



1. What is a transaction?

- A transaction is dialog program that change data objects in a consistant way.

2. What are the requirements a dialog program must fulfill?

A dialog program must fulfil the following requirements

- A user friendly user interface.
- Format and consistancy checks for the data entered by the user.
- Easy correction of input errors.
- Access to data by storing it in the data bases.

3. What are the basic components of dialog program?

- Screens (Dynpros)
- Each dialog in an SAP system is controlled by dynpros.A dynpros consists of a screen

And its flow logic and controls exactly one dialog step.

- ABAP/4 module Pool.

Each dynpro refers to exactly one ABAP/4 dialog program .Such a dialog program is also called a module pool ,since it consists of interactive modules.

4.What is PBO and PAI events?

PBO- Process Before Output-It determines the flow logic before displaying the screen.

PAI-Process After Input-It determines the flowlogic after the display of the screen and after receiving inputs from the User.

5. What is dynpro?What are its components ?

- A dynpro (Dynamic Program) consists of a screen and its flow logic and controls exactly one dialog steps.
- The different components of the dynpro are :



Flow Logic: calls of the ABAP/4 modules for a screen .

Screen layout: Positions of the text, fields, pushbuttons and so on for a screen

Screen Attributes: Number of the screen, number of the subsequent screen, and others

Fields attributes: Definition of the attributes of the individual fields on a screen.

6. What is a ABAP/4 module pool?

-Each dynpro refers to exactly one ABAP/4 dialog program.Such a dialog program is also called a module pool ,since it consists on interactive modules.

7..Can we use WRITE statements in screen fields?if not how is data transferred from field data to screen fields?

-We cannot write field data to the screen using the WRITE statement.The system instead transfers data by comparing screen fields names with ABAP/4 variable names.If both names are the same,it transfers screen fields values to ABAP/4 programs fields and Vice Versa.This happens immediately after displaying the screen.

8.Can we use flow logic control key words in ABAP/4 and vice-versa?

- The flow control of a dynpro consists os a few statements that syntactically resemble ABAP/4 statements .However ,we cannot use flow control keywords in ABAP/4 and vice-versa.

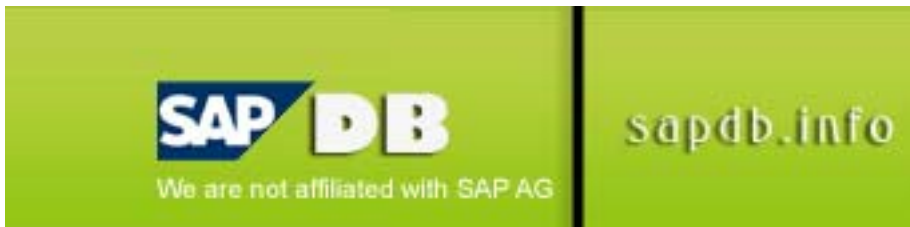
9.What is GUI status? How to create /Edit GUI status?

-A GUI status is a subset of the interface elements used for a certain screen.The status comprises those elements that are currently needed by the transaction .The GUI status for a transaction may be composed of the following elements:

-Title bar.

-Mneu bar.

-Application tool bar



-Push buttons.

To create and edit GUI status and GUI title,we use the Menu Painter.

10. How does the interection between the Dynpro and the ABAP/4 Modules takes place?

-A transaction is a collection os screens and ABAP/4 routines, controlled and executed by a Dialog processor. The Dialog processor processes screen after the screen, thereby triggering the appropriate

ABAP/4 processing of each screen .For each screen,the system executes the flow logic that contains the corresponding ABAP/4 processing.The controls passes from screen flow logic to ABAP/4 code and back.

11. How does the Dialog handle user requests?

- when an action is performed ,the system triggers the PROCESS AFTER INPUT event.The data passed includes field screen data data entered by the user and a function code. A functioncode is a technical name that has been allocated in a screen Painter or Menu Painter to a meny entry,a push button,the ENTER key or a function Key of a screen.An internal work field(ok-code)in the PAI module evaluates the function code,and the appropriate action is taken.

12. What is to be defined for a push button fields in the screen attributes?

- A function code has to be defined in the screen attributes for the push buttons in a screen.

