

SAP ABAP Interview Questions, Answers, and Explanations: ABAP Certification Review

By Barry Fewer

SAP ABAP Interview Questions

Please visit our website at www.sapcookbook.com

© 2006 Equity Press all rights reserved.

ISBN 1-933804-06-8

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without either the prior written permission of the publisher or a license permitting restricted copying in the United States or abroad.

The programs in this book have been included for instructional value only. They have been tested with care but are not guaranteed for any particular purpose. The publisher does not offer any warranties or representations nor does it accept any liabilities with respect to the programs.

Trademark notices

SAP, SAP ABAP, Netweaver, and SAP New Dimension are registered trademarks of SAP AG. This publisher gratefully acknowledges SAP permission to use its trademark in this publication. SAP AG is not the publisher of this book and is not responsible for it under any aspect of the law.

SAP ABAP Interview Questions

Table of Contents

SAP ABAP Interview Questions, Answers, and Explanations.....	1
Motivation	7
Introduction	9
The ABAP Guru has Spoken!	9
Part I: Database Updates and Changing the SAP Standard.....	10
Question 1: Lock Objects and Function Modules	11
Question 2: Update function modules.....	13
Question 3: ABAP Memory Exchanging	14
Question 4: Authorization Objects.....	15
Question 5: Modifications	16
Question 6: Modification Assistant.....	17
Question 7: Function Module Exit	18
Question 8: Transaction Variants	19
Question 9: Changing SAP	21
Question 10: Adding Fields	23
Question 11: What is a BADI?	24
Part II: List Processing.....	25
Question 12: SAP Query.....	26
Question 13: List Generation	27
Question 14: Output an Icon.....	28
Question 15: List Processing	29
Question 16: System internal table?	30
Question 17: Selection Screen Number	33
Question 18: Logical Database.....	34
Question 19: Native and Open SQL.....	36
Question 20: AT LINE-SELECTION.....	37
Part III: Internal Tables & ALV Grid Control	38
Question 21: Internal Tables.....	39

SAP ABAP Interview Questions

Question 22: COLLECT Statement.....	41
Question 23: AT NEW Control Structure	42
Question 24: ALV Grid Control	44
Question 25: Code Example	45
Question 26: Field Descriptions	46
Question 27: ALV Grid and Events.....	47
Part IV: Dialog programming.....	48
Question 28: Module Pool	49
Question 29: Important TCODES.....	50
Question 30: Inner and Outer Join	51
Question 31: GUI Status.....	53
Question 32: SY-UCOMM.....	55
Question 33: Dialog Execution	56
Question 34: Search Helps.....	57
Question 35: Subscreen Call	58
Question 36: Subscreen OK_CODE	59
Question 37: Table Control	60
Part V: ABAP Objects.....	61
Question 38: Object Oriented Languages.....	62
Question 39: Class and Objects	64
Question 40: READ-ONLY	65
Question 41: Public versus private attributes....	66
Question 42: Static Attributes.....	67
Question 43: Instance v. Static Method	68
Question 44: Object Instantiation	69
Question 45: Garbage Collector	70
Question 46: Constructor Method	71
Question 47: Multiple Inheritance	72
Question 48: Redefined Method	73
Question 49: Interfaces	75
Question 50: Abstract Classes.....	77
Part VI: Data Transfer	78
Question 51: Data Transfer.....	79

SAP ABAP Interview Questions

Question 52: BDCDATA.....	81
Question 53: What is LSMW?	82
Question 54: Batch Input	83
Part VII: BASIS.....	84
Question 55: Client Definition.....	85
Question 56: Tiered Architecture.....	86
Question 57: Dispatcher	87
Question 58: Work Processes	88
Question 59: RFCs.....	89
Question 60: SAPNet	91
Question 61: CCMS.....	92
Question 62: Transport Change Request.....	93
Question 63: Internet outages.....	94
Question 64: OK Codes.....	95
Question 65: Transaction Codes.....	96
Question 66: STMS Importing	96
Question 67: Work Processes	97
Question 68: Patch Level for GUI	97
Question 69: Web help installation	98
Question 70: Web Resources	99
Question 71: Instance installation	99
Question 72: Client copy.....	100
Question 73: Table T000.....	101
Question 74: Table USR02.....	101
Question 75: Passwords.....	102
Question 76: Table TADIR.....	102
Question 77: Table TDEVC.....	103
Question 78: Change Requests.....	103
Question 79: User Access	104
Question 80: Transport object keys.....	104
Question 81: Transport in progress	105
Question 82: Repaired Objects.....	105
Question 83: Disable Multiple Logins	106

SAP ABAP Interview Questions

Question 84: Locked Transactions	107
Part VIII: ABAP Development	108
Question 85: Chained Statements	109
Question 86: Defining Constants.....	110
Question 87: Return Codes	111
Question 88: SELECT SINGLE.....	112
Question 89: SELECT SEMANTICS.....	113
Question 90: Array Fetch.....	114
Question 91: Dialog Data Transport.....	115
Question 92: Domain v. Data Element	116
Question 93: Table Buffering.....	117
Question 94: CONDENSE.....	119
Question 95: CLEAR and FREE.....	120
Question 96: Passing Parameters.....	121
Question 97: Assigning Attributes.....	123
Question 98: Function Group.....	124
INDEX.....	125
ABOUT THE AUTHOR	131

Motivation

My motivation for writing this book is based on two different reasons resulting from my experience working with SAP over the last eight years.

The first reason stems from my recent re-certification in ABAP. Anyone who has ever written an SAP certification knows the enormous amount of material that must be reviewed in order to prepare for the exam. In my most recent endeavor I found that there was a definite lack of material covering the type of questions presented in the ABAP exam. It is this experience that has led me to write this book that provides sample questions and related material.

The second reason is to provide technical managers a guide for interviewing ABAP resources. The ABAP language provides a wealth of functionality and tools for developers. It is difficult to remember "all the right questions" that should be asked in an interview especially as not all SAP projects use all of the features, tools and techniques that the product offers. This book enables the interviewer to ask the right questions in order to select the best candidate.

I am confident that this book will provide an excellent precursor for an ABAP developer that is planning on writing the certification exam. I am equally sure that it will assist technical managers in completing a thorough interview covering all needed aspects of the language in the hopes of finding the best possible candidate.

Barry Fewer

Ottawa, Ontario

January 2006

SAP ABAP Interview Questions

Introduction

This book is divided into the following sections:

- I. Database Updates and Changing the SAP Standard
- II. List Processing
- III. Internal Tables & ALV Grid Control
- IV. Dialog Programming
- V. ABAP Objects
- VI. Data Transfer
- VII. BASIS
- VIII. ABAP Development

Each question has a question and an answer – that is pretty straightforward – but when you see the guru icon – this is information that represents the highest degree of knowledge in a particular area. So if you’re looking for an “ABAP Objects guru” be sure to listen for an answer similar to those given under the guru icon.



Don't be bamboozled!

The ABAP Guru has Spoken!



Part I: Database Updates and Changing the SAP Standard

Question 1: Lock Objects and Function Modules

What are the names of the function modules that will be generated upon activation of a lock object?

A: Prior to creating setting any locks in an ABAP program, a lock object must be created in the ABAP dictionary. When the lock object is created, the system will automatically generate the two function modules required for lock management.

The first function module used to set locks is the ENQUEUE_<Lock Object Name>. This function module is used to generate a lock entry in the lock table. If the lock cannot be set for any reason then this would be reflected in the return code.

The second function module used to release locks is the DEQUEUE_<Lock Object Name>. This function module will remove a lock entry from the lock table.

These function modules are used to set and release locks in an ABAP program. The programmer

SAP ABAP Interview Questions

simply performs the specific “CALL FUNCTION...” statement for each function module.



These lock function modules are executed in a specific work process with the SAP system dedicated to lock management. This work process is run on a single server that maintains the central lock table for the entire SAP system.

There are two types of locks that can be set:

Shared lock – is a read only lock that prevents data from being changed while a user is reading it.

Exclusive lock – is a write lock that prevents data from being changed while a user is already modifying it.

Question 2: Update function modules

Update function modules are classified as either V1 or V2. Which type of update is performed first and in what mode (Asynchronously, Synchronously or locally) can each type be processed in?

A: V1 updates take priority over V2 updates and are therefore processed prior to V2 updates. V1 updates can be performed asynchronously, synchronously or locally. V2 updates always run asynchronously.

Question 3: ABAP Memory Exchanging

What two statements would be used to exchange data between programs using ABAP memory?

A: EXPORT to MEMORY ID <id> will copy data to ABAP memory and IMPORT from MEMORY ID <id> is used to copy the data from ABAP memory into a program.

The data that is being exchanged via ABAP memory must be declared in the two programs involved with exactly the same data declarations.

Question 4: Authorization Objects

What are authorization objects and what statement is used to perform an authorization check in an ABAP program?

A: Authorization objects are composed of a grouping of fields. The values in these fields will be used in an authorization check.

AUTHORITY-CHECK is the statement used in an ABAP program to perform an authorization check against an authorization object. In the AUTHORITY-CHECK statement, all fields of the object must be addressed or the keyword DUMMY is used to bypass checking on a field.



There can be a maximum of 10 fields defined on an authorization object.

Question 5: Modifications

What is the definition of a modification in an SAP system and how do they impact an upgrade?

A: A modification is a change made by a customer to an SAP delivered repository object. Modifications must be reviewed during upgrade time to determine if the new SAP object should be used or if the modified object can still be used with or without further modifications.

Question 6: Modification Assistant

What is the Modification Assistant?

A: The modification assistant is a tool introduced in release 4.5 to simplify the upgrade process as it relates to modifications. The modification assistant can be triggered through the ABAP editor and it will assist in logging modifications for any changes that are made to the system.

The modification assistant provides support for modification made through the ABAP Editor, Screen Painter, Menu Painter, text element maintenance, Function Builder and ABAP Dictionary.

