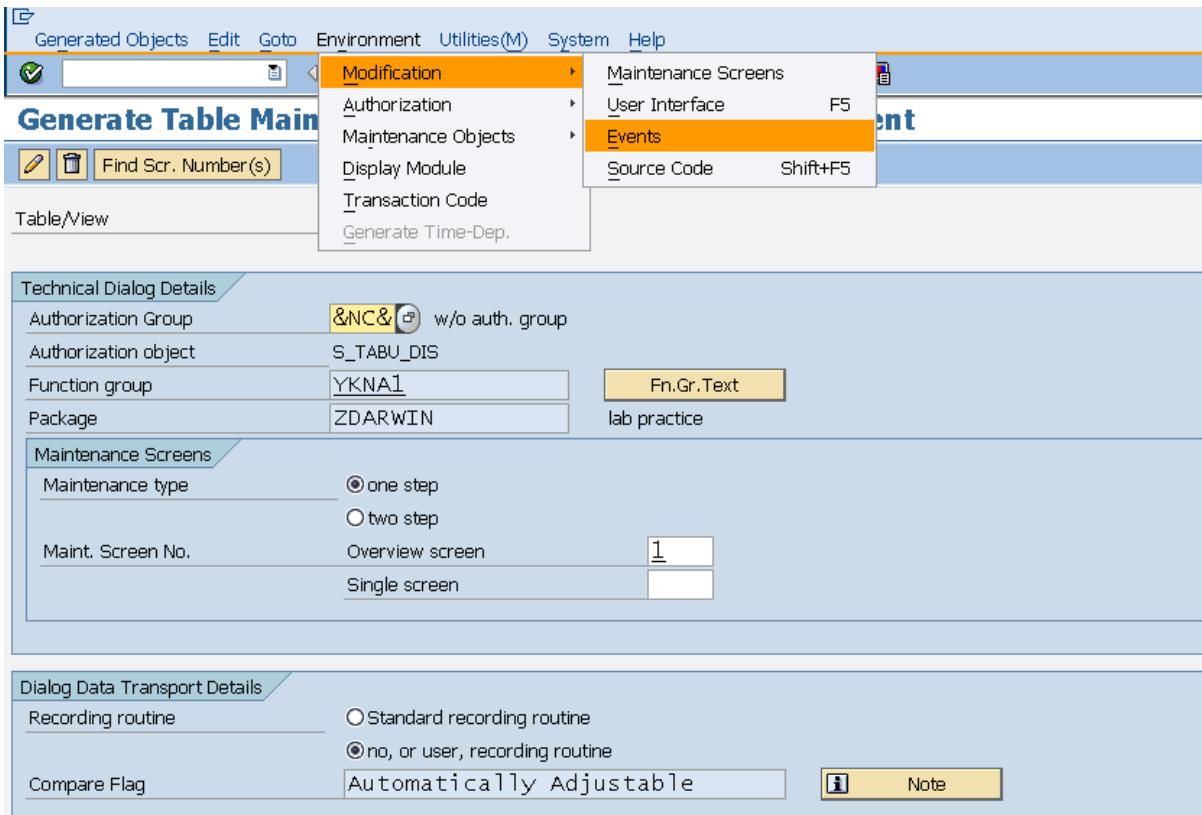
Field	Key	Initial...	Data element	Data Type	Length	Decimal...	Short Description
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0	Client
KUNNR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	KUNNR	CHAR	10	0	Customer Number 1
NAME1	<input type="checkbox"/>	<input type="checkbox"/>	NAME1	CHAR	30	0	Name
LAND1	<input type="checkbox"/>	<input type="checkbox"/>	LAND1	CHAR	3	0	Country Key

- + Create an event as below.
→ Click on utilities → TMG → TMG table is opened.



The screenshot shows the SAP Dictionary: Maintain Table interface with the Utilities menu open. The 'Table Maintenance Generator' option is highlighted. The main area displays the same table of fields as the previous screenshot. The Utilities menu also lists other options like Settings..., Display Object List, Worklist, etc.

→ Click on Environment Menu → Modification → Events.



- ⊕ A new screen is opened.
- ⊕ Click on new entries button.
- ⊕ Press F4 on the first field and select an event.
Eg: 05 → Create new entry.

- ⊕ Give the form routine name.
- ⊕ Press Enter.
- ⊕ ABAP editor icon is displayed.

Change View "FORM routines to be called from view maintenance"

The screenshot shows a table titled 'FORM routines to be called from view maintenance'. The table has two columns: 'T' (Type) and 'FORM routine'. The first row contains '05' and 'YKNA1_NAME1'. The second row is empty. The table includes standard SAP navigation icons for sorting and filtering.

T	FORM routine	Editor
05	YKNA1_NAME1	

- Click on the icon and write the below code.

The screenshot shows the SAP ABAP Editor interface. The title bar reads "ABAP Editor: Display Include LYKNA1F01". The toolbar has various icons for file operations like Open, Save, Print, and Help. The menu bar includes "Program", "Edit", "Goto", "Utilities(M)", "Environment", "System", and "Help". The status bar at the bottom says "Active". The code area contains the following:

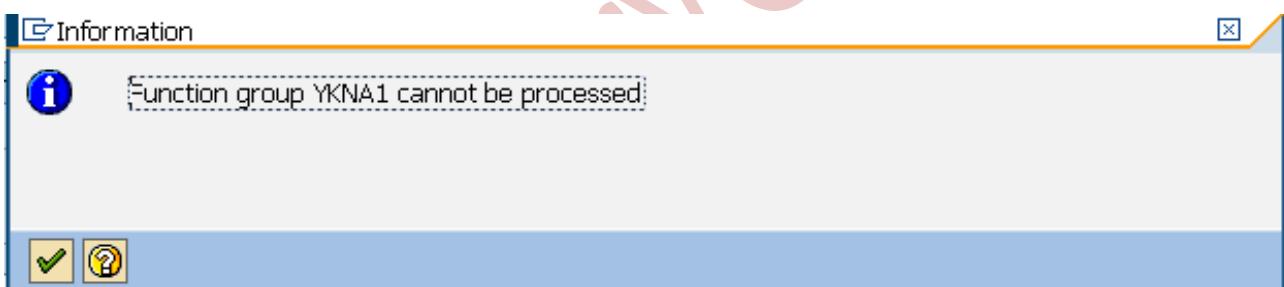
```

*-
***INCLUDE LYKNA1F01 .
*-

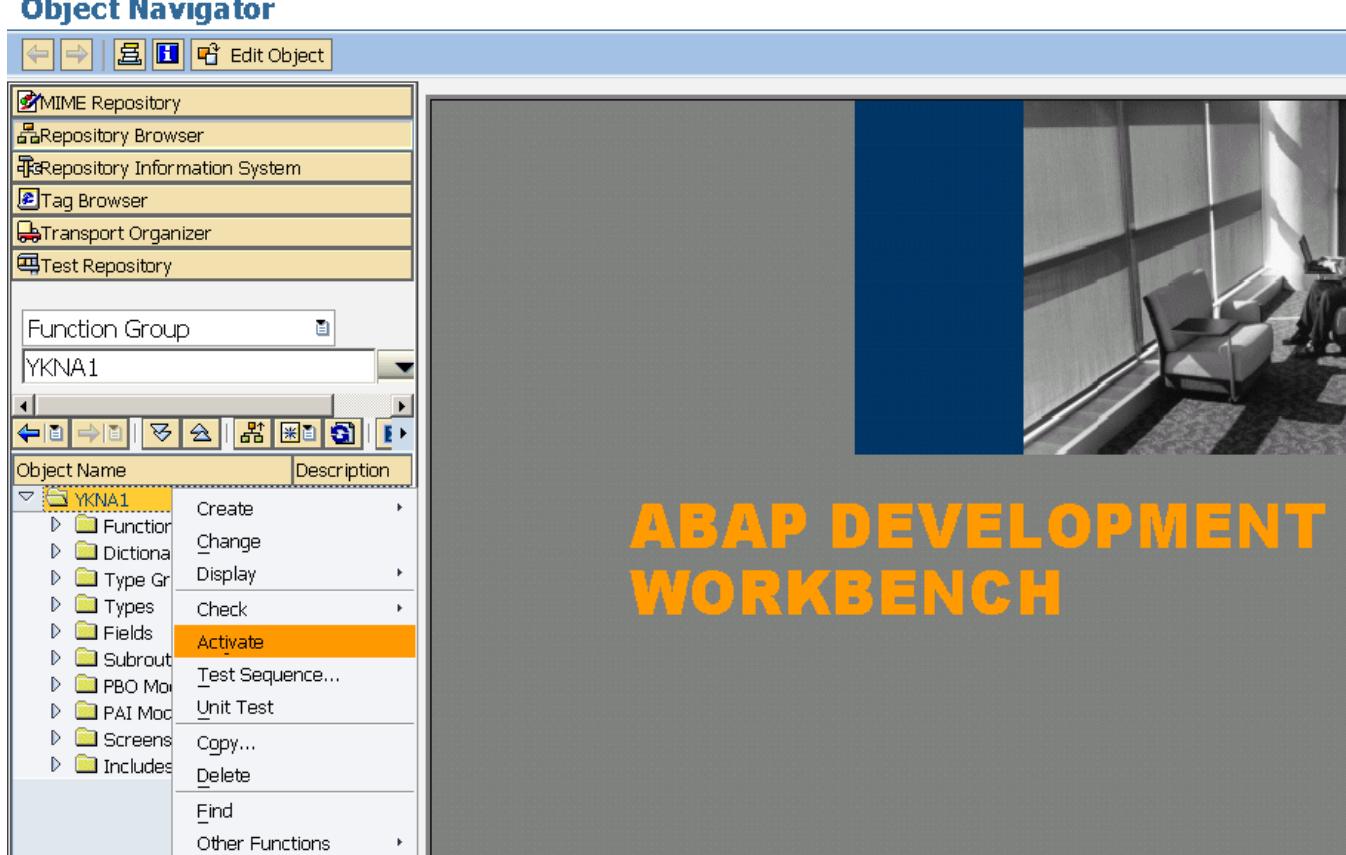
FORM YKNA1_NAME1.
IF YKNA1-NAME1 = ' '.
BREAK-POINT.
MESSAGE 'name of customer is mandatory' TYPE 'E'.
ENDIF.
ENDFORM.

```

- Save it and activate it.
- Click on back and save it.
- Again click on back for 2 times.
- An information box will be displayed with the below msg.



- Press Enter
- Click on back and make the table to display mode.
- Open a new session
- Goto SE80.
- Select function Group from the list.
- Give the function Group name i.e., table name.
- Press Enter.
- Right click on the function group name and select activate.



- Now test the table by creating a new record without specifying the name1.
 - An error message will be displayed at the status Bar. [New Entries: Overview](#)

NOTE:

02 EVENT

FORM XXX

MESSAGE ‘SUCCESSFULLY UPDATED’ TYPE ‘I’.

END FORM.

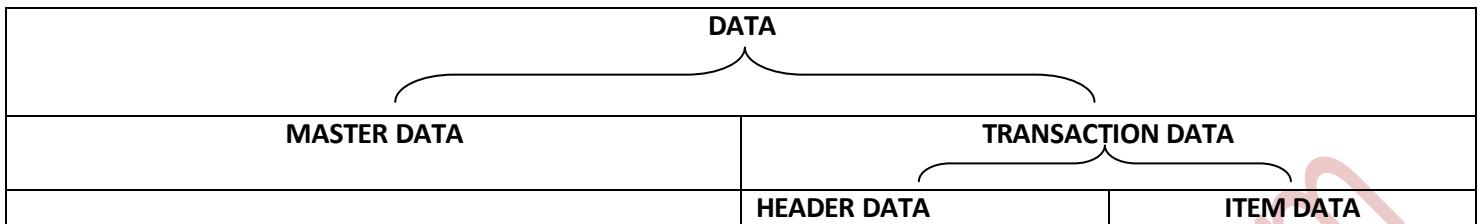
04 EVENT

FORM XXXX

MESSAGE 'DELETED SUCCESSFULLY TYPE 'I'.

ENDFORM.

TYPES OF DATA



MASTER DATA:

NEVER CHANGES (OR) CHANGES VERY RARELY.

TRANSACTION DATA:

DAY-TO-DAY TRANSACTION.

HEADER DATA:

DATA RELATED TO ORGANIZATION.

EG: SALES ORGANIZATION, DISTRIBUTION CHANNEL, DIVISION.

ITEM DATA:

DATA RELATED TO ITEMS (OR) PRODUCTS.

EG: ITEM NUMBER, MATERIAL NUMBER, QUANTITY, PRICE ETC.,

#=#=#=#=#=#=TYPES OF TABLES #=#=#=#=#=#=#=#

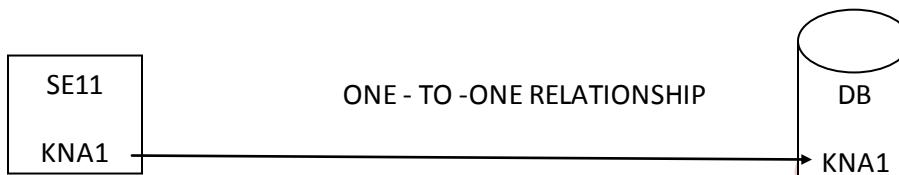
TYPES OF TABLES

We have 3 types of tables

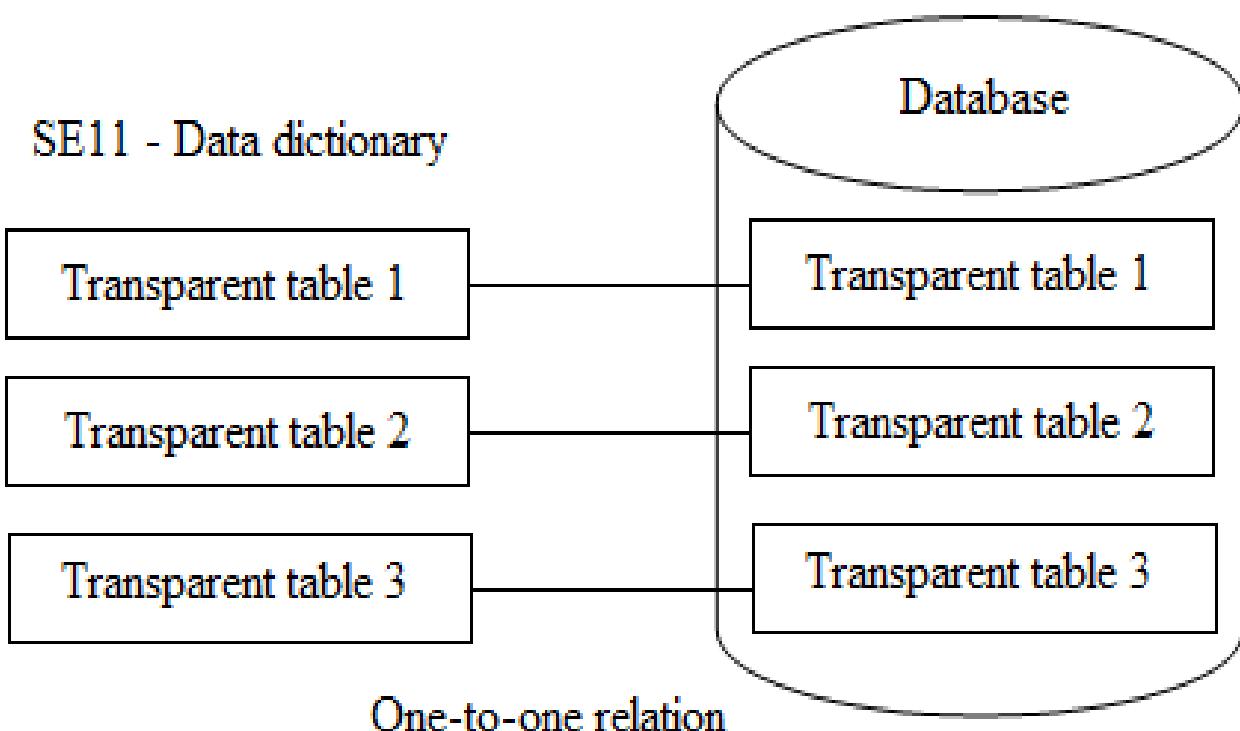
- 1) TRANSPARENT TABLE
- 2) POOLED TABLE
- 3) CLUSTER TABLE

TRANSPARENT TABLES:

- IT FORMS ONE-TO-ONE RELATIONSHIP WITH TABLE DEFINITION IN THE DATABASE.
- TRANSPARENT TABLES ARE USED TO HOLD APPLICATION DATA, WHICH REPRESENTS MASTER DATA OR TRANSACTION DATA.
- EG: FOR MASTER DATA:
 - TABLE OF VENDORS (VENDOR MASTER DATA)
 - TABLE OF CUSTOMER (CUSTOMER MASTER DATA)
- EG: FOR TRANSACTION DATA:
 - ORDER PLACED BY A CUSTOMER OR
 - ORDER PLACED BY A VENDOR.



- USER CAN USE OPEN SQL & NATIVE SQL ALSO.
- BY DEFAULT ALL THE TABLES CREATED BY US ARE CALLED AS TRANSPARENT TABLES.



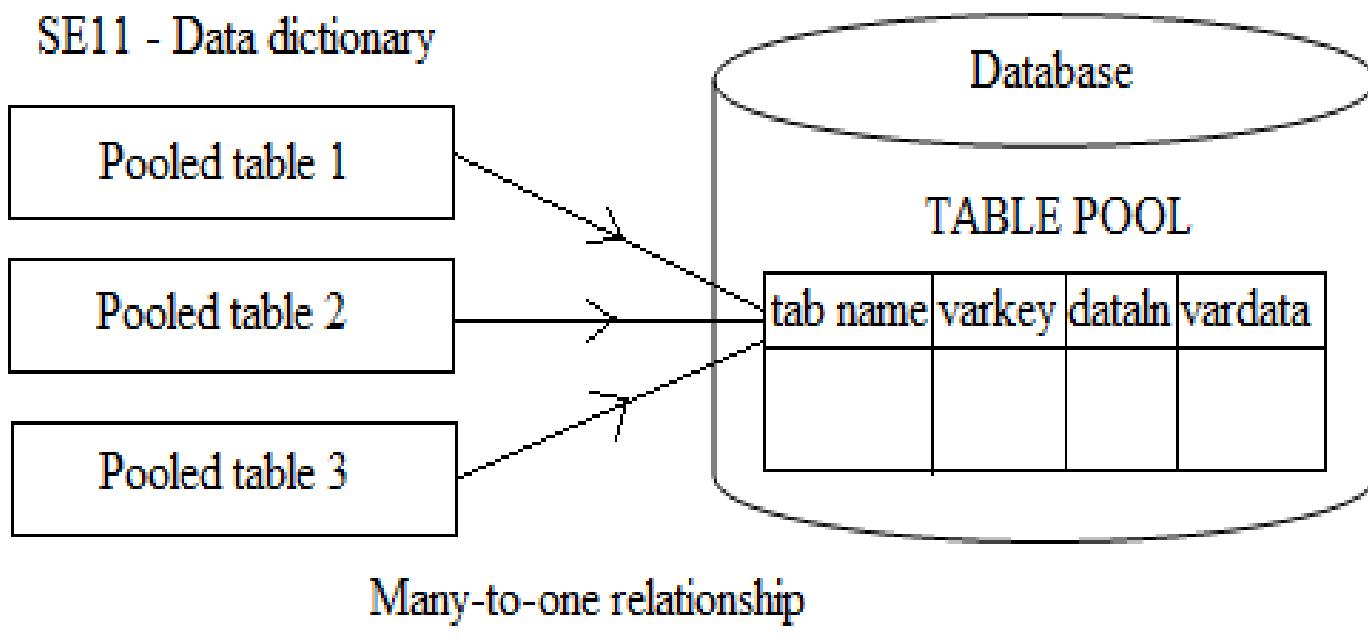
POOLED TABLES:

- FORMS MANY-TO-ONE RELATIONSHIP WITH TABLE DEFINITION IN SAP DB.
- ALL THE POOLED TABLES IN SAP WILL BE STORED IN A BIG TABLE BY NAME TABLE-POOL IN DB. IT MEANS THAT FOR A SINGLE TABLE DEFINED IN THE DB, THERE ARE MANY TABLES IN ABAP DICTIONARY.

THE STRUCTURE OF TABLE-POOL IS:

TAB NAME	: CHAR(10)	:	NAME OF POOLED TABLE IN SAP	KEY FIELDS
VAR KEY	: CHAR(N)	:	STORES KEY FIELDS OF POOLED TABLE	
DATA LN	: INT2(5)	:	STORES THE LENGTH OF TABLE DATA.	
VAR DATA	: RAW(N)	:	STORES THE DATA OF POOLED DATA.	

SE11 - Data dictionary



- FOLLOWS LINEAR SEARCH.
- ONLY OPEN SQL STATEMENTS.
- POOLED MAY HAVE / MANY NOT HAVE COMMON PRIMARY KEY.
- POOLED ARE USED TO STORE INTERNAL CONTROL INFORMATION, SCREEN SEQUENCES, PROGRAM PARAMETERS AND TEMPORARY DATA.
- SECONDARY INDEXES CANNOT BE CREATED.

EXAMPLES:

- T138B - MATERIALMASTER SCREEN SEQUENCE CONTROL
- T588A - TRANSACTION CODE FOR HR INFO TYPE INFORMATIONS
- T030 - STANDARD ACCOUNTS TABLE

- THE DATA IN THE TABLE-POOL IS STORED CONTINUOUSLY WITHOUT ANY PAGE BREAKS (SAME LIKE NOTEPAD FILES)
- WE DON'T CREATE ANY POOLED TABLES.

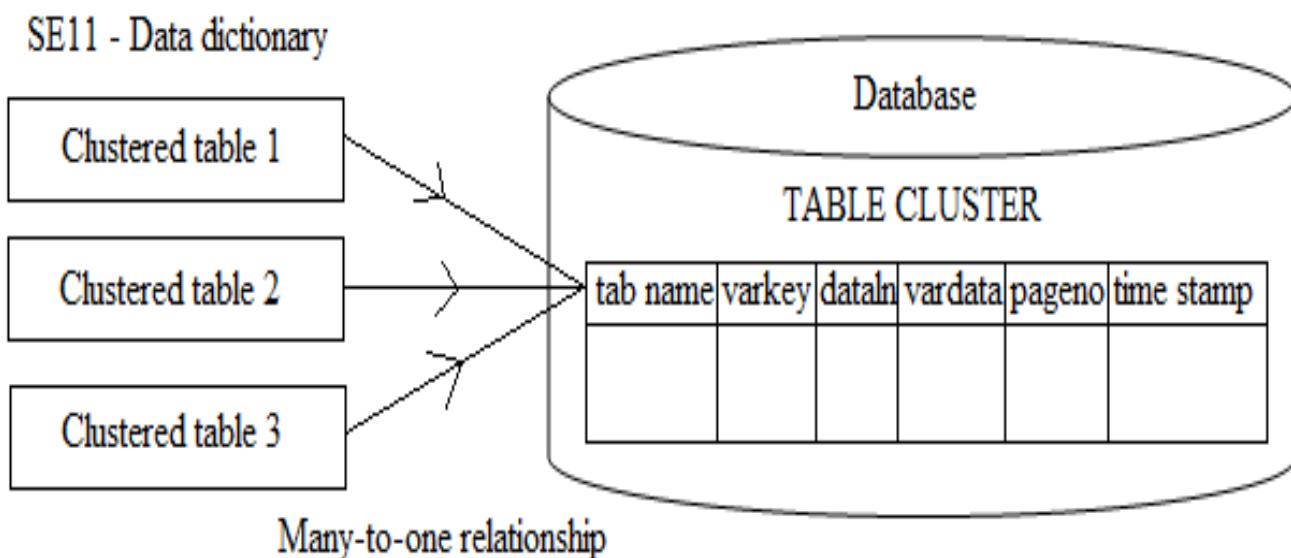
CLUSTER TABLES:

- FORMS MANY-TO-ONE RELATIONSHIP WITH TABLE DEFINITION IN SAP DB.
- ALL THE CLUSTER TABLES IN SAP WILL BE STORED IN A BIG TABLE BY NAME TABLE-CLUSTER IN DB. IT MEANS THAT FOR A SINGLE TABLE DEFINED IN THE DB, THERE ARE MANY TABLES IN ABAP DICTIONARY.

THE STRUCTURE OF TABLE-POOL IS:

TAB NAME	:	CHAR(10)	:	NAME OF POOLED TABLE IN SAP	KEY FIELDS
VAR KEY	:	CHAR(N)	:	STORES KEY FIELDS OF POOLED TABLE	
DATA LN	:	INT2(5)	:	STORES THE LENGTH OF TABLE DATA.	
VAR DATA	:	RAW(N)	:	STORES THE DATA OF POOLED DATA.	
PAGE NO	:		:	STORES PAGE NUMBER OF DATA	
TIME-STAMP	:		:	STORES THE TIME STAMP OF DATA (I.E. DATE / TIME / WHO CREATED)	

- DATA IS STORED IN FORM OF PAGES (MS-WORD)
- SECONDARY INDEX IS NOT CREATED



- FOLLOWS BINARY SEARCH.
- ONLY OPEN SQL STATEMENTS.
- ALL THE CLUSTER TABLES HAVE COMMON PRIMARY KEY.
- CLUSTER TABLES ARE USED TO STORE INTERNAL CONTROL INFORMATION, SCREEN SEQUENCES, PROGRAM PARAMETERS, TEMPORARY DATA.
- SECONDARY INDEXES CANNOT BE CREATED.

EXAMPLES:

BKPG - DOCUMENT HEADER TABLE IN FI MODULE

B SEG - DOCUMENT ITEM TABLE IN FI MODULE

THE TABLE CLUSTER RFBLG HOLDS DATA FOR 5 TRANSPARENT TABLE BSEC, BSED, BSEG, BSES AND BSET.

TRANSPARENT	POOL	CLUSTER
Contain a single table. Used to store master data	They are used to hold a large number of very small tables(stores system data)	They are used to hold data from a few number of large tables.(stores system data)
It has a one-to-one relationship with a table in the database	It has a many-to-one relationship with a table in the database	It has a many-to-one relationship with table in the database
For each transparent table there is one associated table in the database	It is stored with other pooled tables in a single table called table pool in the database	Many cluster tables are stored in a single table in the database called a table cluster
The database table has the same name, same number of fields and the fields have the same names	The database table has different name, different number of fields and fields have different names	The database table has different name, different number of fields and fields have different names
There is only a single table	Table pools contain more tables than table clusters	Contains less tables than table pools
Secondary indexes can be created	Secondary indexes cannot be created	Secondary indexes cannot be created

STRUCTURES

USAGE OF STRUCTURE	
PROGRAM	DATA DICTIONARY
<ul style="list-style-type: none"> ➤ IN PROGRAM, STRUCTURES ARE USED TO HOLD (NOT STORE) 1 RECORD OF DATA AT RUNTIME. ➤ STRUCTURES CANNOT CARRY MULTIPLE RECORDS. 	<ul style="list-style-type: none"> ➤ IT IS USED TO ADD FIELDS TO TABLES OR STRUCTURE ➤ SIGNIFICANCE OF STRUCTURE IS DATA DICTIONARY IS RE-USABILITY.

SIGNIFICANCE = IMPORTANT, WITH RESPECT TO WHAT.

STRUCTURE:

- IT IS GROUP OF FIELDS. STRUCTURE DOES NOT HAVE MEMORY IN THE DATABASE LIKE TABLE.
- DURING RUNTIME, IT HOLDS ONLY ONE RECORD.
- AT DATA DICTIONARY LEVEL BOTH TABLE AND STRUCTURE ARE THE ONE AND THE SAME.
- THE DIFFERENCE LIES IN THE DATABASE IN TERMS OF MEMORY CREATION.

DIFFERENCE BETWEEN STRUCTURE AND TABLE?

DATABASE TABLE	STRUCTURE (PART OF DATA TYPE)
IT HAS MEMORY IN THE DATABASE	DOES NOT HAVE MEMORY IN THE DATABASE
STORES PERMANENTLY MULTIPLE RECORDS OF DATA	HOLDS SINGLE RECORD AT RUNTIME
NEED TO SPECIFY TECHNICAL SETTINGS	SINCE THERE IS NO MEMORY THERE IS NO NEED TO SPECIFY TECHNICAL SETTINGS
TABLE WILL HAVE PRIMARY KEY AND HAVE PRIMARY INDEX.	STRUCTURE DOES NOT HAVE PRIMARY KEY AND PRIMARY INDEX.
DELIVERY MAINTENANCE UNDERLYING	NO CONCEPT OF DELIVERY MAINTENANCE
TMG CAN BE CREATED	TMG CANNOT BE CREATED

- FIELDS IN STRUCTURE ARE CALLED AS COMPONENTS.
- NO MANDT IN STRUCTURE.
- STRUCTURE IS ALSO A DATATYPE.

USING STRUCTURES IN TABLES:

THERE ARE TWO WAYS TO USE STRUCTURES

APPEND STRUCTURE	.INCLUDE STRUCTURE
APPEND STRUCTURE CAN BE CREATED FOR BOTH STANDARD TABLES AND CUSTOM TABLE. BUT MOSTLY WE USE FOR STANDARD TABLES.	INCLUDE STRUCTURE IS ONLY FOR CUSTOM TABLES. WE CAN INCLUDE STANDARD STRUCTURE OR CUSTOM STRUCTURE.
ADD THE FIELDS AT THE END OF THE TABLE	STRUCTURE CAN BE INCLUDED AT ANY POSITION
APPEND STRUCTURE FOR A TABLE CANNOT BE USED FOR OTHER TABLES I.E, WE CREATE A STRUCTURE DURING CREATION OF APPEND STRUCTURE. ONE TIME USAGE.	SAME STRUCTURE CAN BE USED AS INCLUDE STRUCTURE FOR MULTIPLE TABLES.
JUST CLICK ON APPEND STRUCTURE BUTTON AND GIVE NAME AND FIELDS.	GIVE THE FIELD NAME AS .INCLUDE AND DATA ELEMENT AS <STRUCTURE NAME>

RESTRICTIONS FOR APPEND STRUCTURE:

IF A TABLE IS HAVING A FIELD OR FIELDS WITH DATA TYPES LCHAR, VCHAR, RAW THESE FIELDS SHOULD BE ALWAYS BE LAST FIELDS OF YOUR TABLE. FOR SUCH FIELDS WE CANNOT MAKE APPEND STRUCTURE.

ACCESS KEY IS OF TWO TYPES:

- DEVELOPER ACCESS KEY
- COMPONENT ACCESS KEY.

DEVELOPER ACCESS KEY	COMPONENT ACCESS KEY
ASKS FOR ONE TIME REGISTER YOUR USER ID AS DEVELOPER ALLOWS YOU TO DO CUSTOM DEVELOPMENTS.	EXISTS FOR EVERY STANDARD COMPONENT WHEN YOU OPEN A STANDARD COMPONENT IN CHANGE MODE.

STEPS TO CREATE A STRUCTURE

- ☞ GO TO SE11.
- ☞ SELECT DATA TYPE
- ☞ GIVE STRUCTURE NAME, EG: ZADDRESS
- ☞ CLICK ON **CREATE**.
- ☞ GIVE DESCRIPTION.
- ☞ GIVE THE FIELDS AS BELOW.

COMPONENT	COMPONENT TYPE
LAND1 (FIELD NAME)	LAND1 (DATA ELEMENT)
ORT01 (FIELD NAME)	ORT01 (DATA ELEMENT)
PSTLZ (FIELD NAME)	PSTLZ (DATA ELEMENT)

- ☞ SAVE AND ACTIVATE.

STEPS TO INSERT INTO A TABLE

- ☞ GO TO SE11.
- ☞ OPEN THE CUSTOM TABLE.
- ☞ GIVE THE FIELD NAME AS .INCLUDE
- ☞ GIVE DATA ELEMENT AS STRUCTURE NAME.
- ☞ PRESS ENTER.
- ☞ ALL THE FIELDS WILL BE COPIED.
- ☞ CLICK ON  EXPAND ICON TO DISPLAY FIELDS.
- ☞ SAVE AND ACTIVATE.

NOTE:

- ❖ SOMETIMES WE WILL BE GETTING AN ERROR WITH A MESSAGE STRUCTURE CHANGE AT THE FIELD LEVEL.
- ❖ TO FIX THE ERROR FOLLOW THE BELOW STEPS.

- ☞ GO TO SE14.
- ☞ GIVE TABLE NAME.
- ☞ PRESS ENTER.
- ☞ CLICK ON **ACTIVATE & ADJUST DATABASE TABLE**.
- ☞ THE TABLE ERRORS WILL BE AUTOMATICALLY FIXED AND IT MOVES INTO ACTIVE STATE.

USING APPEND STRUCTURE

BUSINESS REQUIREMENT:

ENHANCE THE STANDARD SAP TABLE MARA TO INSERT A NEW FIELD BY NAME MATERIAL DESCRIPTION.

- ☞ GO TO SE11.
- ☞ GIVE THE TABLE NAME AS MARA.
- ☞ CLICK ON DISPLAY.
- ☞ CLICK ON APPEND STRUCTURE.
- ☞ GIVE THE APPEND NAME, EG: ZAPPEND.
- ☞ PRESS ENTER, GIVE DESCRIPTION.
- ☞ GIVE THE FIELD NAME AS MAT_DES.
- ☞ GIVE THE DATA ELEMENT AS MAKTX.
- ☞ SAVE AND ACTIVATE.

TO SEE THE FIELD CLICK ON APPEND STRUCTURE AND SELECT THE STRUCTURE NAME **ZAPPEND** AND PRESS ENTER.

CURRENCY AND QUANTITY FIELDS

CURRENCY FIELDS:

- A FIELD WHICH IS USED TO STORE AMOUNT DATA IS CALLED CURRENCY FIELD. THE DATA TYPE FOR CURRENCY FIELD IS CURR.
- FOR EACH CURRENCY FIELD WE SHOULD SPECIFY THE CORRESPONDING CURRENCY KEY TO TELL WHETHER IT IS INR, USD, EUR, ETC., THE DATA TYPE FOR CURRENCY KEY IS CUKY.
- FINALLY WE SHOULD LINK CURRENCY FIELD AND CURRENCY KEY FIELD USING REFERENCE TABLE NAME AND REFERENCE FIELD NAME.

QUANTITY FIELDS:

- A FIELD WHICH IS USED TO STORE QUANTITY DATA IS CALLED QUANTITY FIELD. THE DATA TYPE FOR QUANTITY FIELD IS QUAN.
- FOR EACH QUANTITY FIELD WE SHOULD SPECIFY THE CORRESPONDING QUANTITY UNITS. THE DATA TYPE IS UNIT.
- FINALLY, WE SHOULD LINK QUANTITY FIELD AND UNITS FIELD USING REFERENCE TABLE NAME AND REFERENCE FIELD NAME.

STEPS:

- DEFINE A CURRENCY (CURR) OR QUANTITY (QUAN) FIELD.
- DEFINE A CURRENCY KEY (CUKY) OR QUANTITY UNITS (UNIT) FOR ABOVE FIELDS
- FINALLY CLICK ON CURRENCY / QUANTITY FIELDS TAB AND LINK THEM BY USING REFERENCE TABLE NAME AND REFERENCE FIELD NAME.

EXAMPLE:

- ☞ CREATE A TABLE WITH THE FIELDS: CUSTNO, CNAME, AMT, QTY.
- ☞ CREATE A FIELD BY CUSTNO, CNAME WITH RESPECTIVE DATA ELEMENTS AND DOMAINS.
- ☞ GIVE FIELD NAME AS AMOUNT.
- ☞ GIVE DATA ELEMENT AS ZAMT.
- ☞ GIVE DOMAIN NAME AS ZAMT.
- ☞ SPECIFY THE DATA TYPE AS CURR, LENGTH AS 5.
- ☞ SAVE AND ACTIVATE THE DOMAIN, DATA ELEMENT.
- ☞ SIMILARLY CREATE ANOTHER FIELD AMT_CURR.
- ☞ GIVE THE DATA ELEMENT AS ZCURR.
- ☞ GIVE THE DOMAIN AS ZCURR.
- ☞ SPECIFY THE DATA TYPE AS CUKY, LENGTH AS 5.
- ☞ SAVE AND ACTIVATE DOMAIN, DATA ELEMENT.
- ☞ FINALLY CLICK ON CURRENCY / QUANTITY FIELDS AND SPECIFY REFERENCE TABLE NAME AND FIELD NAME.
- ☞ REPEAT THE SAME PROCEDURE FOR QUANTITY FIELDS ALSO.

FIELD	DATA ELEMENT	DATA TYPE	LENGTH	DECIMALS	DESCRIPTION
AMT	ZAMT	CURR	5	2	AMOUNT
AMT_CURR	ZCURR	CUKY	5	0	CURRENCY
QTY	ZQTY	QUAN	5	0	QUANTITY
QTY_UNITS	ZUNITS	UNIT	3	0	UNITS

DATA VALIDATIONS

A **primary key** is a field or group of fields that uniquely identify a record in a table. Primary Key fields cannot be NULL and cannot contain duplicate values. If you want to link two tables, the primary key of one table will be added to another table where primary key of first table will become the **foreign key** of second table.

FOREIGN KEY RELATIONSHIP:

A RELATION BETWEEN TWO TABLES FOR VALIDATING THE DATA IS CALLED FOREIGN KEY RELATIONSHIP.

VALIDATION:

CHECKING THE VALUE WHETHER IT IS CORRECT OR NOT IS CALLED VALIDATION.

FOREIGN KEY TABLE:

A TABLE WHICH IS RELATED OR LINKED WITH CHECK TABLE FOR VALIDATING THE DATA IS CALLED FOREIGN KEY TABLE.

FOREIGN KEY FIELDS:

A FIELD IN THE FOREIGN KEY TABLE WHICH IS LINKED WITH A FIELD IN THE CHECK TABLE IS CALLED FOREIGN KEY FIELD.

EXAMPLES FOR CHECK TABLES : KNA1, LFA1, MARA, T001W, TCURR.

EXAMPLES FOR FOREIGN KEY TABLES : KNBK, KNB1, LFB1, LFBK, MAK, MARC, MARD

- THE DATA VALIDATIONS OF A FIELD ARE SPECIFIED / MAINTAINED AT DOMAIN LEVEL.
- IN DOMAIN, IT IS MAINTAINED AT 'VALUE RANGE'.

VALUE RANGE		
SINGLE / FIXED VALUES	INTERVALS	VALUE TABLE
VALUES ARE FIXED & FEW IN NUMBER EG: SGENDER, SBRANCH	IF THE GIVEN VALUE IS A VALUE WITHIN A RANGE. EG: SNO	WHEN LIST OF POSSIBLE VALUES ARE HUGE IN NUMBER, WE MAINTAIN THE DATA IN A SEPERATE TABLE AND ASSIGN THAT TABLE AS VALUE TABLE.

- TABLE IS INDEPENDENT OF DOMAIN.
- VALUE TABLE DOES NOT PROHIBIT THE VALUES WHICH ARE GIVEN OTHER THAN THE VALUES GIVEN IN VALUE TABLE.
- BOTH FIXED VALUES AND INTERVALS STRICTLY PROHIBIT THE VALUES THAT ARE EITHER OUTSIDE THE RANGE NOT SPECIFIED IN THE FIXED VALUE.
- BUT VALUE TABLE DOES NOT PROHIBIT THE VALUES THAT ARE NOT MAINTAINED IN VALUE TABLE.

VALUE TABLE:

IT IS A TABLE WHICH CONTAINS LIST OF POSSIBLE VALUES FOR A TABLE.

IT IS MAINTAINED AT DOMAIN LEVEL.

HERE IT IS EASY TO MAINTAIN AT VALUE TABLE THAN AT FIXED VALUE AT DOMAIN LEVEL.

THE DATABASE WE USE IN SAP IS RELATIONAL DATABASE.
HERE TABLES ARE RELATED TO EACH OTHER.

DATABASE	
LOGICAL DATABASE	RELATIONAL DATABASE
TABLES ARE ARRANGED IN TREE STRUCTURE. ONLY IN HR MODULE, LOGICAL DATA BASE IS STORED	TABLES ARE HORIZONTALLY RELATED TO EACH OTHER. EG:SD, MM, PP, QA, FI TABLES.

SAP HAS A RELATIONAL DATABASE.

- IN RELATION DATABASE, THE TABLES ARE RELATED TO EACH OTHER AND THE RELATIONSHIP IS ESTABLISHED BY MEANS OF COMMON FIELDS BETWEEN THE TABLES.
- THE RELATIONSHIP BETWEEN THE TABLES IS BY MEANS OF FOREIGN KEY RELATIONSHIP.
- WHEN TWO TABLES ARE RELATED TO EACH OTHER BY MEANS OF COMMON FIELDS AND IF THE COMMON FIELD MAKES A PRIMARY KEY IN ONE OF THE TABLE THEN OBVIOUSLY THIS COMMON FIELD WILL BE A NON-KEY FIELD IN OTHER TABLE (CAN BE PRIMARY KEY ALSO).
- THE TABLE IN WHICH THE COMMON FIELD MAKES A PRIMARY KEY IS CALLED AS CHECK TABLE & THE OTHER TABLE IS CALLED AS FOREIGN KEY TABLE.
- THE COMMON FIELD IN THE FOREIGN KEY TABLE IS CALLED AS FOREIGN KEY.
- THIS RELATIONSHIP IS CALLED AS FOREIGN KEY RELATIONSHIP.

FOREIGN KEY RELATIONSHIP:

RELATION BETWEEN 2 TABLES FOR VALIDATING THE DATA IS CALLED FOREIGN KEY RELATIONSHIP.

VALIDATION:

CHECKING THE VALUE WHETHER IT IS CORRECT OR NOT IS CALLED VALIDATION.

FOREIGN KEY TABLE:

A TABLE WHICH IS RELATED OR LINKED WITH CHECK TABLE FOR VALIDATING THE DATA IS CALLED FOREIGN KEY TABLE.

FOREIGN KEY FIELD:

A FIELD IN THE FOREIGN KEY TABLE WHICH IS LINKED WITH A FIELD IN THE CHECK TABLE IS CALLED FOREIGN KEY FIELD.

EG: CHECK TABLES: KNA1, LFA1, MARA, T001W, TCURR.

EG: FOREIGN KEY TABLES: KNBK, KNB1, LFB1, LFBK, MAKT, MARC, MARD.

CHECK TABLE:

IT IS A TABLE THAT CONTAINS LIST OF POSSIBLE VALUES FOR A FIELD (FOREIGN KEY).

NOTE:

FOREIGN KEY, WHEN A VALUE IS ENTERED, IT MAKES A CHECK FOR THE VALUE IN ITS CHECK TABLE. IT ACCEPTS THE VALUE ONLY IF IT IS PRESENT IN ITS CHECKTABLE.

