A methodology for performing meta-analyses of developers attitudes towards programming practices

Abstract. Programming practices are often labelled "best practice" and "bad practice" by developers. This label can be subjective but we can see trends among developers. A methodology for performing meta-analyses of articles discussing any given practice was created to determine programmers overall attitudes towards any given practice while accounting for factors such as whether they considered alternative approaches.

Introduction

Programming practices can often be described as *bad practice* or *best practice* depending on whether they have a positive or negative impact on the maintainability of the code in which they are used[1].

For software developers looking for information regarding any given programming practice, sources will vary in their level of detail. For example, a manual page will demonstrate how to use a practice but will not weigh in on the discussion of when or if the practice should be used. Opinion pieces may go into significantly more detail with discussions about pros/cons of the practice, where it's applicable and alternative approaches that can be used to solve the same problem.

If one article labels a programming practice "bad practice" and another "good practice" which should the reader believe?

The level of detail of an article can be used to determine academic rigour. An article suggesting to use the practice but without discussing alternative approaches is not making as strong a case for its use as a similar article which compares the practice to alternatives and explains why one approach is preferred over others.

A scoring system will be created to grade articles on their academic rigour. Once articles are graded, it will be possible to compare two or more articles based on their academic rigour and then perform a meta-analysis of any number of articles discussing a specific bad practice.

If academic rigour were ignored and a meta-analysis carried out using a simple tally of articles with positive/negative/neutral opinions a different conclusion may be drawn compared with an analysis including academic rigour. Academically rigourous articles may be more or less likely to have a favourable opinion of the practice.

Aims and Objectives

1. Create a scoring system which can be used to:
    1. Grade the analytic rigour of an article/book/paper discussing a particular programming practice.
    2. Compare the academic rigour of different articles for the purposes of meta-analysis.
    3. Compare the overall quality of discussions about a specific

programming practice.

2. Calculating the score should not require reading the article in detail to calculate the score and anything used to calculate the score should be a binary choice.
3. With a scoring system in place, perform proof-of-concept meta-analyses on practices which are well known to be described as "good" and "bad" to demonstrate that the meta-analysis framework is fit for purpose.

Methodology

1. Metric for comparing analytical rigour in programming articles

Differing methodological rigor in sources is a problem which exits exists when doing any kind of meta-analysis. When performing meta-analysis of clinical trials the Cochrane Collaboration[2] consider methodological rigour an important part of their meta-analysis.

Rather than simply counting the number of trials which show a positive outcome and counting the number of trials which show a negative outcome, they weigh the trials based on methodological rigour. For example in a meta-analysis of a drug they may find that 3 trials show that it is an effective treatment and 8 which say that it is not. Instead of simply counting the numbers on each side, they look at the academic rigor of each study and use that as a factor when building their conclusion of the overall efficacy of the treatment.

In a meta-analysis of the efficacy of homeopathic treatments[3] they found that trials of homeopathy with a poor methodology are much more likely to show a positive outcome whereas trials with a robust methodology are much more likely to conclude that homeopathy is no better than placebo.

This is because methodological rigour can affect the outcome. For example, by putting the most healthy patients in the experimental group and putting the least healthy patients in the control group it's likely that the experimental group will see significant improvement over the control group regardless of whether the drug being tested has any effect[4].

For programming articles, academic rigour can be plotted against whether the article recommends using or avoiding the practice to create a meta-analysis in a similar manner.

It should be possible to draw conclusions such as *as an article's academic rigour increases, it is more likely to recommend using the practice in question*

The created metric was based on the Jadad Scale[5] used for analysis of clinical trials in medicine. The Jadad Scale is a 5 point scale using a 3 question questionnaire which can be used to quickly assess the methodological rigour used in a clinical trial. The questions asked are: *Was the study described as randomized?, Was the study described as double blind? and Was there a description of withdrawals and dropouts?*. These are then used to generate a score from zero (very poor) to five (rigorous). By citation count the Jadad Scale is the most widely used method of comparing clinical trials in the world[6].

As the Jadad Scale is not applicable for anything other the clinical trials, a new metric was created based on the principles of the Jadad scale to be used in determining the academic rigour of any given article about a programming practice. A seven point scale was chosen with a point awarded if the article does each of the following:

1. Describes how to use the practice
2. Provides a code example of using the practice
3. Discusses potential negative/positive implications of using the practice
4. Describes alternative approaches to the same problem
5. Provides like for like code samples comparing the practice to alternative approaches
6. Discusses of pros/cons of the compared approaches
7. Offers a conclusion on when/where/if the practice is suitable

Using this metric, a manual page that describes a practice and provides a sample of how to use it would score two whereas an article that discussed the pros/cons of different approaches and made a recommendation would score seven.

2. Meta-analysis

For any meaningful conclusions to be drawn, two axis are required. Clinical trials could be separated by their Jadad score but this alone tells us nothing about the efficacy of the treatment. To produce a conclusion we need to plot the Jadad Score of a trial against outcome.

For example, a set of trials studying the same treatment can be analysed and observations drawn such as *trials with lower Jadad scores are more likely to produce a positive result*, indicating that the stronger the methodological rigor the less likely the treatment is to be shown to be effective.

Programming articles do not produce a result, but they can offer a recommendation to use or avoid the practice being discussed. A manual page won't make a recommendation but an opinion piece will discuss if/when the practice being described should be used.

A five point scale was used to model the recommendation made by an article:

1. Always favour this practice over alternatives
2. Favour this practice over alternatives unless specific (defined*) circumstances apply
3. Neutral - No recommendation (e.g. a manual page) or no conclusion drawn
4. Only use this practice in specific (defined*) circumstances
5. Always favour alternative approaches

A five point scale was chosen over a three point scale as there may be cases where an article is concluded with a discussion of trade-offs. For example where an approach may be faster but less flexible an author may conclude their article with something like "use this practice unless performance is a priority".