Question 7: Function Module Exit

What statement will be found in an SAP application program that implements a function module exit?

A: Function module exits will exist in some SAP application programs to allow a customer to add some functionality to the SAP program. Search the SAP application program globally looking for the following search string: "CALL CUSTOMER" to determine if an exit exists.

Question 8: Transaction Variants

What is a transaction variant and why are they used?

A: A transaction variant is a set of screen variants that is used to predefine screen behavior and defaults. Fields, subscreens and full screens that may not be required by a user can be suppressed from that users view through the use of variant functionality. Default values can also be set for any input fields and fields can have the “ready for input” status revoked.

A transaction variant can only be created for a dialog or reporting transaction and only normal, subscreen and dialog screens can be included in the variant.



The GuiXT script language allows a developer to modify a screen through transaction

SAP ABAP Interview Questions

variant maintenance providing more flexibility and power to the developer. Screen layouts can be modified by inserting buttons, value helps, moving objects, inserting screens and much more

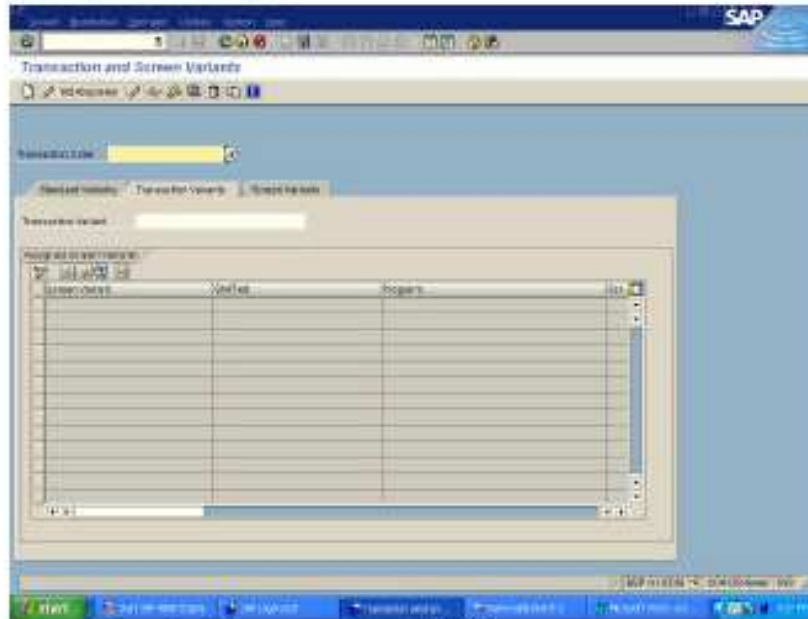


Fig. 1: Transaction SHDo

Question 9: Changing SAP

List some of the many different ways that the SAP standard functionality can be changed by a customer?

A: The SAP standard can be changed through personalization, customizing, modifications, enhancements and custom ABAP programs developed by the customer development team. Examples of these different approaches for changing the standard are as follows:

Personalization techniques include creation of variants, set/get parameters and activity groups.

Customizing is the most common way for changing the SAP standard through the use SAP tools such as the R/3 reference model and the Implementation Guide. Customizing could be considered mandatory in order to implement R/3 and is usually performed by the functional team.

Enhancements are usually performed by the development team and include activities such as dictionary enhancements, function module exits, menu and screen exits and Business add-ins (BADI's)

SAP ABAP Interview Questions

Custom ABAP programs are developed for a wide variety of reasons and can work with standard SAP objects or custom developed objects.

Modifications to SAP objects are strongly discouraged. All other techniques should be explored before modifying an SAP object. The SAP Software Change Registration (SSCR) is a procedure for registering all manual modification to SAP objects.

Question 10: Adding Fields

What are two different ways to add fields to SAP tables?

A: An append structure or a customizing include can be used to add fields to a table. Append structures are created by adding fields to the end of a table while Customizing includes are specified by the SAP developer in advance to allow for the customer to create new fields.

Question 11: What is a BADI?

A: BADI stands for Business add-in. BADI's are a relatively new enhancement concept that make use of ABAP Object technology. This is a true object oriented approach to SAP enhancements using classes, interfaces and methods to implement the BADI

In order to enhance a SAP application program, the BADI must first be defined. The developer will create an interface for the BADI. An adapter class is created from this that will implement the interface. Developers will then create an instance of the adapter class in the SAP application program and create method calls as required.

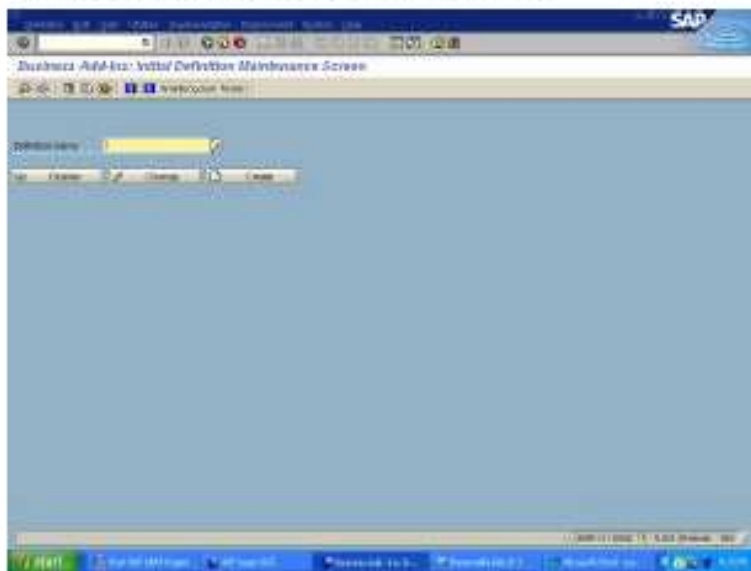


Fig. 2: SE18 – BADI Maintenance



Part II: List Processing

Question 12: SAP Query

Explain the relationship between a functional area, user group and query when developing queries using the SAP Query tool?

A: Functional areas and user groups are organizational elements used in SAP Query that must first be created in order to use the query environment.

The data that a query will be based on is defined in a functional area. A functional area is typically based on a logical database but can be any table and fields within that table.

Users are assigned to user groups. Functional areas are also assigned for the user group allowing users to create and start queries based on the data in that functional area. Multiple functional areas can be assigned to a user group and a functional area can be assigned to several different groups.

Question 13: List Generation

What ABAP statement will trigger list generation?

A: The first WRITE statement encountered in an ABAP program will trigger list processing. Upon completion of the program, the list is output. The system will generate some standard headings for list output that includes the program name and a page number in the heading.

Question 14: Output an Icon

What two statements are required in an ABAP program to output an icon using a write statement?

A: There are 2 statements required in the ABAP program:

The INCLUDE <ICON> or INCLUDE <LIST> statement is required in the program.

The WRITE statement will have the following syntax: WRITE <Icon-Name> AS ICON.

The Include files contain the names of constants that represent all system defined icons. <LIST> contains only icons that are relevant for list processing while <ICON> will contain all system defined icons.

A: SY-LINSZ – the line width from REPORT statement LINE-COUNT.

Dictionary: Display Structure

Structure: DST Info

Short Text: ABAP Dictionary

Navigation: Back Components Display Structure Component Short Text

Table: 28 / 1 (2)

Component	DT	Component Type	Data Type	Length	Occs	Short Text	Group
DST001	DST001	1674	16	16	16	Return value, internal	
DST002	DST002	1674	16	16	16	Return	
DST003	DST003	1674	16	16	16	Return value	
DST004	DST004	1674	16	16	16	Return processing, internal	
DST005	DST005	1674	16	16	16	Return Value, Return Value Mod ABAP Statements	
DST006	DST006	1674	16	16	16	Return	
DST007	DST007	1674	16	16	16	Return, it is not a supported for ABAP	
DST008	DST008	1674	16	16	16	Return, it is not a supported for ABAP	
DST009	DST009	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST010	DST010	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST011	DST011	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST012	DST012	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST013	DST013	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST014	DST014	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST015	DST015	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST016	DST016	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST017	DST017	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST018	DST018	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST019	DST019	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST020	DST020	1674	16	16	16	Return processing, it is not a supported for ABAP	
DST021	DST021	1674	16	16	16	Return processing, it is not a supported for ABAP	

Navigation: Back Components Display Structure Component Short Text

Status: SAP (7) (600) 11 10/21/2008 8:02

- 29 -

Question 16: What is the format of the system generated SELECT-OPTIONS internal table?

A: The SELECT-OPTIONS statement is usually used to create selection criteria on an input screen. The statement will create a input field(s) on a screen that will allow a user to input complex selection criteria. The basic syntax is as follows:

SELECT-OPTIONS: <name> FOR < field-name>

The <name> is the name of the field to appear on the selection screen while the <field-name> is the name of the field (which could be from a database table) for which the selection criteria is being created.

The SELECT-OPTIONS statement will trigger the system to create an internal table that will have the following structure:

Sign	Option	Low	High
I	EQ	A	V

The low and high values are entered by the user on the screen. These are the low and high values for the field of which the SELECT-OPTION is defined. Every line of data in the internal table will formulate a condition based on the sign and option fields and data values.

The sign field can have one of the following values:

I – Include OR E- Exclude

SAP ABAP Interview Questions

The option field can have one of the following values:

NE	Not Equal	GT	Greater than
EQ	Equal	LT	Less than
LE	Less than equal	BT	Between
GE	Greater than equal	NB	Not between
CP	Contains pattern	NP	Not contains pattern

The data selected will be the union of all the “Includes” minus the union of all the “Excludes”. If the table only consists of rows that have a sign of ‘E’ then the data selected will be all data outside of these conditions.