CREATING TABLES AND MAINTAINING THE FOREIGN KEY RELATIONSHIP BETWEEN THEM

Business Scenario:

Suppose we are Maintaining Employee Details and Every Employee has to be assigned to a Corresponding Department. So the prerequisite is, all the departments should be available first and then when an Employee master data is created, we can assign only the available Department to the Employee.

Table 1: Department Table (Independent Table / Master Table / Check Table)

Table 2: Employee Table (Dependent Table / Child Table / Foreign Key Table)

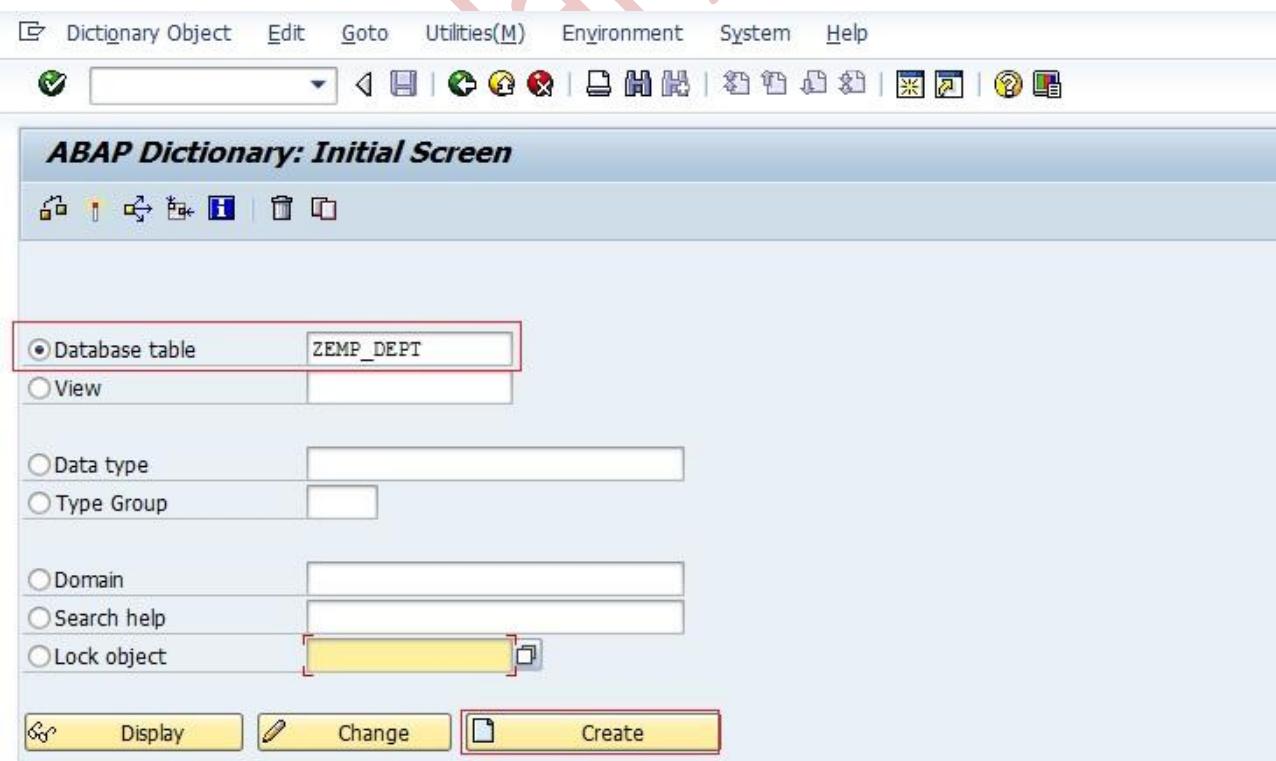
Let's create the tables with all field details,
Create the Table Maintenance Generator,
Maintaining the Foreign Key Relationships,
Maintaining the Entries and Validating the Entries.

CREATING THE DEPARTMENT TABLE: ZEMP_DEPT

Step1

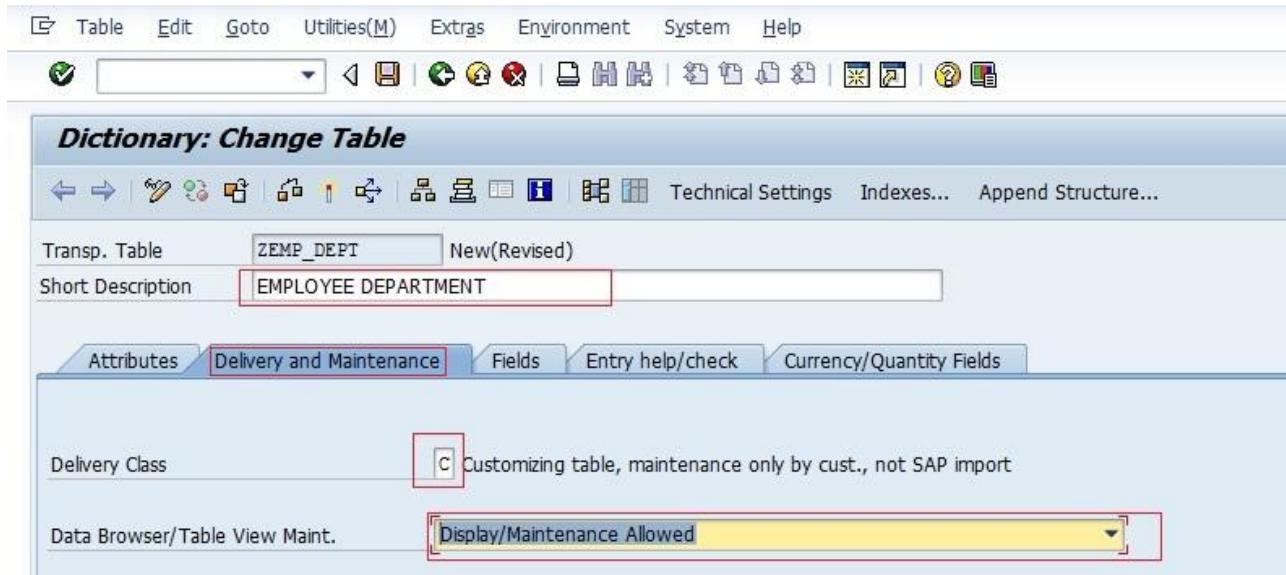
Go to TCODE- SE11

Select the Database Table Radio Button and Provide the Table Name: ZEMP_DEPT and Click on the 'CREATE' Button.



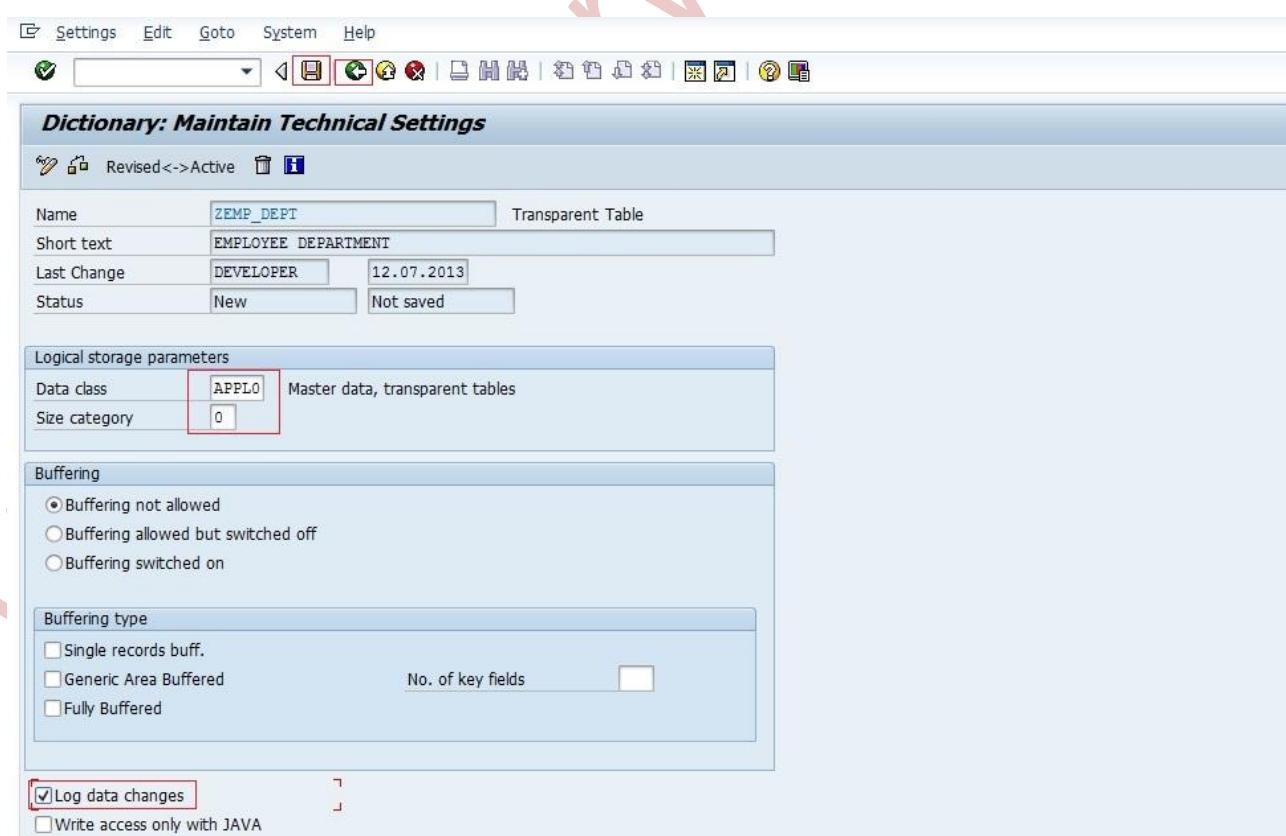
Step 2

Provide the Short Description; Click the 'Delivery and Maintenance' tab. Provide the Delivery Class as 'C' and in the Data Browser/Table View Maintenance Select the Display / Maintenance allowed. At last click on the Technical Settings Button on the Application tool bar.



Step 3

Provide the Data Class 'APPLO' and the Size Category as: '0'. Click on the SAVE button and click on the BACK button.



Step 4

First create all the required Domains and Data Elements as per the required data types and length so that we can assign the data element against the fields. Provide all the fields and corresponding data elements. Mark on fields MANDT and DEPT_ID as primary key for this Department table. Finally click on the Activate Button to activate the Table.

Field	Key	Data element	Data Type	Length	Deci...	Short Description
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> MANDT	CLNT	3	0	Client
DEPT_ID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> ZDEPT_ID	CHAR	20	0	DEPT_ID
DEPT_NAME	<input type="checkbox"/>	<input type="checkbox"/> ZDEPT_NAME	CHAR	15	0	name of the dept.

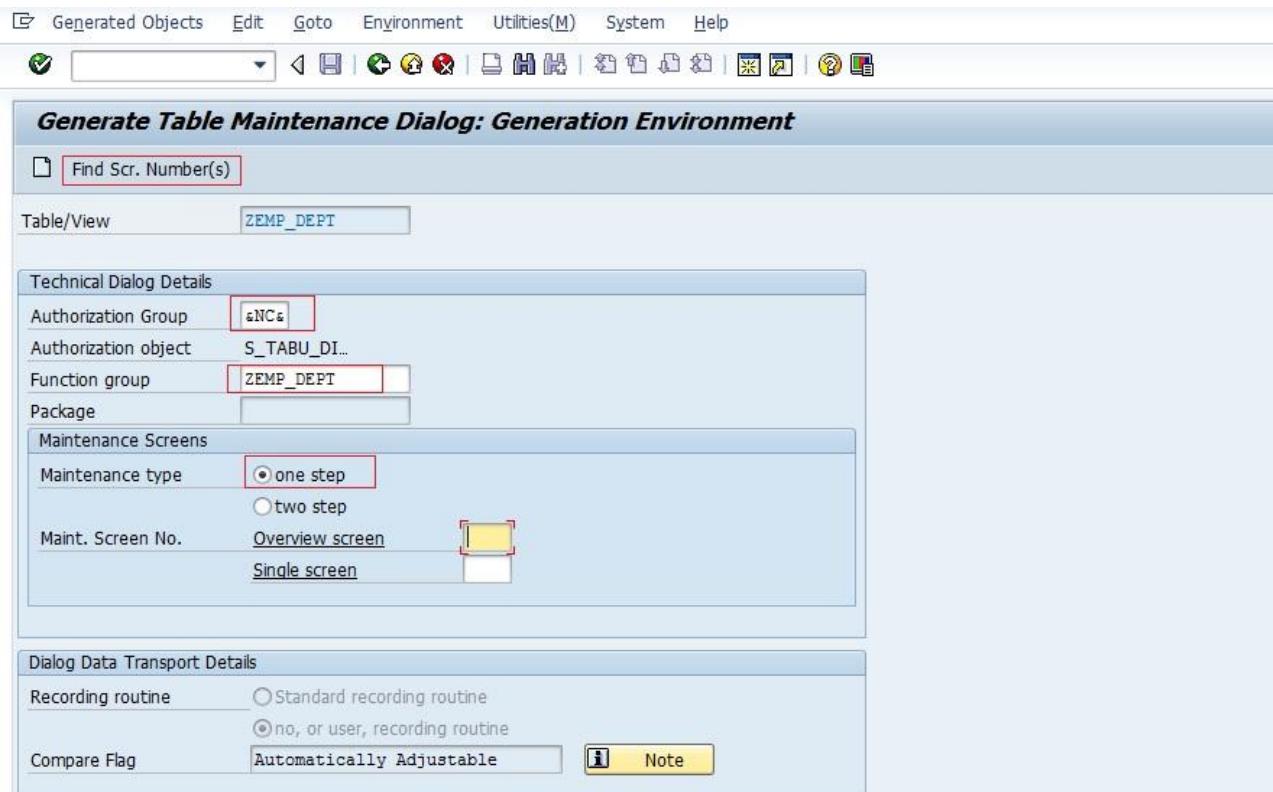
Step 5

In order to maintain multiple entries in the table at a time, we need to create a Table Maintenance Generator (TMG). Click on the Utilities, and then click on the Table Maintenance Generator Menu Bar.

- Utilities(M)
 - Table Maintenance Generator

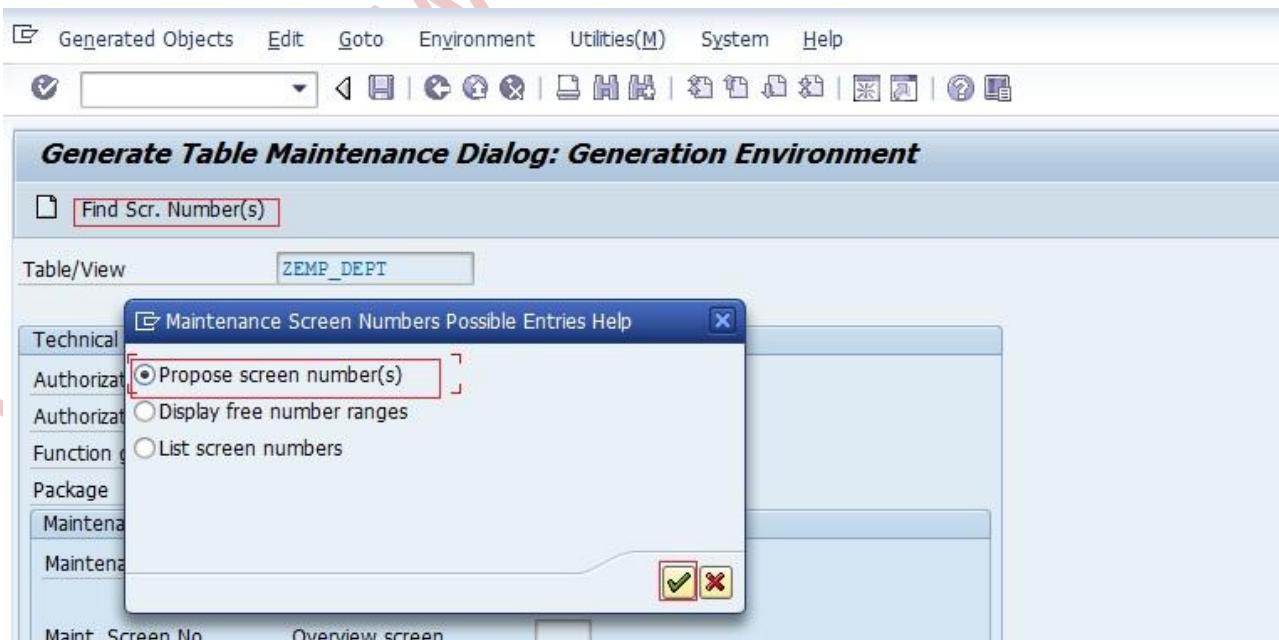
Step 6

First provide the Authorization Object: '&NC&', then provide the Function Group Name as 'ZEMP_DEPT' (The TMG Function group name normally same as the table name and it is going to hold all the screen logics that are automatically generated when TMG is created). Select Maintenance type as One Step and then click on the button 'Find Scr. number(s)'.



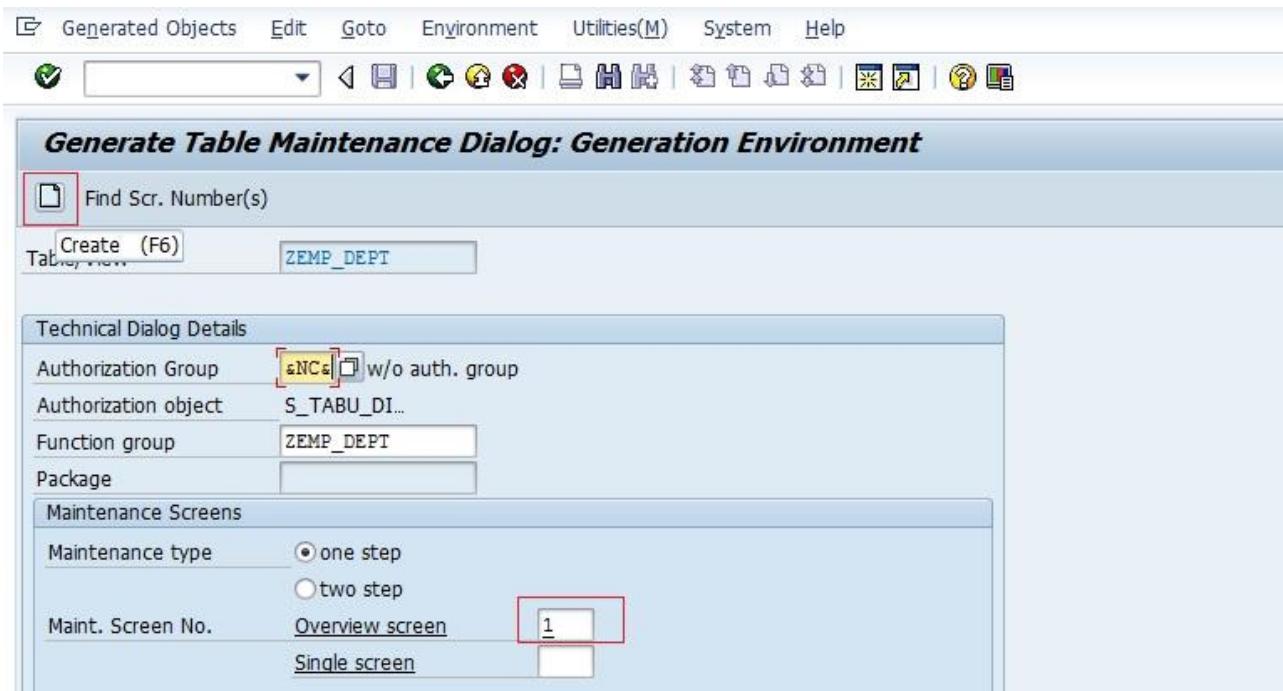
Step 7

Select the First radio button 'Propose Screen number(s)' and click on the Tick Button.



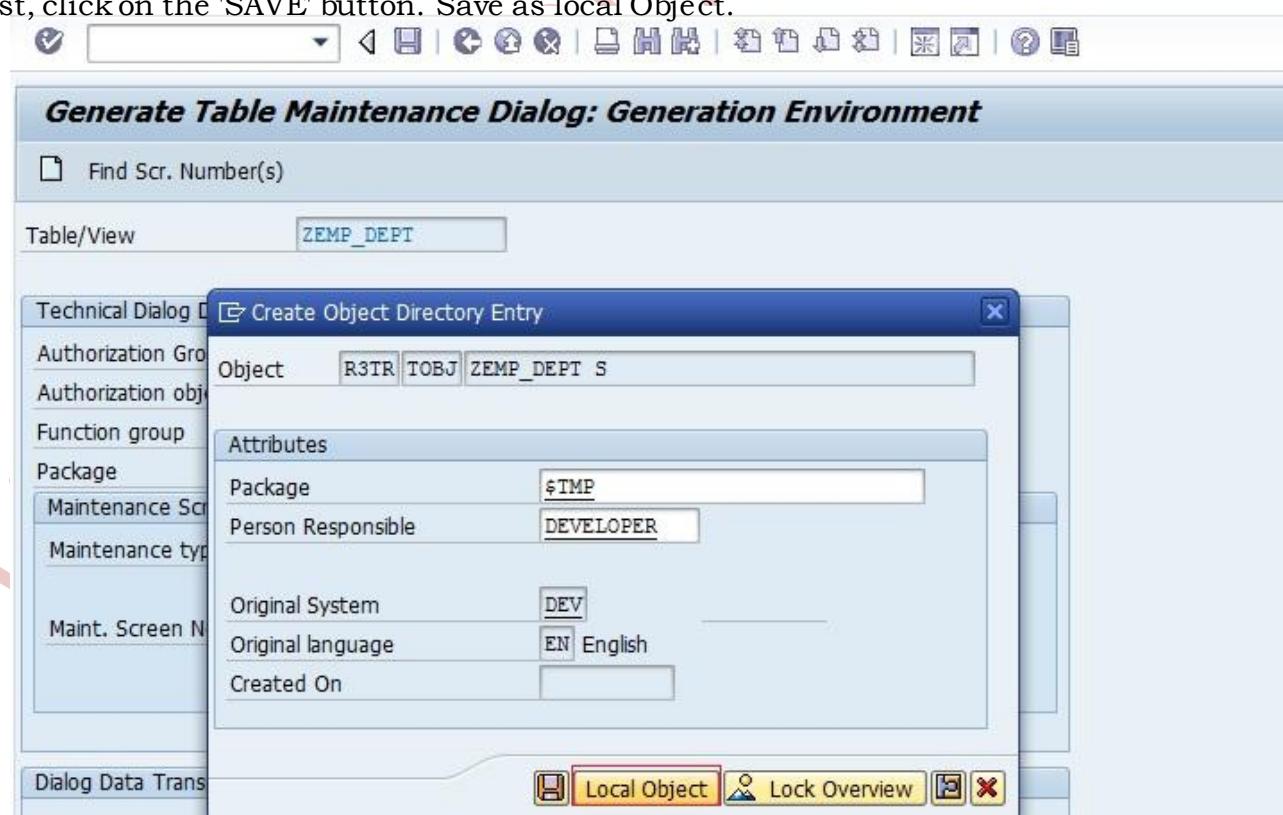
Step 8

Automatically the Overview Screen is filled with some value and then click on the 'CREATE' button to create the TMG as shown on the below diagram.



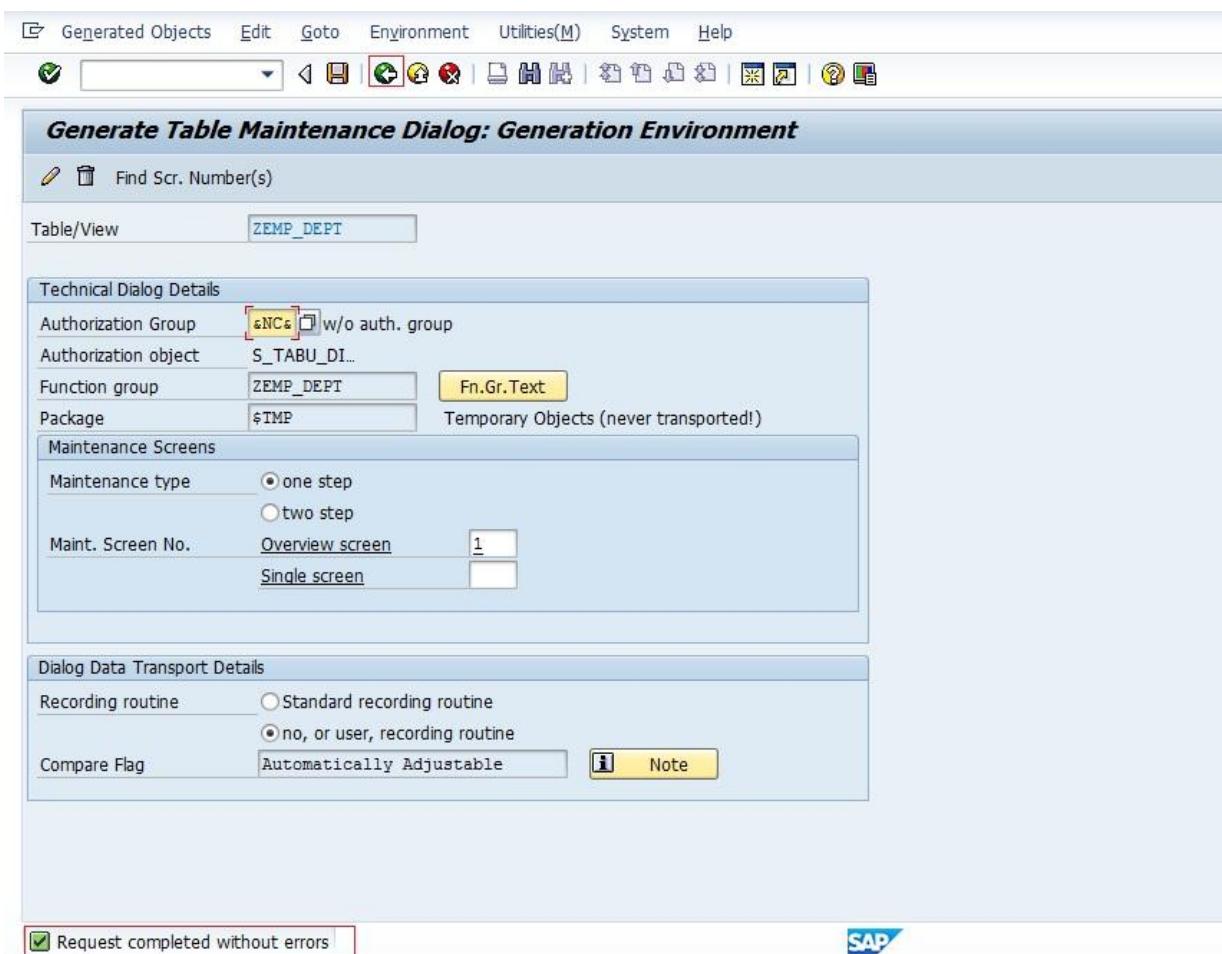
Step 9

At last, click on the 'SAVE' button. Save as local Object.



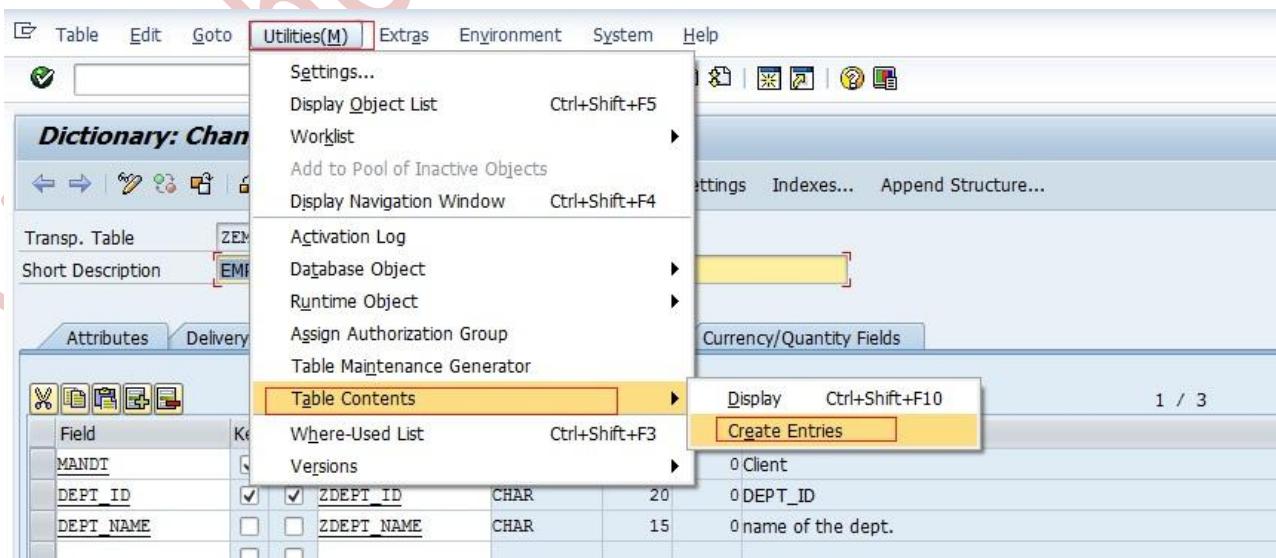
Step 10

If the TMG for the Table is created successfully then a status message is shown as below: 'Request Completed without Errors' and then click on the back button.



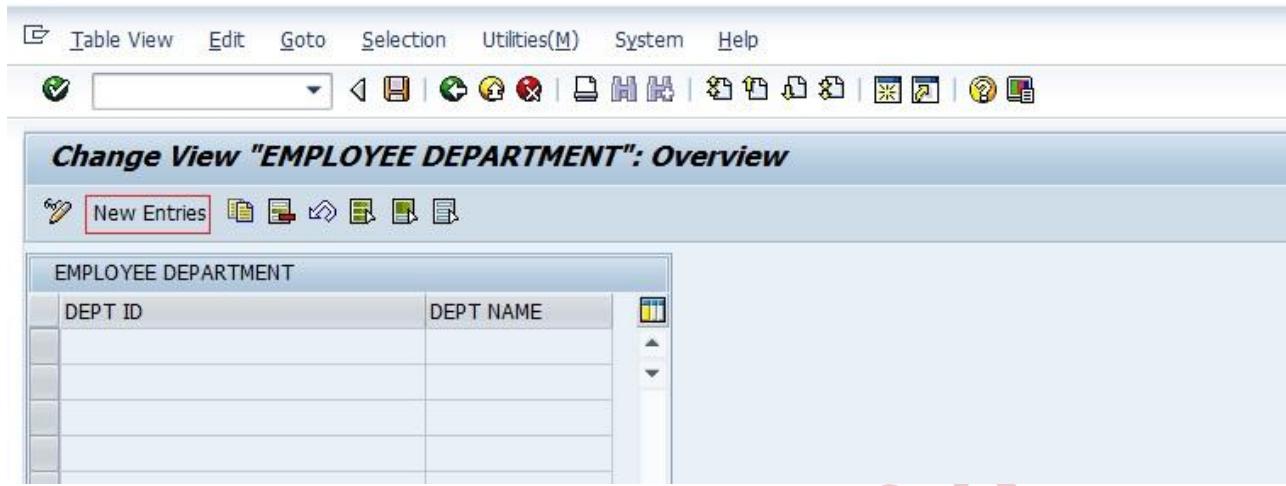
Step 11

Go to UTILITIES, then TABLE CONTENTS and then click on CREATE ENTRIES.



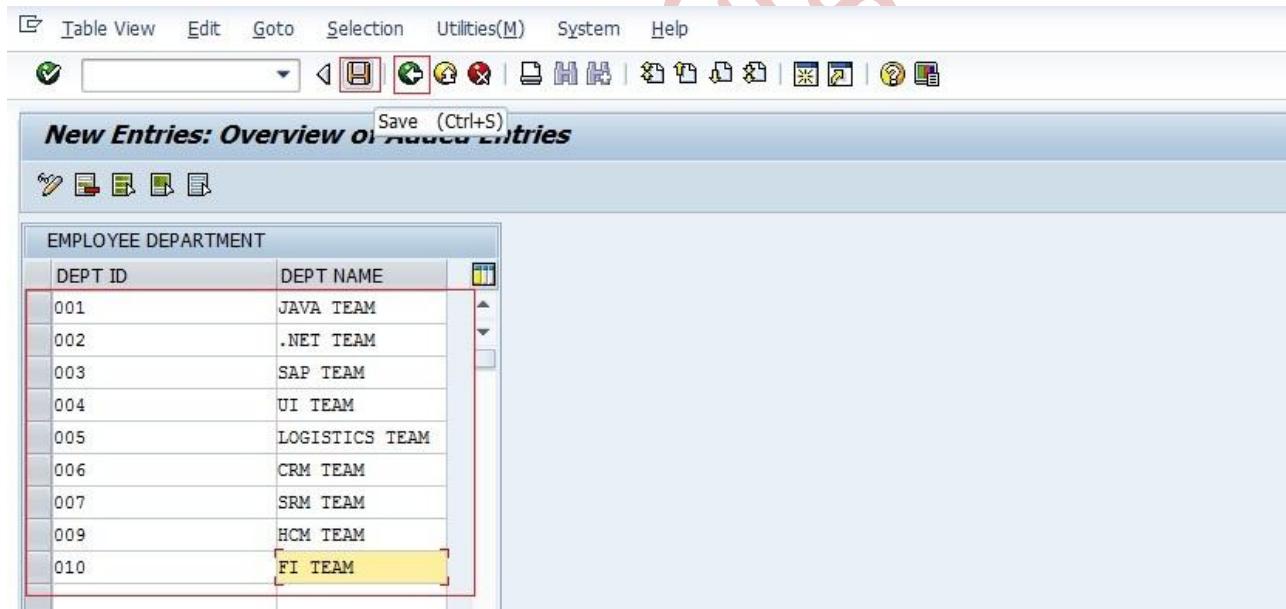
Step 12

Click on NEW ENTRIES button.



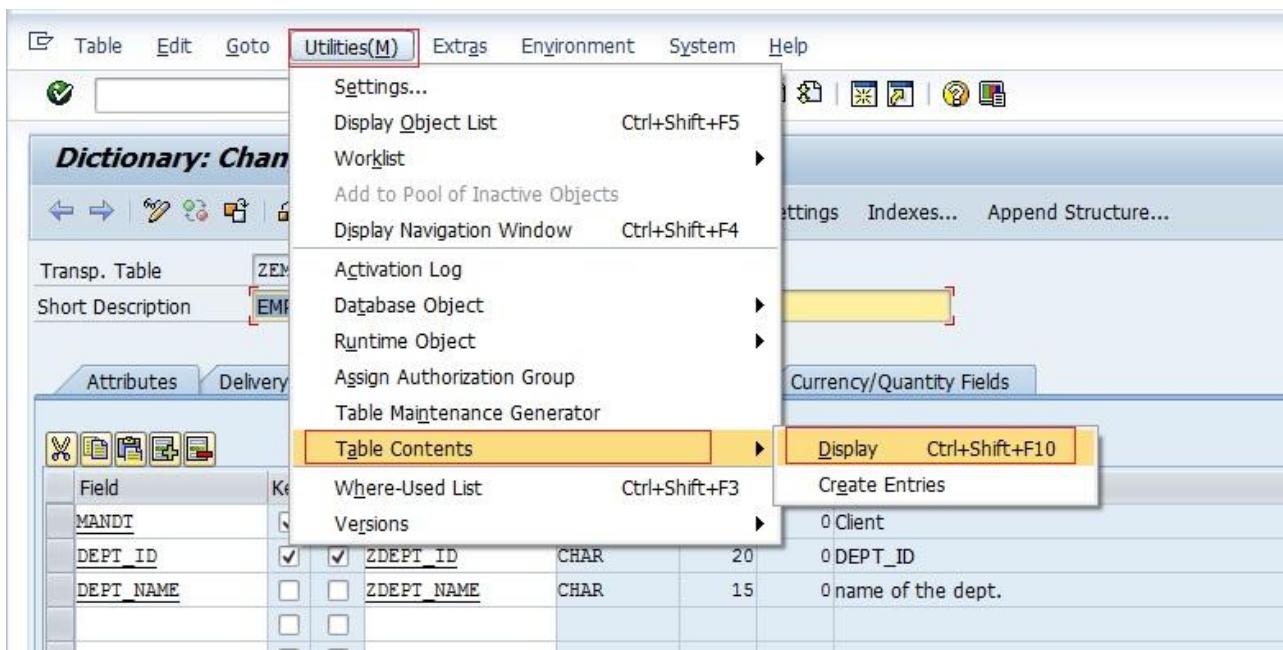
Step 13

Maintain the DEPT ID and DEPT NAME and click on the 'SAVE' button and at last click on the BACK button.



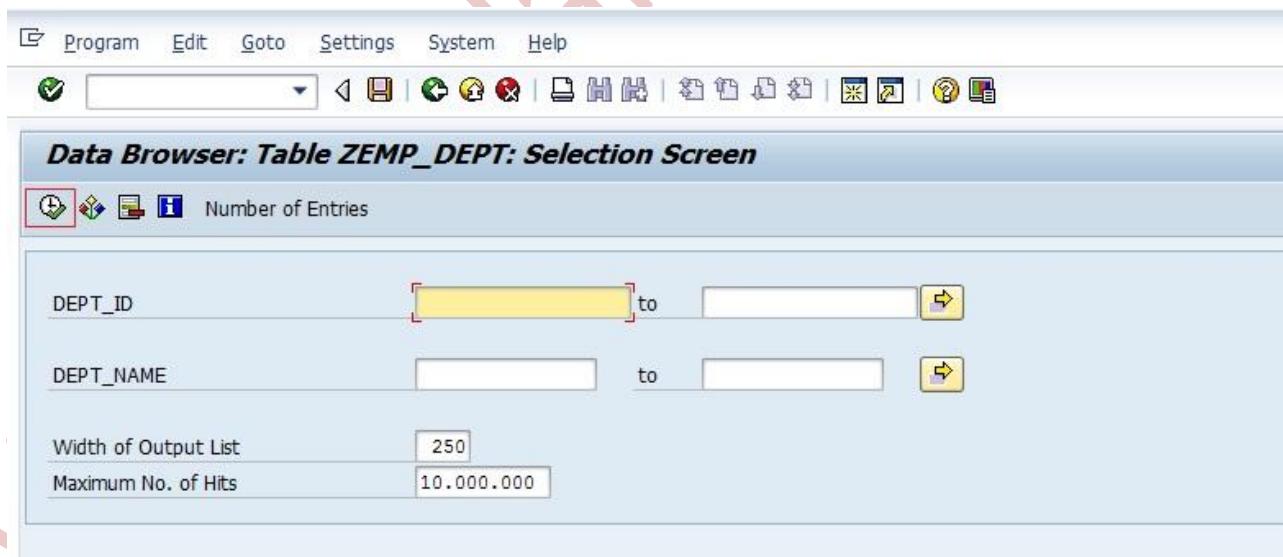
Step 14

Now click on 'UTILITIES', then 'TABLE CONTENTS' and then click on 'DISPLAY'.



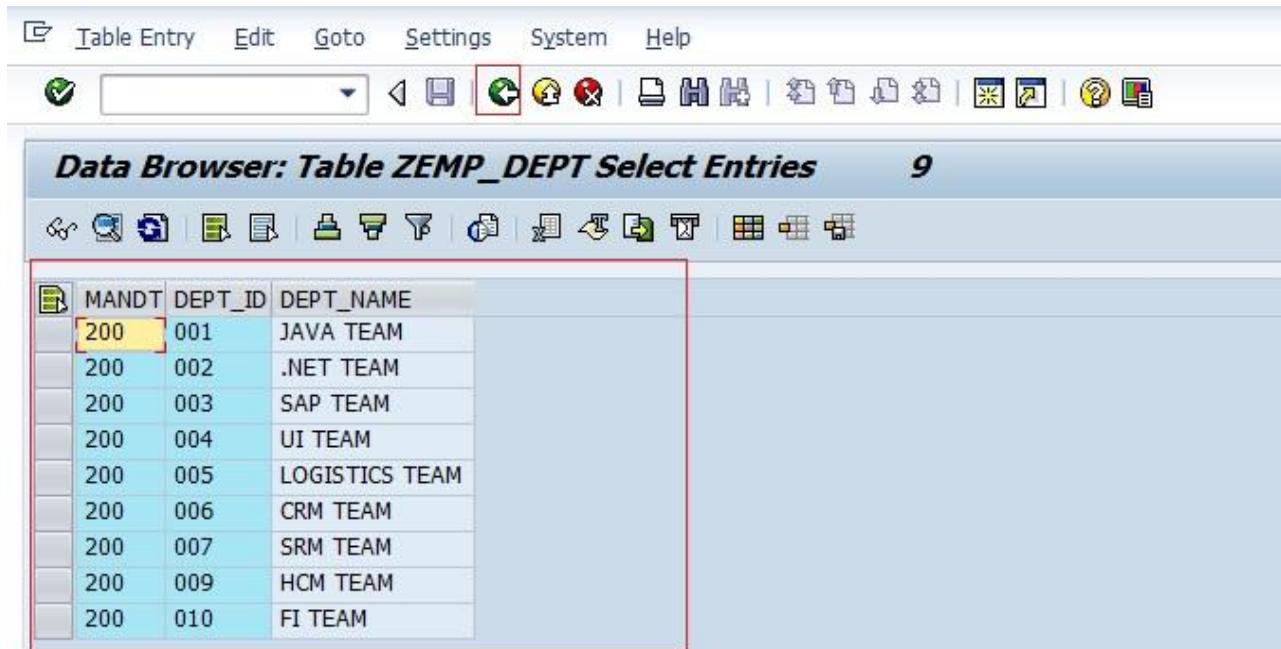
Step 15

Click on the 'EXECUTE' button or else click the F8 button to see all the entries of the table.



Step 16

Now all the required Departments are available with their DEPT ID AND DEPT NAME.



