

ABAP Development: abapOpenChecks - Open source checks SAP Code Inspector / ATC

Posted by [Lars Hvam](#) Mar 8, 2015

Over the last 6 months I've been working on a set of custom checks for code inspector, the project is available on [github](#).

Currently it contains the following general purpose checks, some complex and some simple.

- CHECK_01 - IF in IF
- CHECK_02 - EXIT outside of loop
- CHECK_03 - TRY without CATCH
- CHECK_04 - Line length
- CHECK_05 - 7 bit ASCII
- CHECK_06 - Check for use of pretty printer
- CHECK_07 - Functional writing style for CALL METHOD
- CHECK_08 - Obsolete statement
- CHECK_09 - Tab instead of spaces
- CHECK_10 - Use icon_ constants
- CHECK_11 - Max one statement per line
- CHECK_12 - Specify SORT order
- CHECK_13 - Sequential blank lines
- CHECK_14 - Commented code
- CHECK_17 - Definitions in top of routine
- CHECK_18 - Empty branch
- CHECK_19 - Use LINE OF
- CHECK_20 - Bad indentation
- CHECK_21 - Unused FORM parameter
- CHECK_22 - Conditions contain identical code
- CHECK_23 - CHECK outside of loop
- CHECK_24 - Identical code blocks
- CHECK_25 - Selection screen data not referenced statically
- CHECK_26 - No direct changes to standard tables

You can configure [code inspector](#) or [ATC](#) to use these checks in your default variant.

	D..	...	E...	Tests	Error	Warn...	Infor...
				List of Checks	3198	0	0
				abapOpenChecks	3198	0	0
				No direct changes to standard tables	0	0	0
				Selection screen data not referenced statically	2	0	0
				Identical code blocks	697	0	0
				CHECK outside of LOOP	561	0	0
				Conditions contain identical code	9	0	0
				Unused FORM parameter	0	0	0
				Bad indentation	861	0	0
				Use LINE OF	552	0	0
				Empty branch	14	0	0
				Definitions in top of routine	59	0	0
				LEAVE is obsolete	0	0	0
				IS REQUESTED is obsolete	0	0	0
				Commented code	86	0	0
				Sequential blank lines	16	0	0
				Specify SORT order	71	0	0
				Max one statement per line	17	0	0
				Use icon_ constants	0	0	0
				Tab instead of spaces	29	0	0
				REFRESH is obsolete	89	0	0
				Functional writing style for CALL METHOD	0	0	0
				Check pretty printer use	4	0	0
				7 bit ASCII	38	0	0
				Line length	7	0	0
				TRY without CATCH	0	0	0
				EXIT outside of LOOP	75	0	0
				IF in IF	11	0	0

The checks are provided "AS IS", some might not fit your development requirement, and some might be slow or not even work.

This is a box of chocolates; take the ones you like 😊. Some of the checks contains settings so it is possible to change to fit your requirements.

Feel free to submit pull requests or open issues. More checks will be added over time.

<https://github.com/larshp/abapOpenChecks>

2790 Views Tags: [abap](#), [community_projects](#), [open_source](#), [code_inspector](#), [abap_test_cockpit](#), [atc](#), [github](#)



Praveer Kumar Sen

Mar 26, 2015 11:20 AM

nice post Lars, really help full and contains lots of information to reduce extra works after code writing.



Happy Learning.


Praveer.



Lars Hvam in response to [Peter Inotai](#) on page 2


Mar 9, 2015 5:35 PM

One note on message priorities for abapOpenChecks, the "Management of" -> "Message Priorities" is not supported. Instead its possible to define the message priority in the check variant, this allows users to have different variants with different message priorities for the same check.


 [Peter Inotai](#) in response to [Lars Hvam](#) on page 3

Mar 9, 2015 5:27 PM

You're right. Changing priorities is not supported for "Extended Program Check" (not even in 7.40 system). I

didn't notice it before. 


Peter

 [Lars Hvam](#) in response to [Peter Inotai](#) on page 4

Mar 9, 2015 5:15 PM

These 2 checks are part of "Extended Program Check" -> "Programming Guidelines".

There are some of the checks in "Programming Guidelines" that I personally don't like, whereas I like the CHECK / EXIT checks. Having these checks in abapOpenChecks allows for enabling and changing the message priority, as changes of message priorities for "Extended Program Check" is not supported(see note 1887255). I'll keep the note on the wiki page for the 2 checks so people can choose the approach they like.

 [Peter Inotai](#) in response to [Peter Inotai](#) on page 4

Mar 9, 2015 3:03 PM


Found also an example for "EXIT statement outside a loop":

2	Extended Program Check	Exception handling is empty
2	Extended Program Check	EXIT statement outside a loop
2	Extended Program Check	EXIT statement outside a loop
2	Extended Program Check	EXIT statement outside a loop
2	Extended Program Check	EXIT statement outside a loop
2	Extended Program Check	Non-class-based exception used



Details


	Location / Finding
Package	/EBY/PDBO_USER_MANAGEMENT
Class Pool	/EBY/CL_PDBO_USER_SAP
Method	/EBY/IF_OBJECT~EQUALS
Line Number	7
Check Title	Extended Program Check
Check Message	EXIT statement outside a loop
Priority	Priority 2
Found on	09.03.2015 14:37:09
Only use EXIT in loops. Otherwise use RETURN.	
Display Object	Exemptions allowed only for central results

 [Peter Inotai](#) in response to [Lars Hvam](#) on page 4

Mar 9, 2015 2:56 PM

Great, that was fast 😊

Peter

 [Lars Hvam](#) in response to [Peter Inotai](#) on page 4

Mar 9, 2015 12:58 PM

Thanks, I've added your information in the Wiki page for CHECK_23, [ZCL_AOC_CHECK_23](#) · [larshp/abapOpenChecks Wiki](#) · [GitHub](#)

I'll delete the code sometime in the future if it turns out its available on most SAP versions.

 [Florian Henninger](#)

Mar 9, 2015 11:24 AM

Nice one. I'll give it a shot in my testsystem and will report if there is something strange to me 😊
~Florian

 [Peter Inotai](#)

Mar 9, 2015 9:55 AM

Hi Lars,

Thanks a lot for sharing these checks.

I believe in SAP BASIS 7.02 CHECK_02 (EXIT outside of loop) and CHECK_23 (CHECK outside of loop) are part of the standard checks:

Details			
	Location / Finding	Description	Contact Person
Package	/EBY/ICIV_MISC		READSOFT
Class Pool	/EBY/CL_ICIV_TOOLS	Utilities	READSOFT
Method	GET_SELOPT_GUID	Get GUIDs based on selection criterias	INOTAI
Line Number	32		
Check Title	Extended Program Check		
Check Message	CHECK statement outside a loop		
Priority	Priority 3		
Found on	02.03.2015 15:08:29		
Only use CHECK in loops or directly after the start of a procedure.			
Otherwise use RETURN.			
Display Object	Exemptions allowed only for central results		

Anyway in lower releases it might not be there, so it makes sense to implement there.

Peter

 [Sergio Ferrari](#)

Mar 9, 2015 9:22 AM

Thanks for sharing Lars. I'll try as soon as a new inspection project will come.

This is a great example of the value of our SCN.