There are many useful additions that can be applied to the `SELECT-OPTION` statement such as :

1. `OBLIGATORY` – used to make the field mandatory
2. `NO DISPLAY` – used to hide the input field
3. `NO INTERVALS` – used to restrict the display to one single field
4. `NO-EXTENSION` – used to disable multiple selections (the arrow to the right of the select option field on the screen)
5. `DEFAULT`- used to specify default values for the input fields
6. `MATCHCODE OBJECT` – used to specify a matchcode for input field

SAP ABAP Interview Questions

7. MODIF ID – used to assign a screen modification group to the input fields

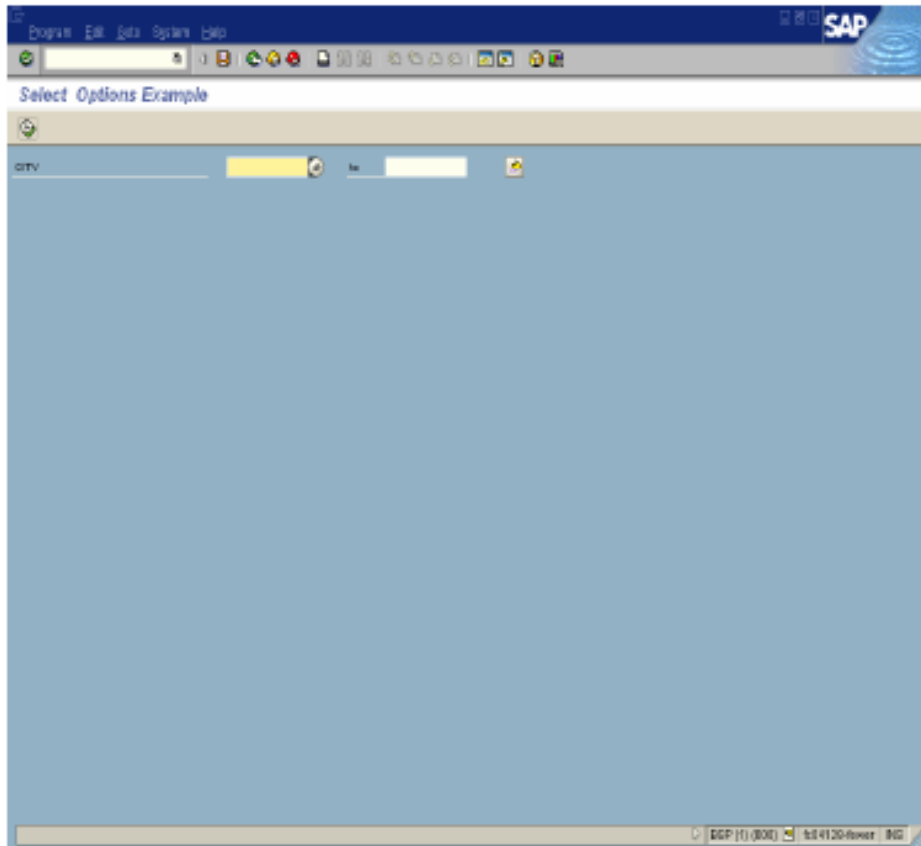


Fig. 4: Select Option field on a screen

Question 17: Selection Screen Number

What is the screen number of the default selection screen in a program?

A: The default selection screen number in an ABAP program is 1000. The screen number can be viewed by choosing System → Status while on the selection screen. The following system status screen is displayed.

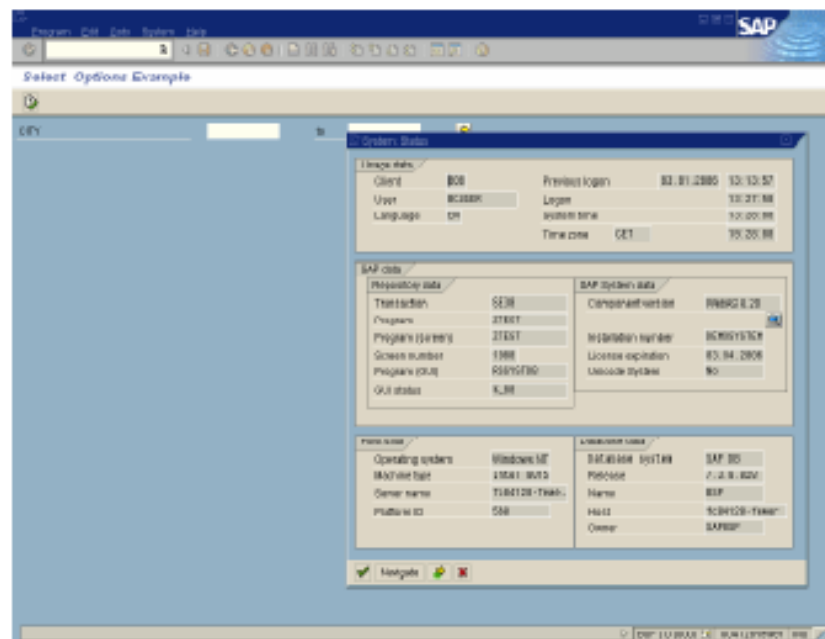


Fig 5: System Status Display

Question 18: Logical Database

What is a logical database?

A: A logical database is an ABAP program that reads data from database tables and makes this available to an ABAP program. The logical database provides a system generated selection screen. The ABAP program must specifically refer to the logical database name in its program attributes.

An ABAP developer does not have to worry about screen design or coding any SQL in order to get the required data. The logical database handles this automatically.

The logical database is hierarchical in structure composed on several nodes. A developer can control the depth of the data retrieval by using the GET <NODE> statement.

It may not be a good idea to use a logical database if the only data your program requires is in the last node of a deep hierarchy. The more performance friendly approach would be to program the data retrieval.

SAP ABAP Interview Questions



A logical database can be called directly from an ABAP program by using a function module called **LDB_PROCESS**.

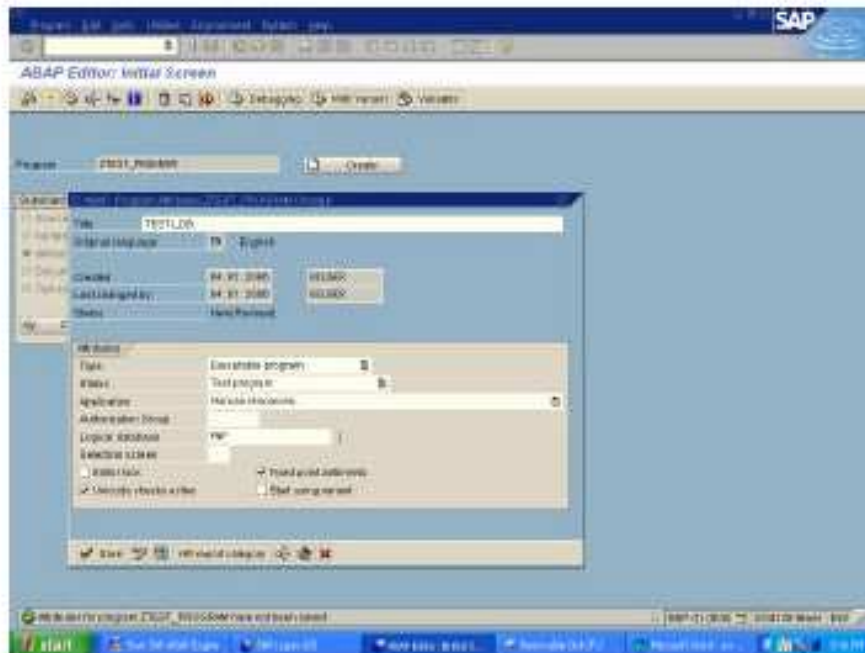


Fig 6: Create program using logical database

Question 19: Native and Open SQL

What is the difference in native SQL and open SQL?

A: Open SQL statements are program independent of the operating system. Native SQL statements are platform dependent. Native SQL is bound into a program using EXEC SQL statements

Question 20: AT LINE-SELECTION

When is the AT LINE-SELECTION event triggered in an ABAP program?

A: The AT LINE-SELECTION event is triggered when a user double clicks on a line in list.



The function code "PICK" is associated to the AT LINE-SELECTION event.



Part III: Internal Tables & ALV Grid Control

Question 21: Internal Tables

What are the types of internal tables that can be defined in ABAP?

A: The types of internal tables that can be defined in ABAP are STANDARD, SORTED, HASHED.

Standard tables have an internal index. Records can be accessed by using the index or the key. The response time for key access depends on the number of rows in the table. A standard table always has a non-unique key. Standard tables can be populated quickly as the system does not have to for existing entries.

Sorted tables also have an internal index and are always sorted by the key. They also have an internal index. Records can be accessed by using the index or the key.. The response time for key access depends on the number of entries in the table. The system uses a binary search for key access. The key can be either unique or non-unique and must be specified when creating the table.

Standard tables and sorted tables are also known as index tables

SAP ABAP Interview Questions

Hashed tables have no internal index and therefore can only be accessed through a key which must be specified as unique. The response time for access depends solely on a hashing algorithm



In addition to these 3 common “fully specified” table types, there are 2 generic table types that can also be used in defining internal tables. The table type INDEX can be used to define a table where index access is permitted. Therefore it can represent either a STANDARD or SORTED table. The table type ANY can be used to represent any table type.

***Question 22: COLLECT
Statement***

What is the purpose of the COLLECT statement?

A: The COLLECT statement is used to fill an internal table with data by summarizing the entries based on key data.

COLLECT <work area> into <internal table>

The contents of the work area are inserted into the internal table by appending the entry to the end of the table if a row does not already exist with the same key data. If a row does exist with the same key data then the numeric fields from the work area are added to the row in the internal table.

Question 23: AT NEW Control Structure

When looping through entries of an internal table, what would the following AT NEW ... ENDAT block be used to accomplish?

LOOP at <Internal Table>.

AT NEW <Field1>.

...

ENDAT.

ENDLOOP.

A: AT...ENDAT statements are used to perform control break processing. Control break processing occurs when the value of a field in the current record is different from the value of the field in the previous record.

In the example above, when the value of <Field1> changes the code between AT...ENDAT is executed. The most common processing that would take place here would be summation of numeric fields.