13. How are the function code handles in Flow Logic?

- When the User selects a function in a transaction ,the system copies the function code into a specially designated work field called OK_CODE.This field is Global in ABAP/4 Module Pool.The OK_CODE can then be evaluated in the corresponding PAI module. The function code is always passed in Exactly the same way , regardless of Whether it comes from a screen's pushbutton,a menu option ,function key or other GUI element.

14.What controls the screen flow?

- The SET SCREEN and LEAVE SCREEN statements controls screen flow.

14. The Function code currently active is ascertained by what Variable?



- The function code currently active in a Program can be ascertained from the SY-UCOMM Variable.

15. The function code currently active is ascertained by what variable ?

- By SY-UCOMM Variable.

16. What are the “field” and “chain” Statements?

- The FIELD and CHAIN flow logic statements let you Program Your own checks.FIELD and CHAIN tell the system Which fields you are checking and Whether the System should Perform Checks in the flow logic or call an ABAP/4 Module.

17. What is an “on input filed” statements?

- ON INPUT

The ABAP/4 module is called only if a field contains the Value other than the initial Value.This initial Value is determined by the filed's Dta Type: blanks for character Fields ,Zeroes for numerics. If the user changes the Fields Value back t o its initial value,ON INPUT does not trigger a call.

18. What is an “on request Field” statement?

- ON REQUEST

The ABAP/4 Module is called only if the user has entered the value in the field value since the last screen display .The Value counts as changed Even if the User simply types in the value that was already there .In general ,the ON REQUEST condition is triggered through any Form of” MANUAL INPUT’.

19. What is an on”*-input filed” statement?

- ON *-INPUT

- The ABAP/4 module is called if the user has entered the “*” in the first character of the field, and the field has the attribute *-entry in the screen Painter.You can use this option in Exceptional cases where you want to check only fields with certain Kinds of Input.

20. What are conditional chain statement?



ON CHAIN-INPUT similar to ON INPUT.

The ABAP/4 module is called if any one of the fields in the chain contains a value other than its initial value(blank or nulls).

ON CHAIN-REQUEST

This condition functions just like ON REQUEST, but the ABAP/4 module is called if any one of the fields in the chain changes value.

21. What is “at exit-command:?”

The flowlogic Keyword at EXIT-COMMAND is a special addition to the MODULE statement in the Flow Logic .AT EXIT-COMMAND lets you call a module before the system executes the automatic fields checks.

22. Which Function type has to be used for using “at exit-command” ?

- To Use AT EXIT – COMMAND ,We must assign a function Type “E” to the relevant function in the MENU Painter OR Screen Painter .

23. What are the different message types available in the ABAP/4 ?

- There are 5 types of message types available.
- E: ERROR
- W-WARNING
- I –INFORMATION
- A-ABNORMAL TERMINATION.
- S-SUCCESS

24. Of the two “ next screen “ attributes the attributes that has more priority is -----.

Dynamic.

25. Navigation to a subsequent screen can be specified statically/dynamically. (TRUE/FALSE).

TRUE.



26. Dynamic screen sequence for a screen can be set using ----- and ----- commands

Set Screen, Call screen.

27. The commands through Which an ABAP/4 Module can “branch to “ or “call” the next screen are

1.-----,2-----,3-----,4-----.

- Set screen<scr no>,Call screen<scr no> ,Leave screen, Leave to screen <scr no>.

28. What is difference between SET SCREEN and CALL SCREEN ?

- With SET SCREEN the current screen simply specifies the next screen in the chain , control branches to this next screen as soon as the current screen has been processed .Return from next screen to current screen is not automatic .It does not interrupt processing of the current screen.If we want to branch to the next screen without finishing the current one ,use LEAVE SCREEN.
- With CALL SCREEN , the current (calling) chain is suspended , and a next screen (screen chain) is called .The called can then return to the suspended chain with the statement LEAVE SCREEN TO SCREEN 0 .Sometime we might want to let an user call a pop up screen from the main application screen to let him enter secondary information.After they have completed their entries, the users should be able to close the popup and return directly to the place where they left off in the main screen.Here comes CALL SCREEN into picture .This statement lets us insert such a sequence into the current one.

29. Can we specify the next screen number with a variable (*Yes/No)?

- Yes



30. The field SY-DYNR refers to-----

Number of the current screen.

31. What is dialog Module?

- A dialog Module is a callable sequence of screens that does not belong to a particular transaction. Dialog modules have their module pools , and can be called by any transaction.