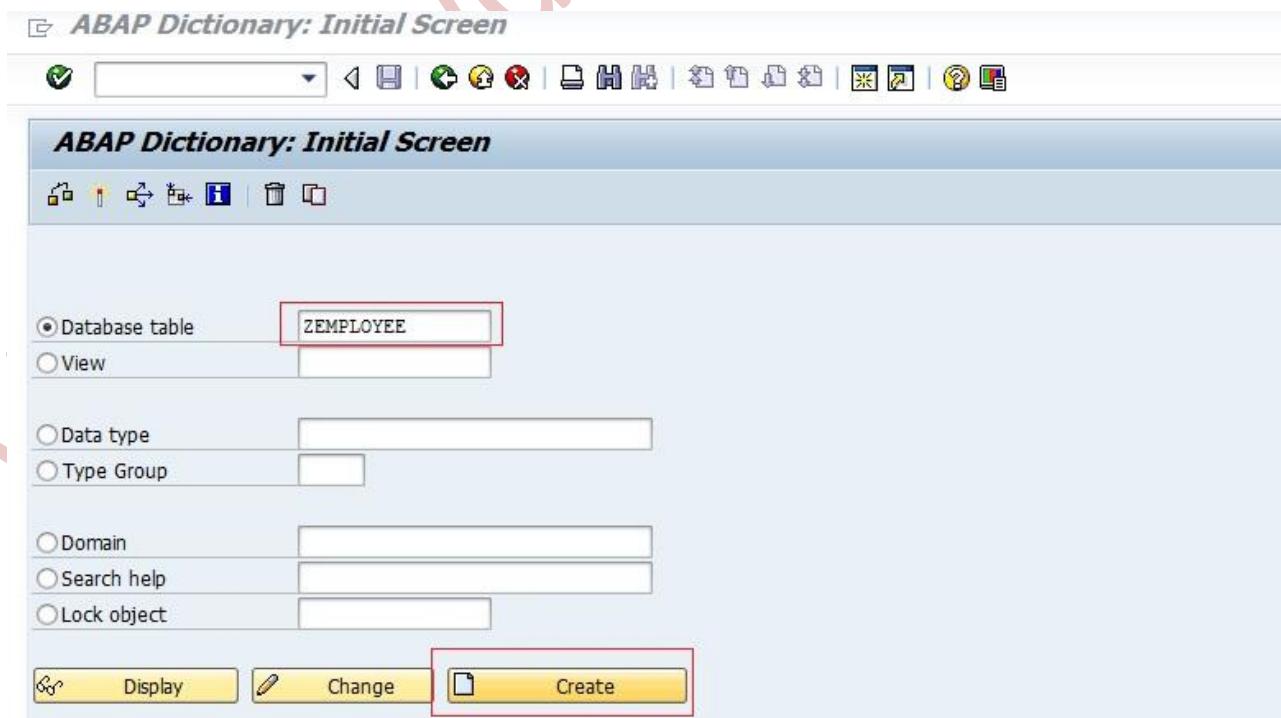
MANDT	DEPT_ID	DEPT_NAME
200	001	JAVA TEAM
200	002	.NET TEAM
200	003	SAP TEAM
200	004	UI TEAM
200	005	LOGISTICS TEAM
200	006	CRM TEAM
200	007	SRM TEAM
200	009	HCM TEAM
200	010	FI TEAM

LET'S CREATE SECOND TABLE

CREATING THE EMPLOYEE TABLE:

Step 1

Go to TCODE- SE11, Select the Database table Radio button and provide the table name 'ZEMPLOYEE' and then click on the CREATE button.



Database table

View

Data type

Type Group

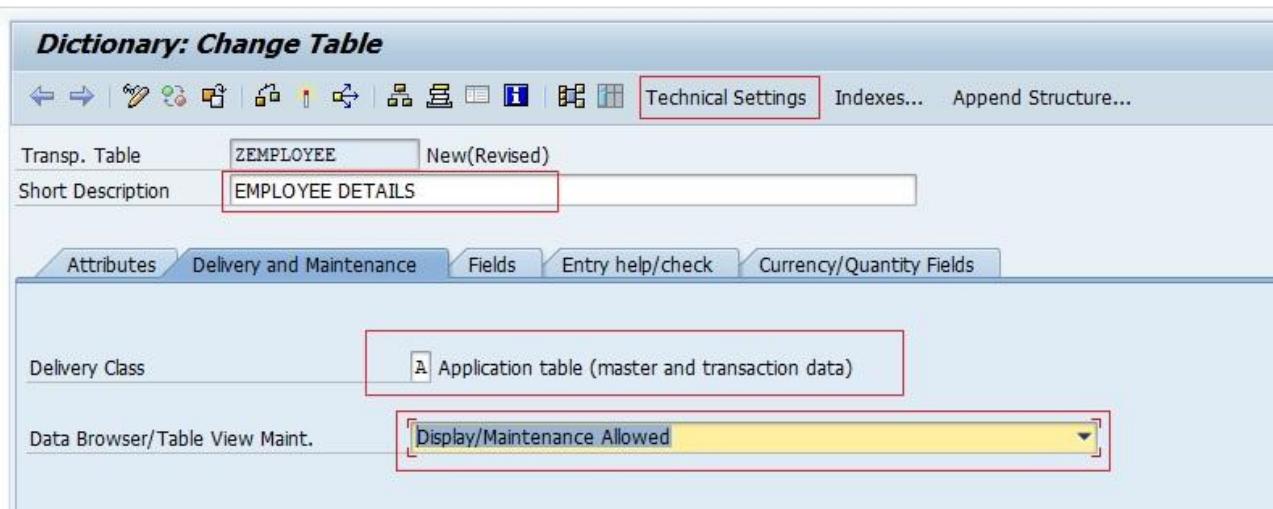
Domain

Search help

Lock object

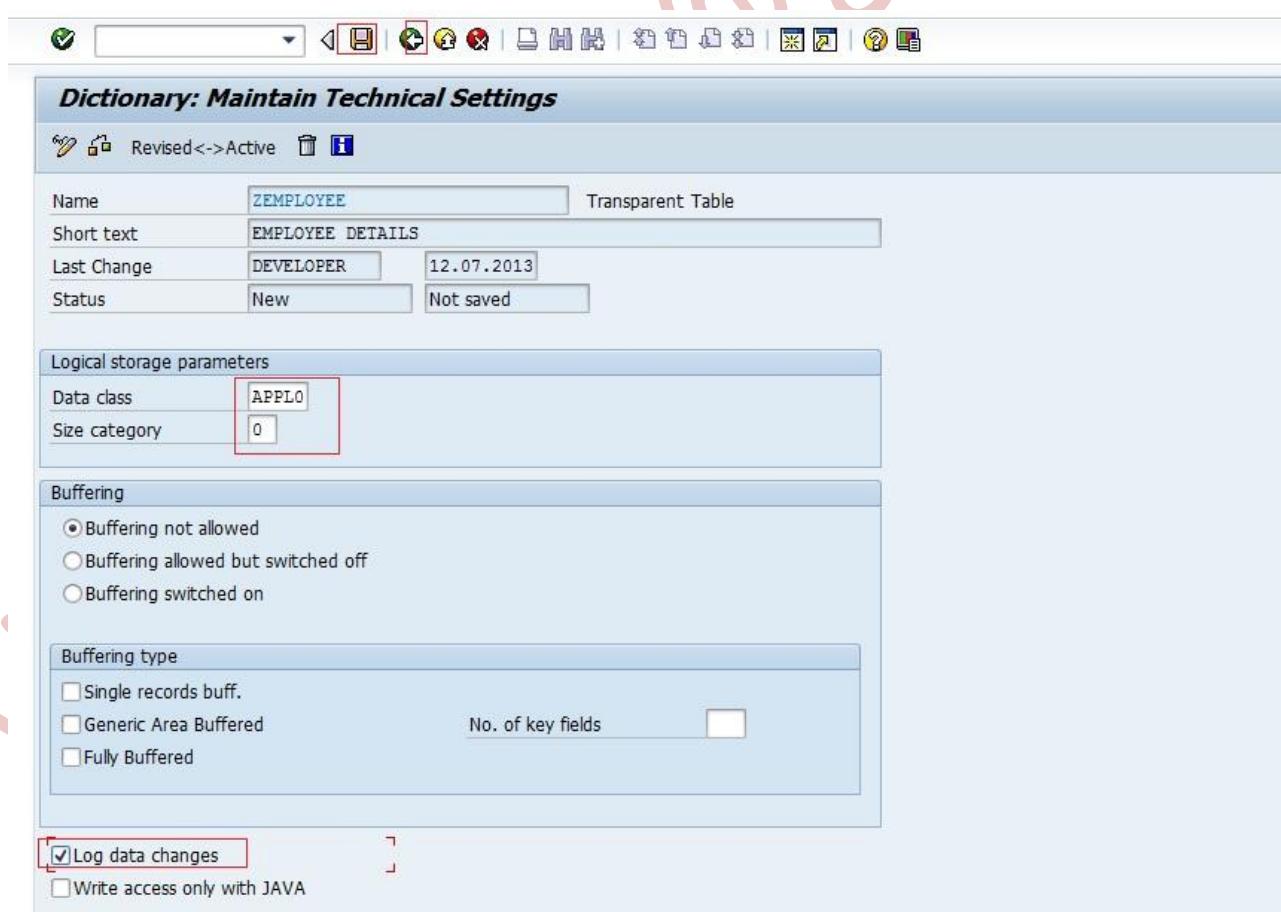
Step 2

Provide the short description, Provide Delivery Class as 'A' and choose 'Display/Maintenance Allowed'. At last click on the 'TECHNICAL SETTINGS' button.



Step 3

Provide the delivery class as 'APPL0' and the Size Category as '0'. click on SAVE and then click on the BACK button.



Step 4

Create all the domains and data elements as per the requirement. Provide the field names with appropriate data elements. Select the MANDT and the EMP_ID field as the Primary key fields. Maintain the data element 'ZDEPT_ID' against the DEPT field, the same data element as used the above DEPT table.

Dictionary: Change Table							
Technical Settings Indexes... Append Structure...							
Transp. Table	ZEMPLOYEE	Active					
Short Description	EMPLOYEE DETAILS						
Attributes	Delivery and Maintenance	Fields	Entry help/check		Currency/Quantity Fields		
			Srch Help		Predefined Type		
1 / 6							
Field	Key	Init...	Expand include	Data Type	Length	Deci...	Short Description
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0	Client
EMP_ID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ZEMP_ID	CHAR	8	0	Employee Id
EMP_NAME	<input type="checkbox"/>	<input type="checkbox"/>	ZEMP_NAME	CHAR	25	0	employee name
D_O_BIRTH	<input type="checkbox"/>	<input type="checkbox"/>	DATUM	DATS	8	0	Date
D_O_JOIN	<input type="checkbox"/>	<input type="checkbox"/>	DATUM	DATS	8	0	Date
DEPT	<input type="checkbox"/>	<input type="checkbox"/>	ZDEPT_ID	CHAR	20	0	DEPT_ID

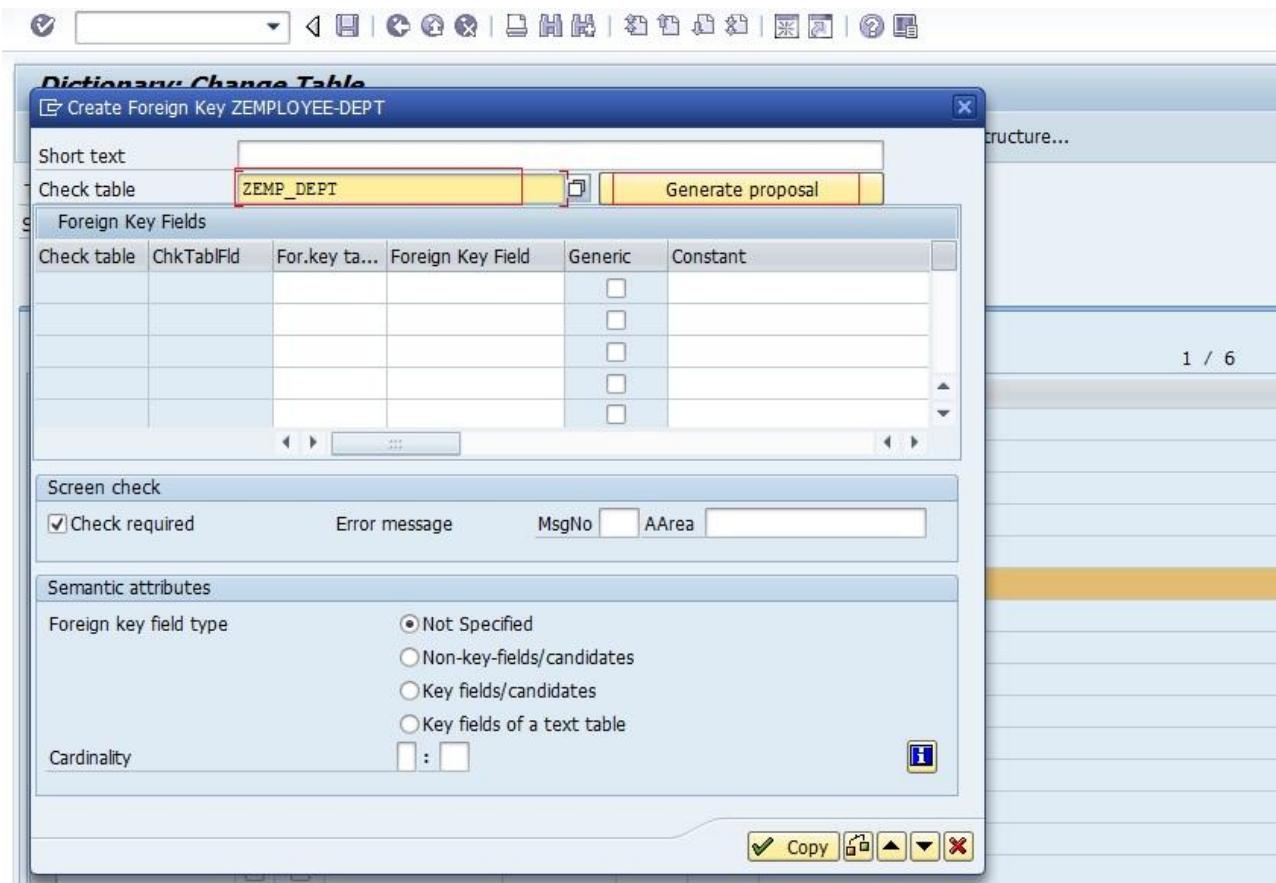
Step 5

Now we have to maintain the foreign key for the DEPT field. So select the particular row and click on the FOREIGN KEY Button.

Dictionary: Change Table							
Technical Settings Indexes... Append Structure...							
Transp. Table	ZEMPLOYEE	Active					
Short Description	EMPLOYEE DETAILS						
Attributes	Delivery and Maintenance	Fields	Entry help/check		Currency/Quantity Fields		
			Srch Help	Predefined Type	1 / 6	Group	
Field	Key	Init...	Data element	Data Type	Length	Deci...	Short Description
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0	Client
EMP_ID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ZEMP_ID	CHAR	8	0	Employee Id
EMP_NAME	<input type="checkbox"/>	<input type="checkbox"/>	ZEMP_NAME	CHAR	25	0	employee name
D_O_BIRTH	<input type="checkbox"/>	<input type="checkbox"/>	DATUM	DATS	8	0	Date
D_O_JOIN	<input type="checkbox"/>	<input type="checkbox"/>	DATUM	DATS	8	0	Date
DEPT	<input type="checkbox"/>	<input type="checkbox"/>	ZDEPT_ID	CHAR	20	0	DEPT_ID

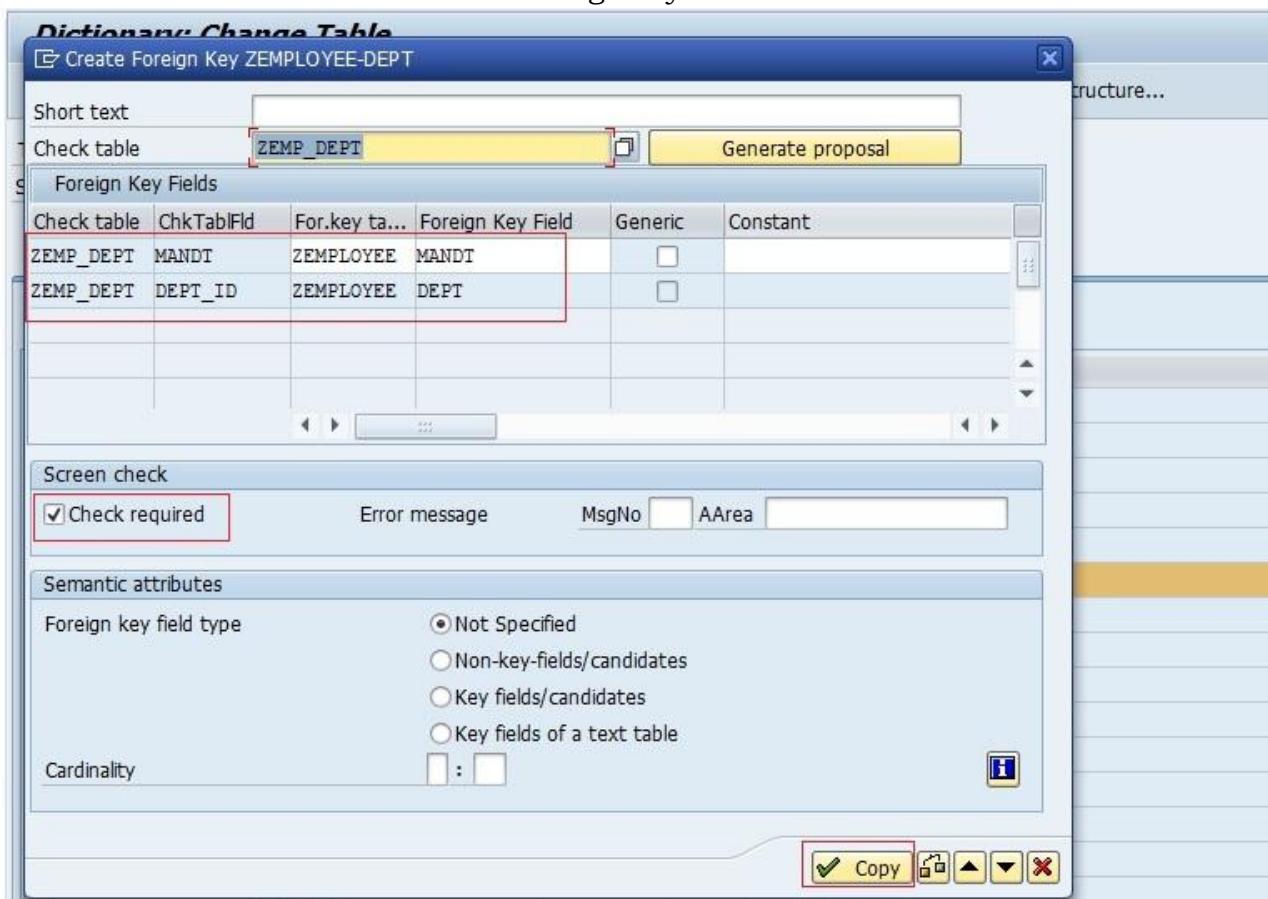
Step 6

Provide the check table name ZEMP_DEPT and click on the GENERATE PROPOSAL button.



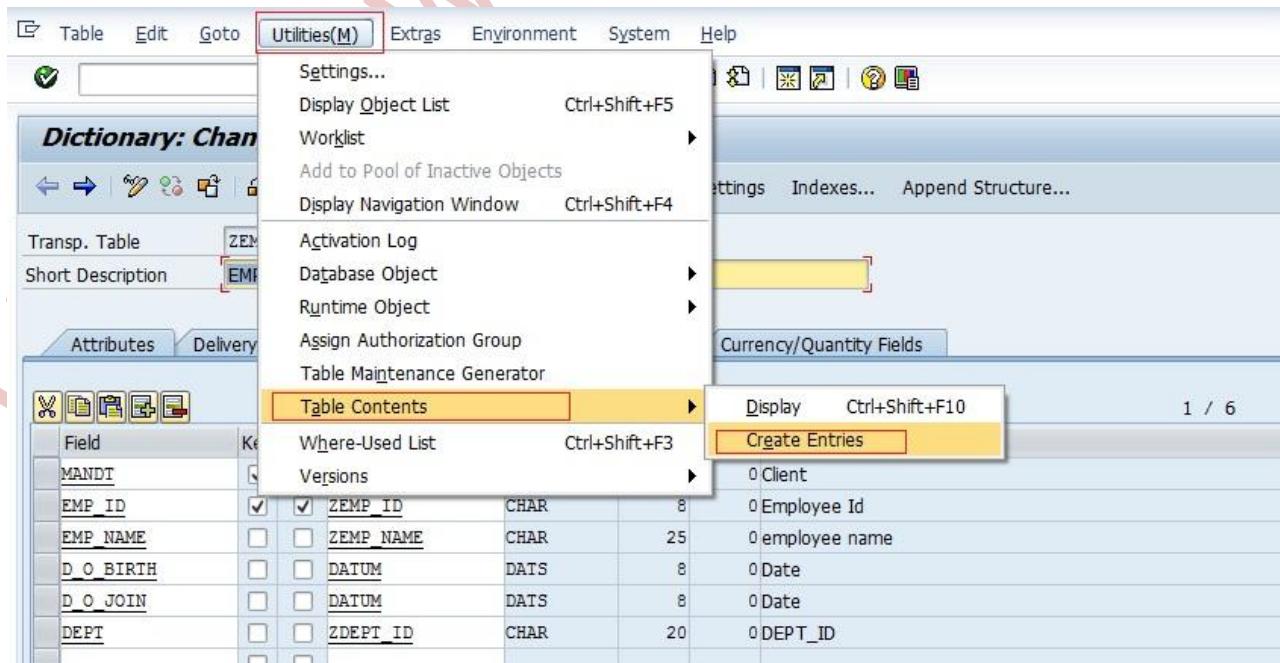
Step 7

After clicking the Generate Proposal button, system will automatically populate the matching domains between the check table and the foreign key table. Just click on the COPY button.



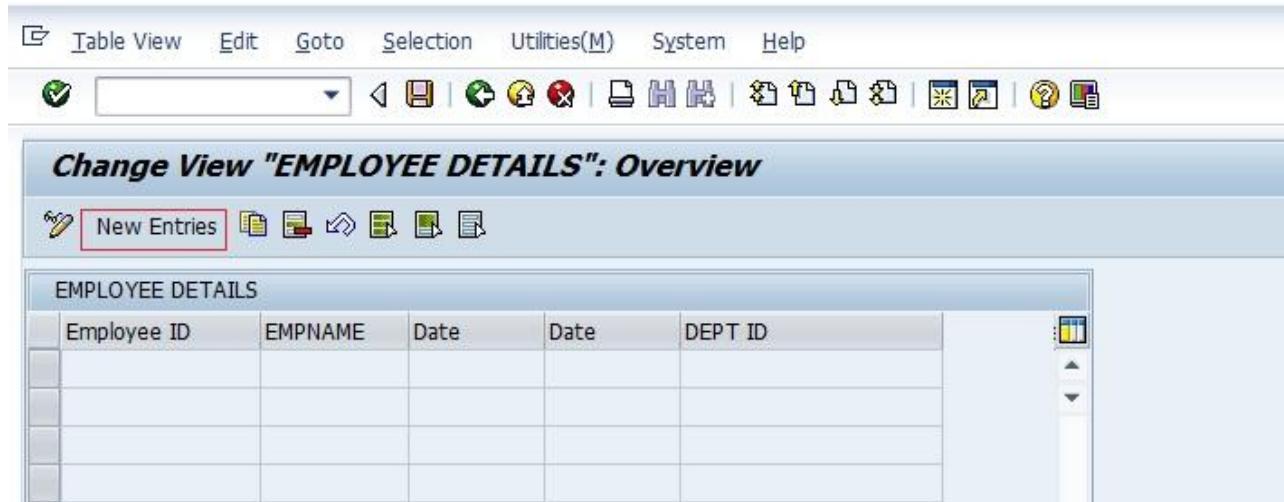
Step 8

Create the TMG for the table and then create the table entries as per the path shown.



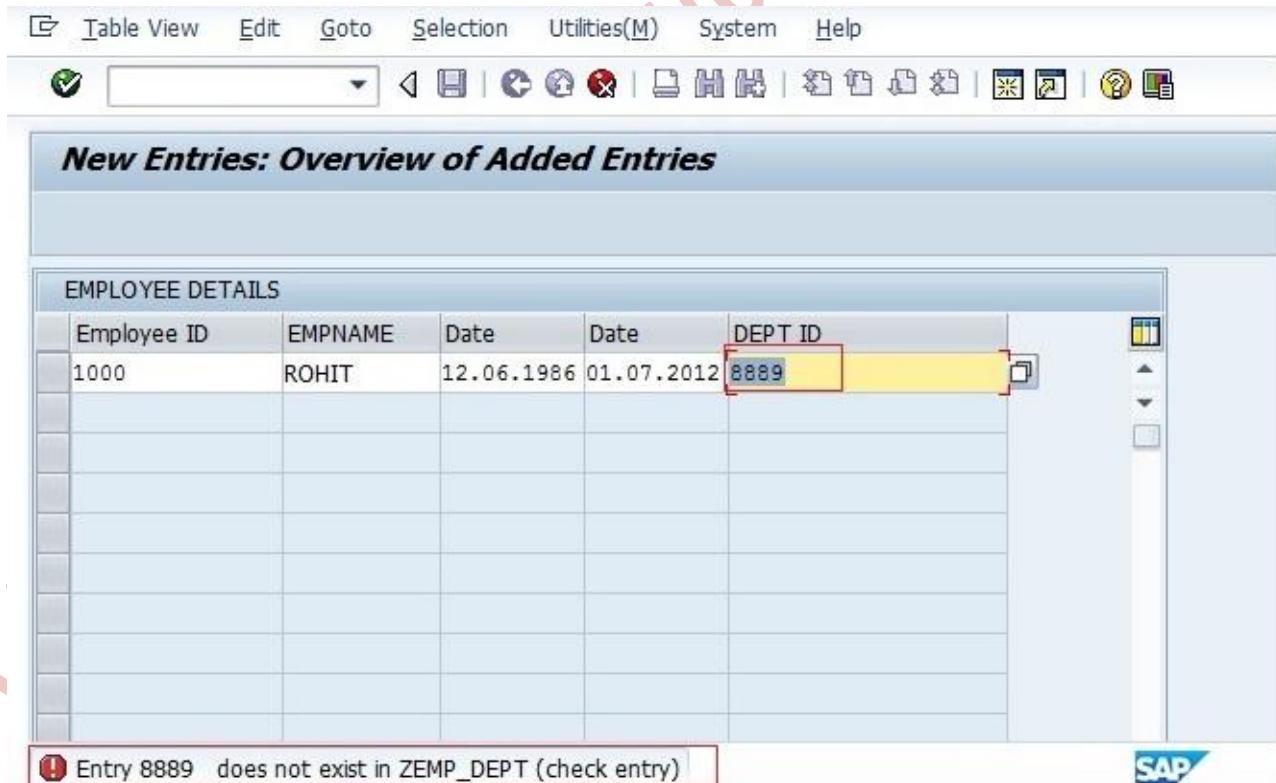
Step 9

Click on CREATE ENTRIES button to create Employee Details.



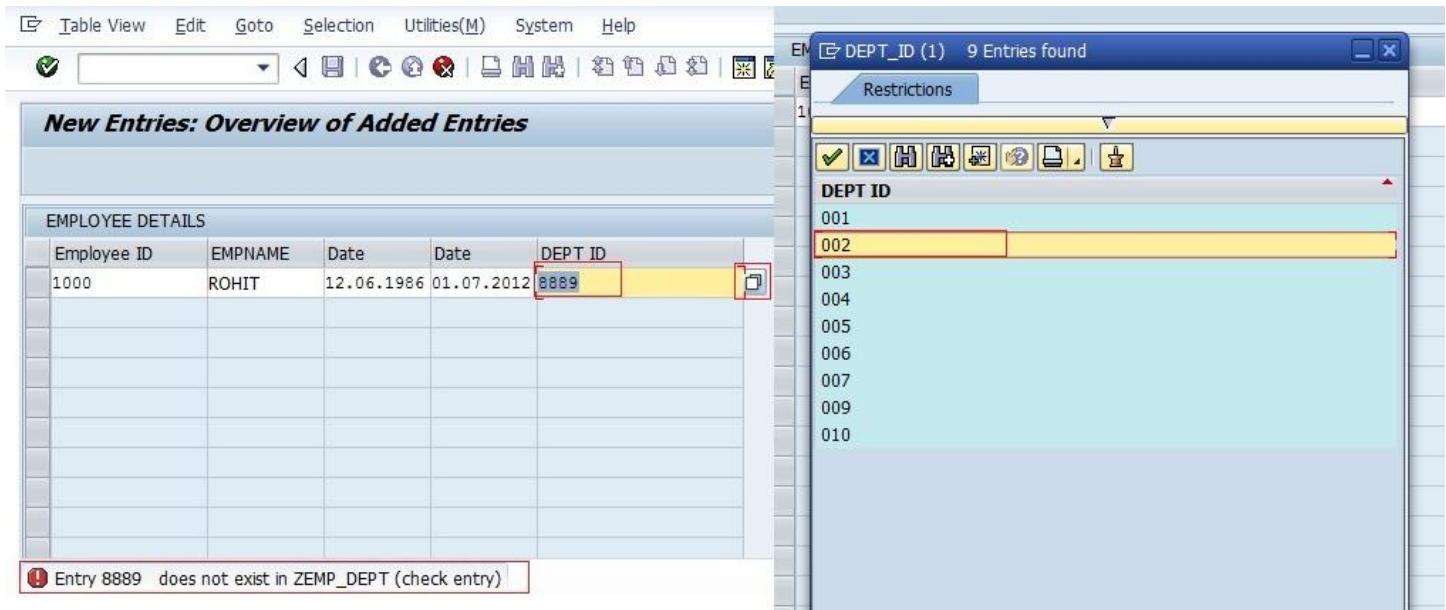
Step 10

Provide the Employee details and provide the DEPT ID as any id that does not exist in the DEPT table and hit the ENTER button and the system will throw the below error message as highlighted below.



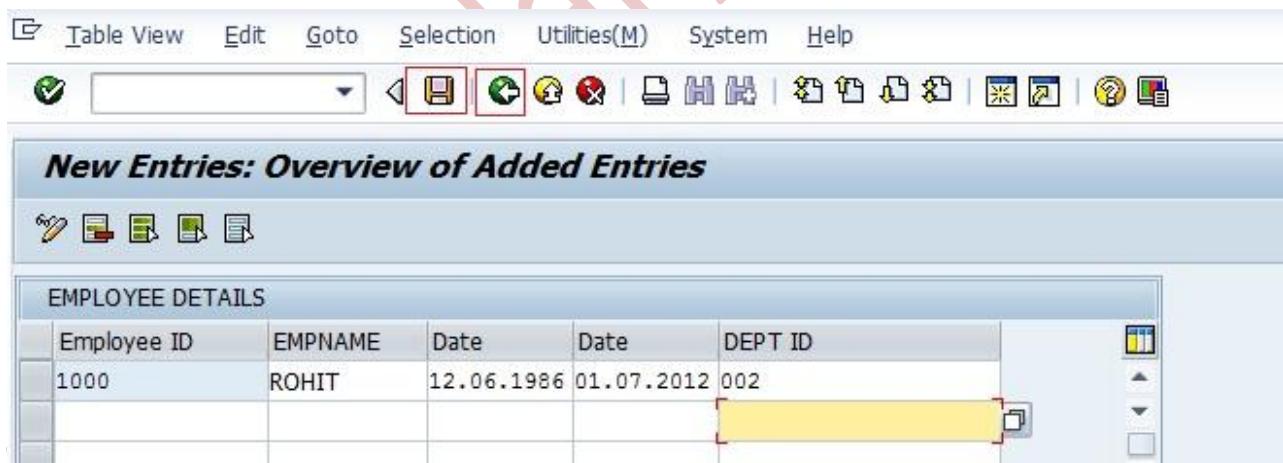
Step 11

So click on the F4 Button and it will show all the possible values from the DEPT table for the field DEPT_ID in the EMPLOYEE TABLE. Choose any one and hit the ENTER button.



Step 12

Save the Entry by clicking on the SAVE button. Similarly create other employee details and SAVE the entries and click on the BACK button.



Step 13

Click on the highlighted button to see the table entries.

Dictionary: Change Table

The screenshot shows the SAP Dictionary interface for the ZEMPLOYEE table. The table has six fields:

Field	Key	Ini...	Data element	Data Type	Length	Deci...	Short Description
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0	Client
EMP_ID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ZEMP_ID	CHAR	8	0	Employee Id
EMP_NAME	<input type="checkbox"/>	<input type="checkbox"/>	ZEMP_NAME	CHAR	25	0	employee name
D_O_BIRTH	<input type="checkbox"/>	<input type="checkbox"/>	DATUM	DATS	8	0	Date
D_O_JOIN	<input type="checkbox"/>	<input type="checkbox"/>	DATUM	DATS	8	0	Date
DEPT	<input type="checkbox"/>	<input type="checkbox"/>	ZDEPT_ID	CHAR	20	0	DEPT_ID

Step 14

Click on the EXECUTE button or hit F8 key.

Data Browser: Table ZEMPLOYEE: Selection Screen

The screenshot shows the SAP Data Browser interface for the ZEMPLOYEE table. It displays selection options for various fields:

Selection Options (F2)	to	
EMP_ID	<input type="text"/>	<input type="button"/>
EMP_NAME	<input type="text"/>	<input type="button"/>
D_O_BIRTH	<input type="text"/>	<input type="button"/>
D_O_JOIN	<input type="text"/>	<input type="button"/>
DEPT	<input type="text"/>	<input type="button"/>

Below the selection options, there are input fields for 'Width of Output List' (250) and 'Maximum No. of Hits' (10.000.000).

Step 15

Now the employee is only assigned to the departments that are only available in the DEPT table.

Data Browser: Table ZEMPLOYEE Select Entries 4

MANDT	EMP_ID	EMP_NAME	D_O_BIRTH	D_O_JOIN	DEPT
200	1000	ROHIT	12.06.1986	01.07.2012	002
200	1001	RAJESH	22.06.1980	01.07.2013	007
200	1002	RAJ	22.06.1976	01.07.2011	005
200	1003	BHUPESH	22.06.1979	01.09.2010	010

DIFFERENCE BETWEEN CHECK TABLE AND VALUE TABLE?

CHECK TABLE	VALUE TABLE
CHECK TABLE CONTAINS THE LIST OF POSSIBLE VALUES FOR A FIELD. CHECK TABLE STRICTLY PREVENTS THE FIELD TO TAKE ANY OTHER VALUE THAT IS NOT THERE IN CHECK TABLE	VALUE TABLE IS A TABLE THAT CONTAINS THE LIST OF POSSIBLE VLAUES FOR A FIELD. FOR THAT FIELD, VALUE TABLE DOESN'T PREVENT TAKING OTHER VALUES WHICH ARE NOT PRESENT IN VALUE TABLE.
DEFINED AT FIELD-LEVEL	DEFINED AT DOMAIN LEVEL.
PROVIDE F4 FUNCTIONALITY	NO F4 FUNCTIONALITY
IT IS USED FOR FIELD VALIDATION	NOT USED FOR VALIDATION. IF WE WANT VALIDATION WE HAVE TO CONVERT VALUE TABLE INTO CHECK TABLE.
VALUE TABLE COMES INTO PICTURE WHERE WE CONSIDER A FIELD IN A TABLE	CHECK TABLE COMES IN TO PICTURE WHERE RELATIONSHIP IS MAINTAINED BETWEEN TABLES.

STEPS TO DEFINE A VALUE TABLE:

STEP1: CREATE A MASTER TABLE FOR CUSTOMER DATA WITH NAME AS YKNA1 & FIELDS AS CUTONO, CNAME, LAND1.

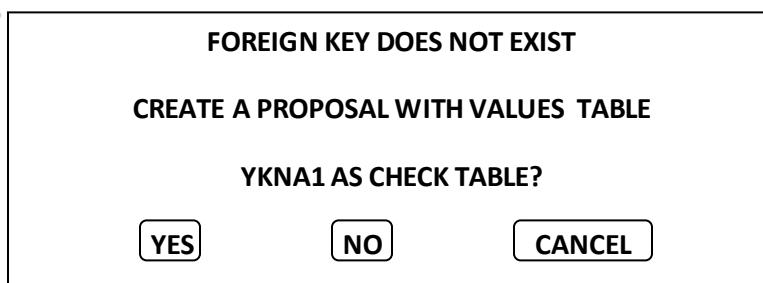
STEP2: CREATE ANOTHER TABLE TO STORE THE BANK DATA BY NAME YKNA_BANK WITH FIELDS AS CNO, BANKS, BANKL.

STEP3: LET US DEFINE THE VALUE TABLE.

- ☞ OPEN THE TABLE YKNA1.
- ☞ DOUBLE CLICK ON THE DATA ELEMENT OF CNO FIELD.
- ☞ DOUBLE CLICK ON THE DOMAIN & CLICK ON VALUE RANGE TAB.
- ☞ DEFINE THE VALUE TABLE AS YKNA1 (MASTER TABLE).
- ☞ SAVE AND ACTIVATE.

CONVERTING VALUE TABLE INTO CHECK TABLE

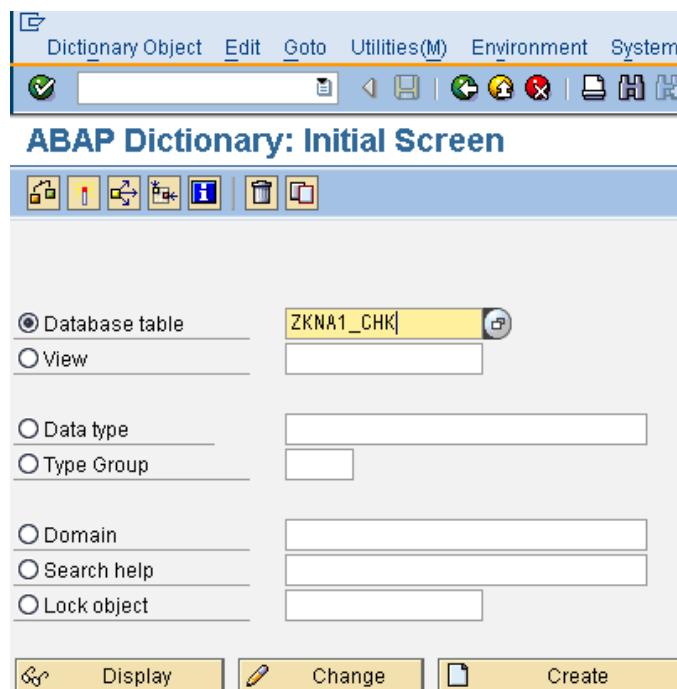
- ☞ GO TO SE11.
- ☞ GIVE THE TABLE NAME AS YKNA_BANK.
- ☞ CLICK ON CHANGE.
- ☞ SELECT THE CNO FIELD CLICK ON FOREIGN KEY BUTTON.
- ☞ A POP UP IS DISPLAYED WITH FOLLOWING MESSAGE



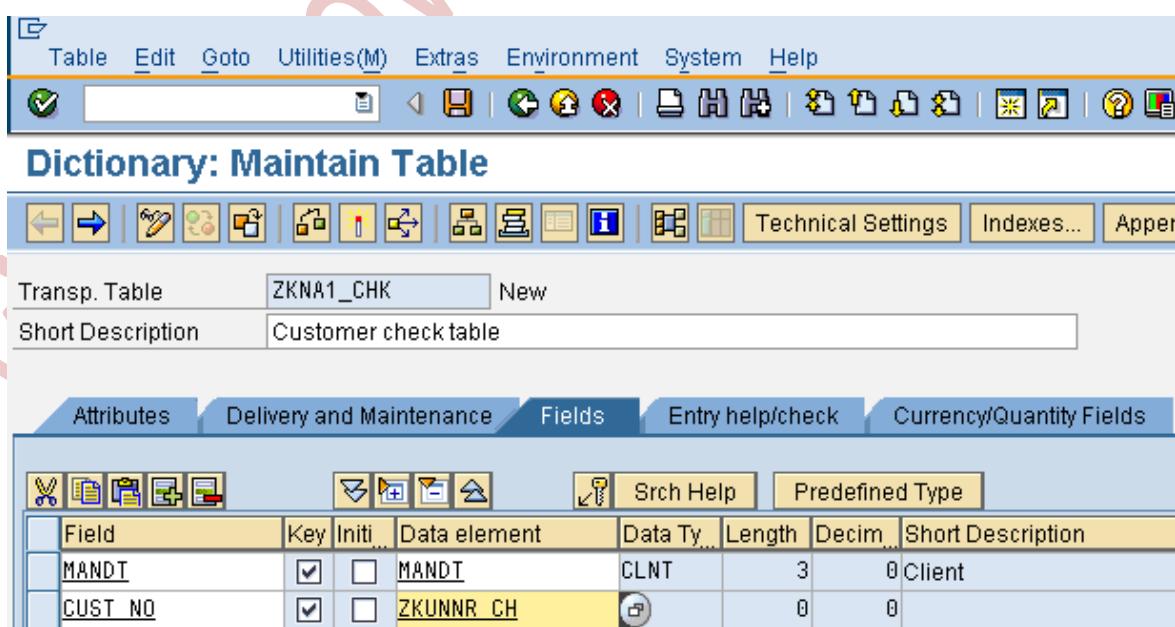
- ☞ IF YOU CLICK ON YES THE VALUE TABLE IS CONVERTED INTO CHECK TABLE.
- ☞ SAVE AND ACTIVATE

STEPS TO CREATE A VALUE TABLE

- ❖ GO TO SE11.
- ❖ GIVE THE TABLE NAME AS ZKNA1_CHK.



- ❖ CLICK ON CREATE.
- ❖ NOW GIVE THE FIELDS AS BELOW.
- ❖ GIVE THE FIRST FIELD AS CUST_NO.
- ❖ GIVE THE DATA ELEMENT AS ZKUNNR_CH AND PRESS ENTER.



- ❖ CREATE THE DATA ELEMENT AND DOMAIN BY THE SAME NAME.

Dictionary: Maintain Data Element

The screenshot shows the SAP Dictionary: Maintain Data Element interface. The data element 'ZKUNNR_CH' is listed with a status of 'Inactive'. The 'Data Type' tab is selected, showing it is a 'CHAR' type with a length of 4, resulting in a 'Character String'. The 'Domain' tab is also visible, showing 'ZKUNNR_CH' is defined as 'DOMAIN FOR CUST.NO'.

Data element	ZKUNNR_CH	Inactive
Short Description	customer number	
<input checked="" type="radio"/> Elementary Type <input type="radio"/> Domain		
ZKUNNR_CH Data Type: CHAR Length: 4 Character String Decimal Places: 0		

- ❖ WHENEVER WE ARE CREATING THE DOMAIN, CLICK ON THE VALUE RANGE TAB. SPECIFY THE VALUE TABLE AS ZKNA1_CH.

Dictionary: Maintain Domain

The screenshot shows the SAP Dictionary: Maintain Domain interface. The domain 'ZKUNNR_CH' is listed with a status of 'New(Revised)'. The 'Value Range' tab is selected, showing the 'Single Vals' and 'Intervals' sections. The 'Value Table' dropdown is set to 'ZKNA1_CHK'.