There are several other AT statements that can be used to perform different control level processing in ABAP programs such as:

1. **AT FIRST** – used to perform specific one time processing when the first record is encountered in the loop

SAP ABAP Interview Questions

2. AT LAST – used to perform specific one time processing when the last record is encountered in the loop.
3. AT END OF <field1> - used to perform specific control level processing at the end of one control level



When a control break is encountered and the code is executed within the AT...ENDAT block, the work area contents will be as follows: all character fields to the left of the control key are filled with an '*' and all others to the right of the control key will contain initial values.

Question 24: ALV Grid Control

What is the relationship between the following two classes that are required to create an ALV grid control?

- 1. CL_GUI_CUSTOM_CONTAINER**
- 2. CL_GUI_ALV_GRID**

A: The CL_GUI_CUSTOM_CONTAINER class is required to create the container control that will hold the ALV Grid Control.

The CL_GUI_ALV_GRID class is required to create the ALV grid control.

Reference variables are actually created using these classes and later instantiated in the program using the CREATE OBJECT statement.

Question 25: Code Example

What does 'MY_CONTAINER' refer to in the following example and where is it defined?

```
DATA: g_container TYPE REF to  
      CL_GUI_CUSTOM_CONTAINER.
```

```
...
```

```
CREATE OBJECT g_container EXPORTING  
      container_name = 'MY_CONTAINER'.
```

A: 'MY_CONTAINER' refers to the name of
subscreen container control that must be created
using the screen painter.

Question 26: Field Descriptions

What are two different ways for specifying field descriptions for an ALV control ?

A: The field descriptions for the ALV control can be taken from a dictionary structure or by defining each specific field by using a field catalog.



The name of the table type used to specify the field catalog is lvc_t_fcat.

The screenshot shows the SAP Class Builder interface for class CL_GUI_ALV_GRID. The 'Methods' tab is active, displaying a list of methods with their signatures and descriptions. The methods are organized into a table with columns: Name, Level, Method, and Description. The methods include various getters and setters for grid properties, as well as methods for handling grid data and events.

Name	Level	Method	Description
IF_GUI_GRID~GET_GRID	Interface	Public	Read Grid Fields
IF_GUI_GRID~GET_GRID_F1	Interface	Public	Get Column 1 Property
IF_GUI_GRID~GET_GRID_F2	Interface	Public	Get Column 2 Property
IF_GUI_GRID~GET_GRID_F3	Interface	Public	Get Column 3 Property
IF_GUI_GRID~GET_GRID_F4	Interface	Public	Get Column 4 Property
IF_GUI_GRID~GET_GRID_F5	Interface	Public	Get Column 5 Property
IF_GUI_GRID~GET_GRID_F6	Interface	Public	Get Column 6 Property
IF_GUI_GRID~GET_GRID_F7	Interface	Public	Get Column 7 Property
IF_GUI_GRID~GET_GRID_F8	Interface	Public	Get Column 8 Property
IF_GUI_GRID~GET_GRID_F9	Interface	Public	Get Column 9 Property
IF_GUI_GRID~GET_GRID_F10	Interface	Public	Get Column 10 Property
IF_GUI_GRID~GET_GRID_F11	Interface	Public	Get Column 11 Property
IF_GUI_GRID~GET_GRID_F12	Interface	Public	Get Column 12 Property
IF_GUI_GRID~GET_GRID_F13	Interface	Public	Get Column 13 Property
IF_GUI_GRID~GET_GRID_F14	Interface	Public	Get Column 14 Property
IF_GUI_GRID~GET_GRID_F15	Interface	Public	Get Column 15 Property
IF_GUI_GRID~GET_GRID_F16	Interface	Public	Get Column 16 Property
IF_GUI_GRID~GET_GRID_F17	Interface	Public	Get Column 17 Property
IF_GUI_GRID~GET_GRID_F18	Interface	Public	Get Column 18 Property
IF_GUI_GRID~GET_GRID_F19	Interface	Public	Get Column 19 Property
IF_GUI_GRID~GET_GRID_F20	Interface	Public	Get Column 20 Property
IF_GUI_GRID~GET_GRID_F21	Interface	Public	Get Column 21 Property
IF_GUI_GRID~GET_GRID_F22	Interface	Public	Get Column 22 Property
IF_GUI_GRID~GET_GRID_F23	Interface	Public	Get Column 23 Property
IF_GUI_GRID~GET_GRID_F24	Interface	Public	Get Column 24 Property
IF_GUI_GRID~GET_GRID_F25	Interface	Public	Get Column 25 Property
IF_GUI_GRID~GET_GRID_F26	Interface	Public	Get Column 26 Property
IF_GUI_GRID~GET_GRID_F27	Interface	Public	Get Column 27 Property
IF_GUI_GRID~GET_GRID_F28	Interface	Public	Get Column 28 Property
IF_GUI_GRID~GET_GRID_F29	Interface	Public	Get Column 29 Property
IF_GUI_GRID~GET_GRID_F30	Interface	Public	Get Column 30 Property
IF_GUI_GRID~GET_GRID_F31	Interface	Public	Get Column 31 Property
IF_GUI_GRID~GET_GRID_F32	Interface	Public	Get Column 32 Property
IF_GUI_GRID~GET_GRID_F33	Interface	Public	Get Column 33 Property
IF_GUI_GRID~GET_GRID_F34	Interface	Public	Get Column 34 Property
IF_GUI_GRID~GET_GRID_F35	Interface	Public	Get Column 35 Property
IF_GUI_GRID~GET_GRID_F36	Interface	Public	Get Column 36 Property
IF_GUI_GRID~GET_GRID_F37	Interface	Public	Get Column 37 Property
IF_GUI_GRID~GET_GRID_F38	Interface	Public	Get Column 38 Property
IF_GUI_GRID~GET_GRID_F39	Interface	Public	Get Column 39 Property
IF_GUI_GRID~GET_GRID_F40	Interface	Public	Get Column 40 Property
IF_GUI_GRID~GET_GRID_F41	Interface	Public	Get Column 41 Property
IF_GUI_GRID~GET_GRID_F42	Interface	Public	Get Column 42 Property
IF_GUI_GRID~GET_GRID_F43	Interface	Public	Get Column 43 Property
IF_GUI_GRID~GET_GRID_F44	Interface	Public	Get Column 44 Property
IF_GUI_GRID~GET_GRID_F45	Interface	Public	Get Column 45 Property
IF_GUI_GRID~GET_GRID_F46	Interface	Public	Get Column 46 Property
IF_GUI_GRID~GET_GRID_F47	Interface	Public	Get Column 47 Property
IF_GUI_GRID~GET_GRID_F48	Interface	Public	Get Column 48 Property
IF_GUI_GRID~GET_GRID_F49	Interface	Public	Get Column 49 Property
IF_GUI_GRID~GET_GRID_F50	Interface	Public	Get Column 50 Property
IF_GUI_GRID~GET_GRID_F51	Interface	Public	Get Column 51 Property
IF_GUI_GRID~GET_GRID_F52	Interface	Public	Get Column 52 Property
IF_GUI_GRID~GET_GRID_F53	Interface	Public	Get Column 53 Property
IF_GUI_GRID~GET_GRID_F54	Interface	Public	Get Column 54 Property
IF_GUI_GRID~GET_GRID_F55	Interface	Public	Get Column 55 Property
IF_GUI_GRID~GET_GRID_F56	Interface	Public	Get Column 56 Property
IF_GUI_GRID~GET_GRID_F57	Interface	Public	Get Column 57 Property
IF_GUI_GRID~GET_GRID_F58	Interface	Public	Get Column 58 Property
IF_GUI_GRID~GET_GRID_F59	Interface	Public	Get Column 59 Property
IF_GUI_GRID~GET_GRID_F60	Interface	Public	Get Column 60 Property
IF_GUI_GRID~GET_GRID_F61	Interface	Public	

A: Events are defined in the global class CL_GUI_ALV_GRID. The class builder, transaction SE24, can be used to view the class and all its components.



Part IV: Dialog programming

Question 28: Module Pool

What is a module pool?

A: A module pool is an ABAP program of type “M” that requires at least one transaction code in order to be executed. The transaction code usually starts a screen and its associated flow logic.

Question 29: Important TCODES

What are some of the important transaction codes to know as an ABAP developer?

A: List of important transaction codes for ABAP developers:

1. SE38 – ABAP editor
2. SE80 – Object navigator
3. SE11 – ABAP dictionary
4. BAPI – BAPI browser
5. SE24 – Class Builder
6. SE18 – Business Add-Ins (BADI)
7. SE37 – Function Builder
8. SE09 – Transport Organizer
9. SE51 – Screen Painter

Question 30: Inner and Outer Join

What is the difference in an inner join and an outer join when creating database views?

A: When creating a database view using a join the base tables of the join condition are combined to create a results table. The results table for an inner join will contain a record where a match is found in each base table for the join condition. In an outer join records are also selected from the left table that do not have a corresponding entry in the right.

Join condition on Col1 & Col2.

Base Table 1

Col1	Col2	Col3
1a	2a	3a
1b	2b	3b
1c	2c	3c

Base Table 2

Col1	Col2	Col4
1a	2a	4a
1a	2b	4b
1c	2c	4c

SAP ABAP Interview Questions

Results Table – INNER JOIN

Col1	Col2	Col3	Col4
1a	2a	3a	4a
1c	2c	3c	4c

Results Table – OUTER JOIN

Col1	Col2	Col3	Col4
1a	2a	3a	4a
1b	2b	3b	
1c	2c	3c	4c

Question 31: GUI Status

What is the purpose of a GUI status?

A: A GUI status is created in an ABAP program using the statement: SET PF-STATUS '<GUI Status Name>'.

A GUI status is composed of a menu bar, standard toolbar, application toolbar and function key settings. All of these elements combine to define the 'look and feel' that will be associated with the user interface for the ABAP program.

