Readability Survey

Hello! This study focuses on the **readability** of Java classes. In this context a class is readable if a developer, different than the developer who wrote the class, is able to design a set of test cases that adequately cover the functionalities of that class.

The survey consists of assessing the readability of six classes from five Github projects (except in *common-cli* and *common-csv* where there are respectively five and four classes). Each page provides links to the classes within a single project.

Take your time to read each class; we recommend no more than five minutes per class, but many classes are short and should take less time. Once you are familiar with the classes, we ask you to assign a readability score, comparing the readability of the classes within the same project.

We thank you in advance for your time!

How many years of Java programming experience do you have?

○ 0 years

○ 1-3 years
○ 4-6 years
○ 6> years

First Project: commons-cli

Classes

CommandLine.java
DefaultParser.java
OptionGroup.java
Options.java
PatternOptionBuilder.java

Assign a score from 1 to 5 for each class (where 1 means "not at all readable" and 5 means "very readable"). You can assign the same score to multiple classes if you think they have a comparable degree of readability.

	1	2	3	4	5
CommandLine.java	○	○	○	○	○
DefaultParser.java	○	○	○	○	○
OptionGroup.java	○	○	○	○	○
Options.java	○	○	○	○	○
PatternOptionBuilder.java	○	○	○	○	○

Second Project: commons-csv

Classes

[CSVParser.java](#)
[CSVPrinter.java](#)
[CSVRecord.java](#)
[ExtendedBufferedReader.java](#)

Assign a score from 1 to 5 for each class (where 1 means "not at all readable" and 5 means "very readable"). You can assign the same score to multiple classes if you think they have a comparable degree of readability.

	1	2	3	4	5
CSVParser.java	○	○	○	○	○
CSVPrinter.java	○	○	○	○	○
CSVRecord.java	○	○	○	○	○
ExtendedBufferedReader.java	○	○	○	○	○

Third Project: commons-lang

Classes

[ArrayFill.java](#)

[ArraySorter.java](#)
[BasicThreadFactory.java](#)
[CharSet.java](#)
[Fraction.java](#)
[Range.java](#)

Assign a score from 1 to 5 for each class (where 1 means "not at all readable" and 5 means "very readable"). You can assign the same score to multiple classes if you think they have a comparable degree of readability.

	1	2	3	4	5
ArrayFill.java	○	○	○	○	○
ArraySorter.java	○	○	○	○	○
BasicThreadFactory.java	○	○	○	○	○
CharSet.java	○	○	○	○	○
Fraction.java	○	○	○	○	○
Range.java	○	○	○	○	○

Fourth Project: gson

Classes

[JsonTreeWriter.java](#)
[JsonArray.java](#)
[JsonPrimitive.java](#)

JsonStreamParser.java
LazilyParsedNumber.java
LinkedHashTreeMap.java

Assign a score from 1 to 5 for each class (where 1 means "not at all readable" and 5 means "very readable"). You can assign the same score to multiple classes if you think they have a comparable degree of readability.

	1	2	3	4	5
JsonTreeWriter.java	○	○	○	○	○
JsonArray.java	○	○	○	○	○
JsonPrimitive.java	○	○	○	○	○
JsonStreamParser.java	○	○	○	○	○
LazilyParsedNumber.java	○	○	○	○	○
LinkedHashTreeMap.java	○	○	○	○	○

Fifth Project: jfreechart

Classes

BoxAndWhiskerItem.java
Outlier.java
OutlierListCollection.java
PieLabelDistributor.java
XYCoordinate.java

Assign a score from 1 to 5 for each class (where 1 means "not at all readable" and 5 means "very readable"). You can assign the same score to multiple classes if you think they have a comparable degree of readability.

	1	2	3	4	5
BoxAndWhiskerItem.java	○	○	○	○	○
Outlier.java	○	○	○	○	○
OutlierListCollection.java	○	○	○	○	○
PieLabelDistributor.java	○	○	○	○	○
XYCoordinate.java	○	○	○	○	○
XYInterval.java	○	○	○	○	○