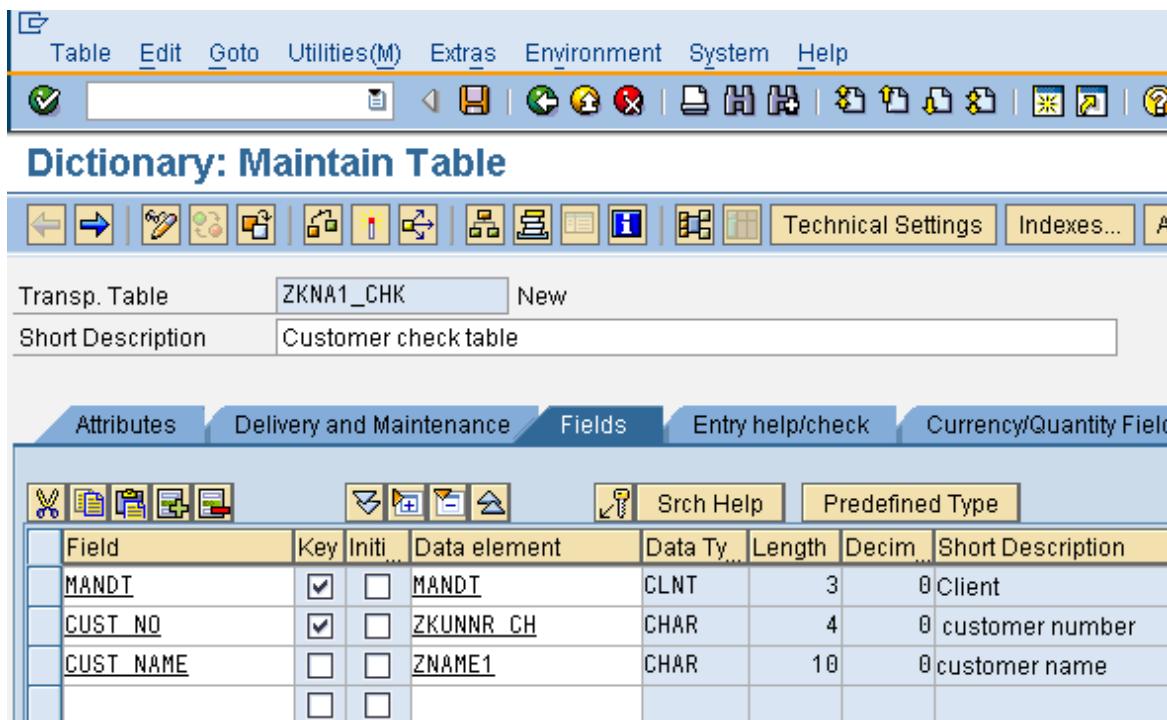
Domain	ZKUNNR_CH	New(Revised)
Short Description	DOMAIN FOR CUST.NO	
<input type="radio"/> Properties <input type="radio"/> Definition <input checked="" type="radio"/> Value Range		
Single Vals Fix.Val.: Short text (Table rows for intervals)		
Intervals Lower limit UpperLimit Short text (Table rows for intervals)		
Value Table: ZKNA1_CHK		

- ❖ SAVE IT, ACTIVATE IT AND COMEBACK.

- ❖ DEFINE THE SECOND FIELD AS BELOW.

FIELD NAME – CUST_NAME

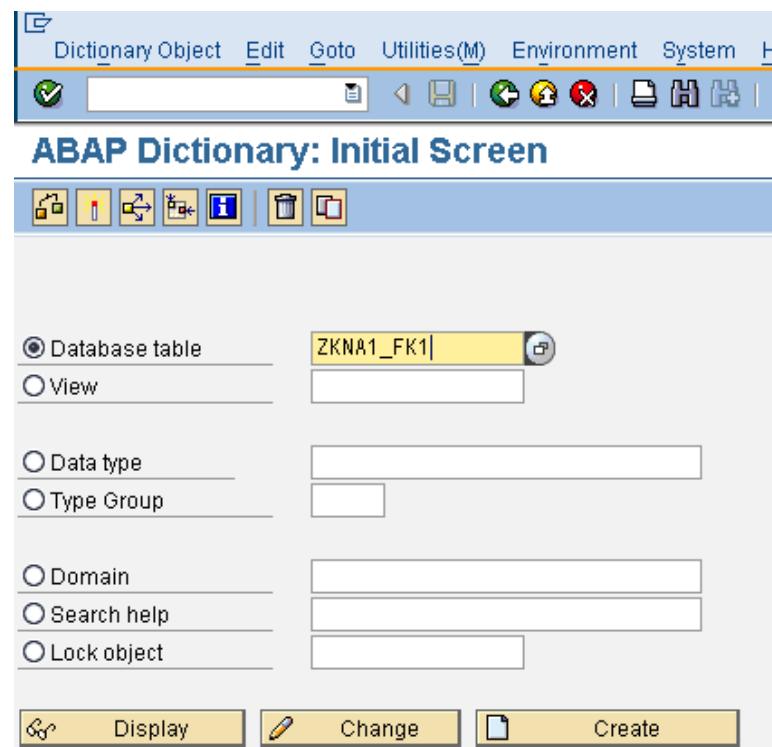
DATA ELEMENT – ZNAME1.



- ❖ SAVE, ACTIVATE AND CREATE 3 RECORDS IN THE TABLE.



CREATE ANOTHER TABLE BY THE NAME ZKNA1_FK1.



GIVE THE FIELDS AS BELOW

FIELDS → CUST_NO, CITY

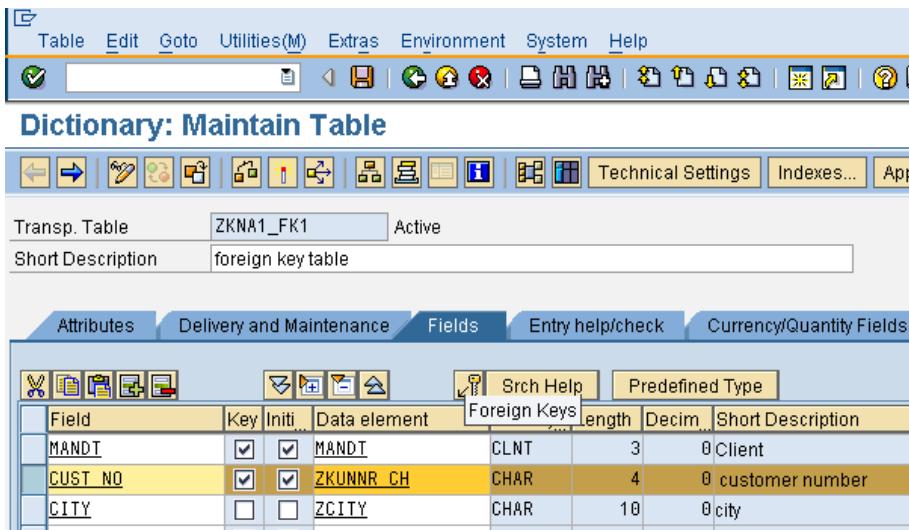
DATA ELEMENT → ZKUNNR_CH, ZCITY.

Transp. Table	ZKNA1_FK1	New					
Short Description	foreign key table						
Attributes Delivery and Maintenance Fields Entry help/check Currency/Quantity Fie							
Field	Key	Initi...	Data element	Data Ty...	Length	Decim...	Short Description
MANDT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	MANDT	CLNT	3	0	Client
CUST_NO	<input checked="" type="checkbox"/>	<input type="checkbox"/>	ZKUNNR_CH	CHAR	4	0	customer number
CITY	<input type="checkbox"/>	<input type="checkbox"/>	ZCITY		10	0	city

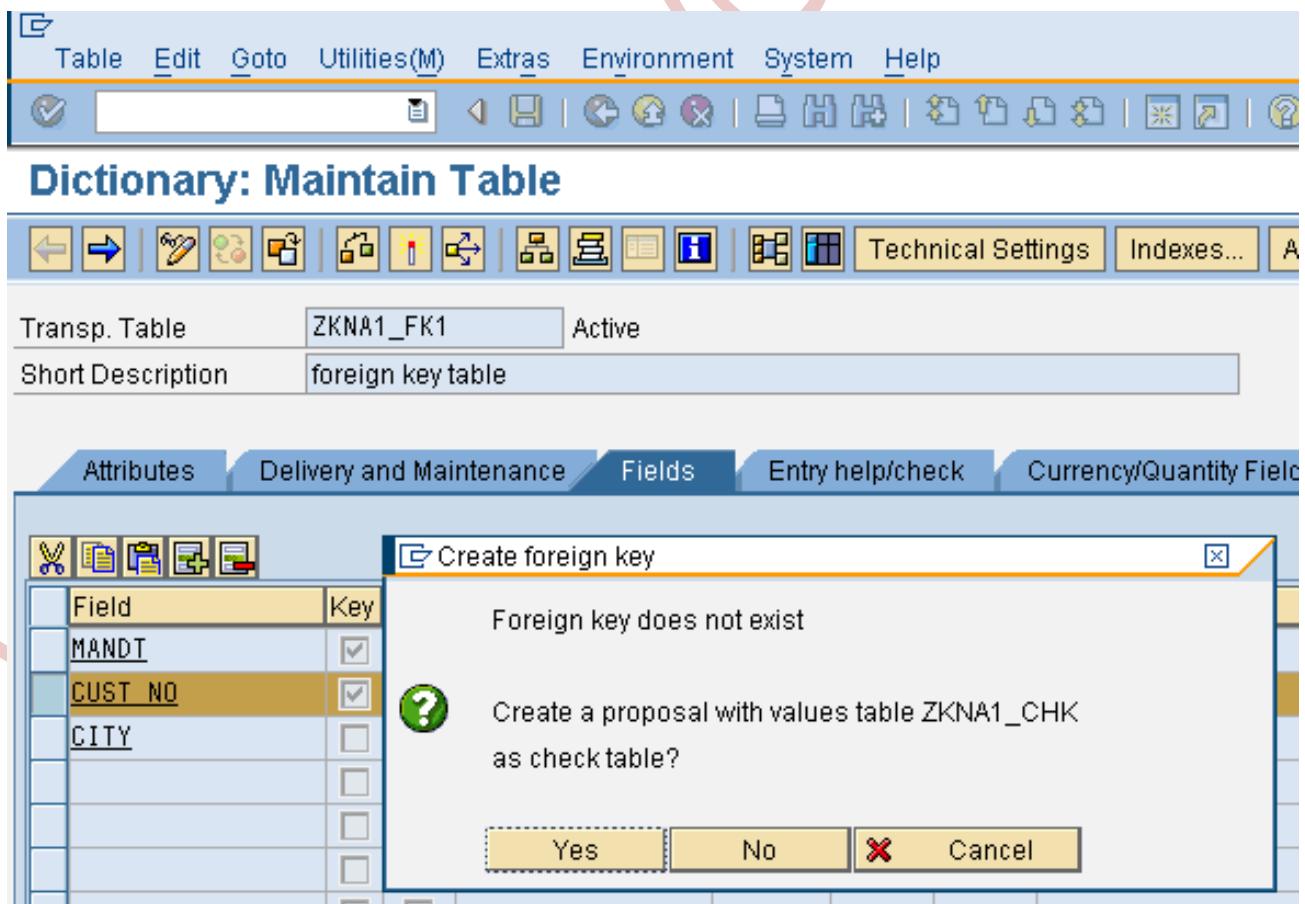
SAVE AND ACTIVATE.

CREATING FOREIGN KEY RELATION USING VALUE TABLE:

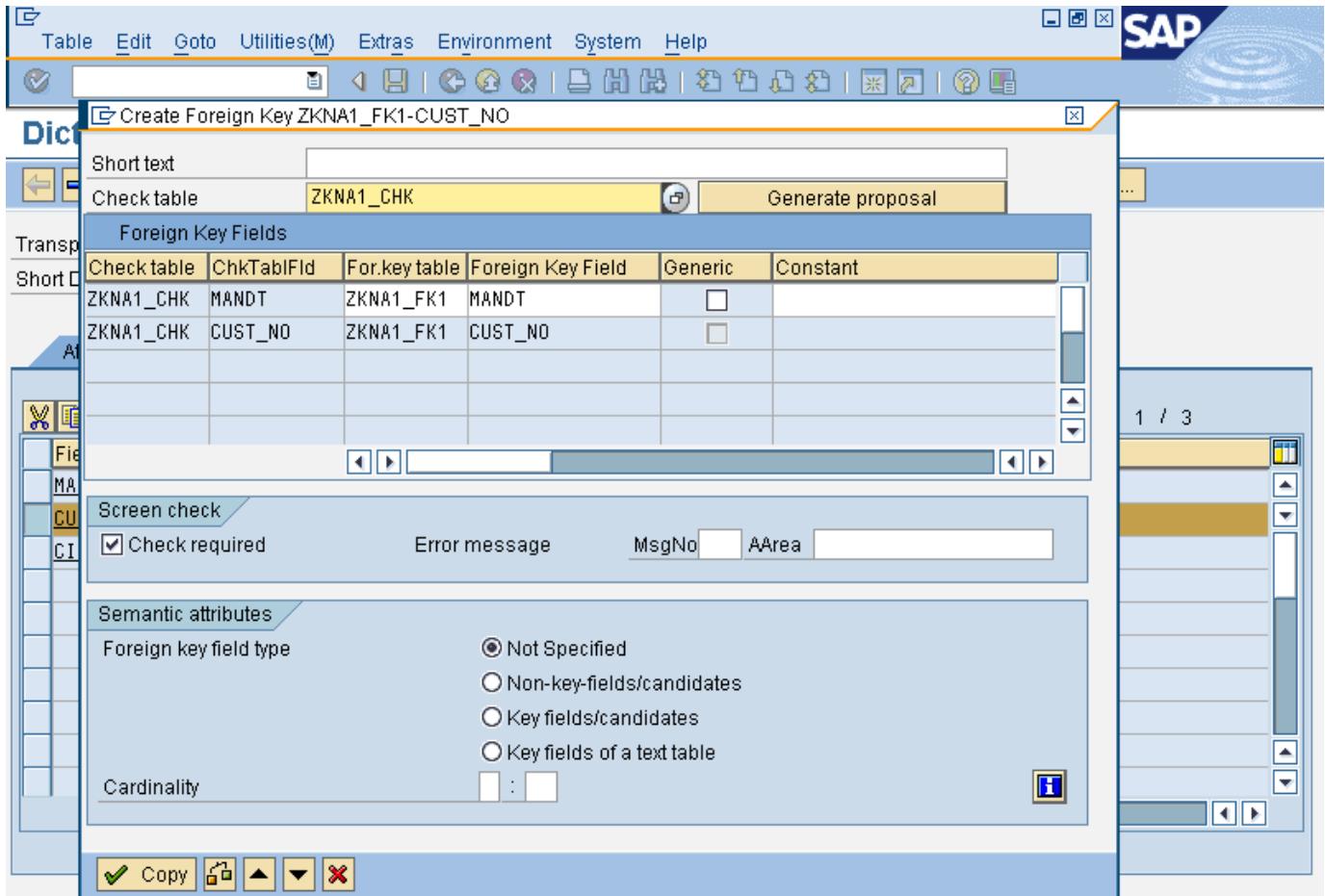
- ❖ GO TO THE FOREIGN KEY TABLE ZKNA1_FK1.
- ❖ CLICK ON CHANGE.
- ❖ SELECT THE FIELD CUST_NO. AND CLICK ON FOREIGN KEY BUTTON.



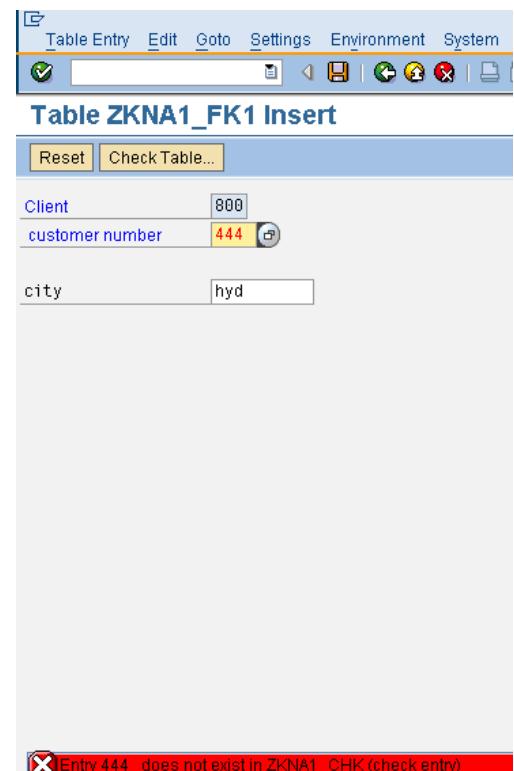
- ❖ A POP WILL BE RAISED WITH THE BELOW MESSAGE.



- ❖ CLICK ON 'YES' BUTTON. A WINDOW WILL BE RAISED AS BELOW.



- CLICK ON 'COPY' BUTTON.
- A FOREIGN KEY WILL BE CREATED FOR ZKNA1_FK1-CUST_NO.
- SAVE, ACTIVATE AND TEST IT.
- IF YOU GIVE THE CUST_NO. OTHER THAN THE CUST_NO. IN THE CHECK TABLE, IT WILL THROW AN ERROR AS BELOW.



INDEX

INDEX IS A COPY OF DATA BASE TABLE. WHEN EVER A TABLE IS CREATED IN DATABASE ALONG WITH THAT ANOTHER TABLE WILL BE CREATED IN DATABASE.

NOTE:

IT IS A CONCEPT TO ARRANGE THE DATABASE TABLE RECORDS IN ASCENDING ORDER FOR QUICK OR FAST ACCESSING OF THE DATA.

THERE ARE TWO TYPES OF INDEXES

1. PRIMARY INDEX
2. SECONDARY INDEX.

PRIMARY INDEX:

AN INDEX CREATED AUTOMATICALLY ON THE KEY FIELDS BY SAP IS CALLED PRIMARY INDEX.

SECONDARY INDEX:

AN INDEX CREATED ON NON-KEY FIELDS BY THE ABAP CONSULTANTS (BASIS CONSULTANTS) IS CALLED SECONDARY INDEX.

USE OF SECONDARY INDEX:

SECONDARY INDEX ARE MAINLY USED OR CREATED ON THE FIELDS WHICH ARE USED IN THE WHERE CLAUSE OF SELECT STATEMENT.

STEPS TO CREATE A SECONDARY INDEX:

BUSINESS REQUIREMENT:

CREATE A SECONDARY INDEX ON THE NON KEY FIELD REGIO.

- GO TO SE11.
- GIVE THE TABLE NAME AS KNA1.
- CLICK ON DISPLAY.
- CLICK ON INDEX BUTTON.
- CLICK ON CREATE INDEX ICON.
- GIVE DESCRIPTION.
- CLICK ON TABLE FIELDS BUTTON.
- SELECT MANDT, KUNNR, REGIO AND PRESS ENTER.
- SAVE AND ACTIVATE.

INITIAL:

- IT IS A CHECKBOX WHICH INDICATES WHETHER A INITIAL VALUE OR DEFAULT VALUE TO BE STORED IN THE FIELD.
- IF INITIAL = 'X', MEANS THE DEFAULT VALUE IS STORED.
- EG:
 - FOR CHAR FIELD IT IS SPACE.
 - FOR INT FIELD IT IS ZERO.
- BY DEFAULT THE INITIAL WILL BE 'X' OR SELECTED FOR ALL THE KEY FIELDS.

{OR}

SECONDARY INDEX:

INDEX: IS COPY OF DB TABLE, WHICH HAS THE DIRECT REFERENCE TO DATA (NO COMPARISON).

PRIMARY INDEX: PRIMARY INDEX IS A COPY OF DB REDUCED TO PRIMARY KEY OR PRIMARY INDEX IS A INDEX THAT HAS ONLY THE VALUES OF COMPLETE PRIMARY KEY.

DIFFERENCE BETWEEN INDEX & DB TABLE?

SEARCH -> COMPARE IN DB TABLE (RECORD BY RECORD).

SEARCH -> DIRECTLY GO TO PARTICULAR RECORD.

SECONDARY INDEX: SECONDARY INDEX IS AN INDEX WHICH IS CREATED OTHER THAN YOUR PRIMARY INDEX.

SUPPOSE IN INTERVIEW IF THEY ASK: DID YOU CREATE SECONDARY INDEX?

SAY NO. YOU WILL BE IN SAFE SIDE.

IF YOU SAY YES. INTERVIEWER WILL ASK WHICH TABLE AND WHAT FIELDS.

WHAT HAPPENS WHEN WE CREATE SECONDARY INDEX?

DUPLICATION, BURDEN ON DB

IMPORTANT POINTS

PRIMARY KEY FIELDS = WE CAN CREATE UP TO 16 KEYFIELDS

SECONDARY INDEX = WE CAN CREATE UP TO 9 INDEXES.

STEPS IN CREATING SECONDARY INDEX

STEP1: CLICK ON INDEXES

The screenshot shows the SAP Dictionary: Change Table interface. The top navigation bar includes icons for back, forward, search, and technical settings, along with buttons for 'Indexes...' (Ctrl+F5) and 'Append Structure...'. Below this, the 'Transp. Table' is set to 'ZVEJAY' and 'Active'. A 'Short Description' field contains the value 'b vejay kumaar', which is highlighted with a red box. The main area displays the 'Fields' tab of the table structure, listing four fields: MANDT, KUNNR, NAME1, and ORTO1. Each field has its key indicator checked, and the 'Data element' column shows the corresponding data elements: MANDT, KUNNR, NAME1, and ORTO1 respectively. The 'Data Type' column lists CLNT, CHAR, CHAR, and CHAR with lengths of 3, 10, 30, and 25 respectively. The 'Dec...' column indicates no decimal places for all fields.

STEP2: CLICK ON CREATE

The screenshot shows the SAP Dictionary: Change Table interface again, but now with a different focus. The 'Indices for Table ZVEJAY' tab is selected. In the toolbar below, the 'Create Index' button is highlighted with a yellow background and a red arrow points to it from the text above. Other buttons like 'Create Extension Index' and 'Create DBS' are also visible.

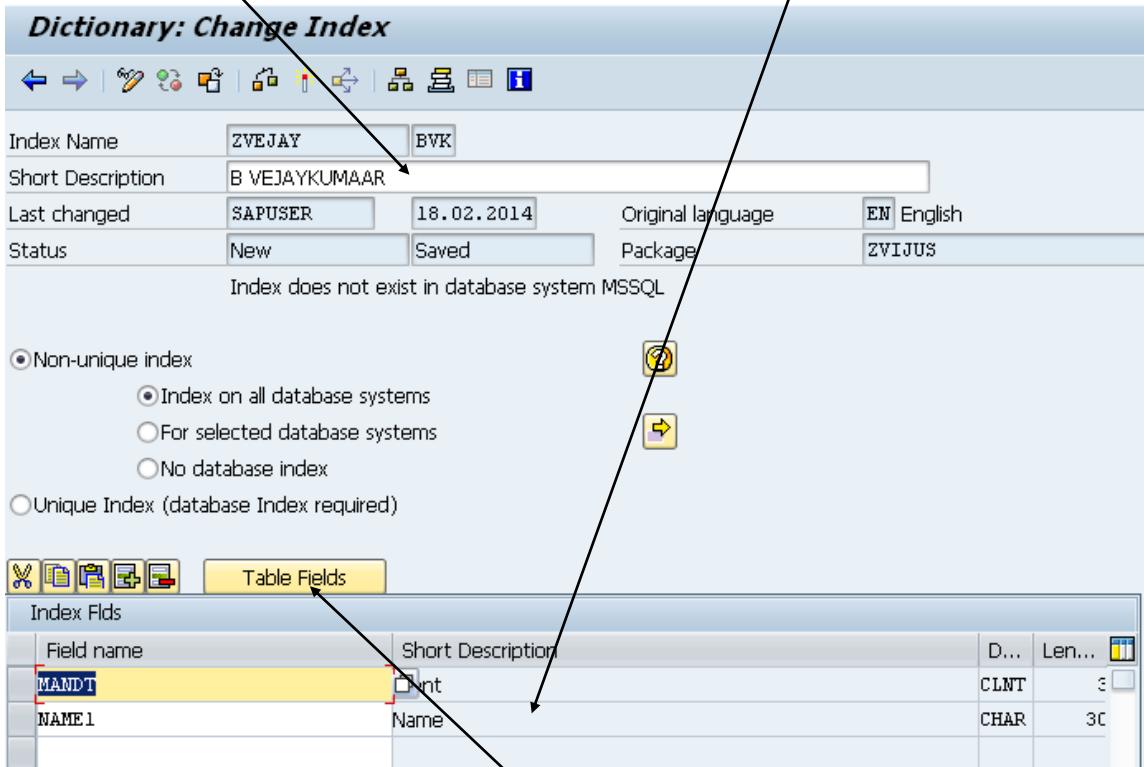
STEP3: GIVE ANY NAME

A dialog box titled 'Create Index' is shown. It has two input fields: 'Table Name' containing 'ZVEJAY' and 'Index Name' containing 'BVK'. Below these fields are two buttons: a green checkmark and a red X. At the bottom right of the dialog is a 'Continue (Enter)' button.

CLICK ON CONTINUE

STEP4:

GIVE SHORT DESCRIPTION AND FIELD NAMES TO WHICH SECONDARY INDEX IS TO BE CREATED.



GIVE DIRECTLY FIELDS NAME OR CLICK ON TABLE FIELDS.

DO NOT FORGET TO ENTER MANDT AS THE FIRST INDEX FIELD IF IT IS A CLIENT DEPENDENT TABLE. SAVE AND ACTIVATE THE INDEX.

NOTE:

WE HAVE TO ACTIVATE AND ADJUST THE DATABASE AFTER CREATION OF THE INDEX.

TO DO THIS

GOTO -> UTILITIES -> ACTIVE AND ADJUST DATABASE. WHICH IS NOTHING BUT **SE14**(DB UTILITIY).

IEWS

VIEW:

- ❖ RETRIEVES DATA FROM DATABASE TABLE. A VIEW IS A VIRTUAL TABLE AS IT DOESN'T STORE THE DATA PHYSICALLY.
- ❖ VIEW IS A MECHANISM I.E., USED TO SEE THE DATA FROM MULTIPLE DATA BASE TABLES SIMULTANEOUSLY.

TYPES OF VIEWS:

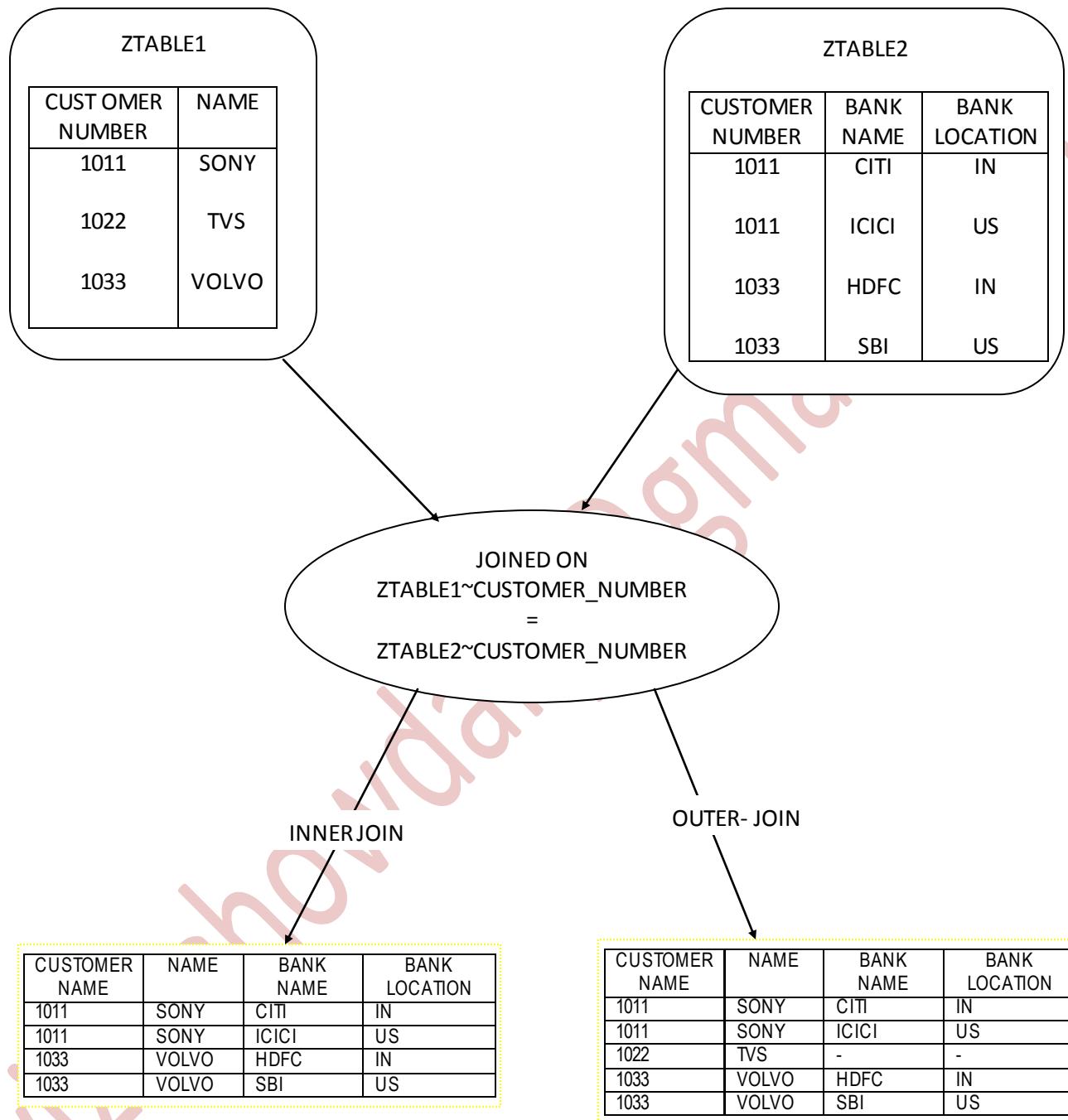
- 1) DATABASE VIEW
- 2) PROJECTION VIEW
- 3) MAINTENANCE VIEW
- 4) HELP VIEW

DATABASE VIEW:

- ❖ A VIEW CREATED ON TWO OR MORE TABLES USING INNER JOIN IS CALLED AS DATABASE VIEW.
- ❖ IN DATABASE VIEW WE CAN ONLY READ THE DATA AND WE CANNOT MAINTAIN ANY DATA.

JOINS:

IT USED TO LINK TWO OR MORE TABLES.



INNER JOIN:

RETURNS ROWS WHEN THERE IS A MATCH IN BOTH TABLES IS INNER JOIN.

LEFT OUTER JOIN:

RETURNS ALL ROWS FROM LEFT TABLE, EVEN IF THERE ARE NO MATCHES IN THE RIGHT TABLE.

EXAMPLE ON DATA BASE VIEW:

BUSINESS REQUIREMENTS:

DEVELOP A DATABASE VIEW WHICH DISPLAYS MATERIAL DETAILS AND DESCRIPTION DETAILS.

TABLE NAMES: MARA, MAKT

FIELD NAMES:
MARA-MATNR
MARA-MTART
MARA-MBRSH
MARA-MEINS
MAKT-SPRAS
MAKT-MAKTX

- ☞ GO TO SE11.
- ☞ SELECT VIEW, GIVE A NAME. EG: ZDB_VIEW
- ☞ CLICK ON CREATE.
- ☞ SELECT DATABASE VIEW, PRESS ENTER.
- ☞ GIVE DESCRIPTION.
- ☞ DEFINE THE TABLE NAMES & JOINING CONDITIONS AS BELOW

TABLES
MARA
MAKT

TABLE	FIELD NAME	=	TABLE	FIELD NAME
MARA	MANDT	=	MAKT	MANDT
MARA	MATNR	=	MAKT	MATNR

- ☞ CLICK ON VIEW FIELDS TAB
- ☞ CLICK ON TABLE FIELDS BUTTON
- ☞ DOUBLE CLICK ON THE TABLE NAME MARA.
- ☞ SELECT MATNR, MTART, MBRSH, MEINS AND CLICK ON COPY BUTTON.
- ☞ AGAIN CLICK ON TABLE FIELDS BUTTON.
- ☞ DOUBLE CLICK ON MAKTX TABLE.
- ☞ SELECT SPRAS, MAKTX AND CLICK ON COPY BUTTON.
- ☞ CLICK ON SELECTION CONDITION TAB.
- ☞ SPECIFY THE CONDITIONS BELOW

TABLE	FIELD NAME	OPERATOR	COMPARISON VALUE	AND
MARA	MEINS	EQ	'KGS'	AND
MAKT	SPRAS	EQ	'E'	

SAVE AND ACTIVATE

CLICK ON THE CONTENTS ICON TO DISPLAY THE DATA.

ASSIGNMENT:

DEVELOP A VIEW ON KNA1 & KNBK TABLES

DEVELOP A VIEW ON LFA1 & LFBK TABLES

DEVELOP A VIEW ON VBAK & VBAP TABLES.

PROJECTION VIEW:

- ❖ PROJECTION VIEW IS MAINLY USED TO CREATE A VIEW (IMAGINARY TABLE) WITH THE REQUIRED FIELDS BY REMOVING UNWANTED FIELDS.
- ❖ WE CAN READ THE DATA AND MAINTAIN THE DATA BECAUSE IT IS A VIEW ON SINGLE TABLE.

EXAMPLE ON PROJECTION VIEW:

BUSINESS REQUIREMENT: CREATE A PROJECTION VIEW ON MARA TABLE.

- ☞ GO TO SE11.
- ☞ GIVE THE VIEW NAME, EG: ZPRJ_VIEW
- ☞ CLICK ON CREATE.
- ☞ SELECT PROJECTION VIEW.
- ☞ GIVE DESCRIPTION.
- ☞ GIVE THE BASIC TABLE NAME AS MARA.
- ☞ CLICK ON TABLE FIELDS BUTTON.
- ☞ SELECT MANDT, MATNR, MTART, MBRSH, MEINS.
- ☞ CLICK ON COPY.
- ☞ CLICK ON MAINTENANCE STATUS TAB.
- ☞ SELECT EITHER READ ONLY OR READ OR CHANGE.
- ☞ SAVE AND ACTIVATE.
- ☞ CLICK ON CONTENTS BUTTON TO DISPLAY THE DATA.

NOTE:

TO MAINTAIN THE DATA SELECT MAINTENANCE STATUS AS READ & CHANGE AND SELECT DISPLAY MAINTENANCE ALLOW. WE CAN CREATE ENTRIES IN THE LIST SCREEN.

ASSIGNMENT:

DEVELOP A PROJECTION VIEW ON KNA1 TABLE.

DEVELOP A PROJECTION VIEW ON LFA1 TABLE.

HELP VIEW:

- ❖ A VIEW CREATED ON 2 OR MORE TABLES USING OUTER JOIN CONCEPT IS CALLED HELP VIEW.
- ❖ HELP VIEWS ARE SPECIALLY DESIGNED FOR SEARCH HELP.
- ❖ WE CANNOT EXECUTE HELP VIEWS DIRECTLY.
- ❖ WE CAN EXECUTE THEM ONLY WITH SEARCH HELP.

EXAMPLE ON HELP VIEW:

BUSINESS REQUIREMENT:

DEVELOP A HELP VIEW ON MAKTABLE & MARA TABLES.

- ☞ GO TO SE11.
- ☞ GIVE THE VIEW NAME AS ZHELP_VIEW.
- ☞ CLICK ON CREATE.
- ☞ GIVE DESCRIPTION.
- ☞ GIVE THE TABLE NAME AS MAKTABLE.
- ☞ CLICK ON RELATIONSHIP BUTTON.
- ☞ SELECT MAKTABLE-MARA
- ☞ CLICK ON COPY BUTTON.
- ☞ THE JOINING CONDITION WILL BE PROPOSED AUTOMATICALLY.
- ☞ CLICK ON VIEW FIELDS TAB.
- ☞ CLICK ON TABLE FIELDS BUTTON, DOUBLE CLICK ON THE TABLE NAME AND SELECT THE LIST OF THE FIELDS.
- ☞ SAVE AND ACTIVATE.
- ☞ WE CANNOT TEST OR EXECUTE HELP-VIEW DIRECTLY.
- ☞ THEY ARE ONLY EXECUTED WITH SEARCH HELP.

MAINTENANCE VIEW:

- ❖ WE CAN SIMULTANEOUSLY MAINTAIN DATA FOR MULTIPLE TABLES / SINGLE TABLE.
- ❖ A VIEW WHICH IS CREATED ON 2 OR MORE TABLES USING INNER JOIN IS CALLED MAINTENANCE VIEWS.
- ❖ IN THIS VIEW WE CAN READ THE DATA AND MAINTAIN DATA.

DIFFERENCE BETWEEN TMG AND MAINTENANCE VIEW?

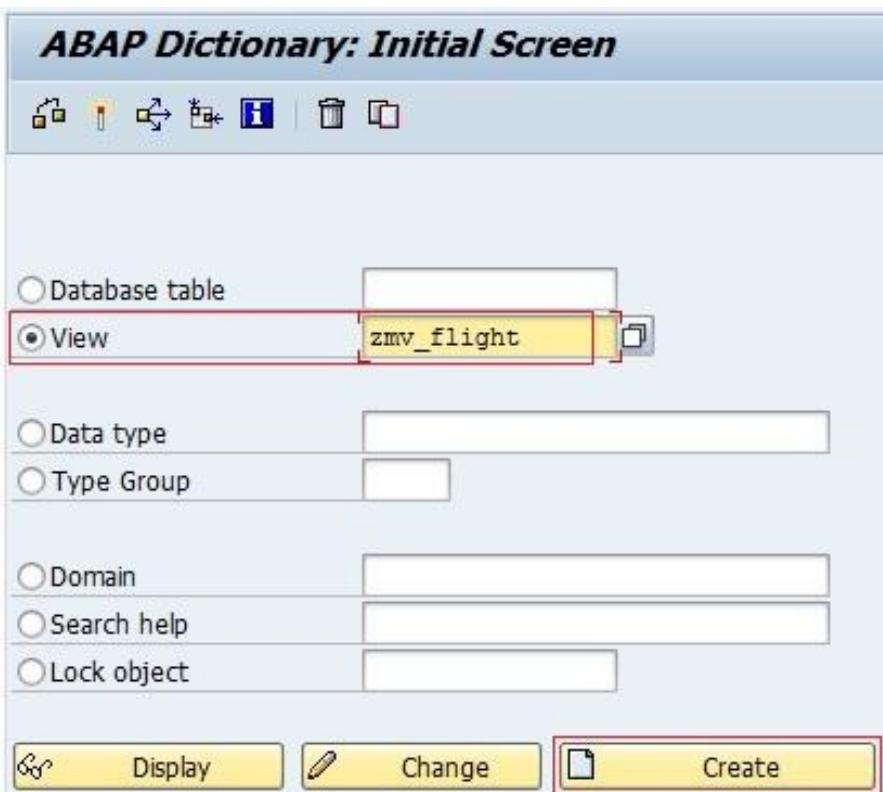
TMG IS NOTHING BUT MAINTENANCE VIEW.

TMG MAINTAINS DATA FOR ONLY A SINGLE TABLE.

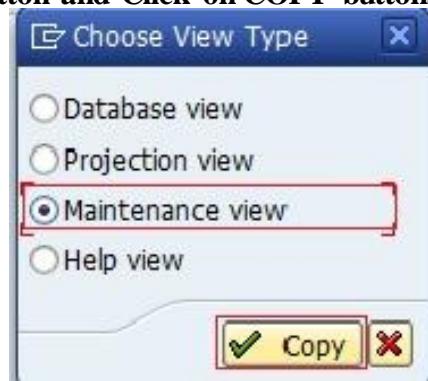
WHERE AS MAINTENANCE VIEW CAN MAINTAIN DATA FOR MULTIPLE TABLES.

Creation of Maintenance View

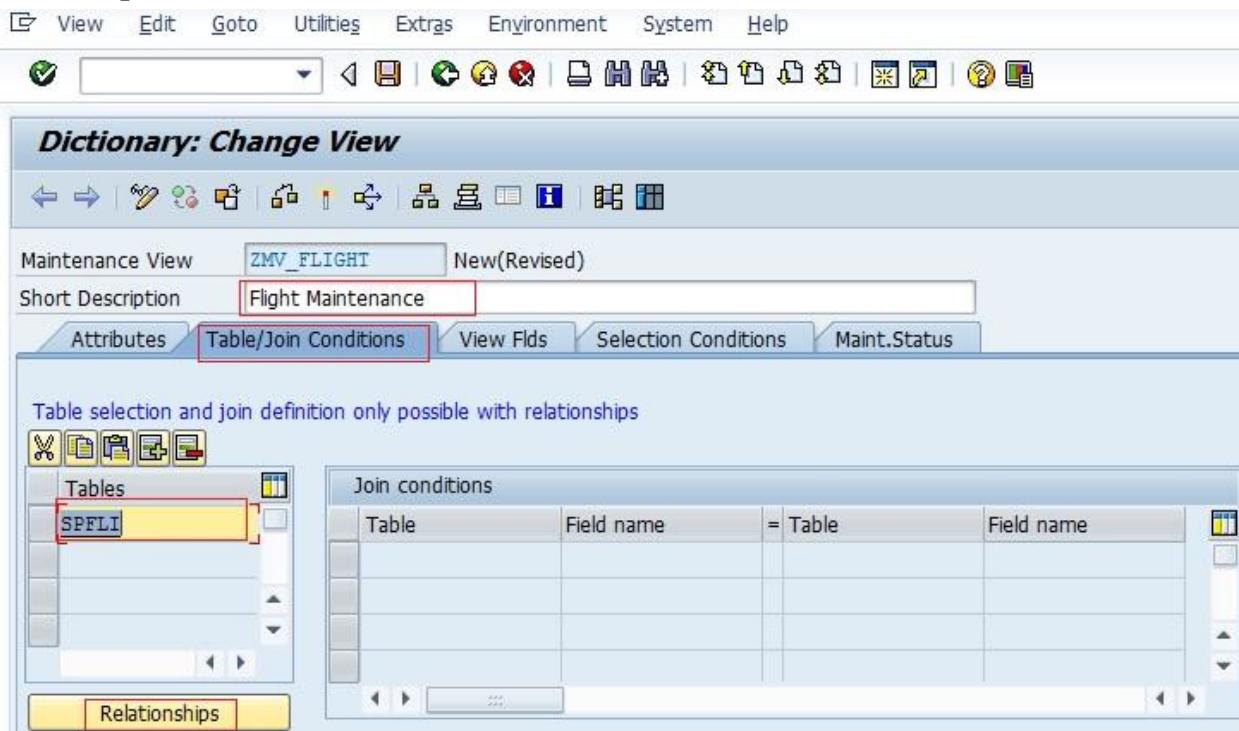
Step 1. Provide the view name as given below and Click on Create Button.