An ABAP program can have multiple GUI statuses associated with it. A developer can create their own GUI statuses, use existing GUI statuses or a combination of both.

If no GUI status is defined in the program then a standard user interface is displayed by default.

SAP ABAP Interview Questions

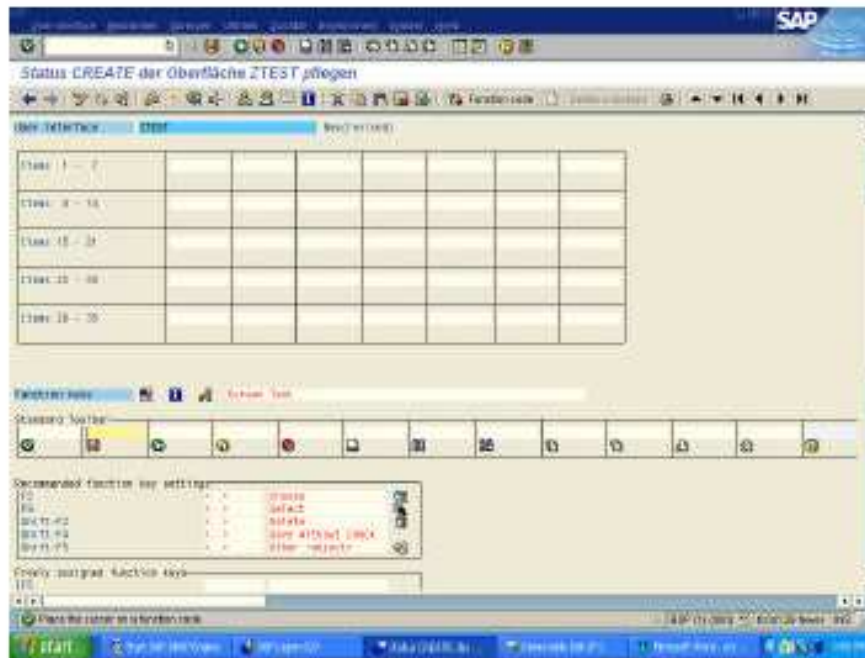


Fig 8: Creating a gui status

Question 32: SY-UCOMM

What value does the system variable SY-UCOMM contain?

A: SY-UCOMM contains the currently selected function code that is normally evaluated within an AT USER-COMMAND event.

Question 33: Dialog Execution

After a screen is displayed and before the PAI is triggered, the system automatically checks the values that a user has entered on the screen. What automatic checks are performed and in what sequence?

A: The automatic input checks are performed in the following sequence:

1. Mandatory fields are first checked to ensure the user has entered a value
2. Field format checks are performed to ensure the values entered correspond the data type defined for the screen field
3. Dictionary checks are triggered for fields defined with reference to a dictionary field. Fixed values of the domain and foreign key relationships are checked.

Question 34: Search Helps

What is the difference in an elementary and collective search help?

A: An elementary search help defines the flow of a standard input help. It is composed of a selection method that defines where to get the data that will make up the hit list, an interface consisting of search help parameters that define the exchange of data between the screen and the selection method and a dialog type that controls how the hit list will be displayed.

A collective search help is a combination of several elementary search helps giving the user different search paths. The interface parameters of the elementary search helps are assigned to the parameters of the collective search help.

Question 35: Subscreen Call

How is a subscreen call implemented within the main screen?

A: The CALL SUBSCREEN statement is placed in both the PBO and PAI of the flow logic of the main screen.



A CALL SUBSCREEN statement cannot be placed between a LOOP...ENDLOOP or CHAIN...ENDCHAIN processing block.

***Question 36: Subscreen
OK_CODE***

What is the name of the OK_CODE field for a subscreen?

A: A subscreen cannot have its own OK_CODE field. Function codes on a subscreen are linked to the OK_CODE field for the main screen.

Question 37: Table Control

In order to process a table control, what code is required in the PBO and PAI events?

A: A LOOP... ENDLOOP structure is required in both the PBO and PAI events when processing table controls.

The loop statement in the PBO transfers data from the ABAP program to the screen after each loop pass, the rest of the fields are filled in as normal at the end of PBO processing.

During PAI processing the contents of the table control are transported back to the ABAP program during the LOOP... ENDLOOP statement.



Part V: ABAP Objects

Question 38: Object Oriented Languages

What are three key concepts that a language must support in order to be a true Object Oriented Language?

A: There key concepts of all Object Oriented languages are:

Encapsulation also known as “information hiding” means the implementation of an object is hidden. The ‘caller’ of any object does not have to know how the object has been implemented

1. Polymorphism refers to the ability to have multiple forms. Objects in different classes react differently to the same message.
2. Inheritance defines the implementation relationship between classes. A subclass can inherit the structure and behavior from a super class and add its own specific behavior.
3. The power of object oriented programming stems from the use of polymorphism and inheritance. These two techniques combine to allow for reuse of individual components.

ABAP Objects refers to the complete set of object-oriented statements that has been introduced into the ABAP language. This object-oriented extension of ABAP builds on the existing language, and is

SAP ABAP Interview Questions

fully compatible with it. ABAP Objects can be used in existing programs and "conventional" ABAP in new ABAP Objects programs.



ABAP objects does not support multiple inheritance. Multiple inheritance is supported by languages such as C++ and refers to the ability of a subclass to inherit from multiple superclasses. The exclusion of multiple inheritance provides an overall simpler class model and object oriented approach.

Question 39: Class and Objects

What is the difference in a class and an object?

A: A class can be considered a blueprint or template for an object that specifies the attributes and behavior that all objects of that class will have.

Objects are instances of the class that are created by an ABAP program at run time.

Question 40: READ-ONLY

What is the purpose of the READ-ONLY addition on a DATA statement in a class definition?

A: The READ-ONLY addition is used to define a public attribute with the DATA statement that can be read from outside the class but can only be changed by methods of the class itself.

Question 41: Public versus private attributes

What is the difference in attributes defined in the public versus private section of a class?

A: Public attributes are accessible by all users of the class. Private attributes can only be accessed from within the class itself.

Question 42: Static Attributes

What is the keyword used to define static attributes?

A: CLASS-DATA is used to define static attributes. The data defined as CLASS-DATA will be the same for all instances of the class.

Question 43: Instance v. Static Method

What is the difference in an instance method and a static method?

A: An instance method can be called using an instance and can use both static and instance components in the implementation.

Instance methods are called using the following syntax:

```
CALL METHOD instancename-  
>instance_method_name
```

A static method (also referred to as a class method) can be called using the class and does not require an instance.

Static methods are called using the following syntax:

```
CALL METHOD  
classname=>class_method_name
```

Class methods are similar to instance methods but can only use static components.

Question 44: Object Instantiation

What statement is used to instantiate an object?

A: CREATE OBJECT is used to create an instance of a class in memory. During the program runtime the class is used to create the specific instance. This is also referred to as instantiation.

Objects are created and addressed using reference variables. These object references allow you to access the attributes and methods of an object.

Question 45: Garbage Collector

What is the purpose of the garbage collector?

A: The garbage collector removes objects from memory that no longer have a reference pointing to it. The garbage collector is an automatic system routine that will delete the object from memory and release the memory.

Question 46: Constructor Method

When is a constructor method executed?

A: A constructor method is executed only once per instance and is automatically called at runtime when the CREATE OBJECT statement is executed.

Question 47: Multiple Inheritance

Is multiple inheritance possible in ABAP Objects?

A: No, only single inheritance is supported much like java. Single inheritance means that a subclass can only have one direct superclass.



Multiple inheritance can be simulated using interfaces.

Question 48: Redefined Method

What is a redefined method and what are some of the characteristics of this type of method?

A: A redefined method is a method in a subclass that provides a new definition of an inherited method from a superclass in order to provide a more specialized implementation in the subclass.

The redefined method will have the same name and interface, but will have a new implementation.

Redefined methods cannot have been defined as final methods in the superclass. Defining a method as final indicates that the method cannot be overridden or redefined.

A constructor method cannot be redefined as it is implicitly a final method.

The method declaration and implementation in the superclass is not affected. The redefined method in the subclass “hides” the original method in the superclass. All references to an object in the subclass will use the redefined method. The system checks for methods first in the current class and then ‘up the hierarchy’.

Only an inherited public or protected instance method can be redefined. Static methods cannot be redefined.



The reference to `SUPER->` in the redefined method is to call the original method of the superclass. This allows you to use the existing functionality in the superclass method in the subclass.

Question 49: Interfaces

What is the purpose of an interface and how is it implemented in ABAP objects?

A: An interface is a declaration of a set of methods with no information given about the implementation of those methods.

Interfaces are used to define common functionality that can be used by different classes. Interfaces are implemented in classes by listing the interface name in the definition part of the class. Example.

```
CLASS lcl_customer DEFINITION.
```

```
    Public section.
```

```
        INTERFACES <interface_name>.
```

```
ENDCLASS.
```

An interface can only be implemented publicly. The methods defined in the interface are implemented in the class. All methods that are defined in the interface must be present in the implementing class.

Much like a class, an interface defines methods. However, unlike a class, an interface never implements methods; instead, classes that implement the interface provide the implementation to the methods defined by the interface.



An interface can be accessed using an object reference as long as that object's class implements the interface. This is done using the interface resolution operator '~' as follows:

```
<objectname>-  
><interfacename>~<methodname>
```


Question 50: Abstract Classes

What is an abstract class?

A: An abstract class cannot be instantiated. It specifies one or more abstract methods that must be implemented in a subclass. It is used as a means to enforce a uniform interface in subclasses.



Part VI: Data Transfer

Question 51: Data Transfer

Describe three standard data transfer techniques supported in the SAP system?

