

### A list of metrics for similar analysis

Index	Name	Description	Used
0	cbo	coupling between objects	Y
1	cbomodified	coupling between object	Y
2	fanin	counts the number of input dependencies a class has	Y
3	fanout	counts the number of output dependencies a class has	Y
4	wmc	it counts the number of branch instructions in a class.	Y
5	dit	depth inheritance tree	Y
6	noc	number of children	Y
7	rfc	counts the number of unique method invocations in a class	Y
8	lcom	lack of cohesion of methods	N
9	numberofmethods	counts the number of methods.	Y
10	numberofstaticmethods	counts the number of static methods	Y
11	numberofpublicmethods	counts the number of public methods	Y
12	numberofprivatemethods	counts the numberof private methods	Y
13	numberofprotectedmethods	counts the numberof protected methods	N
14	numberofdefaultmethods	counts the numberof default methods	Y

15	visiblemethods().size()	counts the number of visible methods.	Y
16	numberofabstractmethods	counts the numberof abstract methods	N
17	numberoffinalmethods	counts the numberof final methods	N
18	numberofsynchronizedmethods	counts the numberof synchronized methods	N
19	numberoffields	counts the numberof fields	Y
20	numberofstaticfields	counts the numberof static fields	Y
21	numberofpublicfields	counts the numberof public fields	Y
22	numberofprivatefields	counts the numberof private fields	Y
23	numberofprotectedfields	counts the numberof protected fields	N
24	numberofdefaultfields	counts the numberof default fields	Y
25	numberoffinalfields	counts the numberof final fields	Y
26	numberofsynchronizedfields	counts the numberof synchronized fields	N
27	nosi	counts the number of invocations to static methods.	N
28	loc	it counts the lines of count, ignoring empty lines and comments.	Y

29	returnqty	the number of return instructions.	Y
30	loopqty	the number of loops	Y
31	comparisonsqty	the number of comparisons (i.e., == and !=).	N
32	trycatchqty	the number of try/catches	Y
33	parenthesizedexps qty	the number of expressions inside parenthesis.	Y
34	stringliteralsqty	the number of string literals (e.g., "john doe").	Y
35	numbersqty	the number of numbers (i.e., int, long, double, float) literals.	Y
36	assignmentsqty	the number of assignments	Y
37	mathoperationsqty	the number of mathoperations	Y
38	variablesqty	the number of variables	Y
39	maxnestedblocks	the highest number of blocks nested together.	Y
40	anonymousclasses qty	quantity of anonymous classes, inner classes, and lambda expressions	Y
41	innerclassesqty	the number of the number of inner classes	Y
42	lambdasqty	the number of lambdas	N
43	uniquewordsqty	number of unique words in the source code.	Y
44	modifiers	public/abstract/private/protected/native modifiers of classes/methods.	N

45	number of log statements	number of log statements in the source code.	Y
----	--------------------------	--	---

---

**Tips:**

The table above refers to the content of M. Aniche, ‘Java code metrics calculator (CK)’, 2015.(<https://github.com/mauricioaniche/ck/>)

Column *Used* is Y means we used this metric to conduct a *similar analysis*, and we do not use this metric when *Used* is N.