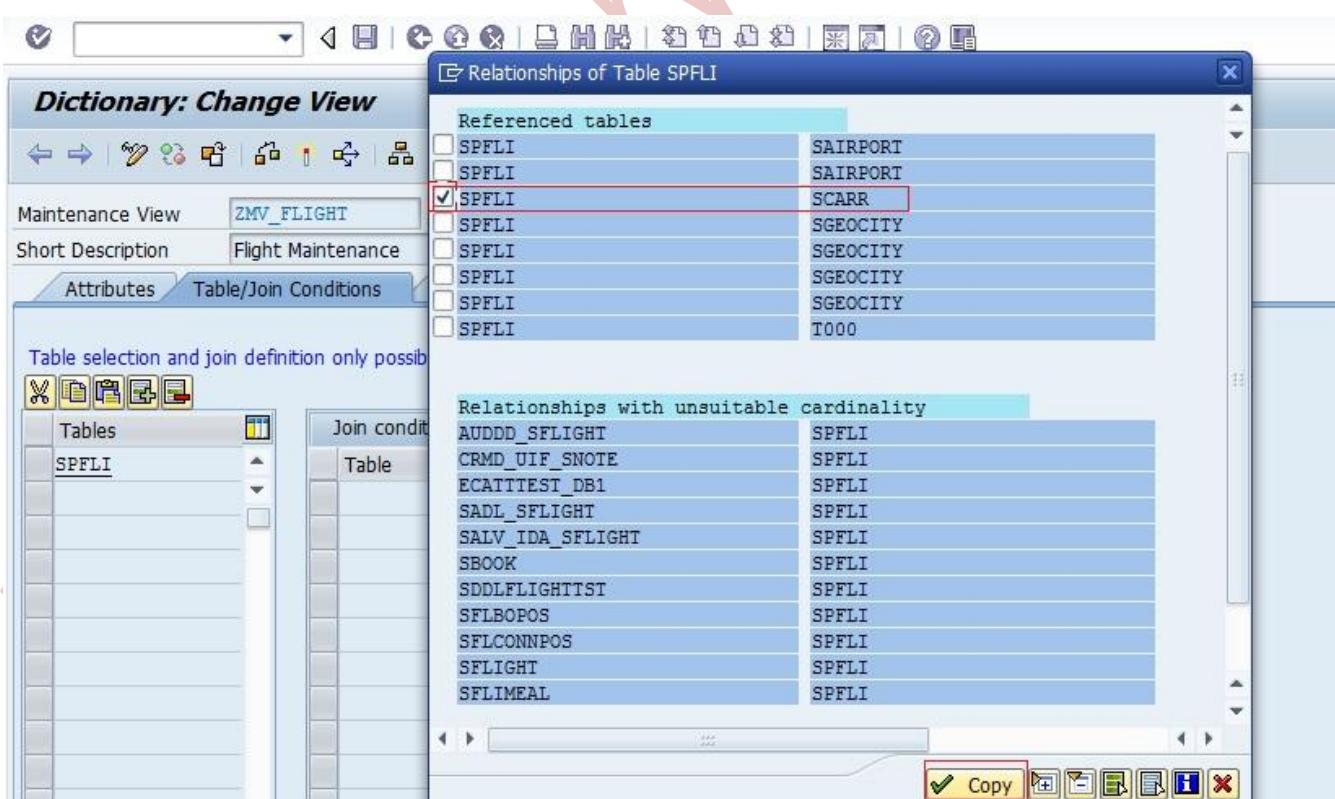
Step 2. Select the Maintenance View Radio Button and Click on COPY button.



Step 3. Provide the short text and in the Tables section provide the table name 'SPFLI' and click on the Relationships Button.



Step 4. Select the combination as marked and click on COPY Button.



Step 5. The Join condition is automatically field based on the two table matching fields. Now click on the View Flds Tab.

Maintenance View: ZMV_FLIGHT New(Revised)

Short Description: Flight Maintenance

Attributes Table/Join Conditions **View Flds** Selection Conditions Maint.Status

Table selection and join definition only possible with relationships

Tables		Join conditions	
		Table	Field name
SPFLI		SCARR	MANDT
SCARR		SCARR	CARRID

Step 6. Click on the Table Fields.

Maintenance View: ZMV_FLIGHT New(Revised)

Short Description: Flight Maintenance

Attributes Table/Join Conditions **View Flds** Selection Conditions Maint.Status

View field	Table	Field	P Key	Data elem.	M...	DTyp	Length	Short description
MANDT	SPFLI	MANDT	<input checked="" type="checkbox"/>	S_MANDT	<input type="checkbox"/>	CLNT	6	Client
CARRID	SPFLI	CARRID	<input checked="" type="checkbox"/>	S_CARR_ID	<input type="checkbox"/>	CHAR	6	Airline Code
CONNID	SPFLI	CONNID	<input checked="" type="checkbox"/>	S_CONN_ID	<input type="checkbox"/>	NUMC	8	Flight Connection Number

Step 7. Select the 'SPFLI' table and click on the Choose Button.

Maintenance View: ZMV_FLIGHT New(Revised)

Short Description: Flight Maintenance

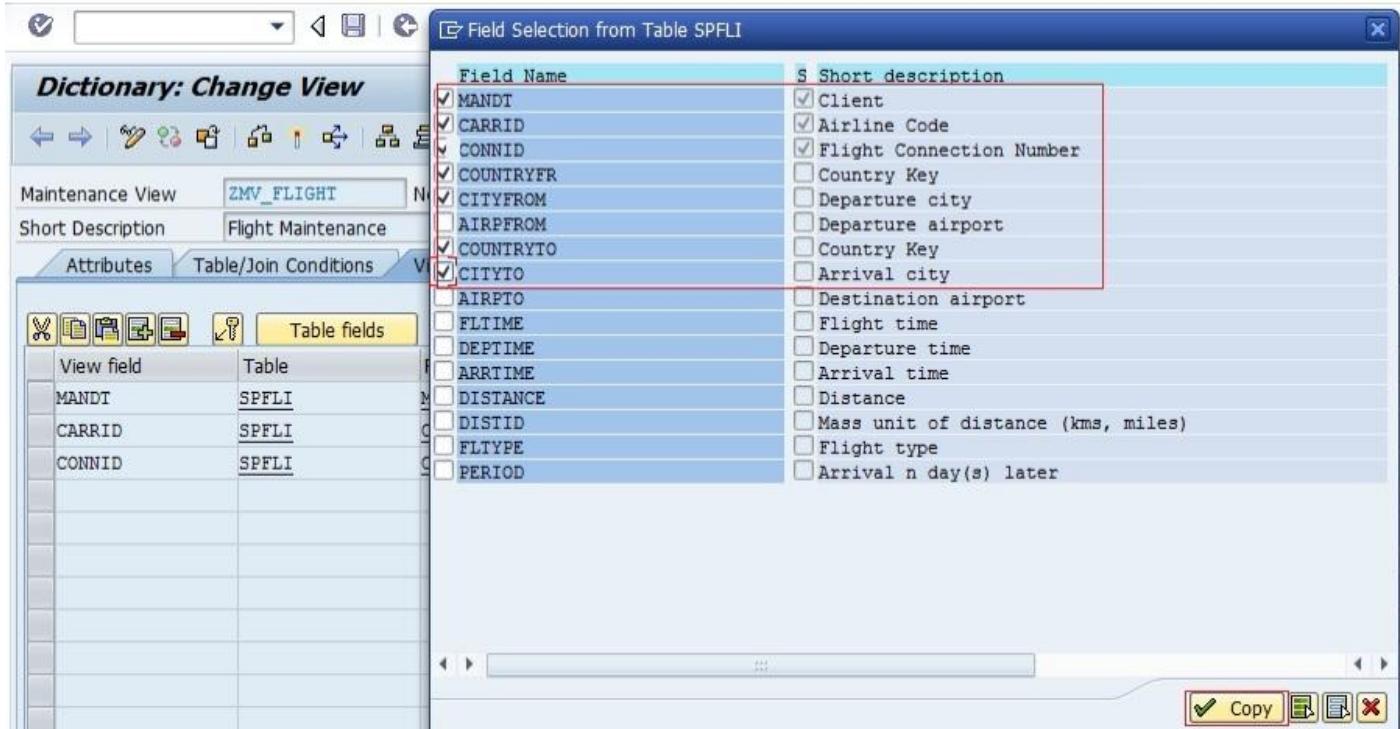
Attributes Table/Join Conditions View Flds

View field	Table	Field
MANDT	SPFLI	MANDT
CARRID	SPFLI	CARRID
CONNID	SPFLI	CONNID

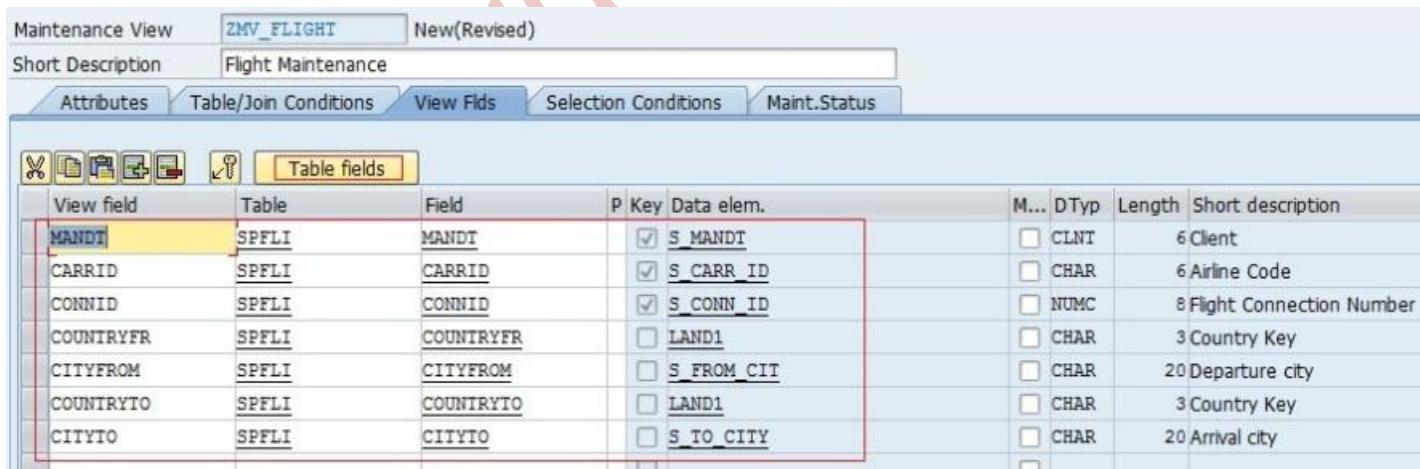
Base Tables
 SPFLI
 SCARR

Choose **X**

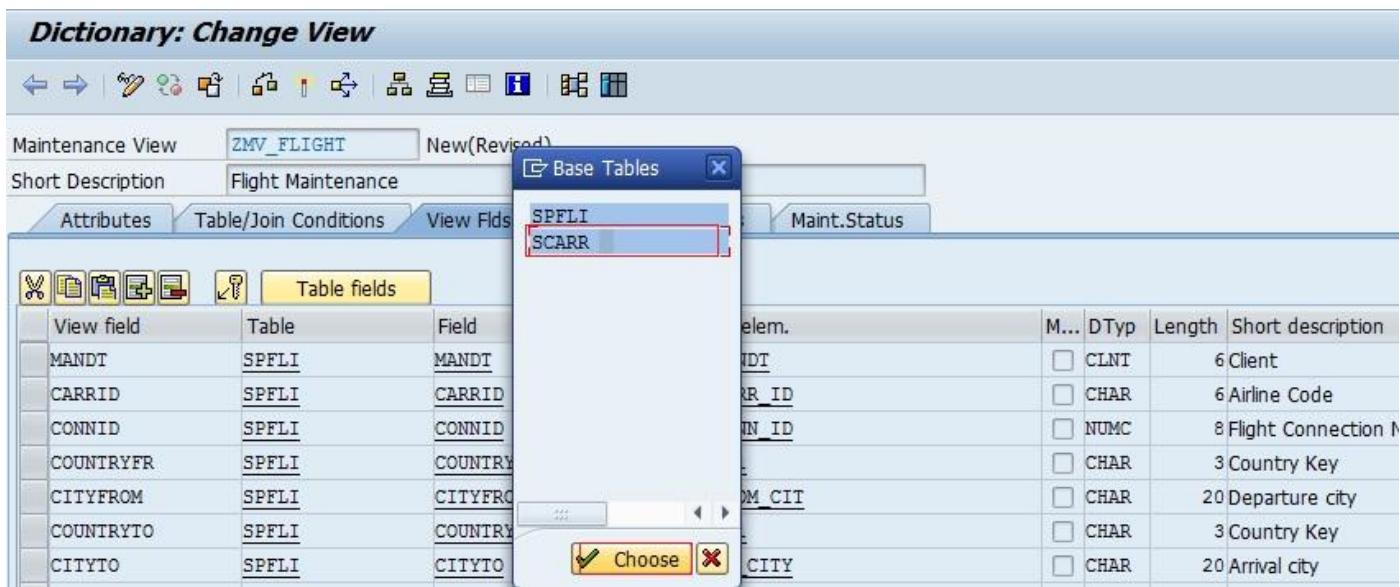
Step 8. Select the required fields From the table to add it in the View and click on the COPY Button.



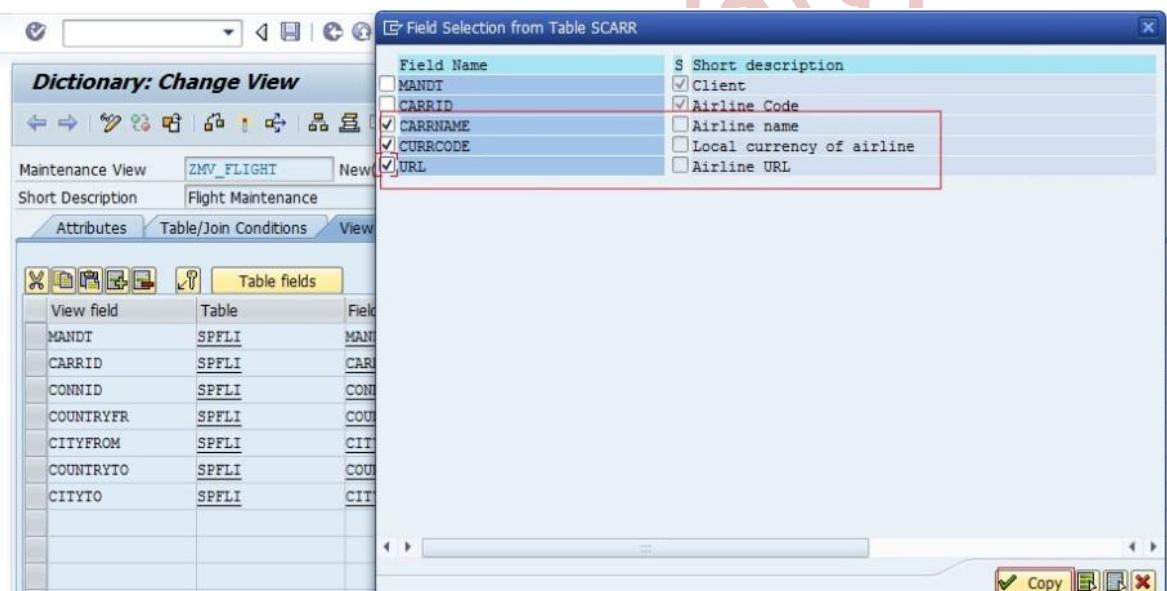
Step 9. All the Selected fields From the Table are populated in the view and then click on the Table Fields Button to add some fields from SCARR Table.



Step 10. Select the SCARR table and click on the Choose Button.



Step 11. Select some fields and then click on the COPY Button.



Step 12. All the fields from two tables are added to the Maintenance View.

View field	Table	Field	P Key	Data elem.	M...	DTyp	Length	Short description
MANDT	SPFLI	MANDT	<input checked="" type="checkbox"/>	S_MANDT	<input type="checkbox"/>	CLNT	3	Client
CARRID	SPFLI	CARRID	<input checked="" type="checkbox"/>	S_CARR_ID	<input type="checkbox"/>	CHAR	3	Airline Code
CONNID	SPFLI	CONNID	<input checked="" type="checkbox"/>	S_CONN_ID	<input type="checkbox"/>	NUMC	4	Flight Connection Number
COUNTRYFR	SPFLI	COUNTRYFR	<input type="checkbox"/>	LAND1	<input type="checkbox"/>	CHAR	3	Country Key
CITYFROM	SPFLI	CITYFROM	<input type="checkbox"/>	S_FROM_CIT	<input type="checkbox"/>	CHAR	20	Departure city
COUNTRYTO	SPFLI	COUNTRYTO	<input type="checkbox"/>	LAND1	<input type="checkbox"/>	CHAR	3	Country Key
CITYTO	SPFLI	CITYTO	<input type="checkbox"/>	S_TO_CITY	<input type="checkbox"/>	CHAR	20	Arrival city
CARRNAME	SCARR	CARRNAME	<input type="checkbox"/>	S_CARRNAME	<input type="checkbox"/>	CHAR	20	Airline name
CURRCODE	SCARR	CURRCODE	<input type="checkbox"/>	S_CURRCODE	<input type="checkbox"/>	CUKY	5	Local currency of airline
URL	SCARR	URL	<input type="checkbox"/>	S_CARRURL	<input type="checkbox"/>	CHAR	255	Airline URL

Step 13. Click on the Maint. Status Tab and select the appropriate Radio Button and activate the View.

Dictionary: Change View

Maintenance View: ZMV_FLIGHT Active
Short Description: Flight Maintenance

Attributes / Table/Join Conditions / View Fds / Selection Conditions / **Maint.Status**

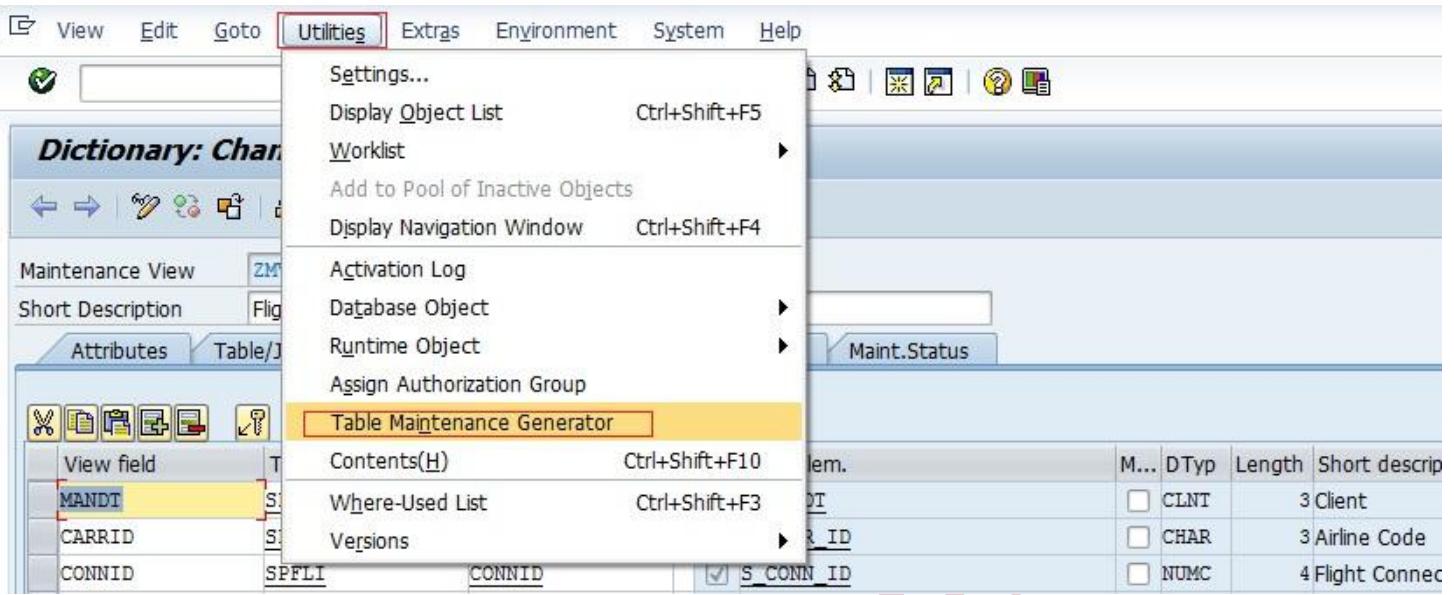
Access

- read only
- read, change, delete and insert
- read and change
- read and change (time-specific views)

Delivery class: A Application table (master and transaction data)

Data Browser/Table View Maint.: Display/Maintenance Allowed

Step 14. To Create the TMG , Click on Utilities and then click on Table Maintenance Generator.



Step 15. Provide the details and click on the Find Scr. Number(s) Button.

Generate Table Maintenance Dialog: Generation Environment

Find Scr. Number(s)

Table/View **ZMV_FLIGHT**

Technical Dialog Details

Authorization Group **&NC&**

Authorization object **S_TABU_DI...**

Function group **ZMV_FLIGHT**

Package

Maintenance Screens

Maintenance type **one step**
 two step

Maint. Screen No. **Overview screen**

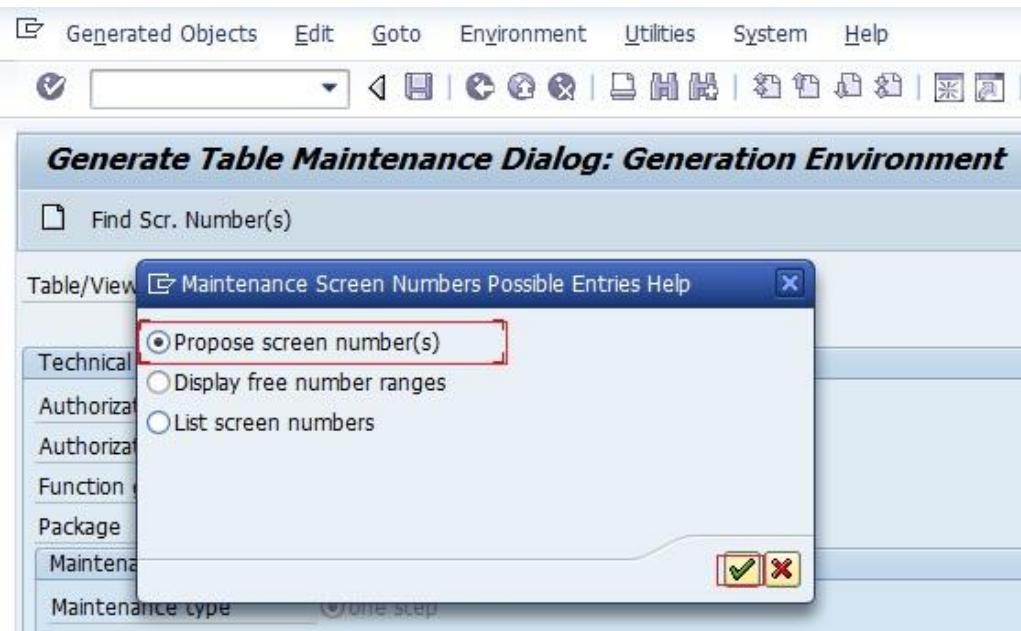
Single screen

Dialog Data Transport Details

Recording routine **Standard recording routine**
 no, or user, recording routine

Compare Flag **Automatically Adjustable** **Note**

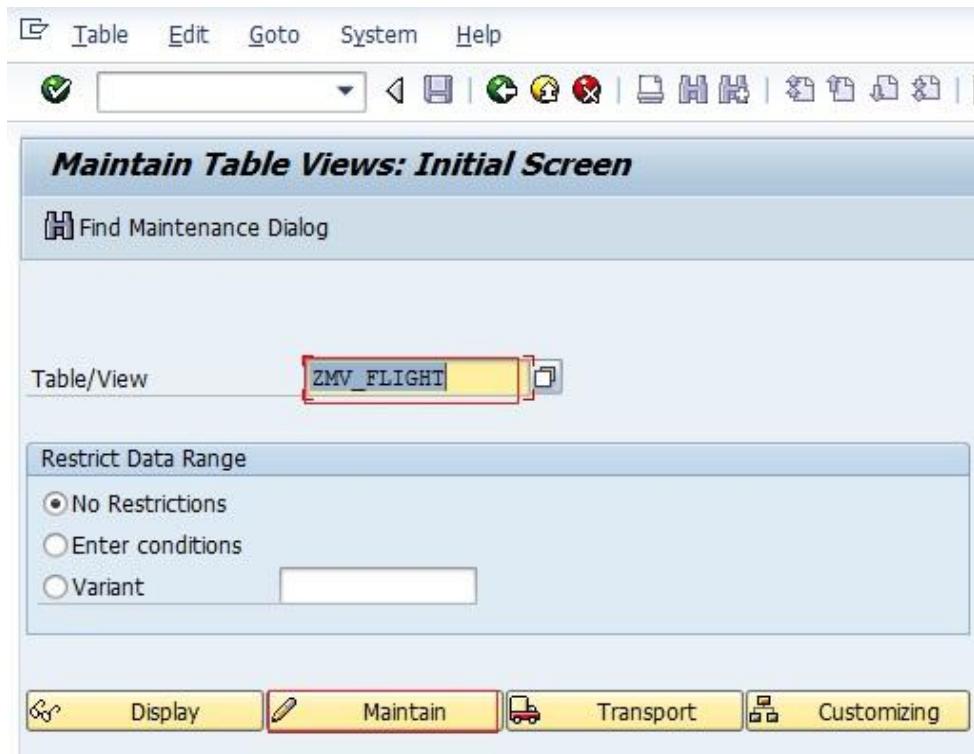
Step 16. Select the First Button and then click on the Tick button.



Step 17. System automatically fills the Overview Screen Number and then click on the Create button.

The screenshot shows the 'Generate Table Maintenance Dialog: Generation Environment' window. The 'Create (F6)' button is highlighted with a red box. In the 'Maintenance Screens' section, the 'Maint. Screen No.' field contains 'Overview screen' and has '1' entered in the adjacent input field. The 'Single screen' field is empty.

Step 18. Now go to TCODE-SM30 and provide the Maintenance View name and click on Maintain button.



Step 19. It shows the data from two tables and here you can create new entries, modify existing entries and delete Entries.

Flight Maintenance						
A...	Co...	Ctr	Depart. city	Ctr	Arrival city	Airline
AA	17	US	NEW YORK	US	SAN FRANCISCO	American Airlines
AA	26	DE	FRANKFURT	US	NEW YORK	American Airlines
AA	64	US	SAN FRANCISCO	US	NEW YORK	American Airlines
AZ	555	IT	ROME	DE	FRANKFURT	Alitalia
AZ	788	IT	ROME	JP	TOKYO	Alitalia
AZ	789	JP	TOKYO	IT	ROME	Alitalia
AZ	790	IT	ROME	JP	OSAKA	Alitalia
AZ	791	JP	OSAKA	IT	ROME	Alitalia
DL	106	US	NEW YORK	DE	FRANKFURT	Delta Airlines
DL	1699	US	NEW YORK	US	SAN FRANCISCO	Delta Airlines
DL	1984	US	SAN FRANCISCO	US	NEW YORK	Delta Airlines
JL	407	JP	TOKYO	DE	FRANKFURT	Japan Airlines
JL	408	DE	FRANKFURT	JP	TOKYO	Japan Airlines
LH	400	DE	FRANKFURT	US	NEW YORK	Lufthansa

SEARCH-HELP

DEFINES INPUT HELP (F4 HELP) FOR THE FIELDS OF DB TABLE. INPUT HELP PROVIDES A POSSIBLE SET OF VALID VALUES THAT YOU CAN ENTER FOR A FIELD.

ELEMENTARY SEARCH HELP:

A SINGLE SEARCH HELP FOR AN INPUT FIELD IS CALLED ELEMENTARY SEARCH HELP.

COLLECTIVE SEARCH HELP:

A COLLECTIVE OR GROUP OF ELEMENTARY SEARCH HELPS IS CALLED AS COLLECTIVE SEARCH HELP.

USING SEARCH HELPS:

THERE ARE 2 STEPS FOR USING THE SEARCH HELPS.

1. CREATE A SEARCH HELP AT SE11 TRANSACTION CODE.
2. ASSIGN THE SEARCH HELP TO A TABLE (OR) PROGRAM FIELD.

ASSIGN SEARCH HELP TO A PROGRAM FIELD.

MATCH-CODE OBJECT

IT IS A KEYWORD WHICH IS USED TO ASSIGN A SEARCH HELP FOR THE PARAMETERS INPUT FIELD ON THE SELECTION-SCREEN.

- ❖ BY DEFAULT THE STANDARD SEARCH HELP WILL BE DISPLAYED.
- ❖ WE CAN USE MATCH-CODE OBJECT TO ASSIGN OUR SEARCH HELP BY OVERWRITING THE STANDARD SEARCH HELP.

BUSINESS REQUIREMENT:

CREATE AN ELEMENTARY SEARCH HELP & ASSIGN IT TO A PROGRAM FIELD.

STEP1: CREATE A SEARCH HELP IN SE11.

- ☞ GO TO SE11.
- ☞ GIVE THE SEARCH HELP NAME EG: ZEL_SH.
- ☞ CLICK ON CREATE.
- ☞ SELECT ELEMENTARY SEARCH HELP.
- ☞ GIVE DESCRIPTION.
- ☞ SPECIFY SELECTION METHOD AS MARA (TABLE NAME).
- ☞ SPECIFY THE SEARCH HELP PARAMETER AS BELOW.

SEARCH HELP PARAMETERS	IMPORT	EXPORT	LIST POSITION	SCREEN POSITION
MATNR	<input type="checkbox"/>	<input type="checkbox"/>	1	1
MTART	<input type="checkbox"/>	<input type="checkbox"/>	2	2
MBRSH	<input type="checkbox"/>	<input type="checkbox"/>	3	3

- ☞ SAVE, ACTIVATE AND TEST.

STEP2: ASSIGN THE SEARCH HELP FOR THE PROGRAM FIELD.

USE THE MATCHCODE OBJECT KEYWORD AND ASSIGN THE SEARCH HELP.

PARAMETER: P_MATNR1 TYPE MARA-MATNR (DISPLAYS STANDARD SEARCH HELP)

PARAMETER: P_MATNR2 TYPE MARA-MATNR **MATCHCODE** OBJECT ZEL_SH.

- ✓ OVERWRITES THE STANDARD SEARCH HELP
- ✓ USED FOR CUSTOM TABLE. FOR CUSTOM TABLES WE NEED TO CREATE SEARCH HELP.
- ✓ DISPLAYS OUR OWN SEARCH HELP BECAUSE WE NEED MATCH CODE OBJECT.

EXAMPLE 2 ON SEARCH HELP

BUSINESS REQUIREMENT:

CREATE A SEARCH HELP FOR THE TABLE AND ASSIGN IT TO TABLE FIELDS.

STEP 1:

CREATE A TABLE BY NAME YYKNA1 WITH CUSTOMER NUMBER, CUSTOMER NAME, COUNTRY.

STEP 2:

CREATE A SEARCH HELP FOR THE ABOVE TABLE.

- ☞ GO TO SE11.
- ☞ GIVE SEARCH HELP NAME AS ZEL_SH1.
- ☞ CLICK ON CREATE AND GIVE DESCRIPTION.
- ☞ GIVE SELECTION METHOD AS YYKNA1 (TABLE NAME).
- ☞ SPECIFY THE SEARCH HELP PARAMETERS AS BELOW.

SEARCH HELP	IMPORTING	EXPORTING	LIST POSITION	SCREEN POSITION
CUSTOMER NO.	<input type="checkbox"/>	<input type="checkbox"/>	1	1
CUSOTMER NAME	<input type="checkbox"/>	<input type="checkbox"/>	2	2
COUNTRY	<input type="checkbox"/>	<input type="checkbox"/>	3	3

- ☞ SAVE AND ACTIVATE
- ☞ TEST IT.

STEP 3:

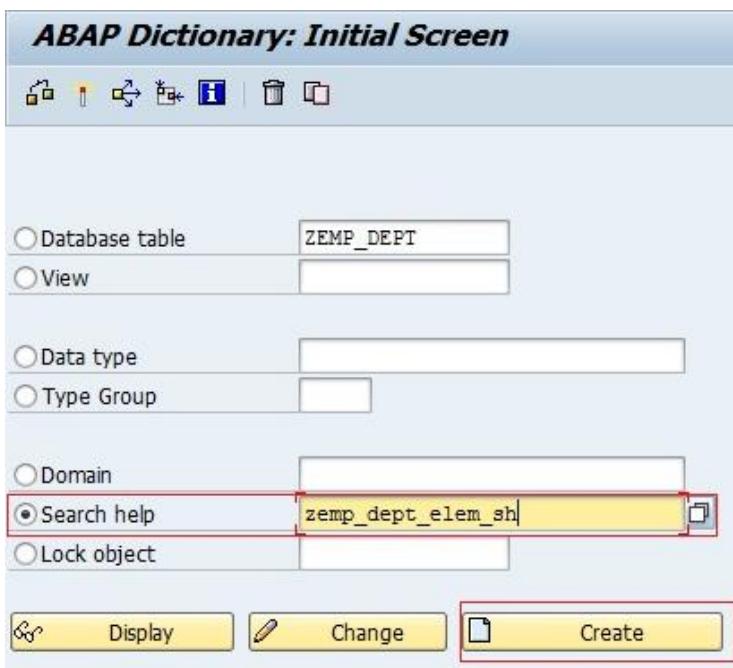
ASSIGN THE SEARCH HELP TO TABLE FIELD.

- ☞ GO TO SE11.
- ☞ GIVE THE TABLE NAME AS YYKNA1.
- ☞ CLICK ON CHANGE.
- ☞ SELECT THE CUSTOMER NO. FIELD AND CLICK ON THE BUTTON SEARCH HELP.
- ☞ GIVE THE SEARCH HELP NAME AS ZEL_SH.
- ☞ PRESS ENTER
- ☞ CLICK ON GENERATE PROPOSAL
- ☞ CLICK ON COPY BUTTON.
- ☞ SAVE AND ACTIVATE.
- ☞ CLICK ON CONTENTS ICON.
- ☞ CLICK ON SEARCH HELP BUTON FOR CUSTOMER NO. FIELD AND SELECT A RECORD.
- ☞ CLICK ON EXECUTE. THE RECORD WILL BE DISPLAYED.

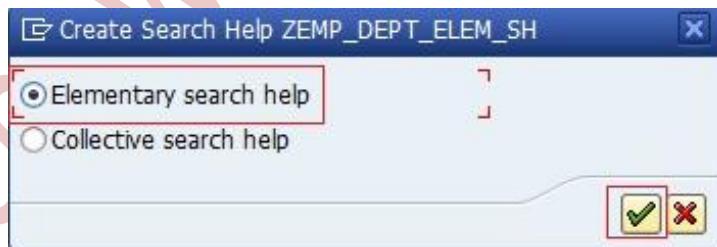
Example 3: Creating Search Help and Assigning to a Table

Creating a search help and Assigning it to the table

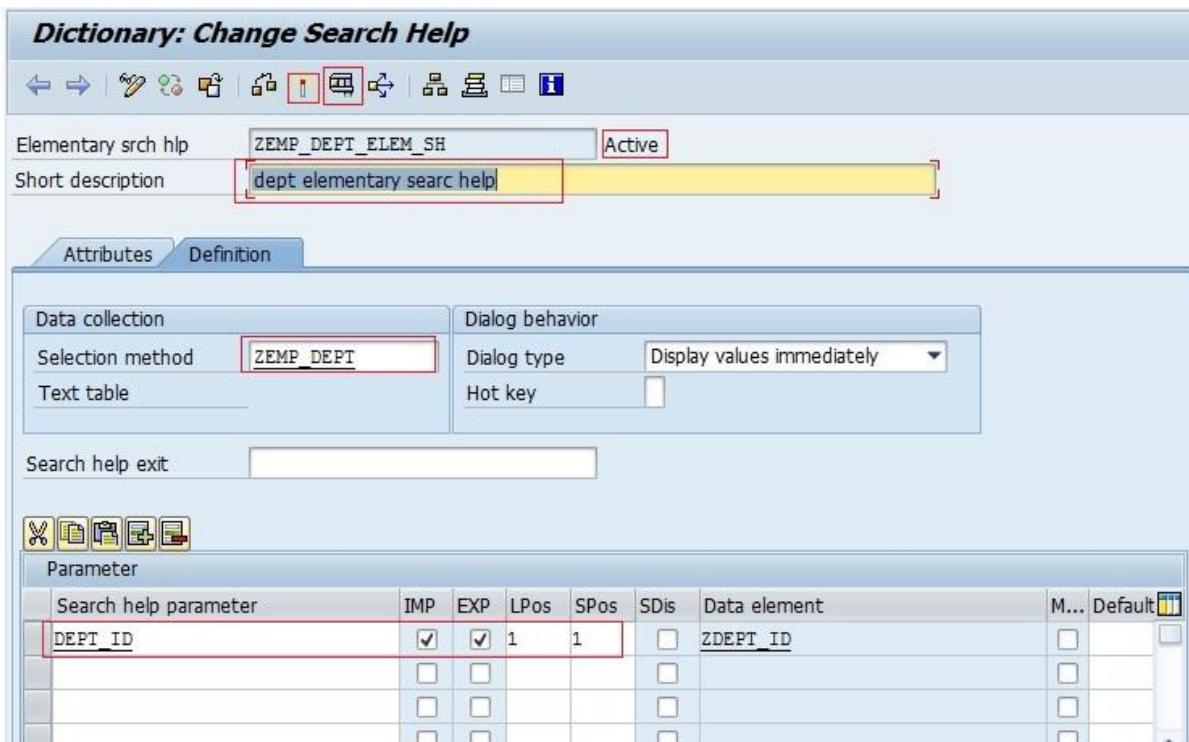
Step 1. Go to TCODE- SE11, Select the Search Help Radio button and Provide the Name and click on Create Button.



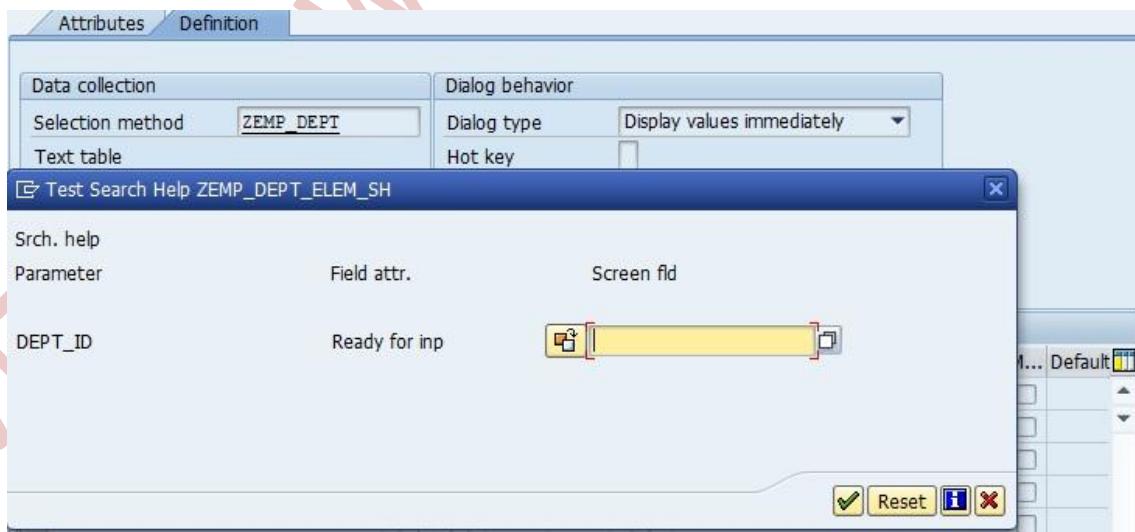
Step 2 Select the Elementary Search Help and Click on the Tick button.



Step 3 Provide the short text. In the selection method provide the table name from where the set of values we will get. In this case the table is 'ZEMP_DEP' is used. (see the table creation of table in the previous post). In the Search help parameter, press F4 button and it will show all the fields of table ZEP_DEPT . Select the DEPT_ID field of the table. Select the IMP and EXP and maintain 1, 1 in the LPos and RPos. Activate the search help and click on the EXECUTE/F8 Button to test the search help.



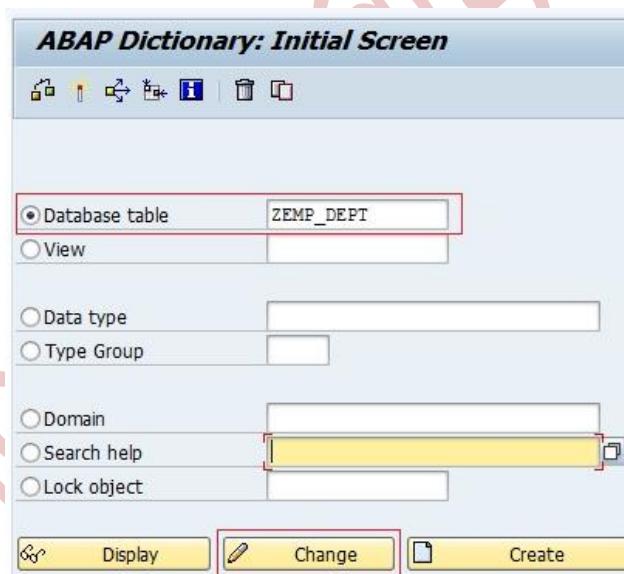
Step 4. Press the F4 button.



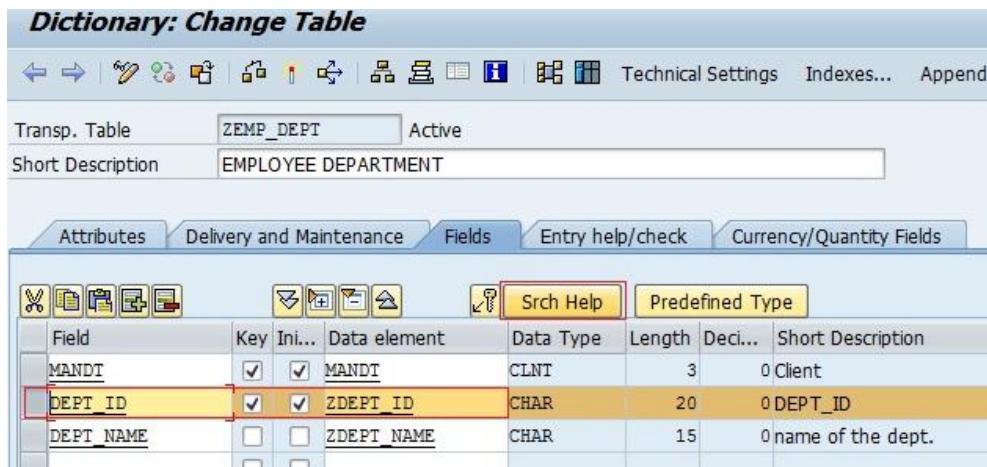
Step 5. Now it shows all the DEPT_ID of the table ZEP_DEPT.