A: In order to load data into an SAP system there are several techniques that can be used. Three of the most common data transfer techniques are as follows:

1. Batch Input, also known as Batch Data Communication (BDC) – This approach is used extensively in ABAP programming and remains a popular choice when no other method exists. The online transaction process is simulated and the data is transferred exactly as if it was entered online. A batch input session is created containing all of the data and then processed using the online transaction in SAP to transfer the data. Once a BDC load program is created any changes to the screens through the IMG or otherwise could cause the BDC to fail. This is one of the disadvantages of the BDC technique.
2. Direct Input – the data in an input file is transferred directly to the SAP database. No screens are involved with direct input making it more efficient than batch input. The direct input program uses function calls to load the data. This method is not available for all data objects.

SAP ABAP Interview Questions

3. Business Application Programming Interfaces (BAPIs) – are standardized APIs that provide external access to SAP objects. BAPIS are defined in the business object repository as methods of an SAP object. When using a BAPI as an interface to SAP, the data to be loaded must be in an Idoc format.

Question 52: BDCDATA

What is the format of the BDCDATA table?

A: The data is structured within the BDCDATA table according to the screens in the transaction that is being simulated.

Each screen that will be processed in the BDC session will have a separate record in BDCDATA identified using the fields PROGRAM, DYNPRO and DYNBEGIN.

The screen record will be followed by a separate record for each field name / field value pair that can be entered on the screen. These records are identified using the FNAM and FVAL fields. The OK code field for each screen will be identified as the BDC_OKCODE field in the BDCDATA table.

PGM	DYNPRO	DYNBEGIN	FNAM	FVAL
<prg name>	<screen no>	'X' for begin of screen	<field name>	<field value>
			<field name>	<field value>
			<field name>	<field value>
			<field name>	<field value>
			BDC_OKCODE	/oo

Question 53: What is LSMW?

A: LSMW is the legacy system migration workbench, a tool that can be used for data conversion from legacy systems into SAP. Data can be transferred using standard transfer programs, IDOCs, BAPIs or creation of a recording using the transaction recorder and generating a batch input session.

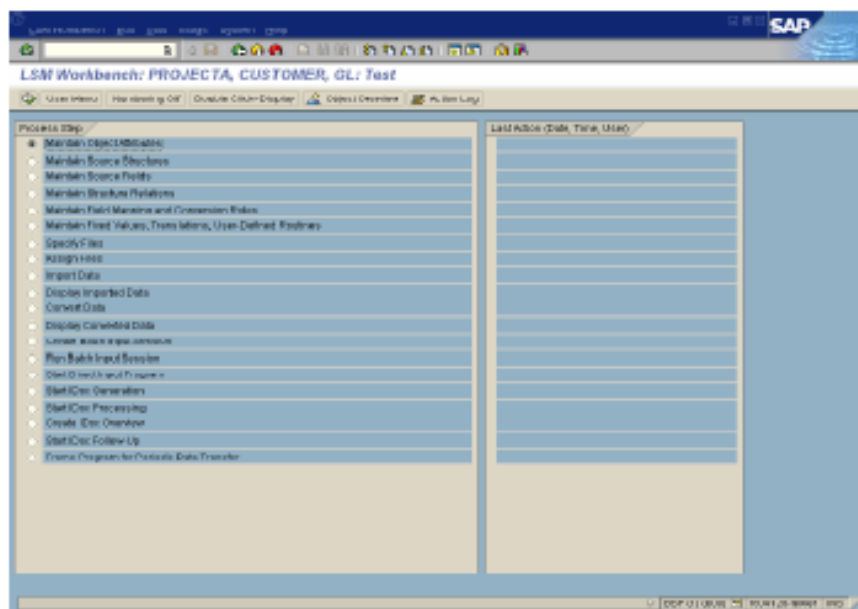


Fig 9: LSMW Workbench

Question 54: Batch Input

What are the three function modules that are used when creating batch input sessions

A: The three SAP function modules used for creating batch input sessions are:

1. **BDC_OPEN_GROUP** – is used to create the new batch input session and will contain general data that is used for the entire session.
2. **BDC_INSERT** – is used to actually insert all of the data into a session that is required for a transaction.
3. **BDC_CLOSE_GROUP** – is used to close the batch input session after all of the data has been inserted and the session is ready for processing.



Part VII: BASIS

Question 55: Client Definition

What is the definition of a client in SAP?

A: A client is an independent unit in the SAP system from an organizational point of view.

Each client has its own data environment. The client concept allows for the joint operation of several enterprises in one system that are independent of each other in business terms.

During each user session, a client is chosen from the main login screen in order to specify which environment a user will access.

Question 56: Tiered Architecture

Explain what a three-tier system configuration is?

A: The core layers of any business application are the presentation layer, application layer and the database layer.

There are several ways to configure these layers in a client/server environment. In a three-tier configuration, individual servers are used for each layer.

ABAP programs run at the application layer. The programs work with data from the database layer. The database layer is where the data is managed. Data includes master, transactional and metadata. The presentation layer contains the user interface.

Question 57: Dispatcher

What is the role of the dispatcher in an SAP system?

A: The dispatcher assists in controlling the resources for the SAP applications. The dispatcher distributes transaction load to the work processes, connects to the presentation layer and organizes communication.

Question 58: Work Processes

What are some of the different types of work processes on an SAP system?

A: Work processes may include:

1. Dialog (D)
2. Update (V)
3. Lock (E)
4. Background (B)
5. Spool (S)



Every dispatcher must have at least 2 dialog work processes, at least 1 spool work process and at least 1 update work process, at least 2 background work processes and only one lock work process.

Question 59: RFCs

What is a Remote Function Call (RFC)?

A: RFC is a communication interface that provides the ability to call a function remotely from another SAP system or Non SAP system.

Each function module has a defined interface through which data, tables and return codes can be exchanged. The only difference between calling a function module remotely and locally is an extra parameter is provided remotely that specifies the target server on which the function should be executed.

There are 3 types of RFCs:

1. Synchronous RFC call: The calling program stops until the function module executes on the target server and returns results.
2. Asynchronous RFC call: The calling program runs in parallel with the function module on the target server. The target system must be available.
3. Transactional RFC call (tRFC) – Several function modules are grouped into a transaction. They are processed on the target system in the sequence in which they

SAP ABAP Interview Questions

are called. The target. System does not have to be available



RFC connections are maintained using transaction SM59.

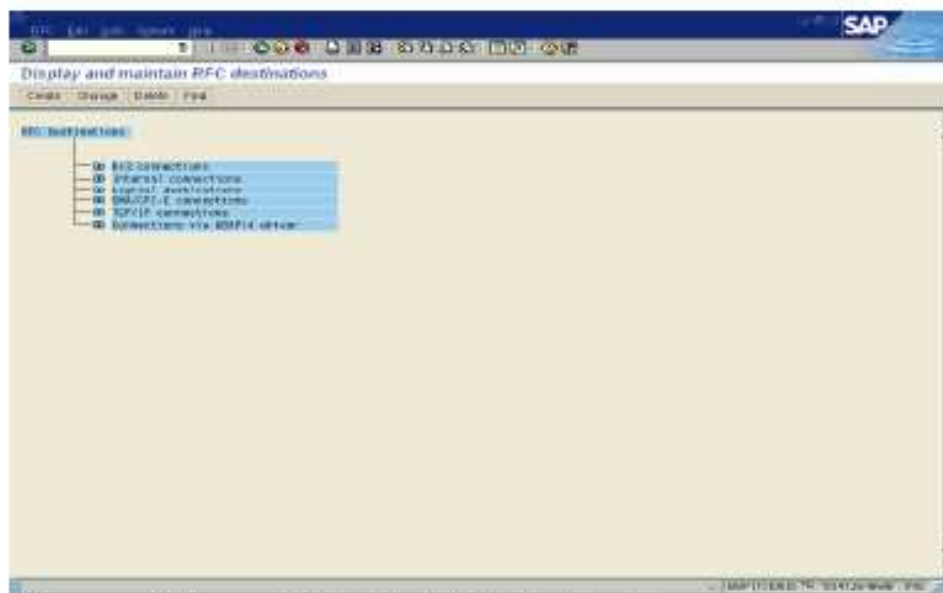


Fig 10: SM59 – Maintain RFC destinations

Question 60: SAPNet

What are some of the tasks that can be performed using SAPNet?

A: SAPNet is an internet portal provided by SAP where customers and partners can:

1. Search the notes database,
2. Write messages to SAP relating to system problems,
3. Request developer keys
4. Allow an SAP employee to access your system for problem investigations.

Question 61: CCMS

What is CCMS and what are some of the useful functions?

A: CCMS is the Computer Center Management System. Some of the functions that can be performed using CCMS are as follows:

1. Starting and Stopping SAP,
2. System monitoring and analysis,
3. Processing and controlling background jobs,
4. Dynamic load balancing.

Question 62: Transport Change Request

If your SAP landscape is composed of a DEV server and PRD server and you need to transport a change request from DEV client 111 to DEV 131 (i.e. transport in the same server). How can this be accomplished?

A: Transaction SCC1



Question 63: Internet outages

We have recently experienced some internet outages caused by our ISP. Although we have resolved our ISP problems, the users refuse to believe that losing the internet connection was to blame for the SAP problems. How can we ensure that this was an internet problem and not an SAP problem?

A: Open a maintenance window, connect to the machine and kill the ISP connection and see if this kills your connection. Repeat at will until they believe that the ISP was the problem.

Question 64: OK Codes

What is an OK code? What are the differences between an OK code to a Transaction code?

A: An OK Code is used by a program to execute a function for example after a pushbutton has been clicked. A transaction code is a “shortcut” that helps a user run a program.



T-Codes are stored in table TSTC

Question 65: Transaction Codes

Where are t-code name and program values stored?
How can I find a list of all the t-codes in the SAP system?

A: You can use transaction st11 to view Table TSTC. You can define a new t-code using transaction se93.

Question 66: STMS Importing

How can one disable the "Import All" button on STMS for the queues?

A: Login to your Transport Domain Controller.
Run STMS->Overview->System.
Choose the System you want to disable import all.
Go to Transport Tool tab.
Add/Create Parameter "NO_IMPORT_ALL" set its value to 1.