32. The Syntex used to call a screen as dialog box (pop up)is-----

CALL SCREEN <screen number.>

STARTING AT <start column><start line>

ENDING AT <end column> <end line>

33. What is “call mode”?

- In the ABAP/4 WORLD each stackable sequence of screens is a “call mode”, This is IMP because of the way u return from the given sequence .To terminate a call mode and return to a suspended chain set the “next screen” to 0 and leave to it:

LEAVE TO SCREEN 0 or (SET SCREEN 0 and LEAVE SCREEN) .When u return to the suspended chain execution resumes with the statement directly following the original CALL SCREEN statement.The original sequence of screens in a transaction (that is , without having stacked any additional call modes),you returned from the transaction altogether.



34. The max number of calling modes stacked at one time is?

- NINE

35. What is LUW or Data base Transaction ?

- A "LUW"(logical unit of work) is the span of time during which any database updates must be performed in an "all or nothing" manner .Either they are all performed (committed),or they are all thrown away (rolled back).In the ABAP/4 world , LUWs and
- Transactions can have several meanings:

LUW (or "database LUW" or "database transaction")

This is the set of updates terminated by a database commit. A LUW lasts, at most, from one screen change to the next (because the SAP system triggers database commits automatically at every screen change).

36. What is SAP LUW or Update Transaction?

Update transaction (or "SAP LUW")

This is a set of updates terminated by an ABAP/4 commit. A SAP LUW may last much longer than a database LUW, since most update processing extends over multiple transaction screens. The programmer terminates an update transaction by issuing a COMMIT WORK statement.

37. What happens if only one of the commands SET SCREEN and LEAVE SCREEN is used without using the other?



If we use SET SCREEN without LEAVE SCREEN, the program finishes processing for the current screen before branching to <scr no>. If we use LEAVE SCREEN without a SET SCREEN before it, the current screen process will be terminated and branch directly to the screen specified as the default next-screen in the screen attributes.

38. What is the significance of the screen number '0'?

In "calling mode", the special screen number 0 (LEAVE TO SCREEN 0) causes the system to jump back to the previous call level. That is, if you have called a screen sequence with CALL SCREEN leaving to screen 0 terminates the sequence and returns to the calling screen. If you have not called a screen sequence, LEAVE TO SCREEN 0 terminates the transaction.

39. What does the 'SUPPRESS DIALOG' do?

Suppressing of entire screens is possible with this command. This command allows us to perform screen processing "in the background". Suppressing screens is useful when we are branching to list-mode from a transaction dialog step.

40. What is the significance of the memory table 'SCREEN'?

At runtime, attributes for each screen field are stored in the memory table called 'SCREEN'. We need not declare this table in our program. The system maintains the table for us internally and updates it with every screen change.

41. What are the fields in the memory table 'SCREEN'?

Name	Length	Description
NAME	30	Name of the screen field
GROUP1	3	Field belongs to field group 1

GROUP2	3	Field belongs to field group 2
GROUP3	3	Field belongs to field group 3
GROUP4	3	Field belongs to field group4
ACTIVE	1	Field is visible and ready for input.
REQUIRED	1	Field input is mandatory.
INPUT	1	Field is ready for input.
OUTPUT	1	Field is display only.
INTENSIFIED	1	Field is highlighted
INVISIBLE	1	Field is suppressed.
LENGTH	1	Field output length is reduced.
DISPLAY_3D	1	Field is displayed with 3D frames.
VALUE_HELP	1	Field is displayed with value help.

42. Why grouping of fields is required? What is the max no of modification groups for each field?

If the same attribute need to be changed for several fields at the same time these fields can be grouped together. We can specify up to four modification groups for each field.

43. What are the attributes of a field that can be activated or deactivated during runtime?

Input, Output, Mandatory, Active, Highlighted, Invisible.

44. What is a screen group? How it is useful?



Screen group is a field in the Screen Attributes of a screen. Here we can define a string of up to four characters which is available at the screen runtime in the SY-DNGR field. Rather than maintaining field selection separately for each screen of a program, we can combine logically associated screens together in a screen group.

45. What is a Subscreen? How can we use a Subscreen?

A subscreen is an independent screen that is displayed in a n area of another (“main”) screen. To use a subscreen we must call it in the flow logic (both PBO and PAI) of the main screen. The CALL SUBSCREEN statement tells the system to execute the PBO and PAI events for the subscreen as part of the PBO or PAI events of the main screen. The flow logic of your main program should look as follows:

PROCESS BEFORE OUTPUT.