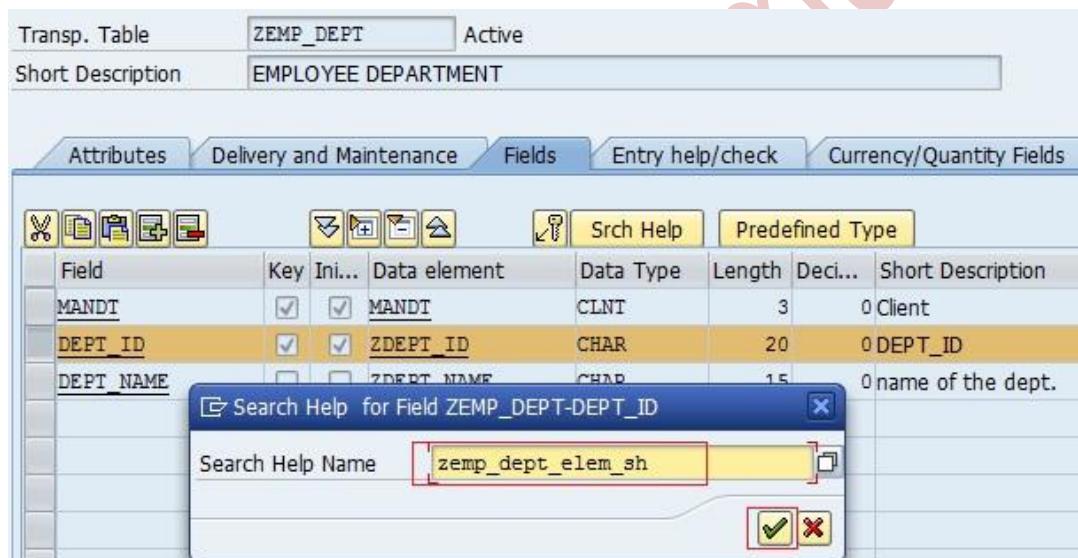
Step 6. Let's assign the created search help to the Table. Go to TCODE- SE11, provide the table name 'ZEMP_DEPT' and click on the change button.



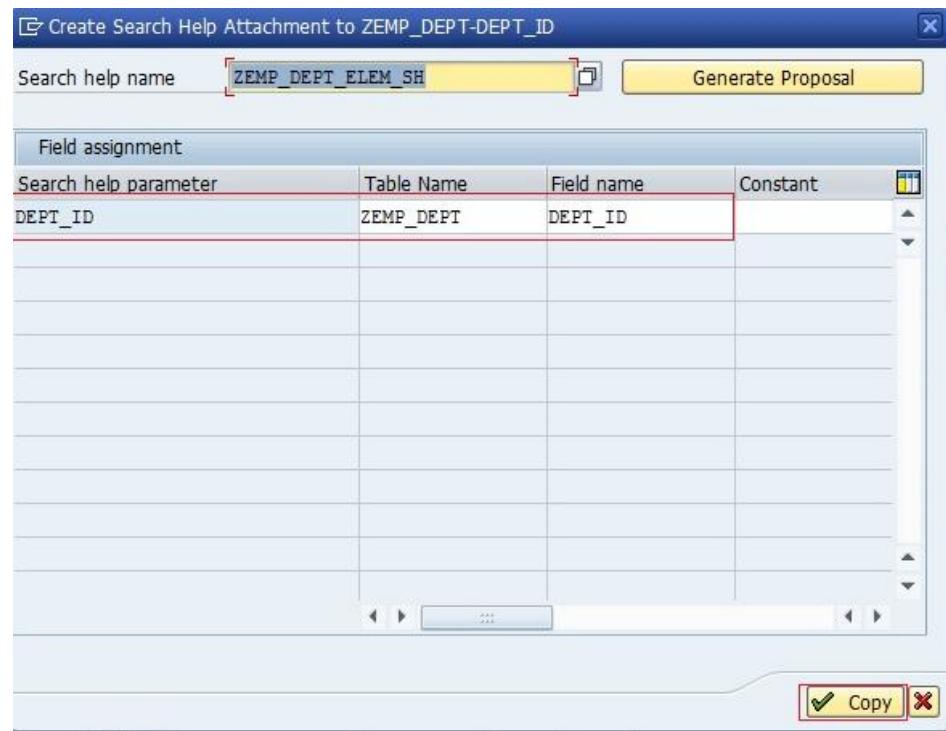
Step 7. Select the Field 'DEPT_ID' and click on the Search Help Button.



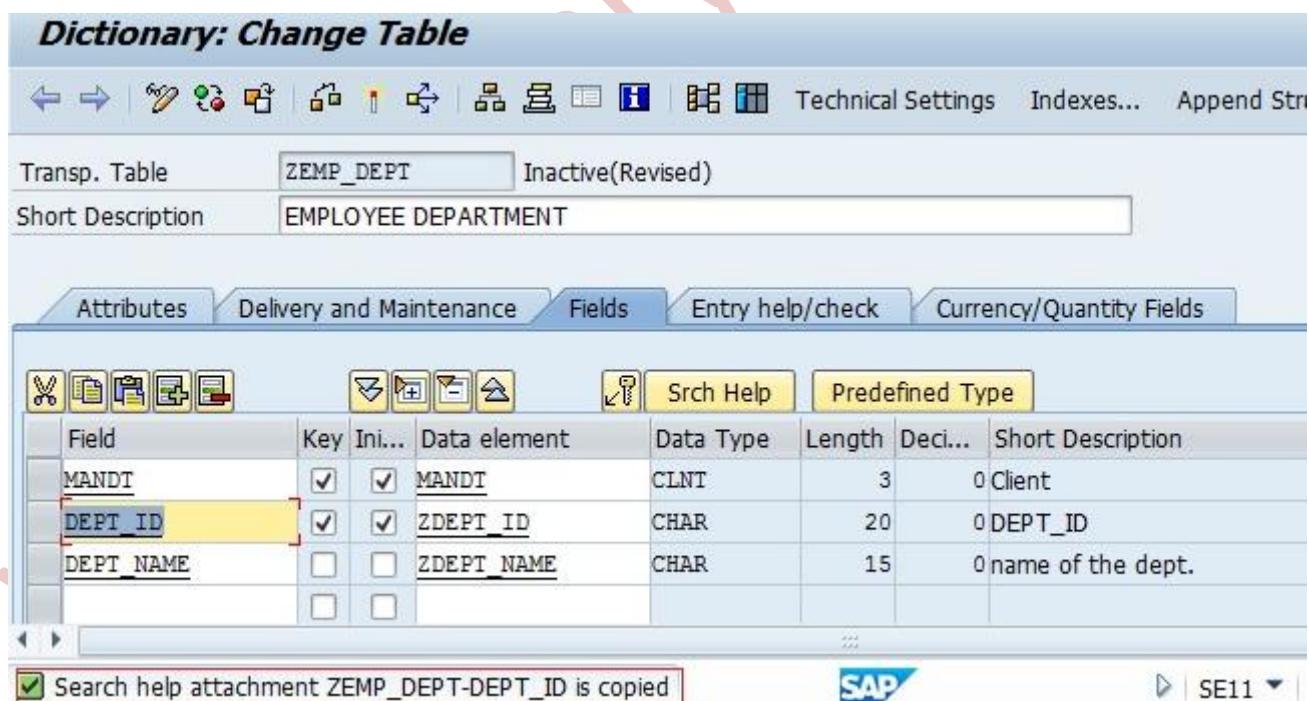
Step 8. Provide the Search help name created above and click on the tick button.



Step 9. See the mapping of the Search help parameter of the Search help to the field of the table. Click on the COPY Button.



Step 10. See the below status message and activate the table.



Step 11. Click on the Entry Help/Check tab, now the search help is assigned to the field.

Transp. Table **ZEMP_DEPT** Inactive(Revised)
Short Description EMPLOYEE DEPARTMENT

Attributes Delivery and Maintenance Fields **Entry help/check** Currency/Quantity Fields

Field	Data element	Data T...	Fo...	C...	Orig...	Srch Help	D...	Domain
MANDT	MANDT	CLNT	<input type="checkbox"/>				<input type="checkbox"/>	MANDT
DEPT_ID	ZDEPT_ID	CHAR	<input type="checkbox"/>		Expli...	ZEMP_DEPT_ELEM_SH	<input type="checkbox"/>	ZDEPT_ID
DEPT_NAME	ZDEPT_NAME	CHAR	<input type="checkbox"/>				<input type="checkbox"/>	ZDEPT_NAME

Step 12. Now click on the Contents button.

Dictionary: Change Table

Transp. Table **ZEMP_DEPT** Active
Short Description EMPLOYEE DEPARTMENT

Contents (Ctrl+Shift+F10)

Attributes Delivery and Maintenance Fields **Entry help/check** Currency/Quantity Fields

Field	Data element	Data T...	Fo...	C...	Orig...	Srch Help	D...	Domain
MANDT	MANDT	CLNT	<input type="checkbox"/>				<input type="checkbox"/>	MANDT
DEPT_ID	ZDEPT_ID	CHAR	<input type="checkbox"/>		Expli...	ZEMP_DEPT_ELEM_SH	<input type="checkbox"/>	ZDEPT_ID
DEPT_NAME	ZDEPT_NAME	CHAR	<input type="checkbox"/>				<input type="checkbox"/>	ZDEPT_NAME

Step 13. Now click on the F4 button of the DEPT_ID field and the search help gives the set of values.

Program Edit Goto Settings System Help

DEPT_ID (2) 9 Entries found

Number of Entries

DEPT_ID

001

002

003

004

005

006

007

009

010

Width of Output List 250

Maximum No. of Hits 10.000.000

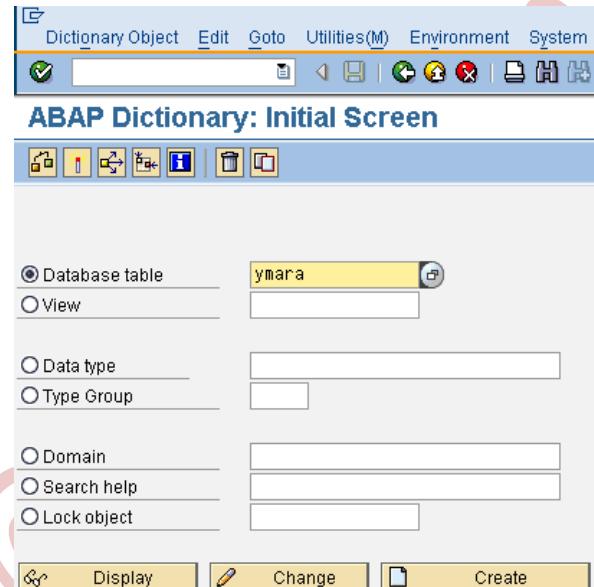
EXAMPLE 4 ON SEARCH HELP

BUSINESS REQUIREMENT:

CREATE A SEARCH HELP FOR THE TABLE AND ASSIGN IT TO TABLE FIELDS.

STEP 1: CREATING A DATABASE TABLE WITH FEW RECORDS.

- ☞ GO TO SE11.
- ☞ GIVE DATABASE TABLE NAME AS YMARA AND CLICK ON CREATE BUTTON.



- ☞ GIVE THE SHORT DESCRIPTION.
- ☞ SPECIFY DELIVERY AND MAINTENANCE.
- ☞ CLICK ON FIELDS TAB, AND SPECIFY THE BELOW FIELDS.
- ☞ CLICK ON TECHNICAL SETTINGS TAB AND SPECIFY DATA CLASS AND SIZE CATEGORY.
- ☞ SAVE AND ACTIVATE THE TABLE.

The screenshot shows the SAP Dictionary: Maintain Table screen. The table 'YMARA' is selected. The 'Fields' tab is active, showing a list of fields: MANDT, MATNR, MTART, MBRSH, and MEINS. Each field has a checkbox for 'Key' and 'Initial'. The 'Data element' column lists the corresponding fields from the table. The 'Data Type' column shows CLNT, CHAR, CHAR, CHAR, and UNIT respectively. The 'Length' column shows 3, 18, 4, 1, and 3. The 'Decim...' column shows 0, 0, 0, 0, and 0. The 'Short Description' column provides a brief explanation for each field.

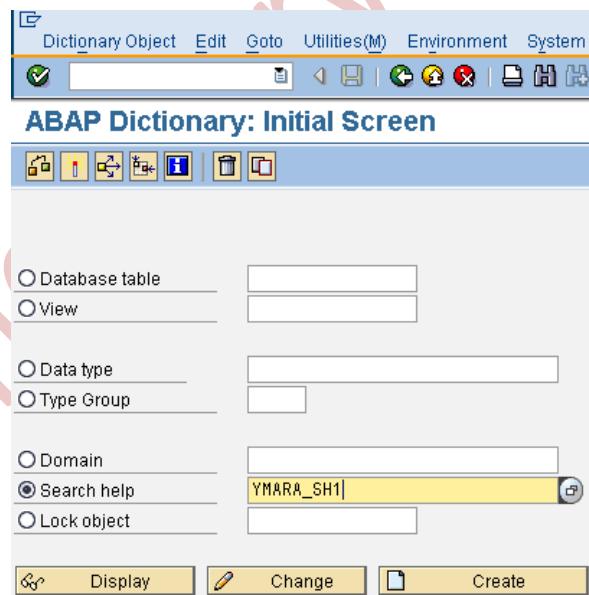
→ UTILITIES → TABLE CONTENTS → CREATE ENTRIES.

→ CREATE 4 TO 5 RECORDS.

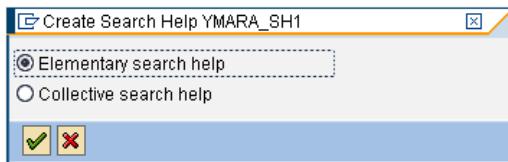
MANDT	MATNR	MTART	MBRSH	MEINS
800	0000000000000000040	DDD	D	
800	00000000000000000500	DDD	D	
800	00000000000000001000	AAA	A	KGS
800	00000000000000002000	BBB	B	KGS
800	00000000000000004000	DDD	D	KG

STEP 2: CREATING ELEMENTARY SEARCH HELP.

- ☞ GO TO SE11,
- ☞ SELECT SEARCH HELP, GIVE SEARCH HELP NAME AS YMARA_SH1
- ☞ CLICK ON CREATE BUTTON.



- ☞ SELECT ELEMENTARY SEARCH HELP AND CLICK ON CONTINUE.



- ☞ GIVE SHORT DESCRIPTION
- ☞ GIVE SELECTION METHOD AS YMARA.
- ☞ DEFINE THE SEARCH HELP PARAMETERS AS BELOW.

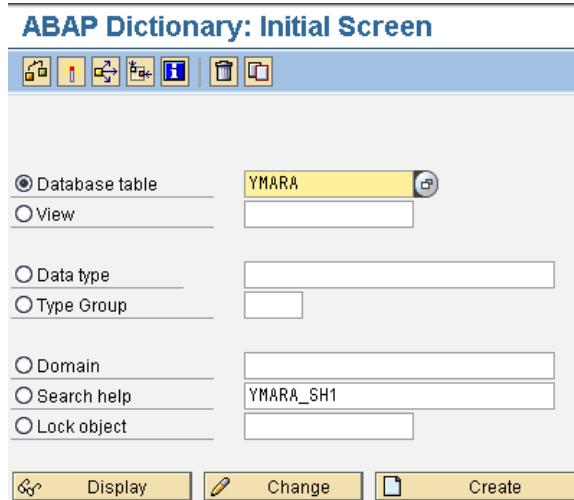
- ☞ SAVE AND ACTIVATE, TEST IT IMMEDIATELY.

Dictionary: Maintain Search Help

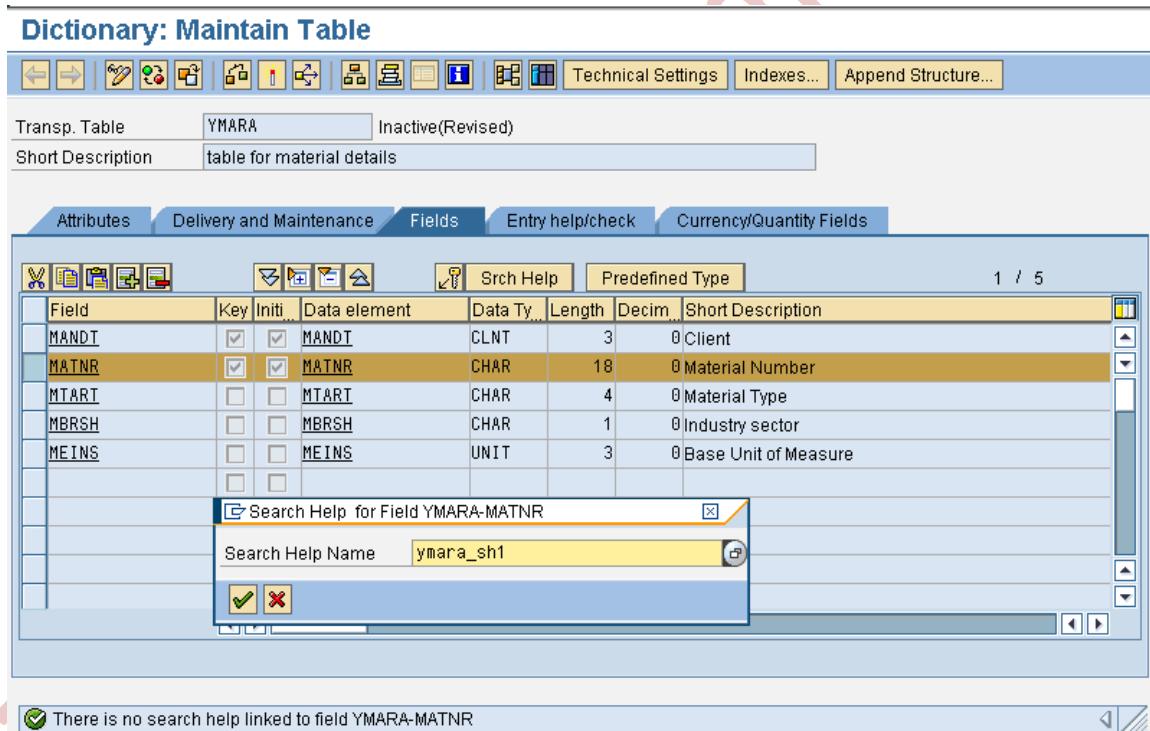
Attributes		Definition																															
Elementary srch hlp	YMARA_SH1	New(Revised)																															
Short description	elementary search help for ymara																																
Data collection	Selection method	YMARA	Dialog behavior																														
	Text table		Dialog type	Display values immediately																													
			Hot key																														
Search help exit																																	
<table border="1"> <thead> <tr> <th colspan="2">Parameter</th> </tr> <tr> <th>Search help parameter</th> <th>IMP</th> <th>EXP</th> <th>LPos</th> <th>SPos</th> <th>SDis</th> <th>Data element</th> <th>M...</th> <th>Default</th> </tr> </thead> <tbody> <tr> <td>MATNR</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>1</td> <td>1</td> <td><input type="checkbox"/></td> <td>MATNR</td> <td><input type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>MTART</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>2</td> <td>2</td> <td><input type="checkbox"/></td> <td>MTART</td> <td><input type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> </tr> </tbody> </table>					Parameter		Search help parameter	IMP	EXP	LPos	SPos	SDis	Data element	M...	Default	MATNR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1	1	<input type="checkbox"/>	MATNR	<input type="checkbox"/>	<input checked="" type="checkbox"/>	MTART	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	2	2	<input type="checkbox"/>	MTART	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Parameter																																	
Search help parameter	IMP	EXP	LPos	SPos	SDis	Data element	M...	Default																									
MATNR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1	1	<input type="checkbox"/>	MATNR	<input type="checkbox"/>	<input checked="" type="checkbox"/>																									
MTART	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	2	2	<input type="checkbox"/>	MTART	<input type="checkbox"/>	<input checked="" type="checkbox"/>																									

STEP 3: ASSIGNING ELEMENTARY SEARCH HELP AT TABLE LEVEL.

GO TO SE11, GIVE THE TABLE NAME AS YMARA, CLICK ON CHANGE BUTTON.

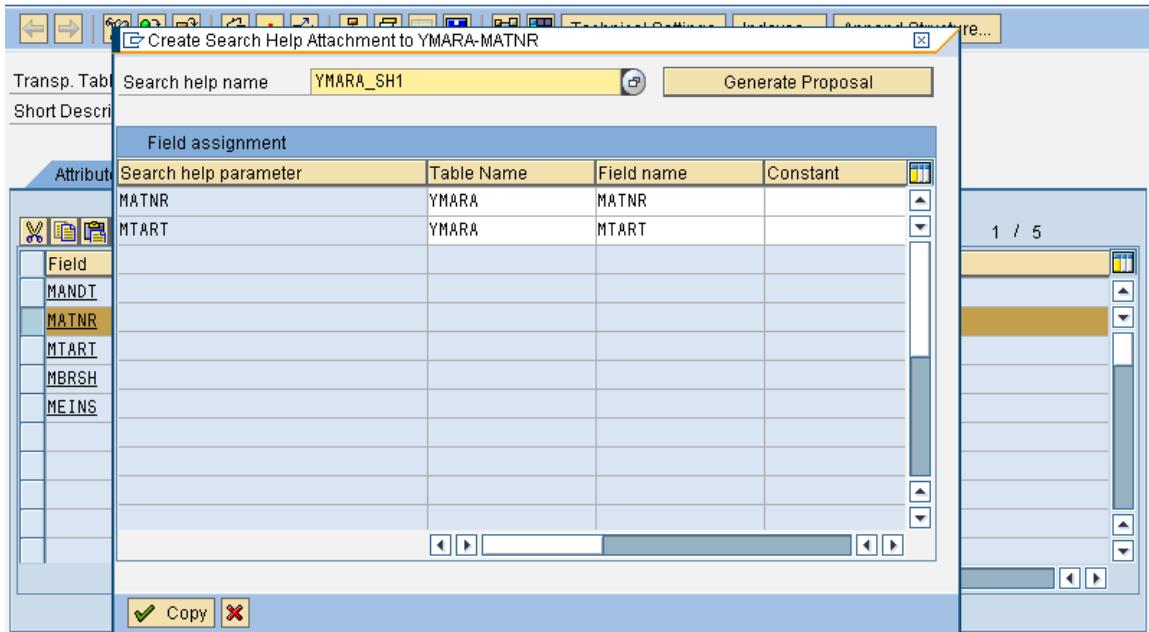


SELECT THE FIELD MATNR, CLICK ON SEARCH HELP BUTTON.



- ☞ GIVE SEARCH HELP NAME AND CLICK ON CONTINUE.
- ☞ CLICK ON GENERATE PROPOSAL.
- ☞ CLICK ON COPY BUTTON.

Dictionary: Maintain Table



- ☞ **SAVE, ACTIVATE, TEST THE TABLE.**
- ☞ **NOW SEARCH HELP WILL BE AVAILABLE ON MATNR FIELD.**

Table YMARA Insert

The screenshot shows the SAP Table YMARA Insert screen. On the left, there are input fields for "MANDT" (value 800) and "MATNR". Below these are "MTART", "MBRSH", and "MEINS" fields. On the right, a search help dialog is open with the title "Material Number (2) 5 Entries found". The dialog has a "Restrictions" tab and a table with columns "Materi..." and "MTyp". The table contains the following data:

Materi...	MTyp
1000	AAA
2000	BBB
40	DDD
4000	DDD
500	DDD

At the bottom of the dialog, it says "5 Entries found".

COLLECTIVE SEARCH HELP

A GROUP OF ELEMENTARY SEARCH HELP'S ARE CALLED COLLECTIVE SEARCH HELP.

EXAMPLE 1:

DEVELOP AN ELEMENTARY SEARCH HELP BY NAME ZEL_SH WITH MATNR, MTART, MBRSH, FIELDS FROM MARA TABLE.

SIMILARLY ONE MORE ELEMENTARY SEARCH HELP BY NAME YEL_SH WITH MATNR, MTART, MEINS FIELDS FROM MARA TABLE.

NOW CREATE A COLLECTIVE SEARCH HELP BY COMBINING THE ABOVE TWO ELEMENTARY SEARCH HELP.

- ☞ GO TO SE11.
- ☞ GIVE SEARCH HELP NAME AS ZCOLL_SH2.
- ☞ CLICK ON CREATE.
- ☞ SELECT COLLECTIVE SEARCH HELP AND GIVE DESCRIPTION.
- ☞ DEFINE THE SEARCH HELP PARAMETERS AS BELOW.

SEARCH HELP	IMPORT	EXPORT	DATA ELEMENT
MATNR	<input type="checkbox"/>	<input type="checkbox"/>	MATNR
MTART	<input type="checkbox"/>	<input type="checkbox"/>	MTART
MBRSH	<input type="checkbox"/>	<input type="checkbox"/>	MBRSH
MEINS	<input type="checkbox"/>	<input type="checkbox"/>	MEINS

- ☞ CLICK ON INCLUDED SEARCH HELP TAB.
- ☞ SPECIFY THE FIRST ELEMENTARY SEARCH ZEL_SH AND PRESS ENTER.
- ☞ CLICK ON PARAMETER ASSIGNMENT.
- ☞ THE SYSTEM WILL GENERATE A PROPOSAL, CLICK ON YES.
- ☞ THE PARAMETER ASSIGNMENT WILL BE AUTOMATICALLY DONE.
- ☞ SIMILARLY GIVE THE SECOND SEARCH HELP NAME AS YEL_SH AND PRESS ENTER.
- ☞ CLICK ON PARAMETER ASSIGNMENT AND ACCEPT THE PROPOSAL.
- ☞ FINALLY SAVE, ACTIVATE AND TEST IT.
- ☞ NOW ASSIGN THIS SEARCH HELP AT THE TABLE LEVEL OR PROGRAM LEVEL.

EG: PARAMTERS: P_MATNR2 TYPE MARA-MATNR MATCHCODE OBJECT ZCOLL_SH.

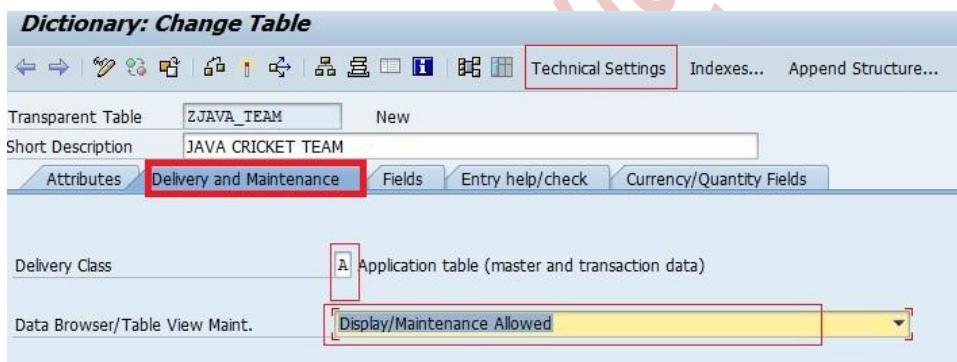
EXAMPLE: Collective Search Help

Business Scenario: Lets to find out the Name we can refer to two different tables which provides two different ways of Name selection. Let's first create two tables and build elementary search help on two tables and use these elementary search help in the Collective Search Help.

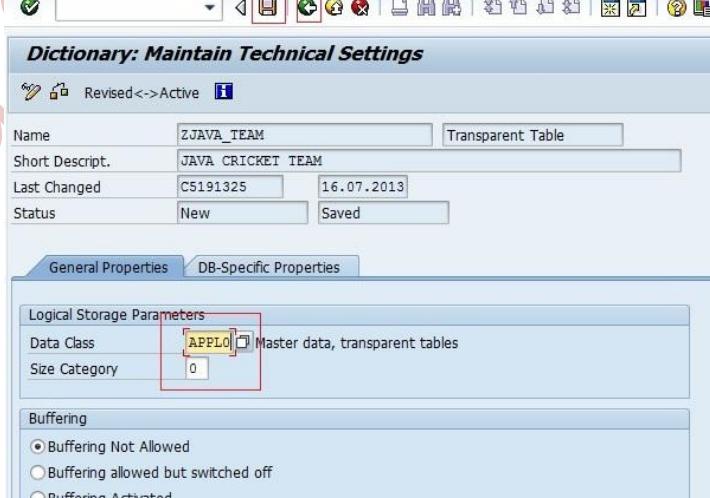
Step 1. Go to TCODE-SE11, Create a table ZJAVA_TEAM.



Step 2. Provide the delivery class, save it local package and click on the Technical settings button.



Step 3. Provide the technical settings details as mentioned below and save it and at last click on the BACK button.



Step 4. Mention the table Fields, data elements, primary keys and activate the table.

The screenshot shows the SAP Dictionary: Change Table interface for the table ZJAVA_TEAM. The table is activated. The fields defined are:

Field	Key	Data element	Data Type	Length	Deci...	Short Description
MANDT	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0	Client
CRID_ID	<input checked="" type="checkbox"/>	ZCRIC_ID	CHAR	3	0	CRCI ID
CRIC_NAME	<input type="checkbox"/>	ZCRIC_NAME	CHAR	20	0	CRICKTER NAME

Step 5. At last mention the records in the table so that we can get some Names.

The screenshot shows the New Entries: Overview of Added Entries screen for the table JAVA CRICKET TEAM. The entries listed are:

01	SACHIN
02	SOURAV
03	DRAVID
04	JADEJA
05	ROBIN

Step 6. After record creation, we have the set of values in the table.

The screenshot shows the Data Browser: Table ZJAVA_TEAM Select Entries screen. The table structure is:

MANDT	CRID_ID	CRIC_NAME
001	01	SACHIN
001	02	SOURAV
001	03	DRAVID
001	04	JADEJA
001	05	ROBIN

Step 7. Let's create another table ZSAP_TEAM with same delivery and technical details. Maintain the same field name, data element, primary key fields and at last activate the table.

Dictionary: Change Table

The screenshot shows the SAP Dictionary: Change Table interface. The table structure is defined as follows:

Field	Key	Inited	Data element	Data Type	Length	Deci...	Short Description
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0	Client
CRID_ID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ZCRIC_ID	CHAR	3	0	CRCI ID
CRIC_NAME	<input type="checkbox"/>	<input type="checkbox"/>	ZCRIC_NAME	CHAR	20	0	CRICKTER NAME

The table is marked as Active and has a short description of JAVA CRICKET TEAM.

Step 8. Let's maintain some records in this table also.

New Entries: Overview of Added Entries

The screenshot shows the New Entries: Overview of Added Entries screen. The table contains the following data:

SAP CRICKET TEAM	
+	+
01	DHONI
02	YUVRAJ
03	RAINA
04	DHAWAN
05	RANINDU

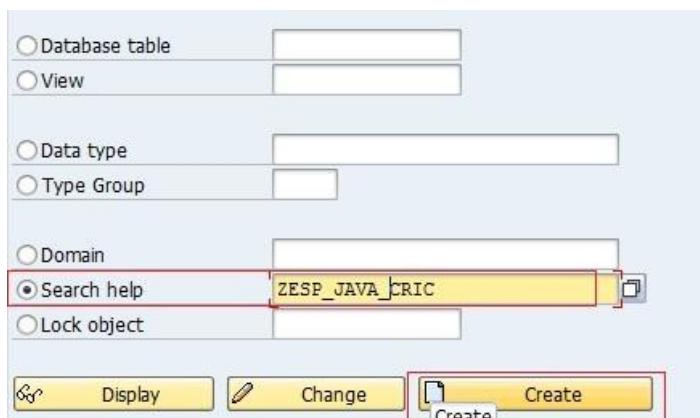
Step 9. All the records created in the table are shown below.

Data Browser: Table ZSAP_TEAM Select Entries 5

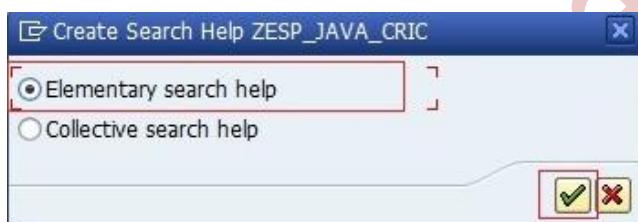
The screenshot shows the Data Browser for the ZSAP_TEAM table. The table structure is:

MANDT	CRID_ID	CRIC_NAME
001	01	DHONI
001	02	YUVRAJ
001	03	RAINA
001	04	DHAWAN
001	05	RANINDU

Step 10. Let's create a elementary search help on table ZJAVA_TEAM. Go to TCODE-SE11, Provide the Elementary search help name and click on create button.



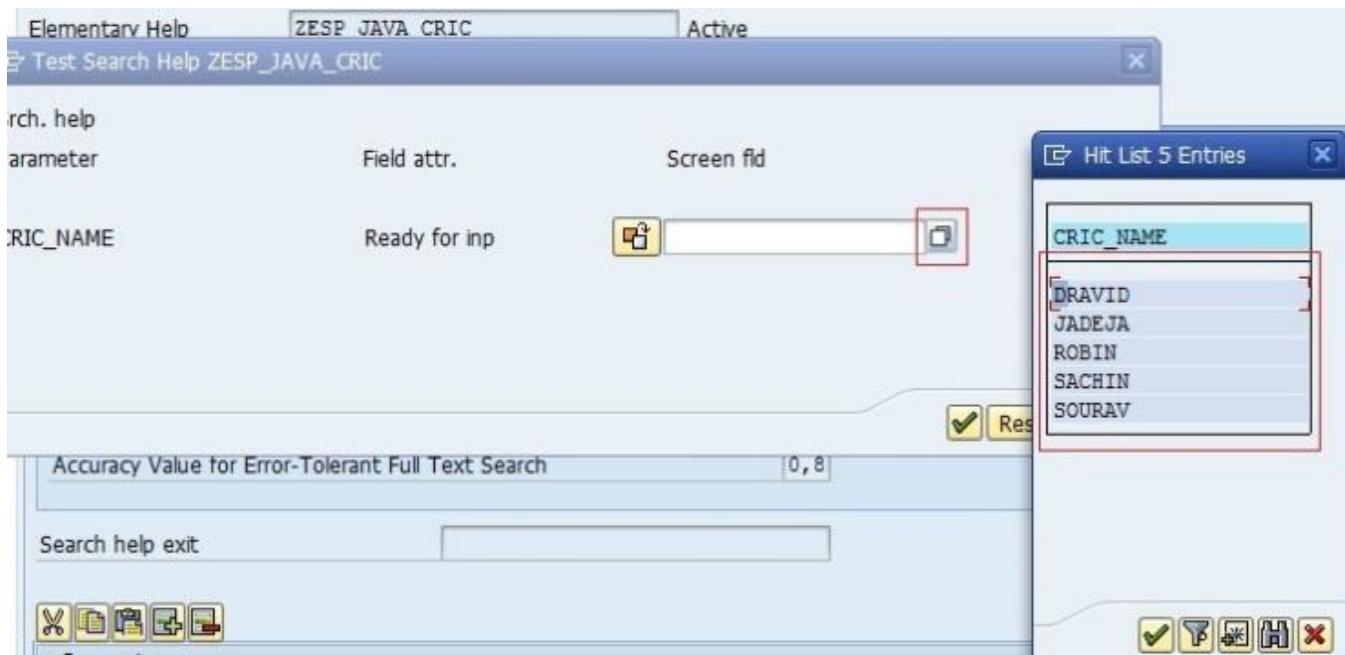
Step 11. Select the Elementary Search Help radio button and then click on the Tick Button.



Step 12. Provide the short text, Under the Definition tab provide the Table name against the Selection method. Mention the Search help parameter as field 'CRIC_NAME', select the Imp and Exp check box and Provide LPos and SPos values as mentioned below and activate the Elementary search help. Now click on the execute button to test the Search Help.

The screenshot shows the SAP SE11 dialog box for defining the search help. The 'Definition' tab is selected. The 'Selection method' is set to 'ZJAVA_TEAM'. The 'Search help parameter' is set to 'CRIC_NAME'. The 'IMP' and 'EXP' checkboxes are checked. The 'Dialog type' is set to 'Display values immediately'. The 'Accuracy Value for Error-Tolerant Full Text Search' is set to '0,8'. The 'Parameter' table shows the search help parameter 'CRIC_NAME' with 'IMP' and 'EXP' checked.

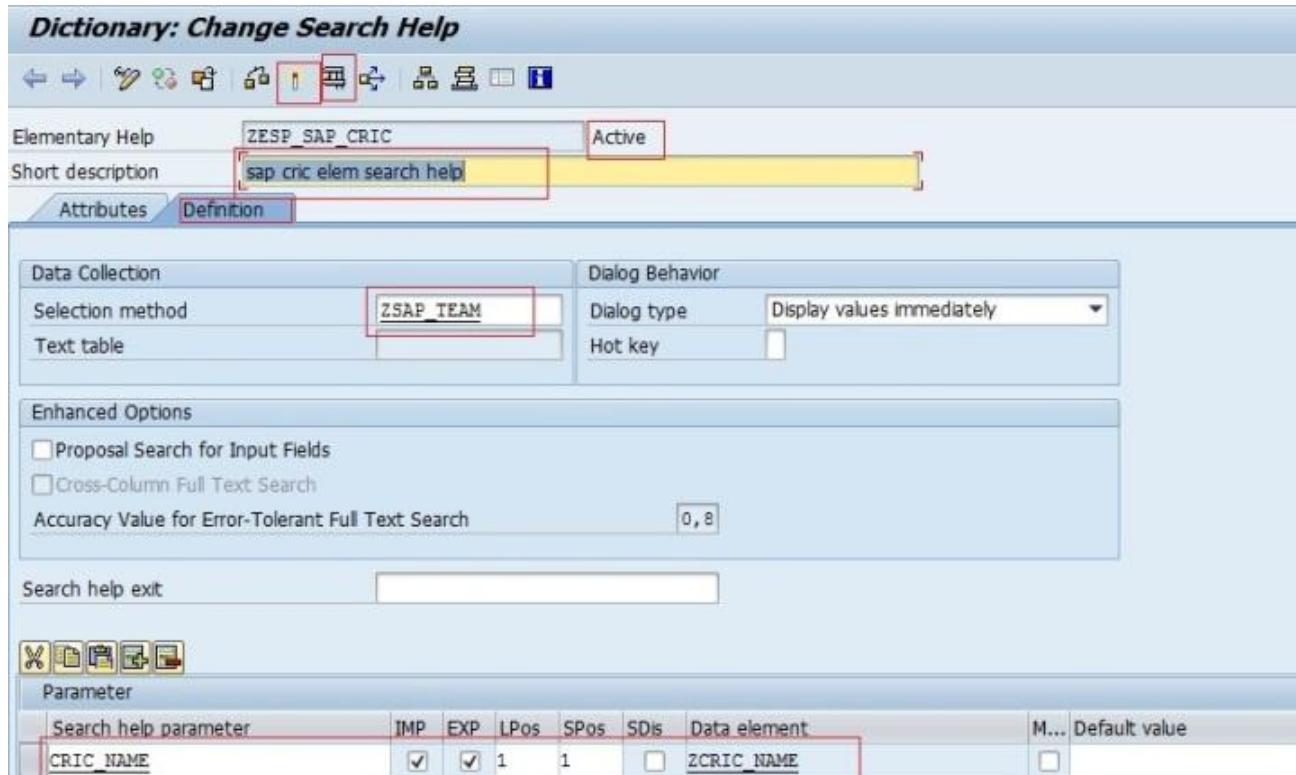
Step 13. Click on the F4 button and it display a set of values.



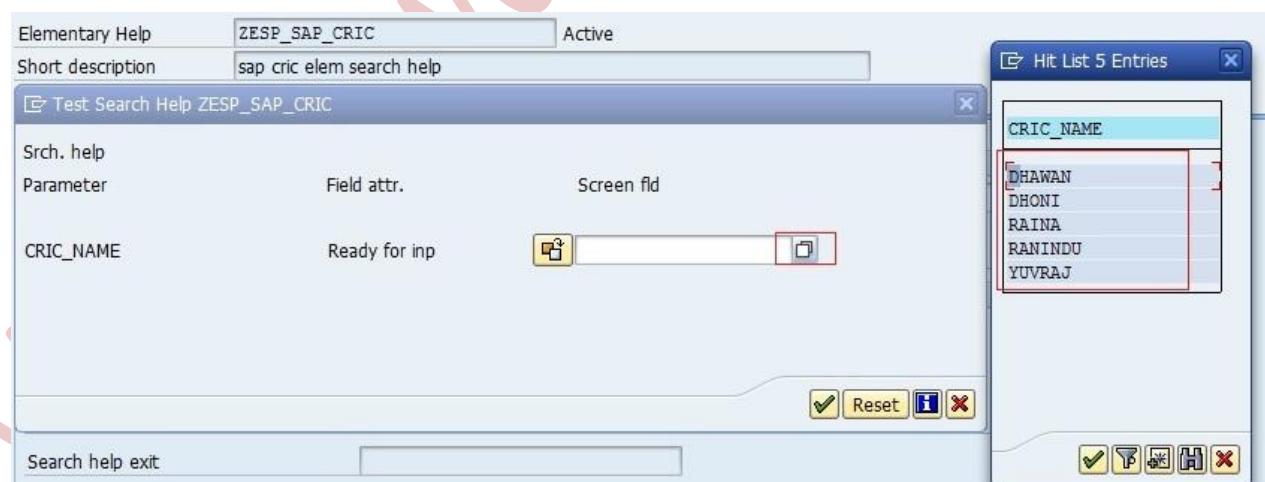
Step 14. Again create elementary Search help, provide the name and click on Create button.



Step 15. Provide the short text, Under the Definition tab, provide the Table name against the Selection method. Mention the Search help parameter as field 'CRIC_NAME', select the Imp and Exp check box and Provide LPos and SPos values as mentioned below and activate the Elementary search help. Now click on the execute button to test the Search Help.