Question 67: Work Processes

What are the different types of work processes in R/3?

A: Dialog (D), Update (U), Enqueue (E), Background(B), and Spool(S).

Question 68: Patch Level for GUI

How can you confirm the patch level for SAP GUI?

A: Log into SAPGUI, and hit Alt-F12 -> About

Question 69: Web help installation

Why can web help be considered easier to install than CHM, the standard delivered SAP help?

A: Web help is easier than trying to get every remote user connected to a network share. If you don't want to use SAP's help site, you can setup your own website to do it. The best route depends on your business' network structure.



Question 70: Web Resources

If you can't find the answer to a question, what are some website you can visit to find the answer?

A:

OSS notes: <http://service.sap.com/notes>

SAP help: <http://help.sap.com>

Google: <http://www.google.com>

Question 71: Instance installation

We want to install another new instance on same development box. Is this possible? What are some of the important considerations?

A: Yes, it is possible to have more than one instance on a single box. The key is to use a different SID and a different system number. It is also important to note that for a 64 bit SAP kernel, SAP recommends a 20 GB swap space for 1st instance and 10 GB per each additional instance.

Question 72: Client copy

What is the difference between a client copy and client refresh?

A: Some times these are the same, For example, if you are performing a copy to an existing client for the express purpose of updating the data, then it is called Client Refresh.

If you are copying to a newly created client then it is more appropriate to say “client copy.”

Another way of thinking about this is that a client copy from production to a QA server or from production to DEV server is really a client refresh.



Question 73: Table T000

What is the purpose of table T000?

A: Table T000 contains a list of defined clients, which you can maintain with transaction SCC4.

Question 74: Table USR02

What is the purpose of table USR02?

A: This table stores User IDs and passwords.

Question 75: Passwords

How do you create a password exception list?

A: Place the answers in table USR40.

Question 76: Table TADIR

What is the purpose of table TADIR?

A: Table TADIR contains object directory entries.

Question 77: Table TDEV

What is the purpose of table TDEV?

A: Table TDEV contains development classes and packages.

Question 78: Change Requests

What are the transaction codes associated with changing requests, request headers, or request object lists?

A: The following tables hold information about change request sets.

E070	Change request headers
E071	Change request object lists

Question 79: User Access

How can you get a list of the users with development access on a particular system?

A: Table DEVACCESS

Question 80: Transport object keys

Where can you find a list of object keys included in a transport?

A: E071K Object keys contained within transports

Question 81: Transport in progress

How can you tell if a transport entry is in the process of being imported?

A: Check table TRBAT

Question 82: Repaired Objects

How can you find a list of objects that have been repaired in the system?

A: ADIRACCESS List of repaired objects and their access keys

Question 83: Disable Multiple Logins

How do you Disable Multiple Logins in the Same Client?

A: To disable multiple user logins within the same client implement this parameter in the instance profile:

login/disable_multi_gui_login = 1

If you do not use this parameter in your system, users have the ability to ignore the warning window at the time they try to login to the same client.

Activating this parameter in your system will make you look good if you get audited!

How about exceptional logins?

In case you're wondering how to allow multiple logins for certain key users you can implement parameter login/multi_login_users. You can list the user IDs that should be ignored if the parameter above is active in your system.

Question 84: Locked Transactions

How can you View Locked Transactions?

A: As you know, you can lock/unlock transaction codes via SMO1. But, how do you go about viewing the transactions that are locked in the system? You need to look in field CINFO, table TSTC.

Within SAP, you can use either SE11 or SE16 to browse the table contents. Make sure you enter "A0" as the "HEX01 data element for SYST" starting value and "A9" as the ending value. This will list all the transactions locked in the system.

Note: The CINFO field description is "HEX01 data element for SYST".



Part VIII: ABAP Development

Question 85: Chained Statements

What is a chained statement in the ABAP language?

A: Consecutive statements with the same initial keyword can be condensed into a single chained statement. The initial part of the statement containing the keyword must be followed by a colon. All elements that follow the colon must be separated by a comma.

Example:

```
WRITE:/ <data element 1> ,  
        <data element 2>,  
        <data element 3>.
```



The system considers each individual part of a chained statement to be a complete and independent statement.

Question 86: Defining Constants

How is a constant defined in ABAP?

A: A constant is defined using the keyword CONSTANT as shown in the following example:

```
CONSTANTS PI TYPE P DECIMALS 2 VALUE  
'3.14'.
```

Question 87: Return Codes

There are several ABAP statements that create a return code depending on detection of an exception. Where is this return code stored?

A: These return codes are stored in the system variable, sy-subrc. A return code of 0 indicates that the statement executed successfully.

Question 88: SELECT SINGLE

When would you use a SELECT SINGLE statement?

A: The SELECT SINGLE statement is used to ensure that only one row of data is read from the database. The search is terminated once that row is found. SELECT SINGLE will produce better performance for single record access rather than using a SELECT loop as long as a value is supplied for all key fields in the WHERE clause

***Question 89: SELECT
SEMANTICS***

During each loop pass of a SELECT loop, what system field will contain the number of records read?

A: The number of records read during each loop pass of a SELECT loop would be stored in the system field: sy-dbcnt.

Question 90: Array Fetch

What is an Array fetch?

A: An array fetch is a SELECT statement that uses the addition “`INTO TABLE <internal table name>`” which causes the system to copy the selected data directly into an internal table.



An Array fetch is not a logical loop and therefore does not require an `ENDSELECT` Statement.

Question 91: Dialog Data Transport

When does data transport take from an ABAP program to a screen and from the screen back to the ABAP program?

A: Data transport between screens and ABAP programs is automatically handled by the system.

The first data transport occurs after all PBO modules have been executed and prior to a screen being displayed. The system will transport the contents of the ABAP work area in the program into the corresponding screen fields.

The second data transport trigger is once all data entry is completed by the user and before the first PAI module is executed. The system will copy the contents of the screen fields back to the corresponding ABAP work area in the program.

Question 92: Domain v. Data Element

What is the difference in a domain and a data element?

A: A domain describes the value range of a field and is assigned to a data element. All fields that use the data element will have the value range of the domain. The value range is defined by specifying the type and length of the domain. Fixed value intervals can also be defined for the domain and are automatically used as check values on screen entries.

A data element describes the business meaning of a domain. Information about the meaning of a field such as its label is assigned to the data element and is accessible from all screen fields that use the data element.

Question 93: Table Buffering

What is the purpose of table buffering?

A: Table buffering is used to increase performance when reading data from a database table. Data from a buffered table is read from a local buffer on the application server thus reducing database access.



There are three different types of table buffering that can be used in SAP:

Full buffering – all records of the table are loaded into the buffer when a record in the table is accessed

Generic buffering – all records with the same key fields are loaded into the buffer when the when a record in the table is accessed.

Single record buffering – the specific record that was accessed is loaded into the buffer.

The screenshot shows the SAP Dictionary: Technische Einstellungen pflegen (Technical Settings Maintenance) screen. The main fields are:

- Name:** Z0010013
- Description:** Erweiterung
- Language:** EN
- Date Format:** DD.MM.YYYY
- Time Zone:** UTC+01:00

Below these fields are several sections for configuration:

- Local Storage Parameters:** Includes fields for Data Source (Z0010013) and Data Category (Z0010013).
- Buffering:** Includes checkboxes for Buffering (checked), Buffering Interval (set to 10), and Buffering Threshold (set to 100).
- Log File Settings:** Includes checkboxes for Single Instance (checked), Create Log File (checked), and Full Path (checked).

The bottom of the screen shows the SAP logo and the text "SAP AG".

- 118 -

Question 94: CONDENSE

What is the purpose of the CONDENSE statement?

A: CONDENSE is used in processing strings in ABAP. The CONDENSE statement will remove all spaces from a given string.

Question 95: CLEAR and FREE

What is the difference in using the CLEAR statement and the FREE statement with an internal table?

A: The CLEAR statement will delete the contents of the internal table. The FREE statement will delete the contents of the internal table and release the memory that was used by the internal table.

Question 96: Passing Parameters

What are the different ways to pass data from the main ABAP program (actual parameters) to a subroutine or FORM statement (formal parameters)?

A: Data is passed from a main program to a subroutine using parameters. The data in the main program would be referred to as the actual parameters. The data in the subroutine (FORM statement) would be known as the formal parameters.

There are three different ways to pass parameters to subroutines:

Pass by reference – parameters are listed after the ‘USING’ or ‘CHANGING’ keyword without the keyword ‘VALUE’. During the call to the subroutine the address of the actual parameter is passed to the formal parameter. The subroutine works with the actual field from the main program. Changing the formal parameter actually changes the actual parameter.

Example:

```
FORM <subroutine name>  
USING Parameter1
```

SAP ABAP Interview Questions

CHANGING Parameter2

Pass by value - parameters are listed after the 'USING' keyword and include the keyword 'VALUE'. The subroutine works with a copy of the actual parameter. Changes to the formal parameter have no impact on the actual parameter

Example:

FORM <subroutine name>

USING VALUE(Parameter1)

Pass by value and result- parameters are listed after the 'CHANGING' keyword and include the keyword 'VALUE'. The subroutine works with a copy of the actual parameter. Upon successful completion of the subroutine, the formal parameter is copied to the actual parameter.

Example:

FORM <subroutine name>

CHANGING VALUE(Parameter1)

Question 97: Assigning Attributes

What are two ways of assigning attributes to screen fields when using the Screen Painter?

A: Field attributes for screens can be adopted from the dictionary or from already defined data objects in the ABAP program.

Question 98: Function Group

What is a function group?

A: A function group is an ABAP program of type 'F' that contains function modules. Function modules that work with the same data objects can be combined to form the function group.



Each function group can contain data objects, subroutines and screens which can be used by the function modules.