CALL SUBSCREEN <area> INCLUDING '<program>' '<screen>'.

PROCESS AFTER INPUT.

CALL SUBSCREEN <area>.

Area is the name of the subscreen area you defined in your main screen. This name can have up to ten characters. Program is the name of the program to which the subscreen belongs and screen is the subscreen's number.

46. What are the restrictions on Subscreens?

Subscreens have several restrictions. They cannot:

- Set their own GUI status
- Have a named OK code
- Call another screen
- Contain an AT EXIT-COMMAND module
- Support positioning of the cursor.



47. How can we use / display table in a screen?

ABAP/4 offers two mechanisms for displaying and using table data in a screen. These mechanisms are TABLE CONTROLS and STEP LOOPS.

48. What are the differences between TABLE CONTROLS and STEP LOOPS?

TABLE CONTROLS are simply enhanced STEP LOOPS that display with the look and feel of a table widget in a desktop application. But from a programming standpoint, TABLE CONTROLS and STEP LOOPS are almost exactly the same. One major difference between STEP LOOPS and TABLE CONTROLS is in STEP LOOPS their table rows can span more than one time on the screen. By contrast the rows in a TABLE CONTROLS are always single lines, but can be very long. (Table control rows are scrollable). The structure of table control is different from step loops. A step loop, as a screen object, is simply a series of field rows that appear as a repeating block. A table control, as a screen object consists of: i) table fields (displayed in the screen) ii) a control structure that governs the table display and what the user can do with it.

49. What are the dynapro keywords?

FIELD, MODULE, SELECT, VALUES and CHAIN are the dynapro keywords.

50. Why do we need to code a LOOP statement in both the PBO and PAI events for each table in the screen?

We need to code a LOOP statement in both PBO and PAI events for each table in the screen. This is because the LOOP statement causes the screen fields to be copied back and forth between the ABAP/4 program and the screen field. For this reason, at least an empty LOOP....ENDLOOP must be there.



51. The field SY-STEPL refers to the index of the screen table row that is currently being processed. The system variable SY-stepl only has a meaning within the confines of LOOP....ENDLOOP processing. Outside the loop, it has no valid value.

52. How can we declare a table control in the ABAP/4 program?

Using the syntax controls <table control name> type tableview using screen <scr no>.

53. Differentiate between static and dynamic step loops.

Step loops fall into two classes: Static and Dynamic. Static step loops have a fixed size that cannot be changed at runtime. Dynamic step loops are variable in size. If the user re-sizes the window the system automatically increases or decreases the number of step loop blocks displayed. In any given screen you can define any number of static step loops but only a single dynamic one.

54. What are the two ways of producing a list within a transaction?

By submitting a separate report.

By using leave to list-processing.

55. What is the use of the statement Leave to List-processing?

Leave to List-processing statement is used to produce a list from a module pool. Leave to list processing statement allows to switch from dialog-mode to list-mode within a dialog program.

56. When will the current screen processing terminates?



A current screen processing terminates when control reaches either a Leave-screen or the end of PAI.

57. How is the command Suppress-Dialog useful?

Suppressing entire screens is possible using this command. This command allows us to perform screen processing "in the background". The system carries out all PBO and PAI logic, but does not display the screen to the user. Suppressing screens is useful when we are branching to list-mode from a transaction dialog step.

58. What happens if we use Leave to list-processing without using Suppress-Dialog?

If we don't use Suppress-Dialog to next screen will be displayed but as empty, when the user presses ENTER, the standard list output is displayed.

59. How the transaction that are programmed by the user can be protected?

By implementing an authority check.

60. What are the modes in which any update tasks work?

Synchronous and Asynchronous.

61. What is the difference between Synchronous and Asynchronous updates?

A program asks the system to perform a certain task, and then either waits or doesn't wait for the task to finish. In synchronous processing, the program waits: control returns to the program only when the task has been completed. In asynchronous processing, the program does not wait: the system returns control after merely logging the request for execution.