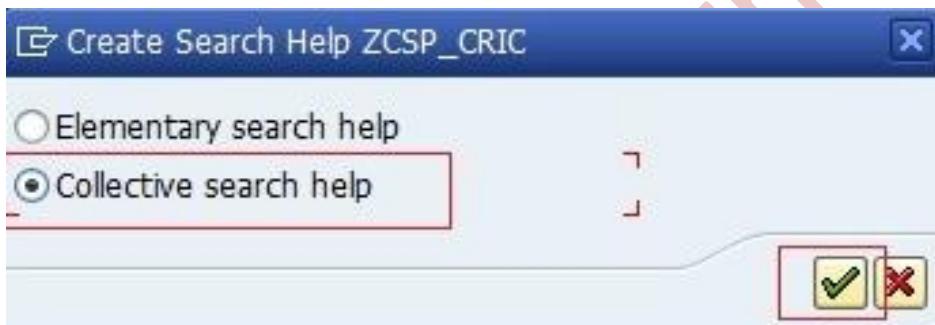
Step 16. Click on the F4 button and it display a set of values.



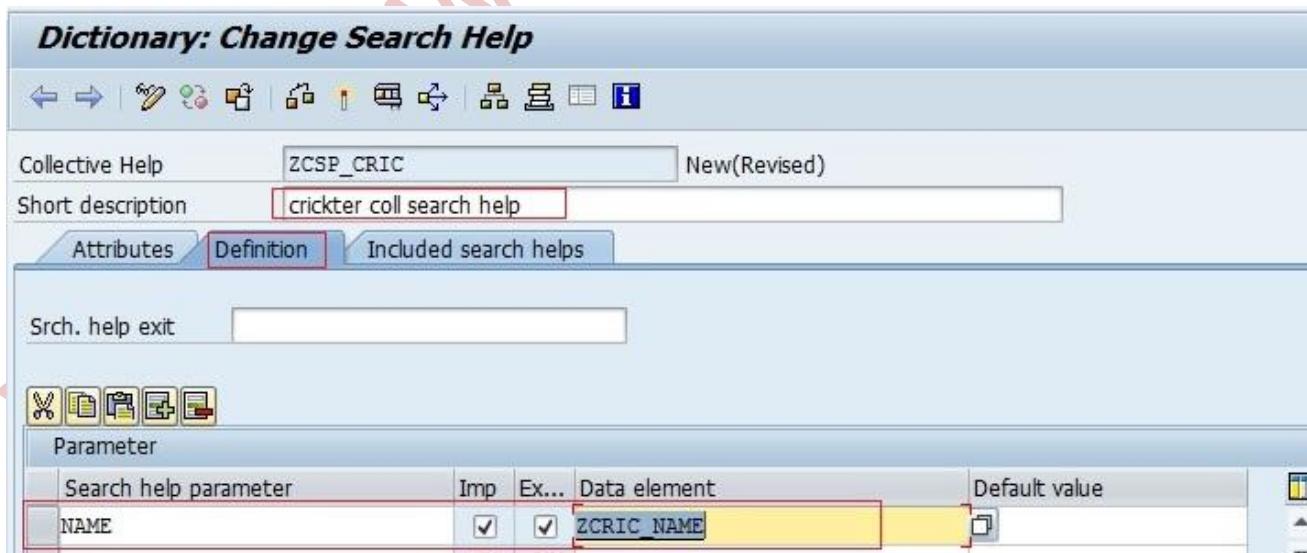
Step 17. Our two elementary search help is ready, Let's create Collective Search Help. Provide the Search help name and click on the Create Button.



Step 18. Select the Collective Search Help Radio Button and click on Tick Button.



Step 19. Provide the Short text . Under the Definition tab, Provide the Search help parameter name and click on the Imp and Exp Check Box. Now Click on the Included search helps tab.



Step 20. Under the Search help, mention previously created two elementary search help Name.

Dictionary: Change Search Help

Collective Help: ZCSP_CRIC (New(Revised))

Short description: crickter coll search help

Included search helps:

Search Help	Hidden	Short text
ZESP_JAVA_CRIC	<input type="checkbox"/>	JAVA CRIC SEARCH HELP
ZESP_SAP_CRIC	<input type="checkbox"/>	sap cric elem search help

Step 21. Now select the first elementary search help and click on the Param. Assignment Button.

Dictionary: Change Search Help

Collective Help: ZCSP_CRIC (Active)

Short description: crickter coll search help

Included search helps:

Search Help	Hidden	Short text
ZESP_JAVA_CRIC	<input type="checkbox"/>	JAVA CRIC SEARCH HELP
ZESP_SAP_CRIC	<input type="checkbox"/>	sap cric elem search help

Step 22. Now system automatically maps the elementary search help reference parameter (CRIC_NAME) to the Collective Search help parameter (NAME) and at last click on Copy button.

Dictionary: Change Search Help

Collective Help: ZCSP_CRIC

Short description: crickter coll search help

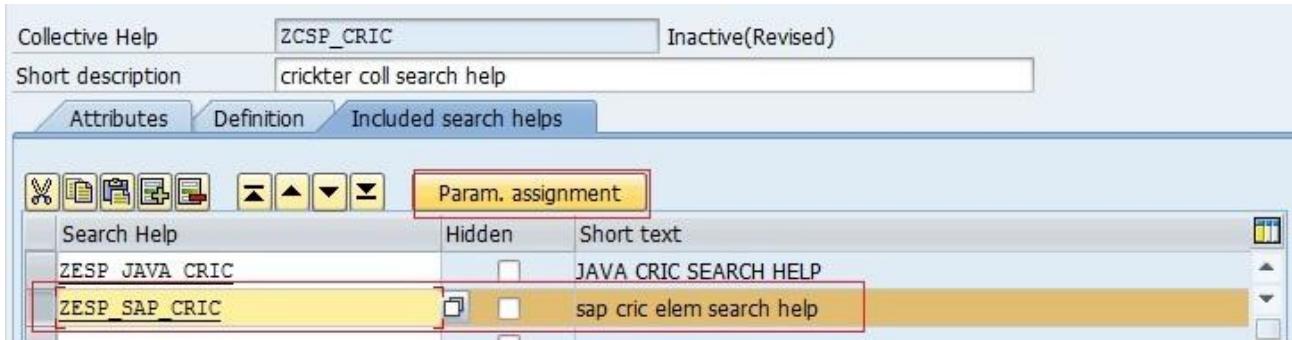
Included search helps:

Parameter Assignment for Search Help: ZESP_JAVA_CRIC

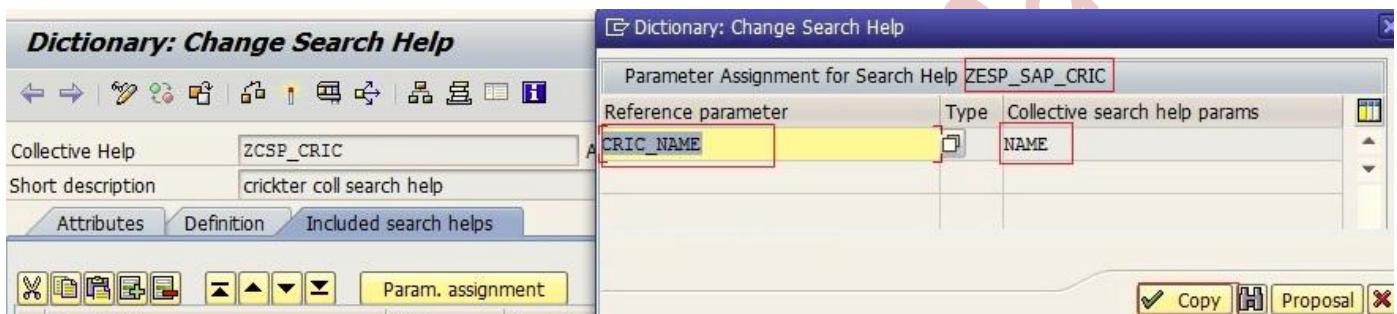
Reference parameter	Type
CRIC_NAME	NAME

Copy | H | Proposal | X

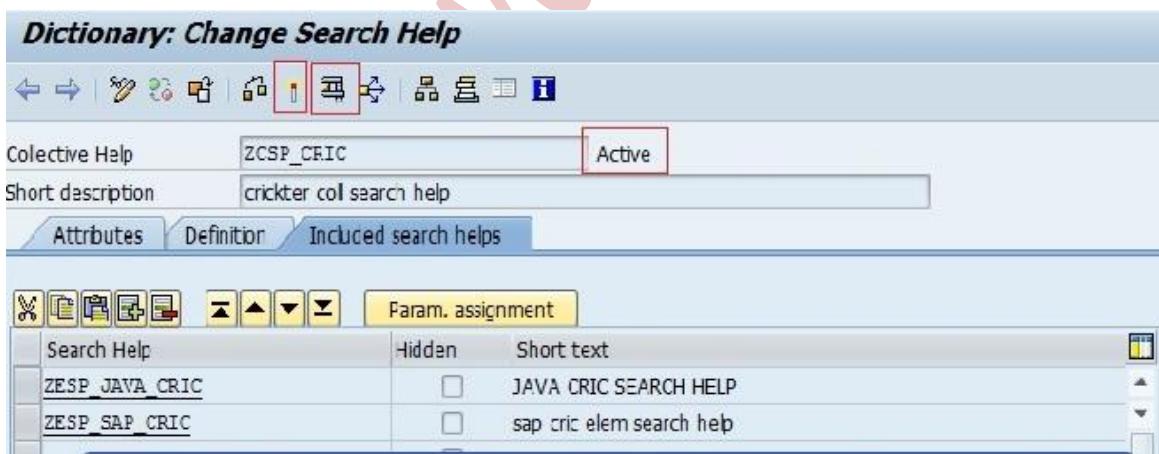
Step 22. Now select the second elementary search help and click on the Param. Assignment Button.



Step 24. Now system automatically maps the elementary search help reference parameter (CRIC_NAME) to the Collective Search help parameter (NAME) and at last click on Copy button.



Step 25. at last Activate the collective search help.



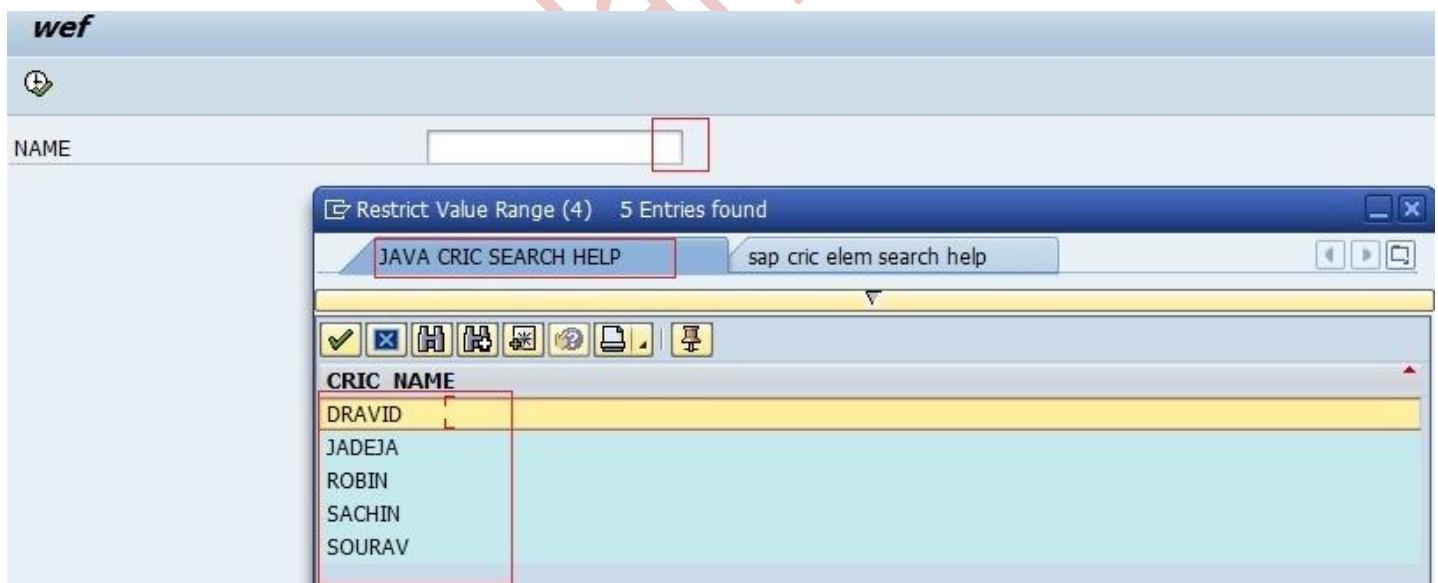
Step 26. Let's create a report program and declare a parameter and assign the collective search help to it by the keyword 'MATCHCODE OBJECT'. Activate the report and execute it.

ABAP Editor: Change Report ZTEST_CSH

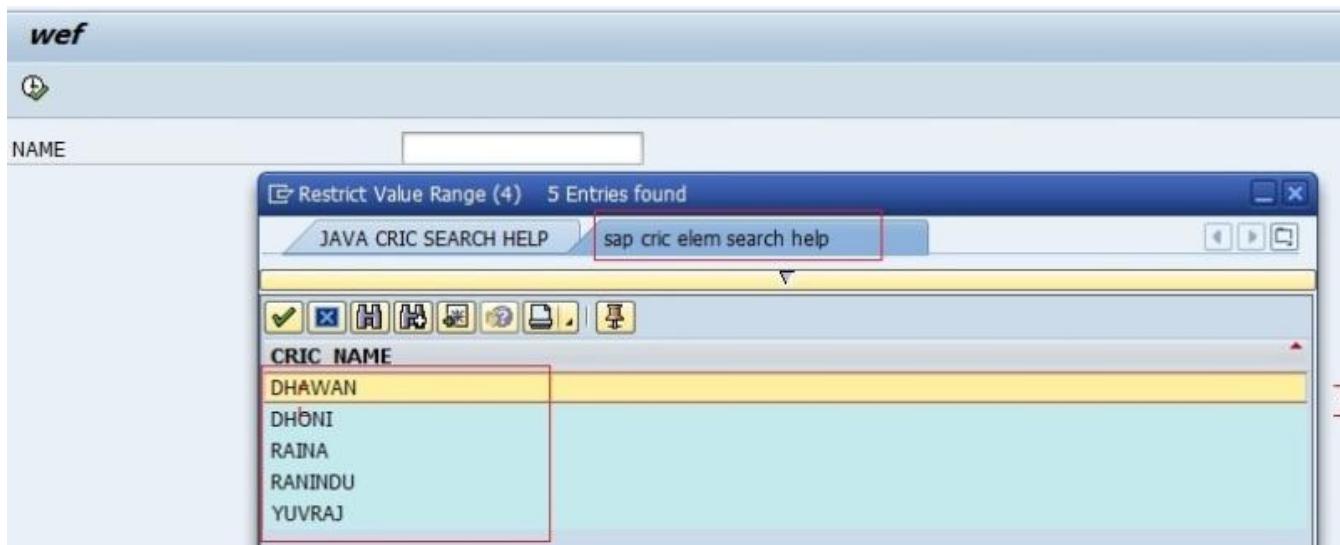


```
Report ZTEST_CSH Active
1  *->-
2  *& Report  ZTEST_CSH
3  *&
4  *&-----*
5  *&
6  *&
7  *&-----*
8
9  REPORT ZTEST_CSH.
10
11  PARAMETERS name TYPE ZCRIC_NAME MATCHCODE OBJECT ZCSP_CRIC.
```

Step 27. Now press **F4** button and the collective search help shows the first set of names from the first elementary search help.



Step 28. Again click on the second tab of the collective search help and it shows the second set of names from the second elementary search help.

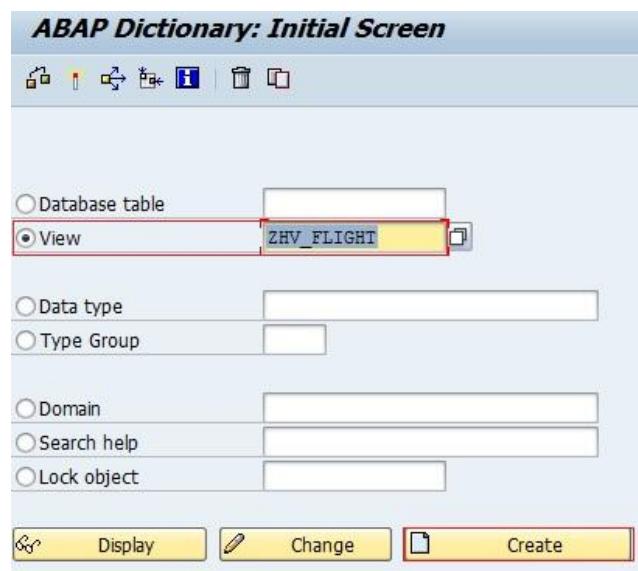


Collective Search help provides multiple ways of data selection and each data selection is created by the elementary search helps added in the Collective search help

USING HELP-VIEW IN SEARCH HELP

Creation of Help View and Using Help View in Creation of Search Help

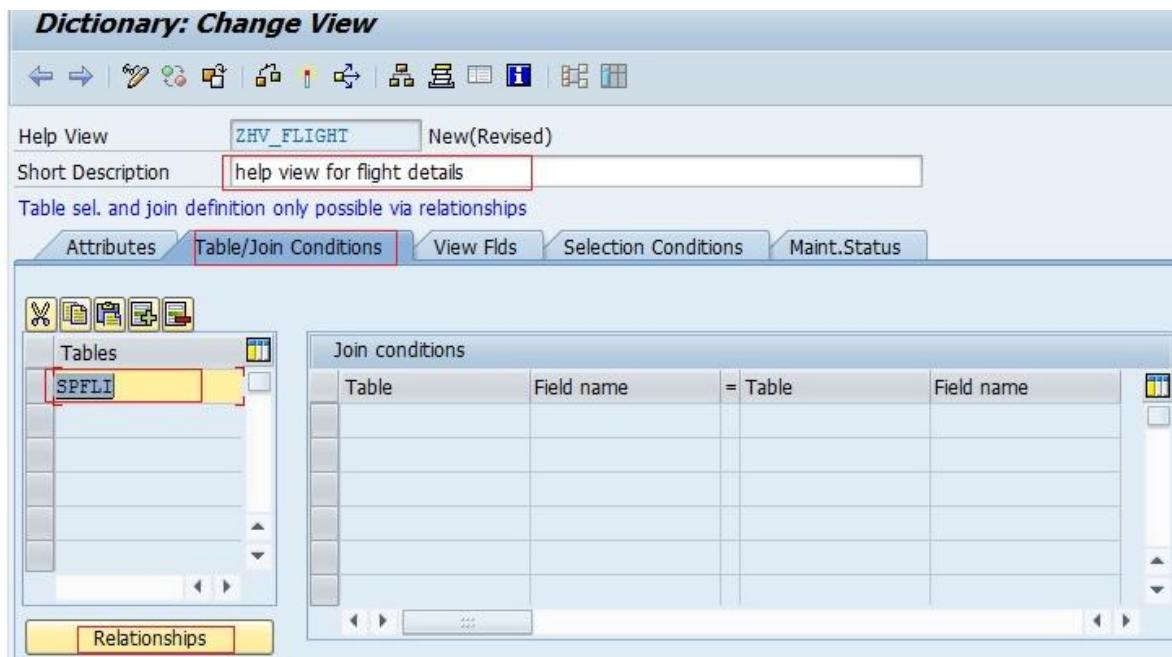
Step 1. Go to TCODE- SE11, Select the view radio button and provide the help view name and click on Create Button.



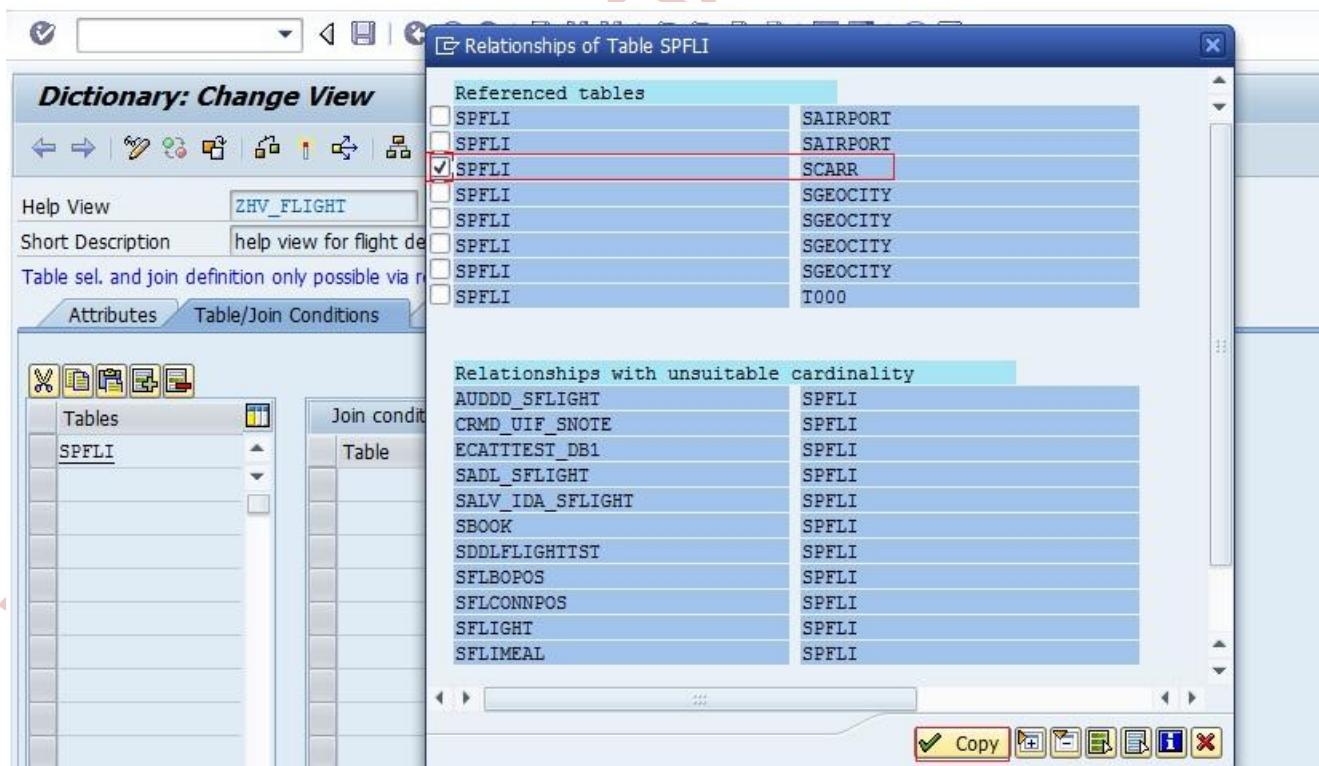
Step 2. Select the Help View radio button and click on Copy button.



Step 3. Provide the short text and in the Tables Section Provide the table Name 'SPFLI' and then click on the Relationships Button.



Step 4. Select the Right Combination 'SPFLI - SCARR' and then click on Copy Button.



Step 5. Based on the common fields The Join Conditions is filled.

Dictionary: Change View

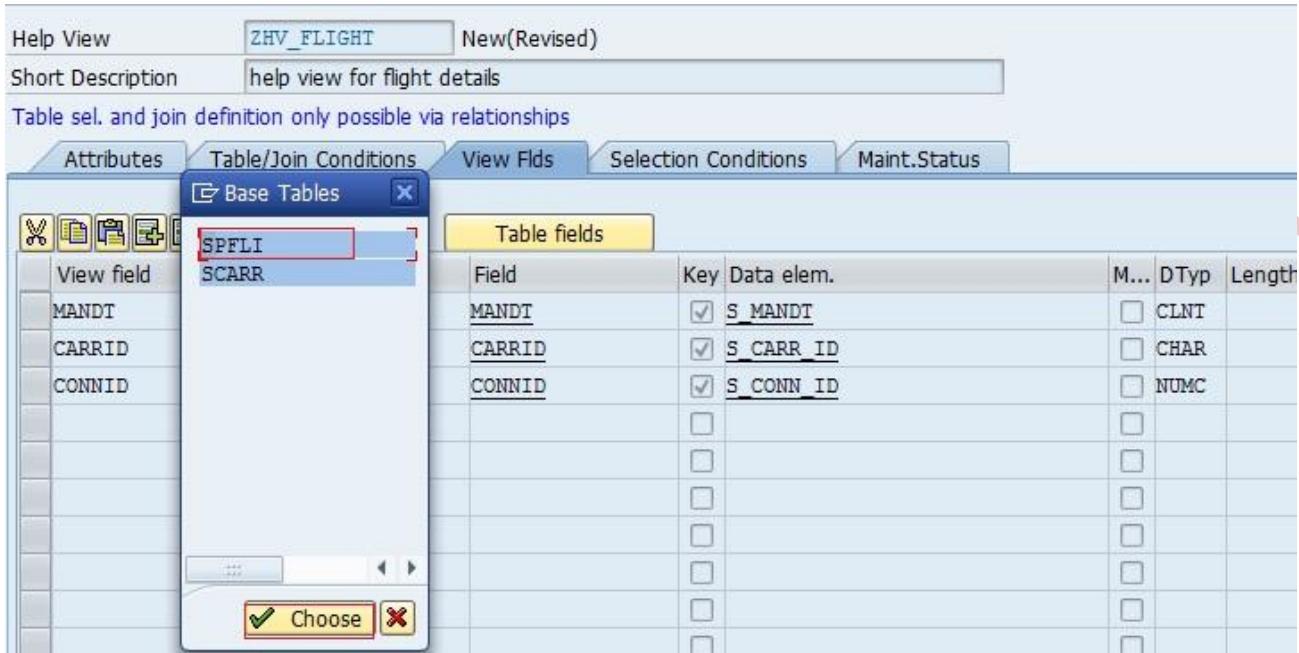
Table	Field name	= Table	Field name
SCARR	MANDT	= SPFLI	MANDT
SCARR	CARRID	= SPFLI	CARRID

Step 6. Now click on the View Flds Tab and then click on the Table Fields button.

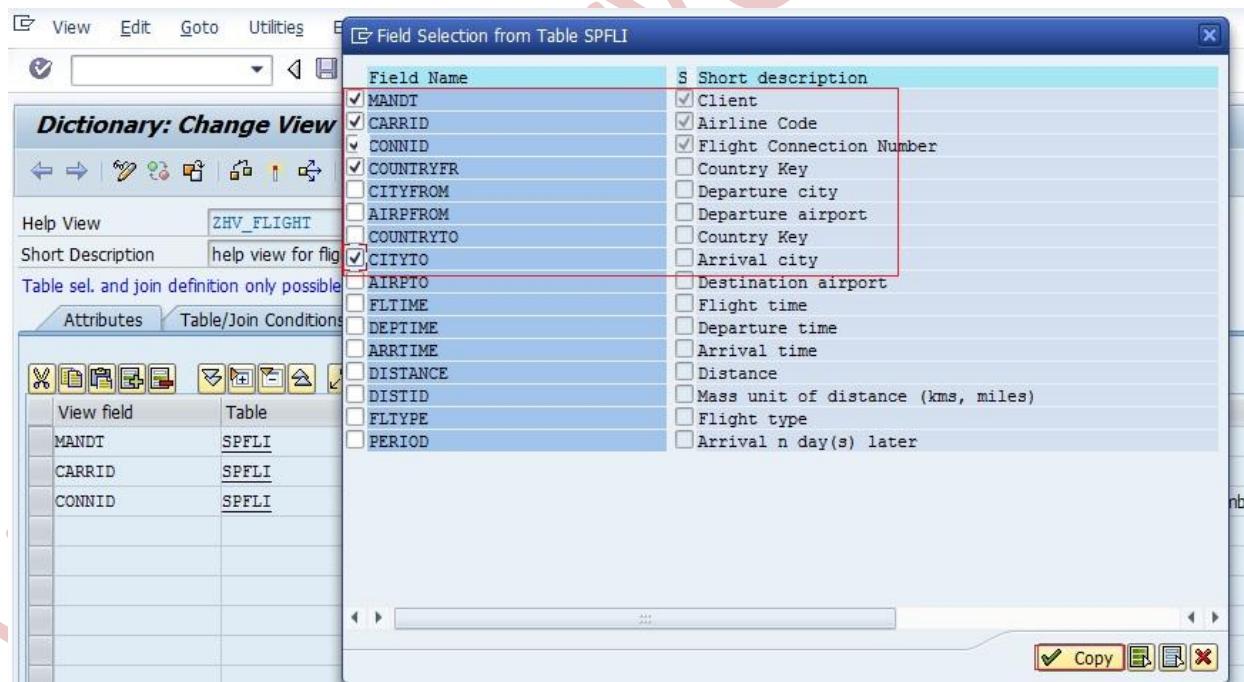
Dictionary: Change View

View field	Table	Field	Key	Data elem.	M...	DTyp	Length	Short desc
MANDT	SPFLI	MANDT	<input checked="" type="checkbox"/>	S_MANDT	<input type="checkbox"/>	CLNT	6	Client
CARRID	SPFLI	CARRID	<input checked="" type="checkbox"/>	S_CARR_ID	<input type="checkbox"/>	CHAR	6	Airline Code
CONNID	SPFLI	CONNID	<input checked="" type="checkbox"/>	S_CONN_ID	<input type="checkbox"/>	NUMC	8	Flight Conn

Step 7. Select the SPFLI table and click on the Choose Button.



Step 8. Select the Appropriate fields and then click on the Copy Button.



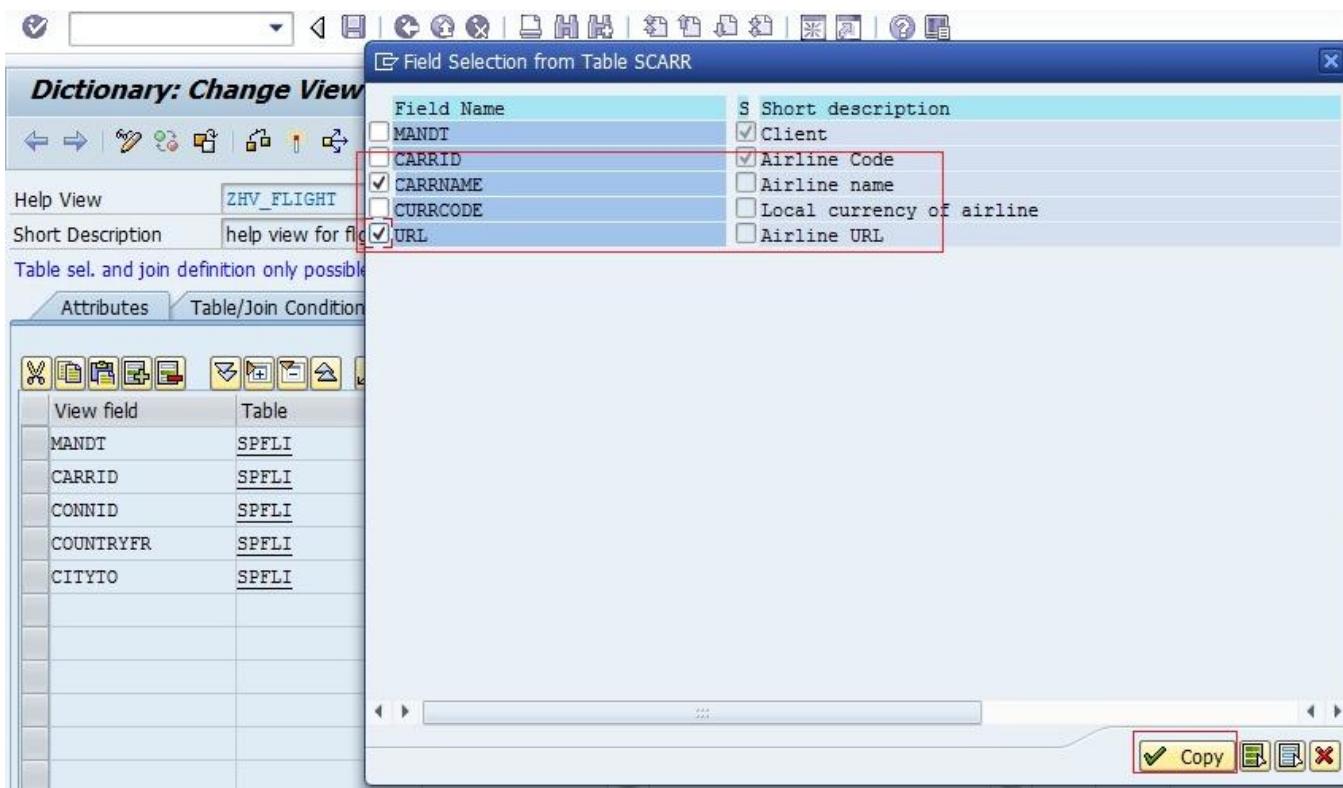
Step 9. All the selected fields from the SPFLI table are populated in the view. Again click on the Table Fields Button.

View field	Table	Field	Key	Data elem.	M...	DTyp	Length
MANDT	SPFLI	MANDT	<input checked="" type="checkbox"/>	S_MANDT	<input type="checkbox"/>	CLNT	6
CARRID	SPFLI	CARRID	<input checked="" type="checkbox"/>	S_CARR_ID	<input type="checkbox"/>	CHAR	6
CONNID	SPFLI	CONNID	<input checked="" type="checkbox"/>	S_CONN_ID	<input type="checkbox"/>	NUMC	8
COUNTRYFR	SPFLI	COUNTRYFR	<input type="checkbox"/>	LAND1	<input type="checkbox"/>	CHAR	3
CITYTO	SPFLI	CITYTO	<input type="checkbox"/>	S_TO_CITY	<input type="checkbox"/>	CHAR	20
					<input type="checkbox"/>		
					<input type="checkbox"/>		
					<input type="checkbox"/>		
					<input type="checkbox"/>		
					<input type="checkbox"/>		

Step 10. Again select the SCARR table and click on the Choose Button.

View field	Table	Field	Key	Data elem.	M...	DTyp	Length
MANDT	SPFLI	MANDT	<input checked="" type="checkbox"/>	S_MANDT	<input type="checkbox"/>	CLNT	
CARRID	SPFLI	CARRID	<input checked="" type="checkbox"/>	S_CARR_ID	<input type="checkbox"/>	CHAR	
CONNID	SPFLI	CONNID	<input checked="" type="checkbox"/>	S_CONN_ID	<input type="checkbox"/>	NUMC	
COUNTRYFR	SPFLI	COUNTRYFR	<input type="checkbox"/>	LAND1	<input type="checkbox"/>	CHAR	
CITYTO	SPFLI	CITYTO	<input type="checkbox"/>	S_TO_CITY	<input type="checkbox"/>	CHAR	
					<input type="checkbox"/>		
					<input type="checkbox"/>		
					<input type="checkbox"/>		
					<input type="checkbox"/>		
					<input type="checkbox"/>		

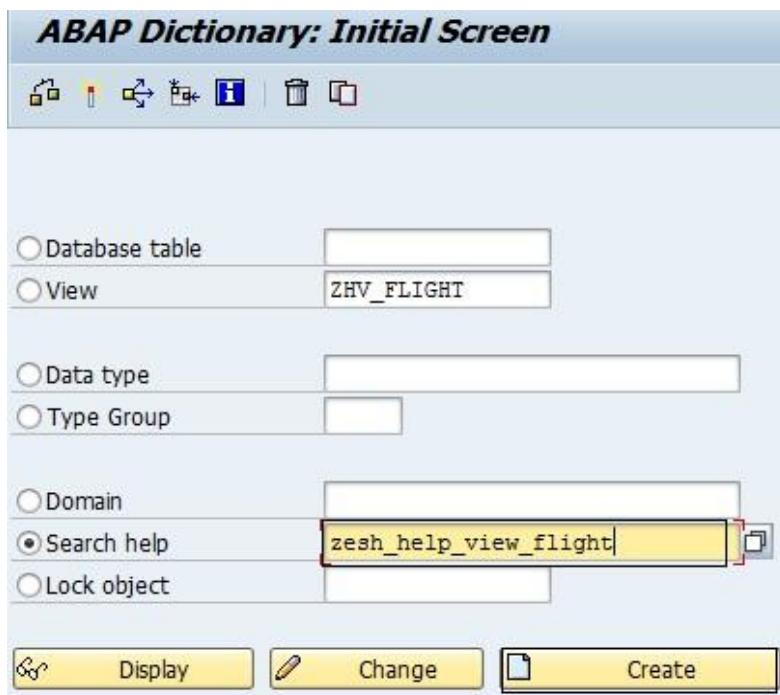
Step 11. Select the required fields for the Help view and click on the Copy Button.



Step 12. As per the fields selected from two tables, the Help View is populated with fields from two tables and at last Activate the Help View.

View field	Table	Field	Key	Data elem.	M...	DTyp	Length	Short description
MANDT	SPFLI	MANDT	<input checked="" type="checkbox"/>	S_MANDT	<input type="checkbox"/>	CLNT	3	Client
CARRID	SPFLI	CARRID	<input checked="" type="checkbox"/>	S_CARR_ID	<input type="checkbox"/>	CHAR	3	Airline Code
CONNID	SPFLI	CONNID	<input checked="" type="checkbox"/>	S_CONN_ID	<input type="checkbox"/>	NUMC	4	Flight Connection Number
COUNTRYFR	SPFLI	COUNTRYFR	<input type="checkbox"/>	LAND1	<input type="checkbox"/>	CHAR	3	Country Key
CITYTO	SPFLI	CITYTO	<input type="checkbox"/>	S_TO_CITY	<input type="checkbox"/>	CHAR	20	Arrival city
CARRNAME	SCARR	CARRNAME	<input type="checkbox"/>	S_CARRNAME	<input type="checkbox"/>	CHAR	20	Airline name
URL	SCARR	URL	<input type="checkbox"/>	S_CARRURL	<input type="checkbox"/>	CHAR	255	Airline URL

Step 13. Now lets Create a search help on the created Help view. Go to Tcode - SE11, select the Search help radio button, provide the name and click on Create button.



Step 14. Select Elementary Search help radio button and then click on the Tick Button as marked.

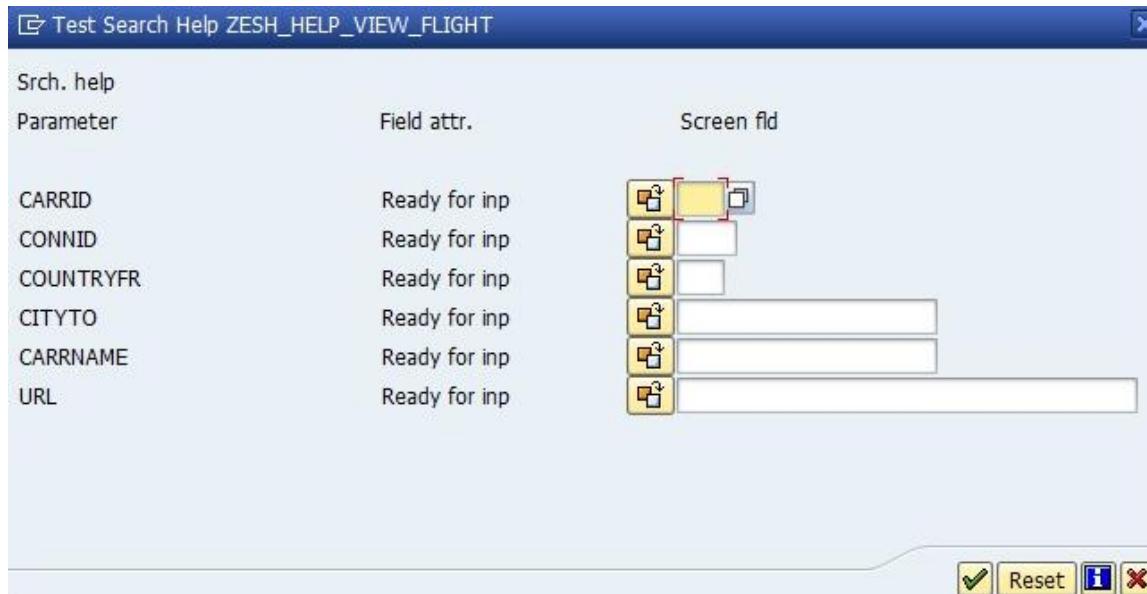


Step 15. Provide the short text and in the Selection Method provide the Help View name created above. In the Search Help parameter section, press F4 button and add all the fields, select the Imp and Exp section, Provide Lpos and Spos numbers and activate the Search Help. At last click on the Execute Button.

The screenshot shows the SAP Fiori Launchpad interface for creating a search help parameter. The top navigation bar includes icons for back, forward, search, and other system functions. Below the bar, the title 'Elementary Help' is displayed, followed by the help view name 'ZESH_HELP_VIEW_FLIGHT' and the status 'Active'. A yellow highlighted area contains the short description 'search help based on help view'. The interface is divided into two main sections: 'Attributes' and 'Definition'. Under 'Attributes', there are sections for 'Data Collection' and 'Enhanced Options'. In 'Data Collection', the 'Selection method' is set to 'ZHV_FLIGHT'. Under 'Enhanced Options', there are checkboxes for 'Proposal Search for Input Fields' and 'Cross-Column Full Text Search', both of which are unchecked. An accuracy value of '0,8' is specified for the error-tolerant full text search. The 'Definition' tab is selected, showing a table titled 'Parameter' with columns: 'Search help parameter', 'IMP', 'EXP', 'LPos', 'SPos', 'SDis', 'Data element', 'M...', and 'Default value'. The table lists several parameters with their corresponding values and data elements:

Search help parameter	IMP	EXP	LPos	SPos	SDis	Data element	M...	Default value
CARRID	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1	1	<input type="checkbox"/> S_CARR_ID	<input type="checkbox"/>	
CONNID	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	2	2	<input type="checkbox"/> S_CONN_ID	<input type="checkbox"/>	
COUNTRYFR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	3	3	<input type="checkbox"/> LAND1	<input type="checkbox"/>	
CITYTO	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	4	4	<input type="checkbox"/> S_TO_CITY	<input type="checkbox"/>	
CARRNAME	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	5	5	<input type="checkbox"/> S_CARRNAME	<input type="checkbox"/>	
URL	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	6	6	<input type="checkbox"/> S_CARRURL	<input type="checkbox"/>	

Step 16. Press F4 button against CARRID field.



Step 17. Now it shows records consists of fields from two different tables.

The screenshot shows the SAP GUI interface with the title bar "Test Search Help ZESH_HELP_VIEW_FLIGHT". Below the title bar is a toolbar with icons for search, help, and other functions. The main area is divided into two sections: a configuration table at the top and a list view at the bottom. The configuration table has three columns: "Parameter", "Field attr.", and "Screen fld". It shows fields CARRID and CONNID with "Ready for inp" status and corresponding graphical representations. Below the configuration table is a header "Hit List 46 Entries". The list view contains a table with columns: ID, No., Ctr, Arrival city, Airline, and Airline URL. The table rows represent flight records. A red box highlights the first few rows of the list. The table has a blue header row and white body rows.