INDEX

ABAP Development	111
ABAP objects	64
Abstract Classes	78
Adding Fields	23
ALV control	47
ALV Grid	3, 4, 8, 39, 45, 48
events	48
ALV Grid Control	45
append structure	23
Array Fetch	117
Asynchronous RFC	91
AT LINE-SELECTION	38
attributes	
public versus private	67
Attributes	4, 5, 68, 126
AUTHORITY-CHECK	15
Authorization	15
Background	90, 100
BADI	24
BAPI	51, 81
BAPIS	84
BASIS	86
Batch Input	80, 85
BDC_CLOSE_GROUP	85
BDC_INSERT	85
BDC_OPEN_GROUP	85
BDCDATA	82
Buffering	

SAP ABAP Interview Questions

tables	120
CCMS.....	5, 94
Chained Statements.....	112
Change Request	95
Change Requests	106
Changing SAP	21
CINFO	110
Class.....	65
CLEAR.....	123
<i>Client copy</i>	103
Client Definition	87
Code Example	46
COLLECT Statement	42
collective search help	58
CONDENSE	122
Constants	
defining	113
Constructor.....	72
constructor method.....	74
Control Structure	
AT NEW	43
customizing include	23
Data Element.....	5, 119
Data Transfer.....	80
database view.....	52
default selection screen.....	34
DEQUEUE	
lock objects	11
DEVACCESS	107
Dialog.....	90, 100
data transport.....	118
Dialog Execution.....	57

SAP ABAP Interview Questions

Direct Input	80
dispatcher	90
Dispatcher	89
Domain	5, 98, 119
Enqueue	100
ENQUEUE	
lock objects	11
Exclusive lock	12
Field Descriptions	47
FREE	123
Function Module	
Exit	18
functional area	26
Garbage Collector	71
GUI Status	54
GuiXT	20
Icon	
outputting	28
Inheritance	63
inner join	52
input help	58
<i>Instance</i>	102
instance method	69
Interfaces	76
Internal Tables	40
Internet outages	96
ISP	96
Join	
inner and outer	52
List Generation	27
List Processing	29
Lock	90
lock object	11

SAP ABAP Interview Questions

Locked Transactions.....	110
Logical Database.....	35
LSMW.....	84
Memory	
exchanging	14
Method	
instance v. static	69
Modification Assistant	17
Modifications	16
Module Pool.....	50
multiple inheritance.....	73
Multiple Inheritance	73
Multiple Logins	109
Object Instantiation.....	70
object keys.....	107
Object Oriented Languages.....	63
Objects	65
OK Code	97
OK_CODE.....	4, 60
Open SQL	37
outer join	52
PAI	57
parameters	
pass by reference	124
Parameters	
passing.....	5, 124
password exception list	105
passwords	104
<i>Patch Level</i>	100
PBO	
table control.....	61
Polymorphism.....	63

SAP ABAP Interview Questions

Product Category.....	<i>See</i> Material Group
READ-ONLY.....	66
redefined method.....	74
Repaired Objects.....	108
Return Codes	114
RFC	91
RFC connections	92
RFC destinations	92
SAP Query.....	26
SAP standard functionality	
changing	21
SAPNet.....	93
SE09	51
SE11.....	51, 110
SE18.....	51
SE24.....	51
SE37.....	51
SE38	51
SE51.....	51
SE80	51
se93.....	98
Search Helps.....	58
SELECT	
semantics	116
SELECT SINGLE.....	115
Selection Screen.....	34
SELECT-OPTION.....	31
SELECT-OPTIONS	30
Shared lock.....	12
Spool.....	90, 100
st11	98
static method	69

SAP ABAP Interview Questions

<i>STMS</i>	98
Subscreen Call.....	59
subscreens	
for transaction variants.....	19
superclass	74
SY-LINSZ.....	29
SY-LISTI	29
SY-LSIND	29
Synchronous RFC	91
SY-UCOMM.....	56
<i>Tooo</i>	104
Table Control	4, 61
TADIR.....	105
TDEVC.....	106
three-tier system configuration	88
Tiered Architecture	88
<i>Transaction Codes</i>	98
Transaction Variants	19
Transactional RFC	91
Transport.....	95, 98, 107, 108
TRBAT.....	108
TSTC	97
Update	90, 100
function module	13
User Access.....	107
user group.....	26
User ID	104
<i>USR02</i>	104
<i>Web help</i>	101
Work Processes.....	90, 100

ABOUT THE AUTHOR

Mr. Fewer has worked in the Information Technology industry for 13 years.

Mr. Fewer is currently a consultant with Fujitsu Consulting Canada. He joined Fujitsu's SAP practice in 1998 and has been working with SAP R/3 since then, attaining two SAP R/3 Academy certifications: ABAP and Financials/Controlling. Mr. Fewer has worked on diverse projects for both public and private sector and has held a variety of roles including developer, tester, business analyst, support analyst, quality assurance analyst, development lead and senior systems architect. He has been involved in implementing various modules such as Financials, Controlling, Materials Management, Sales and Distribution, Plant Maintenance, Production Planning, Project Systems and Human Resources and has experience in various tools and technologies such as ABAP, ITS, SAPSCRIPT, Java, BSP, XI, ALE and LSMW.

Attention SAP Experts

Writing a book can be the best thing for your career.

Have you ever considered writing a book in your area of SAP? Equity Press is the leading provider of knowledge products in SAP applications consulting, development, and support. If you have a manuscript or an idea of a manuscript, we'd love to help you get it published!

Please send your manuscript or manuscript ideas to jim@sapcookbook.com – we'll help you turn your dream into a reality.

Or mail your inquiries to:

Equity Press Manuscripts
BOX 706
Riverside, California
92502

Tel (951)788-0810
Fax (951)788-0812

50% Off your next SAPCOOKBOOK order

If you plan of placing an order for 10 or more books from www.sapcookbook.com you qualify for volume discounts. Please send an email to books@sapcookbook.com or phone 951-788-0810 to place your order.

You can also fax your orders to 951-788-0812

Interview books are great for cross-training

In the new global economy, the more you know the better. The sharpest consultants are doing everything they can to pick up more than one functional area of SAP. Each of the following Certification Review / Interview Question books provides an excellent starting point for your module learning and investigation. These books get you started like no other book can – by providing you the information that you really need to know, and fast.

SAPCOOKBOOK Interview Questions, Answers, and Explanations

ABAP - SAP ABAP Certification Review: SAP ABAP Interview Questions, Answers, and Explanations

SD - SAP SD Interview Questions, Answers, and Explanations

Security - SAP Security: SAP Security Essentials

HR - mySAP HR Interview Questions, Answers, and Explanations: SAP HR Certification Review

BW - SAP BW Ultimate Interview Questions, Answers, and Explanations: SAP BW Certification Review

SAP SRM Interview Questions Answers and Explanations

Basis - SAP Basis Certification Questions: Basis Interview Questions, Answers, and Explanations

MM - SAP MM Certification and Interview Questions: SAP MM Interview Questions, Answers, and Explanations

SAP ABAP Interview Questions

SAP BW Ultimate Interview Questions, Answers, and Explanations



Key Topics Include

- The most important BW settings to know
- BW tables and transaction code quick references
- Certification Examination Questions
- Extraction, Modeling and Configuration
- Transformations and Administration
- Performance Tuning, Tips & Tricks, and FAQ
- Everything a BW resource needs to know before an interview

mySAP HR Interview Questions, Answers, and Explanations



Key topics include:

- The most important HR settings to know
- mySAP HR Administration tables and transaction code quick references
- SAP HR Certification Examination Questions
- Org plan, Compensation, Year End, Wages, and Taxes
- User Management, Transport System, Patches, and Upgrades
- Benefits, Holidays, Payroll, and Infotypes
- Everything an HR resource needs to know before an interview

SAP ABAP Interview Questions

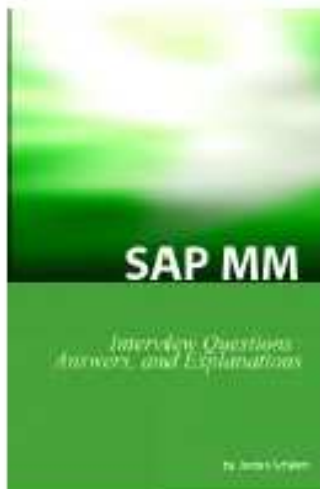
SAP SRM Interview Questions, Answers, and Explanations



Key Topics Include

- The most important SRM Configuration to know
- Common EBP Implementation Scenarios
- Purchasing Document Approval Processes
- Supplier Self Registration and Self Service (SUS)
- Live Auctions and Bidding Engine, RFX Processes (LAC)
- Details for Business Intelligence and Spend Analysis
- EBP Technical and Troubleshooting Information

SAP MM Interview Questions, Answers, and Explanations



- The most important MM Configuration to know
- Common MM Implementation Scenarios
- MM Certification Exam Questions
- Consumption Based Planning
- Warehouse Management
- Material Master Creation and Planning
- Purchasing Document Inforecords

SAP ABAP Interview Questions

SAP SD Interview Questions, Answers, and Explanations



- The most important SD settings to know
- SAP SD administration tables and transaction code quick references
- SAP SD Certification Examination Questions
- Sales Organization and Document Flow Introduction
- Partner Procedures, Backorder Processing, Sales BOM
- Backorder Processing, Third Party Ordering, Rebates and Refunds
- Everything an SD resource needs to know before an interview

SAP Basis Interview Questions, Answers, and Explanations



- The most important Basis settings to know
- Basis Administration tables and transaction code quick references
- Certification Examination Questions
- Oracle database, UNIX, and MS Windows Technical Information
- User Management, Transport System, Patches, and Upgrades
- Backup and Restore, Archiving, Disaster Recover, and Security
- Everything a Basis resource needs to know before an interview

SAP Security Essentials



- Finding Audit Critical Combinations
- Authentication, Transaction Logging, and Passwords
- Roles, Profiles, and User Management
- ITAR, DCAA, DCMA, and Audit Requirements
- The most important security settings to know
- Security Tuning, Tips & Tricks, and FAQ
- Transaction code list and table name references