62. SAP system configuration includes Dialog tasks and Update tasks.

63. Dialog-task updates are Synchronous updates.

64. Update –task updates are Asynchronous updates.

65. What is the difference between Commit-work and Rollback-Work tasks?

Commit-Work statement “performs” many functions relevant to synchronized execution of tasks. Rollback-work statement “cancels: all reuests relevant to synchronized execution of tasks.

66. What are the different database integrities?

- Semantic Integrity.
- Relational Integrity.
- Primary Key Integrity.
- Value Set Integrity.
- Foreign Key integrity and
- Operational integrity.

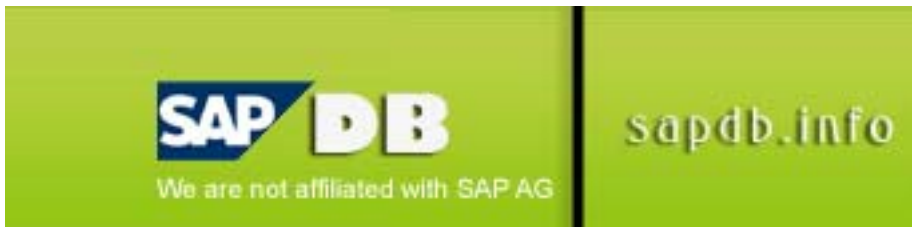
67. All SAP Databases are Relational Databases.

68. What is SAP locking?

It is a mechanism for defining and applying logical locks to database objects.

69. What does a lock object involve?

The tables.



The lock argument.

70. What are the different kinds of lock modes?

Shared lock

Exclusive lock.

Extended exclusive list.

71. How can a lock object be called in the transaction?

By calling Enqueue<lock object> and Dequeue<lock object> in the transaction.

72. What are the events by which we can program "help texts" and display "possible value lists"?

-PROCESS ON HELP-REQUEST (POH).

-PROCESS ON VALUE-REQUEST (POV).

73. What is a matchcode?

A matchcode is an aid to finding records stored in the system whenever an object key is required in an input field but the user only knows other (non-key) information about the object.

74. In what ways we can get the context sensitive F1 help on a field?

- Data element documentation.

- Data element additional text in screen painter.



- Using the process on help request event.

75. What is roll area?

A roll area contains the program's runtime context. In addition to the runtime stack and other structures, all local variables and any data known to the program are stored here.

76. How does the system handle roll areas for external program components?

- Transactions run in their own roll areas.
- Reports run in their own roll areas.
- Dialog modules run in their own roll areas
- Function modules run in the roll area of their callers.

77. Does the external program run in the same SAP LUW as the caller, or in a separate one?

- Transactions run with a separate SAP LUW
- Reports run with a separate SAP LUW.
- Dialog modules run in the same SAP LUW as the caller
- Function modules run in the same SAP LUW as the caller.

The only exceptions to the above rules are function modules called with IN UPDATE TASK (V2 function only) or IN BACKGROUND TASK (ALE applications). These always run in their own (separate) update transactions.

78. What are function modules?

Function modules are general-purpose library routines that are available system-wide.



79. What are the types of parameters in the function modules?

In general, function module can have four types of parameters:

- EXPORTING: for passing data to the called function.
- IMPORTING: for receiving data returned from the function module.
- TABLES: for passing internal tables only, by reference (that is, by address).
- CHANGING: for passing parameters to and from the function.

80. What is the difference between Leave Transaction and Call Transaction?

In contrast to LEAVE TO TRANSACTION, the CALL TRANSACTION statement causes the system to start a new SAP LUW. This second SAP LUW runs parallel to the SAP LUW for the calling transaction.

81. How can we pass selection and parameter data to a report?

There are three options for passing selection and parameter data to the report.

- Using SUBMIT...WITH
- Using a report variant.
- Using a range table.

82. How to send a report to the printer instead of displaying it on the screen?

We can send a report to the printer instead of displaying it on the screen. To do this, use the keywords TO SAP-SPOOL:

SUBMIT RSFLFIND...TO SAP-SPOOL DESTINATION 'LT50'.

83. How can we send data to external programs?

Using SPA/GPA parameters (SAP memory).

Using EXPORT/IMPORT data (ABAP/4 memory)

84. What are SPA/GPA parameters (SAP memory)

SPA/GPA parameters are field values saved globally in memory. There are two ways to use SPA/GPA parameters:

By setting field attributes in the Screen Painter.

By using the SET PARAMETER or GET PARAMETER statements.