ID	No.	Ctr	Arrival city	Airline	Airline URL
AA	0017	US	SAN FRANCISCO	American Airlines	http://www.aa.com
AA	0026	DE	NEW YORK	American Airlines	http://www.aa.com
AA	0064	US	NEW YORK	American Airlines	http://www.aa.com
AZ	0555	IT	FRANKFURT	Alitalia	http://www.alitalia.it
AZ	0788	IT	TOKYO	Alitalia	http://www.alitalia.it
AZ	0789	JP	ROME	Alitalia	http://www.alitalia.it
AZ	0790	IT	OSAKA	Alitalia	http://www.alitalia.it
AZ	0791	JP	ROME	Alitalia	http://www.alitalia.it
DL	0106	US	FRANKFURT	Delta Airlines	http://www.delta-air.com
DL	1699	US	SAN FRANCISCO	Delta Airlines	http://www.delta-air.com
DL	1984	US	NEW YORK	Delta Airlines	http://www.delta-air.com
JL	0407	JP	FRANKFURT	Japan Airlines	http://www.jal.co.jp
JL	0408	DE	TOKYO	Japan Airlines	http://www.jal.co.jp
LH	0400	DE	NEW YORK	Lufthansa	http://www.lufthansa.com

EXAMPLE 2:

vizu.chowdary@gmail.com

PROPERTIES OF SEARCH HELP

SELECTION-METHOD:

- ☞ IT SPECIFIES THE TABLE NAME OR THE VIEW NAME FROM WHICH THE DATA HAS TO BE SELECTED.
- ☞ IF ALL THE FIELDS ARE FROM SINGLE TABLE THEN GIVE THE TABLE NAME.
- ☞ IF THE FIELDS ARE FROM MULTIPLE TABLES THEN GIVE THE VIEW NAME.

IMPORTING:

IF IMPORTING IS SELECTED THE LIST SCREEN WILL IMPORT THE VALUE FROM THE SELECTION SCREEN AND DISPLAY THE LIST OF THE VALUES BASED ON IMPORTING PARAMETERS.

EXPORTING:

IF EXPORTING IS SELECTED THE LIST SCREEN WILL EXPORT THE SELECTED VALUE BACK TO THE SELECTION SCREEN.

SELECTION - SCREEN

CNO:	<input type="text"/>
LAND	<input type="text" value="US"/>
NAME1:	<input type="text"/>

THIS VALUE IS IMPORTED
FROM SELECTION SCREEN
INTO THE LIST SCREEN

CNO	LAND	NAME1
1001	US	
1002	US	
1003	US	
1004	IN	VEJAY

THIS VALUE IS EXPORTED
FROM LIST SCREEN BACK TO
SELECTION SCREEN

SPOS:

IT SPECIFIES THE POSITION OF THE FIELD ON THE SELECTION-SCREEN.

LPOS:

IT SPECIFIED THE POSTION OF THE COLUMN ON THE LIST SCREEN.

DISPLAY VALUES IMMEDIATELY

THE VALUES WILL BE IMMEDIATELY DISPLAYED ON LIST SCREEN WITH OUT ANY DIALOGUE TYPE OR SCREEN.

DISPLAY WITH DIALOGUE TYPE:

THE VALUES WILL NOT BE DISPLAYED DIRECTLY INSTEAD A DIALOGUE SCREEN DISPLAYED FIRST. BASED ON THE VALUE ENTER ON THE DIALOGUE TYPE, THE VALUES WILL BE DISPLAYED ON LIST SCREEN.

DIALOGUE DEPENDS ON SET OF VALUES:

- 1) IF THE VALUES ARE < 500 ON THE LIST SCREEN THEN THE LIST SCREEN WILL BE DISPLAYED IMMEDIATELY.
- 2) IF THE VALUES ARE MORE THAN 500 THEN THE DIALOGUE SCREEN WILL BE DISPLAYED FIRST AND THEN THE LIST SCREEN IS DISPLAYED.

SEARCH HELP EXIT:

IT IS AN EXIT OR A FUNCTION MODULE WHICH CONTAINS THE CUSTOM ABAP CODE FOR DISPLAYING VLAUES ON THE LIST SCREEN.

EG: FFIF_SHLP_EXIT_EXAMPLE (SE37)

TYPE GROUP:

IT IS AN OBJECT WHICH CONTAINS THE REUABLE DATA DECLARATIONS. SO THAT THEY CAN BE USED BY MULTIPLE PROGRAMS.

EG:

SLIS IS A TYPE GROUP WHICH CONTAINS THE REUSABLE DATA DECLARATIONS FOR DEVELOPING ALV REPORTS.

WE DONT CREATE ANY TYPE GROUP INTHE REAL TIME.

TYPE-POOLS:

IT IS A KEYWORD WHICH IS USED TO ASSIGN THE TYPE GROUP INTO OUR PROGRAM. SO THAT OUR PROGRAM CAN REUSE THE DATA DECLARATIONS.

LOCK OBJECTS

LOCK OBJECTS:

- ❖ IT IS AN OBJECT WHICH DISPLAYS A LOCK ON THE TABLE DATA. SO THAT NO OTHER USER ACCESS SAME SET OF DATA.

DEFINITION:

SYNCHRONIZES SIMULTANEOUS ACCESS OF THE SAME SET OF DATA RECORDS BY MULTIPLE USERS.

TYPES OF LOCKS

- ❖ THREE TYPES OF LOCKS ARE AVAILABLE
 - 1) EXCLUSIVE LOCK
 - 2) SHARED LOCK
 - 3) EXCLUSIVE BUT NOT CUMMULATIVE

EXCLUSIVE LOCK:

- A. IN THIS TYPE OF LOCK A SINGLE USER CAN READ THE DATA AND WRITE THE DATA.
- B. NO OTHER USER CAN USE THE SAME SET OF DATA.
- C. ALL THE OTHER USERS REQUEST FOR SAME SET OF DATA WILL REJECTED BY THE SYSTEM.

SHARED LOCK:

- A. IN THIS TYPE OF LOCK EVERY USER CAN READ THE DATA
- B. AS SOON AS ONE USER STARTS EDITING THE DATA, NO OTHER USER CAN READ OR WRITE THE DATA. ALL REQUESTS ARE REJECTED BY THE SYSTEM.

EXCLUSIVE BUT NOT CUMMULATIVE:

- A. IN THIS TYPE OF LOCK A SINGLE USER CAN LOCK THE TABLE DATA FOR INSERTION OR UPDATION.
- B. SUPPOSE IF THE SAME USER TRIES TO USE THE SAME TABLE DATA SECOND TIME, THE REQUEST WILL BE REJECTED.

LOCK OBJECT FUNCITON MODULES:

- ❖ WHENEVER WE CREATE LOCK OBJECTS TOW FUNCTION MODULES WILL BE CREATED AUTOMATICALLY.
 - 1) ENQUEUE_<LOCK_OBJECT_NAME>-> TO DEFINE A LOCK.
 - 2) DEQUEUE_<LOCK_OBJECT_NAME>-> TO RELEASE A LOCK.

WHERE DO WE USE LOCK OBJECTS

USED WHEN EVER WE WORK WITH OPEN SQL STATEMENTS I.E. INSERT, UPDATE, MODIFY, DELETE.

IMPORTANT NOTE:

THE NAMING STANDARD FOR LOCK OBJEC IS:
IT SHOULD START WITH EZ OR EY

NOTE:

EXCLUSIVE -> ONE PERSON

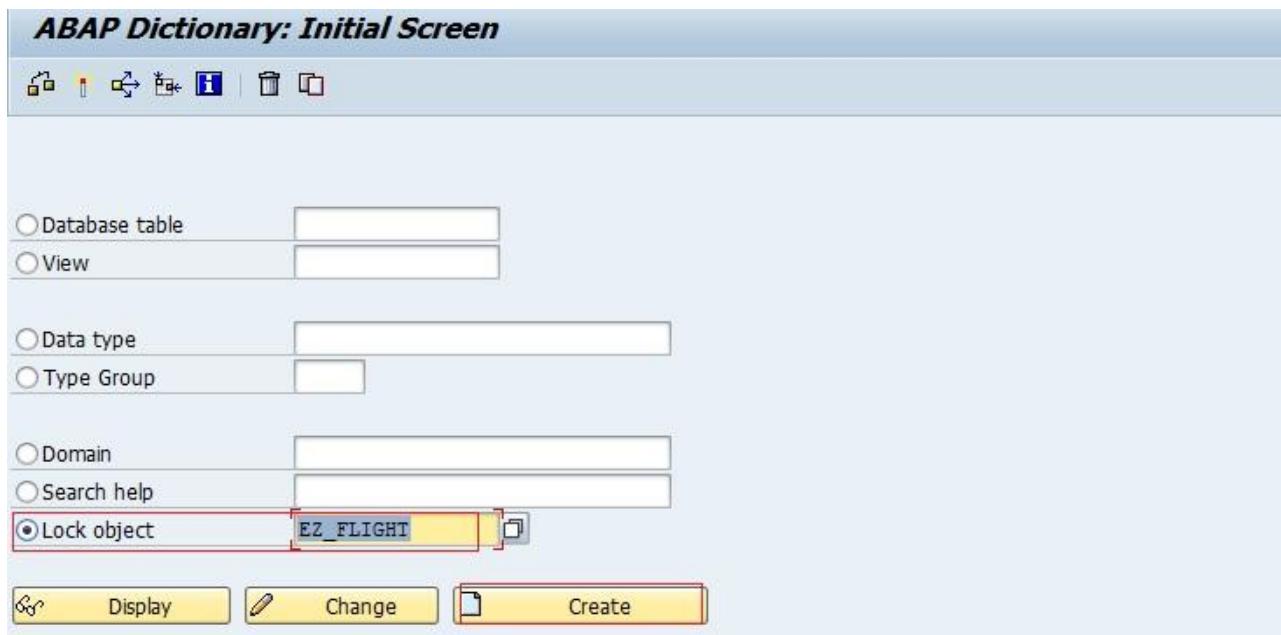
SHARED -> EVERY PERSON CAN READ

EXCLUSE BUT NOT CUMMULATIVE -> ACCESS ONLY ONE TIME FOR ONE PERSON, REQUEST AGAIN, IT WILL REJECT.

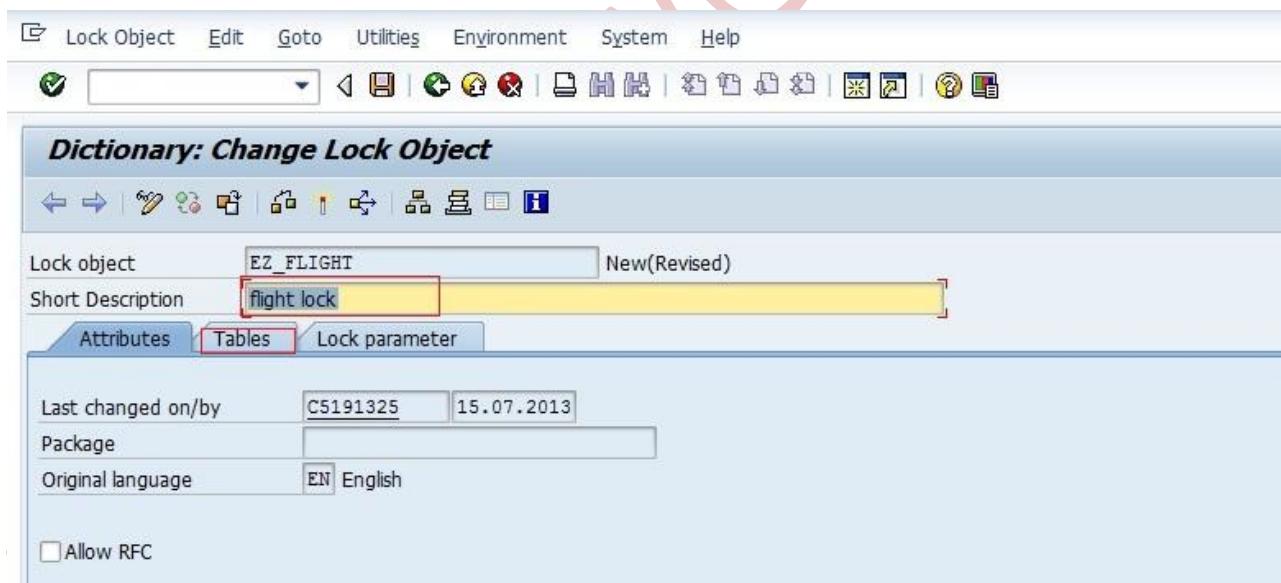
EXAMPLES ON LOCK OBJECT

Creating Lock Object and Using Lock in Program

Step 1. Go to TCODE-SE11, Provide the lock object name and Click on Create Button.



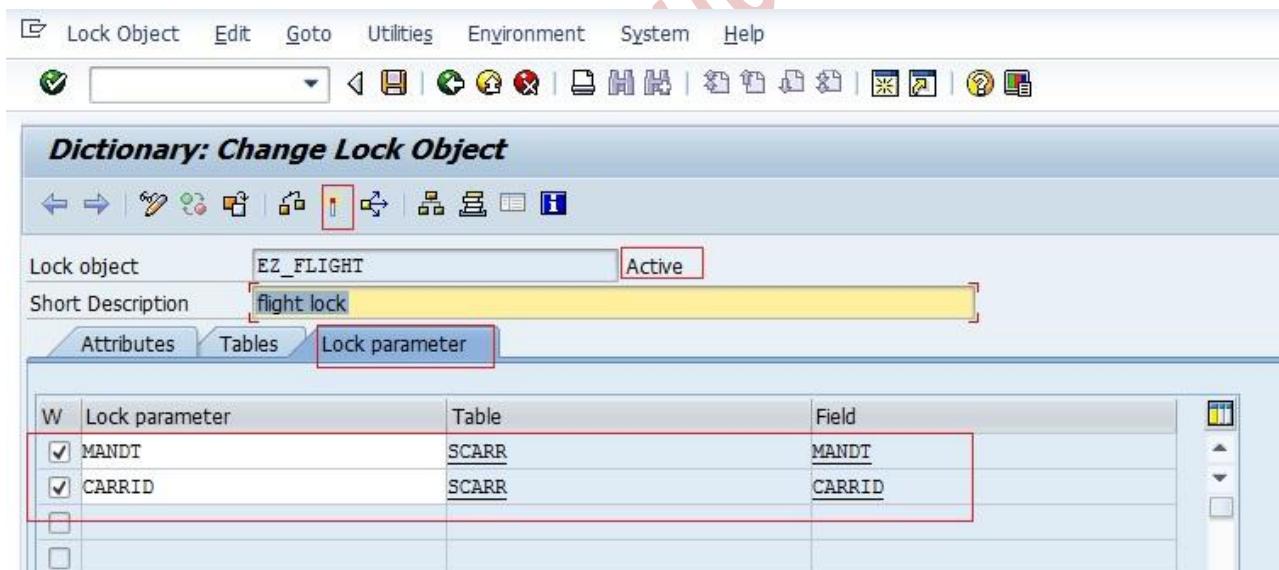
Step 2. Provide the Short text and click on the Tables Tab.



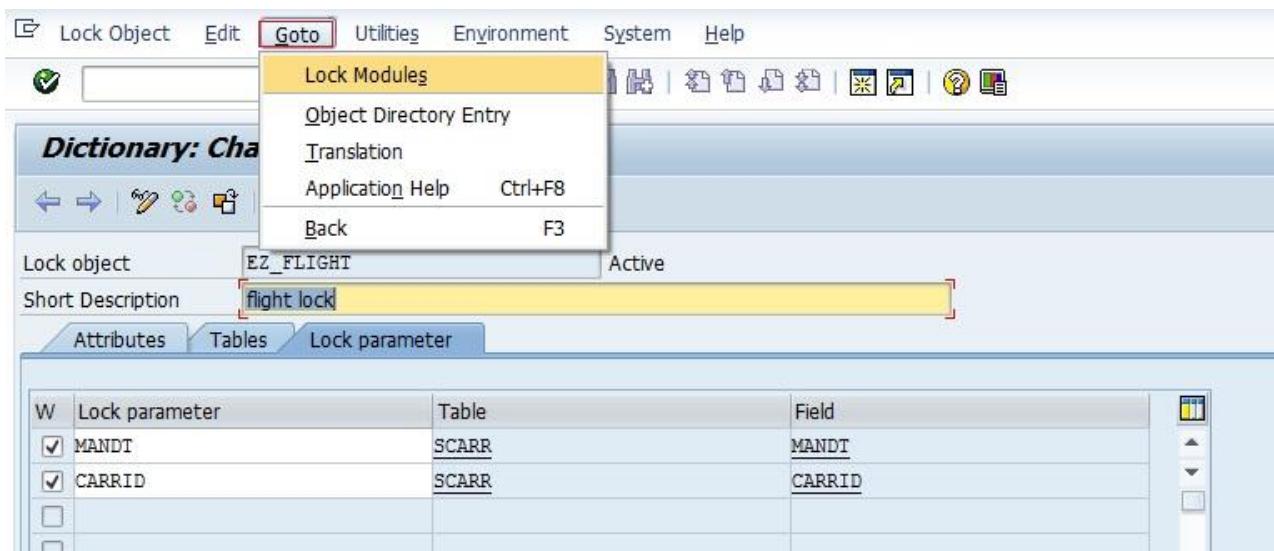
Step 3. Provide the table name and Select the Lock mode as Write Lock and Click on the Lock Parameter Tab.



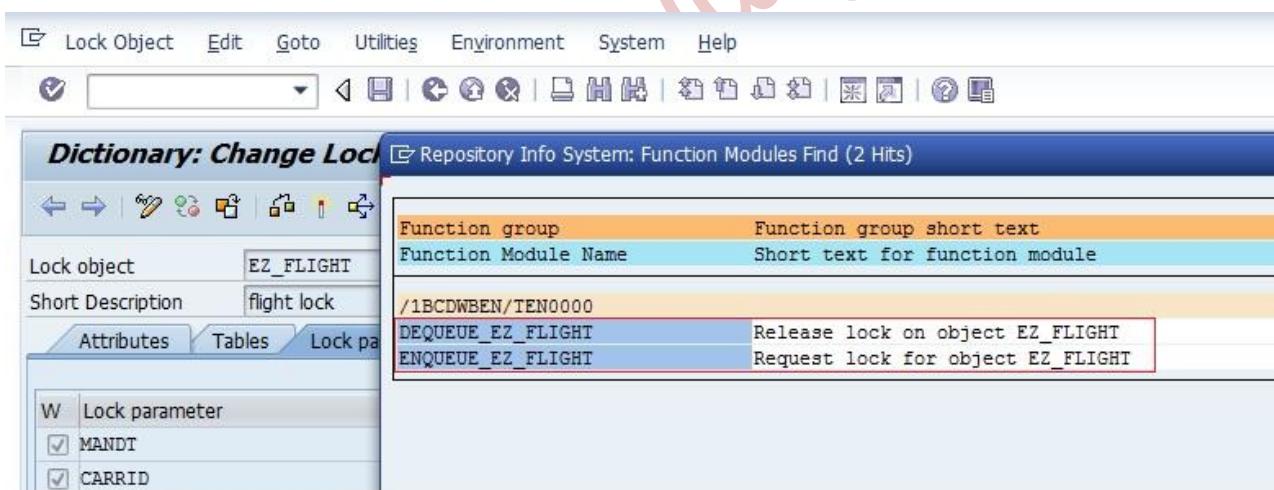
Step 4. All the primary key fields of the given table are added under the Lock parameter section. Activate the Lock Object.



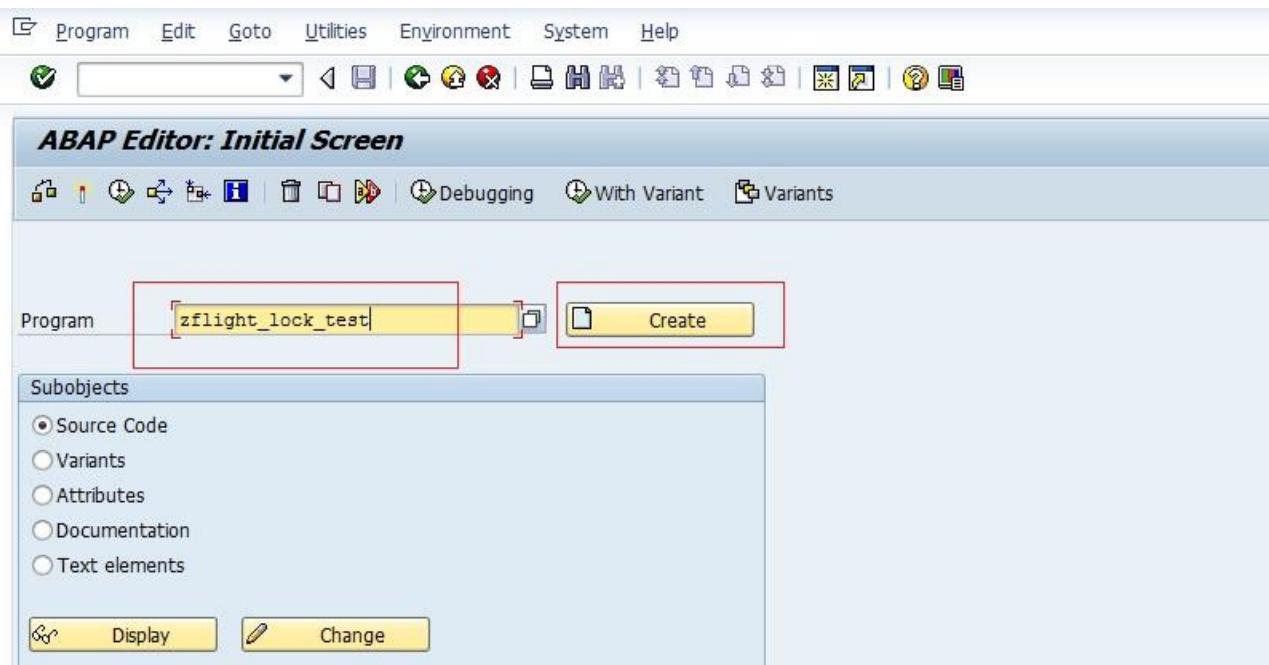
Step 5. When the lock object is activated it creates two function modules 'Enqueue & Dequeue'. Navigate along the below shown path to get the generated function modules.



Step 6.



Step 7. Let's use the generated function module in a program.
Go to TCODE- SE38, provide a report program name and click on create button.

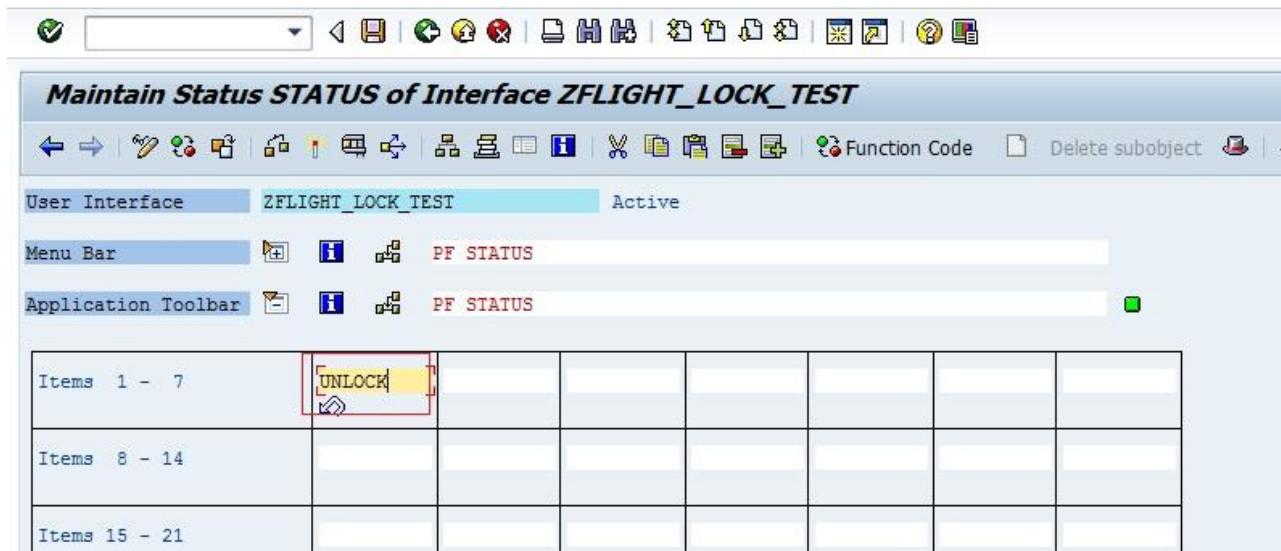


Step 8. Call the Enqueue Function Module to lock the Single record.

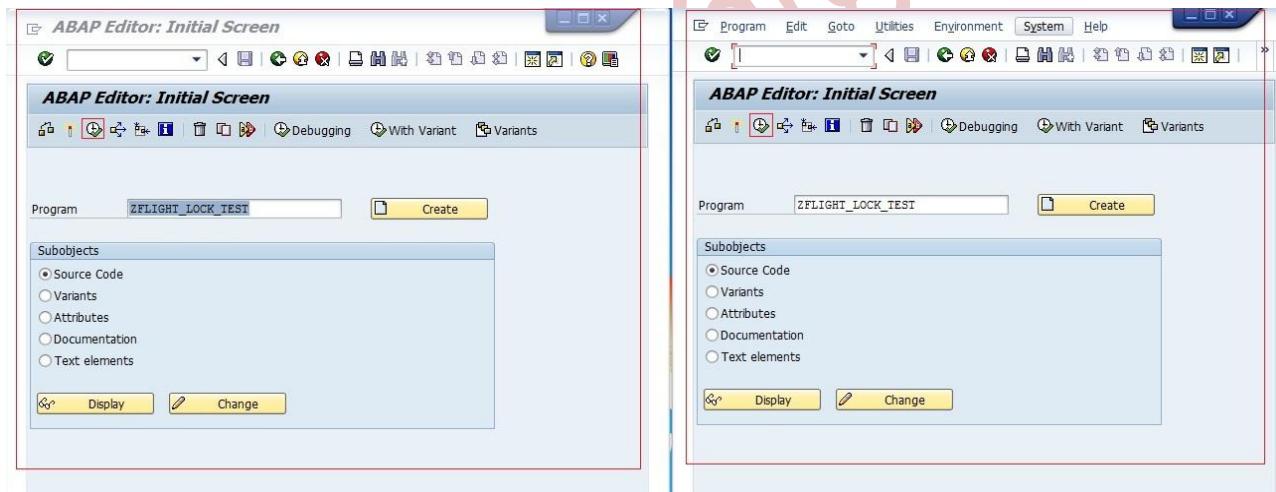
Report ZFLIGHT_LOCK_TEST Active

```
6      *->
7      *->-----*
8
9 REPORT zflight_lock_test.
10 DATA : ls TYPE scarr.
11 PARAMETERS : p_carr TYPE scarr-carrid.
12 START-OF-SELECTION.
13 SET PF-STATUS 'STATUS'.
14 SELECT SINGLE * FROM scarr INTO ls WHERE carrid = p_carr.
15 IF sy-subrc = 0.
16   CALL FUNCTION 'ENQUEUE_EZ_FLIGHT'
17     EXPORTING
18       mode_scarr      = 'E'
19       mandt           = sy-mandt
20       carrid          = p_carr
21     EXCEPTIONS
22       foreign_lock    = 1
23       system_failure  = 2
24       OTHERS           = 3.
25 IF sy-subrc = 1.
26   MESSAGE 'Other User/ Program is currently editing the record' TYPE 'E'.
27 ENDIF.
28 WRITE :/ ls-carrid, ls-carrname, ls-currcode, ls-url.
29 ENDIF.
30
31 AT USER-COMMAND.
32 CASE sy-ucomm.
33 WHEN 'UNLOCK'.
34   CALL FUNCTION 'DEQUEUE_EZ_FLIGHT'
35     EXPORTING
36       mode_scarr      = 'E'
37       mandt           = sy-mandt
38       carrid          = p_carr.
39 ENDCASE.
```

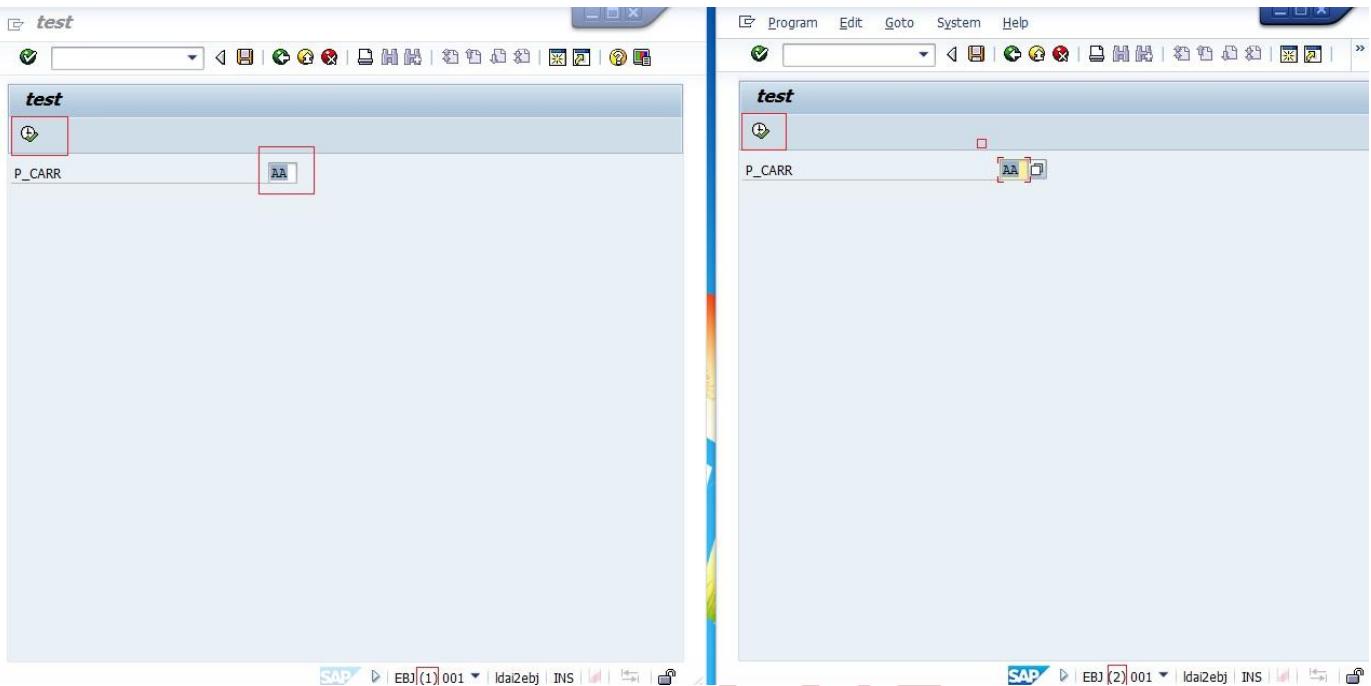
Step 9. In the program PF Status Create the Application tool bar button as shown below.



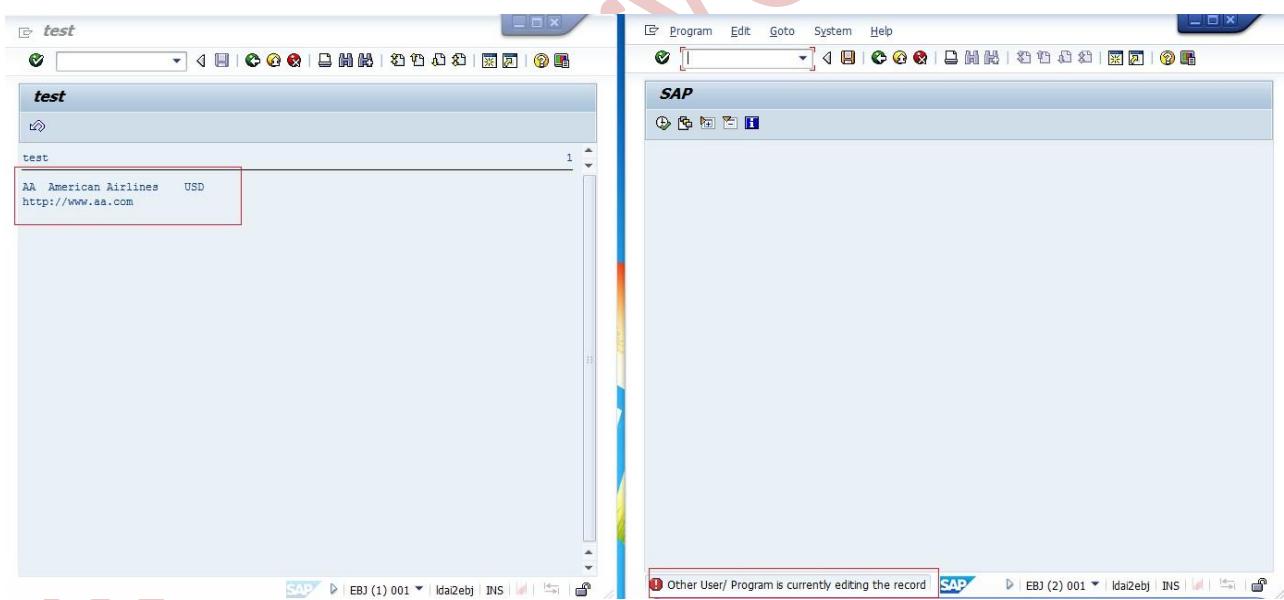
Step 10. Activate the report. Open two different session at a time and run the same report.



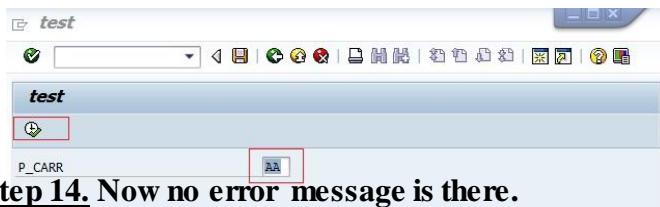
Step 11. Provide the CARRID as 'AA' in both sessions.



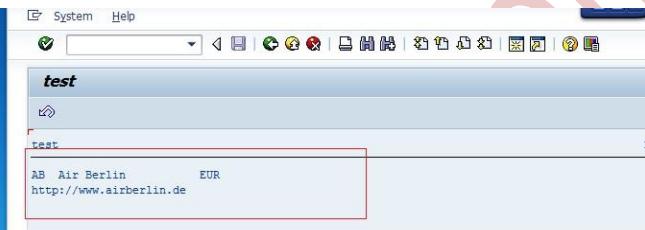
Step 12. Run the first session and we have the out put. Execute the same report in second session, now we have the error message given below as the same record is locked in first session.



Step 13. Now again run the same report in two different sessions and provide different inputs and execute it.



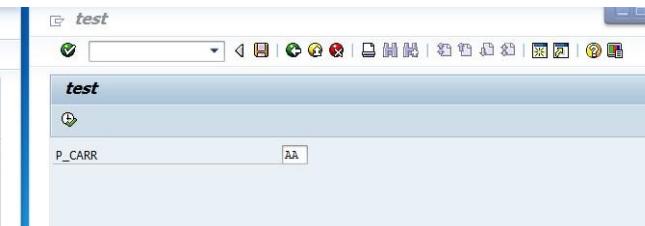
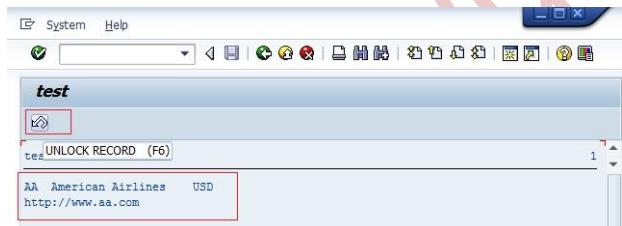
Step 14. Now no error message is there.



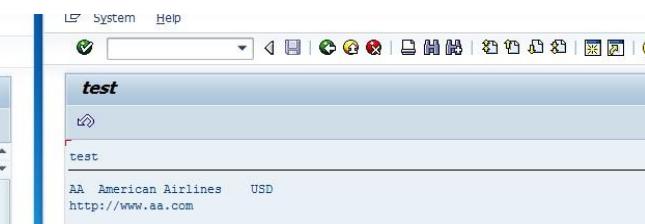
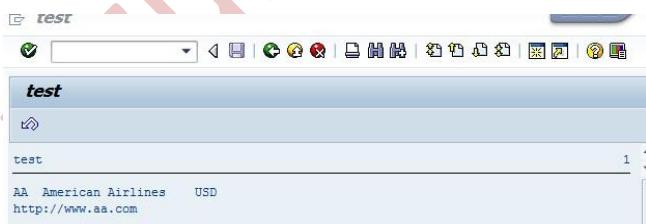
Step 15. Now again run the same report in two different session and provide the same input. Run the program in first session.



Step 16. Now the record is locked. Now Unlock the Record By clicking on the Application toolbar button that is created in the program PF Status. Now run the Report in the Second session.



Step 17. The same record is shown in the second session without error.



INTERVIEW QUESTIONS

- 1) WHAT IS ABAP (OR) DATA DICTIONARY? & TRANSACTION CODE? WHAT IS DB TABLE? IN HOW MANY WAYS WE CAN CREATE A TABLE?
- 2) HOW DO YOU DEFINE A TABLE? WHAT IS DOMAIN AND DATA ELEMENT?
- 3) CAN YOU CREATE A TABLE WITH OUT A KEY FIELD? (NO)
- 4) HOW MANY KEY FIELDS CAN WE CREATE IN A TABLE? (16)
- 5) WHAT IS A KEY FIELD?
- 6) WHAT IS DELIVERY CLASS? OPTIONS USED?
- 7) EXPLAIN DISPLAY & MAINTENANCE (DATA BROWSER / TABLE VIEW MAINTENANCE)?
- 8) WHAT IS DATA CLASS? OPTIONS USED?
- 9) WHAT IS SIZE CATEGORY? OPTIONS USED?
- 10) DO YOU CREATE A NEW DATA ELEMENT & DOMAIN (OR) REUSE THE EXISTING ONE?
- 11) WHAT IS BUFFER & EXPLAIN FUNCTIONALITY? OPTIONS & TYPES?
- 12) WHAT ARE LOG DATA CHANGE, CLIENT NUMBER AND MANDT?
- 13) HOW MANY TYPES OF TABLES ARE THERE BASED ON CLIENTS?
- 14) DIFFERENCE BETWEEN CLIENT DEPENDENT AND INDEPENDENT TABLE?
- 15) DO YOU CREATE ANY CLIENT INDEPENDENT TABLE? (NO)
- 16) WHAT IS CLIENT NUMBER THAT YOU ARE USING IN PREVIOUS / PRESENT PROJECT?
- 17) WHAT IS REFERENCE TABLE NAME & REFERENCE FIELD NAME IN CURRENCY AND QUANTITY FIELDS?
EXPLAIN CURRENCY AND QUANTITY FIELDS?
- 18) WHAT IS STRUCTURE? DIFFERENCE BETWEEN TABLE AND STRUCTURE?
- 19) HOW DO WE USE STRUCTURES IN TABLE?
- 20) DIFFERENCE BETWEEN .INCLUDE STRUCTURE AND APPEND STRUCTURE?
- 21) TELL ME SOME TABLE NAMES, USED IN YOUR PROJECT?
- 22) WHAT IS FOREIGN KEY RELATIONSHIP? CHECK TABLE? FOREIGN KEY TABLE? FOREIGN KEY FIELD? WHY DO WE CREATE FOREIGN KEY RELATIONSHIP?
- 23) WHAT IS VALUE TABLE? DIFFERENCE BETWEEN CHECK TABLE AND VALUE TABLE?
- 24) I WANT TO SPECIFY FIXED VALUES FOR DOMAIN, IS IT POSSIBLE?
- 25) CAN I STORE A NEGATIVE VALUE FOR A NUMBER?
- 26) I WANT TO STORE THE NAME OF A CUSTOMER IN SMALL AND CAPITAL LETTERS. IS IT POSSIBLE, HOW?
- 27) WHAT IS CONVERSION ROUTINE?
- 28) WHAT IS AN AUTHORIZATION GROUP AND FUNCTION GROUP?
- 29) WHAT IS TMG? TRANSACTION CODE? HOW MANY EVENTS ARE THERE? (39)
- 30) DIFFERENT TYPES OF DB TABLES WITH EXAMPLES?
- 31) WHAT IS A VIEW? DIFFERENCE BETWEEN VIEW AND TABLE?
- 32) WHAT IS DATABASE VIEW? CAN I MAINTAIN DATA IN DATABASE VIEW?
- 33) CAN I CREATE A VIEW ON SINGLE TABLE? IF YES HOW?
- 34) CAN I MAINTAIN DATA IN A VIEW? IF YES HOW?
- 35) WHAT IS A HELP VIEW? WHERE DO WE USE IT?
- 36) WHAT IS A HELP VIEW? WHERE DO WE USE IT?
- 37) WHAT IS MAINTENANCE VIEW? DID YOU WORK ON IT?
- 38) DIFFERENCE BETWEEN INNER JOIN & OUTER JOINS?
- 39) WHAT IS A SEARCH HELP AND IT'S TYPES?
- 40) EXPLAIN ABOUT COLLECTIVE SEARCH HELP?

- 41) WHERE DO WE USE SEARCH HELP?
- 42) HOW DO WE USE HELP VIEW IN SEARCH HELP?
- 43) EXPLAIN ABOUT IMPORTING, EXPORTING, SPOS, LOPS?

->SELECTION METHOD

- DISPLAY VALUES IMMEDIATELY
- DISPLAY WITH DIALOGUE TYPE
- DIALOGUE DEPENDS ON SET OF VALUES

- 44) WHAT IS SEARCH HELP EXIT? MATCH CODE OBJECT?
- 45) WHAT IS A TYPE GROUP WITH EXAMPLE? WHAT ARE TYPE-POOLS?
- 46) LOCK OBJECT? HOW MANY TYPES? EXPLAIN?
- 47) LOCK OBJECTS FUNCTION MODULES? WHERE DO WE USE LOCK OBJECTS?
- 48) WHAT IS HEADER DATA AND ITEM DATA?
- 49) WHAT IS MASTER DATA AND TRANSACTION DATA?
- 50) HOW DO WE CREATE A CHECK TABLE?
- 51) HOW TO CREATE A CURRENCY FIELD?
- 52) WHAT IS AN INDEX? HOW MANY TYPES? WHERE DO WE USE INDEX?
- 53) ADVANTAGES OF DATA ELEMENT & DOMAIN?
- 54) WHAT IS INITIAL FIELD IN DATABASE TABLE?