

What is debugger?

→ The ABAP debugger is a tool of the ABAP with which you can stop running program and then execute them line-by-line or section-by-section.

Types of Debugging techniques in ABAP-

① classic debugger

② New ABAP debugger (enhanced version of the classic debugger)

③ Debugger of web dynpro applications.

Shortcut used for ABAP debugger-

① F5 - set a breakpoint at the current line of code.
(line by line debugging)

② F6 - step over the current line of code without stepping into any function or method calls.
(Execute function module or perform with one shot)

③ F7 - step into the current function or method call.
(Return cursor)

④ F8 - skip the current function or method call.
(directly execute entire program)

Breakpoints-

A breakpoint is a signal at a particular point in the program that tells the ABAP runtime processor to interrupt processing and start the debugger. The debugger is activated when the program reaches this point.

Types of Breakpoints-

① Static Breakpoint

② Dynamic Breakpoint

① Static Breakpoint

Static Breakpoints are always user independent if there is no specification of a user name. Once a user has inserted the statement Break-point in an ABAP program, the system always interrupts the program at that point for that user for the user name.

② Dynamic Breakpoint

Dynamic breakpoints are user-specific. We use them if you only want the program to be interrupted when you run it yourself, not when it is being executed by other users.

Dynamic breakpoints are more flexible than static breakpoints because you can deactivate or delete them at runtime.

Watchpoints

A ~~Gen~~ watchpoint is a conditional breakpoint that is only defined in the ABAP Debugger during a running debug session.

It is one of the runtime utilities provided to stop and debug ABAP programs. When debugging ABAP code, you can use watchpoints to track the value of individual ABAP variables.

The ABAP debugger stops as soon as the value of a watched variable has changed.

Watchpoint Features

- ① Watchpoints can be created while ABAP debugger is on and are removed automatically when that debug session is terminated.
- ② Logical conditions may be specified for watchpoints.
- ③ Like breakpoints, watchpoints may be deleted as needed.
- ④ Once a watchpoint is reached, it is indicated on the program statement and corresponding alert is given.

Procedure -

① setting watchpoints for currently selected field in the source editor.

Ⓐ In the ABAP source code editor of the debug perspective, position the cursor on the field in question.

Ⓑ open the context menu & select the option set watchpoint.

② setting watchpoints in the variables view -

Ⓐ In the Variables view of the debug perspective, select the node for the variable in question.

Ⓑ open the context menu and select the option set watchpoint.

③ specifying condition for watchpoints -

Ⓐ open the Breakpoints view of the debug perspective.

Ⓑ select the node for the watched variable.

Ⓒ Enter a valid condition in the corresponding entry field.