This meta-analysis will focus on flexibility. If a conclusion is drawn that you should use a practice when flexibility is preferred over performance (or any other consideration) then the article would be awarded a score of *2* and considered as "Favour this practice unless performance is a paramount concern".

The focus of the analysis could be be changed to performance, security or any other metric and results gathered in the same manner.

* For scores 2 and 4, the specific circumstances have to be described rather than alluded to.

For example Buss[7] writes:

When designing a system, it's important to pick the right design principle for your model. In many circumstances, it makes sense to prefer composition over inheritance.

This article only alludes to when using inheritance is preferable and provides only examples where composition is preferred. In this case the article is given a 5 despite the conclusion saying "many circumstances" rather than "all circumstances".

On the other hand, Ericson[8] says:

If you aren't sure if a class should inherit from another class ask yourself if you can substitute the child class type for the parent class type. For example, if you have a Book class and it has a subclass of ComicBook does that make sense? Is a comic book a kind of book? Yes, a comic book is a kind of book so inheritance makes sense. If it doesn't make sense use association or the has-a relationship instead.

In this instance, the author clearly states a situation where inheritance should be used over composition so would be given a recommendation score of 4.

3. Collecting data

Non-academic sources

In 1950, a vote at the meeting of the British Association for the Advancement of Science showed that about half those present now embraced the idea of continental drift. [...] Interestingly, oil company geologists had known for years that if you wanted to find oil you had to allow for precisely the sort of surface movements that were implied by plate tectonics. But oil geologists didn't write academic papers; they just found oil.

Bryson[9]

The Singleton has been regarded as bad practice in industry since at least 2003[10] with developers denouncing it ever since[11-21] yet where it is mentioned in academia it is only discussed as having been utilised while developing software rather than discussing whether it should or should not have been used[23-23]. Given the scale of negativity towards the pattern in industry and lack of discussion in academia, it's unsurprising that patterns lesser known within industry are never even mentioned in academic works.

This is likely because practices are encountered by people who spend 8 hours a day working on large projects where they are likely to encounter problems that academics focussing mostly on theory will not. Industry experts tend to work on large software projects which require constant maintenance and enhancement for years or even decades. They are able to determine which practices prevent them performing maintenance efficiently.

Although, like the oil geologists, they don't tend to write academic papers, many developers post articles on websites run by themselves or the company they work for discussing these issues.

As there is little discussion of the merits of most practices in academia yet there are many articles written by companies, developers and technology journalists, a wider search was conducted using Google. As a Google search for *singleton pattern* yields over half a million hits, a complete systematic review was not feasible. Instead, the first 100 relevant results from a Google search for the relevant practice will be used as the sample.

A *relevant result* is defined as an article which is written by a single author or organisation describing or discussing the singleton pattern. Discussion forums, posts on social media and question & answer sites will not be included as these pages will include multiple opinions. Comments sections on articles will be omitted for the same reason. Any article which has a *Jadad* style score of zero will also be deemed irrelevant.

Google was used to act as a randomization tool. A search returns any articles discussing the practice regardless of whether they are for or against its use.

Each article was then given a *Jadad* style score from 0-7 and a score for its recommendation.

3.1 Additional considerations

There are several practical issues with collecting data in this manner:

1. To minimise the effect of Google giving user-specific results based on previous searches, results were collected while logged out and using the browser's private browsing mode and closing the browser between each search term.
2. Search results will not be truly random due to the way Google's algorithm works and results will be sorted by *relevance* and the way Google sorts the results may have implicit bias: The most popular links and most cited links will appear first. Although not truly random, this gives a better overview of the zeitgeist than a genuinely randomised sample by putting the most read/cited articles ahead of less read/cited pages. Articles which are widely shared and linked to will be more likely to appear in the first 100 results.
3. A practice may have more than one common name. When this is the case, each name will be searched for and 100 results collected in total. If a practice is known by 4 different names, the first 25 relevant results for each practice were used. If a result lists both names it will only be counted once.
4. Other search engines may yield different results. Google was chosen because of its dominance and likelihood to have indexed more results. Using a search engine such as Qwant[24] which does not offer personalised results would make the results easier to replicate but may not offer as comprehensive results. Regardless of which search engine is used, results will change over time.

Further research is required to determine the extent of which these factors may affect results.

However, regardless of these factors, results should be indicative of developers attitudes towards the programming practice being analysed.

4. Test methodology

To verify that the suggested meta-analysis methodology produces meaningful results, a meta-analysis was performed on two practices where the result can be anticipated with a high degree of certainty. If the methodology works as intended, the following hypotheses should be proven true.

4.1.0 Singleton pattern

The singleton pattern is well known as being considered bad practice among developers[19] and will act as a good benchmark for testing the meta-analysis methodology.

4.1.1 Hypothesis

Before the results were collected it was expected that articles which had a higher *Jadad style score* (higher academic rigour) would be more likely to suggest avoiding the practice.

4.2.0 Dependency Injection

Dependency Injection is antithesis to the Singleton Pattern and is much more flexible. Although there are some practical considerations when using Dependency Injection and there is widespread discussion about the best way to implement it, it's widely considered the best approach for flexibility[25].

4.2.1 Hypothesis

Dependency Injection a well established method of increasing flexibility in code[26]. Because of this, it was expected that there would be few to no negative recommendations and as the *Jadad* style score increases articles should be more likely to suggest favouring dependency injection over alternative approaches.

5. Results

5.1 Singleton

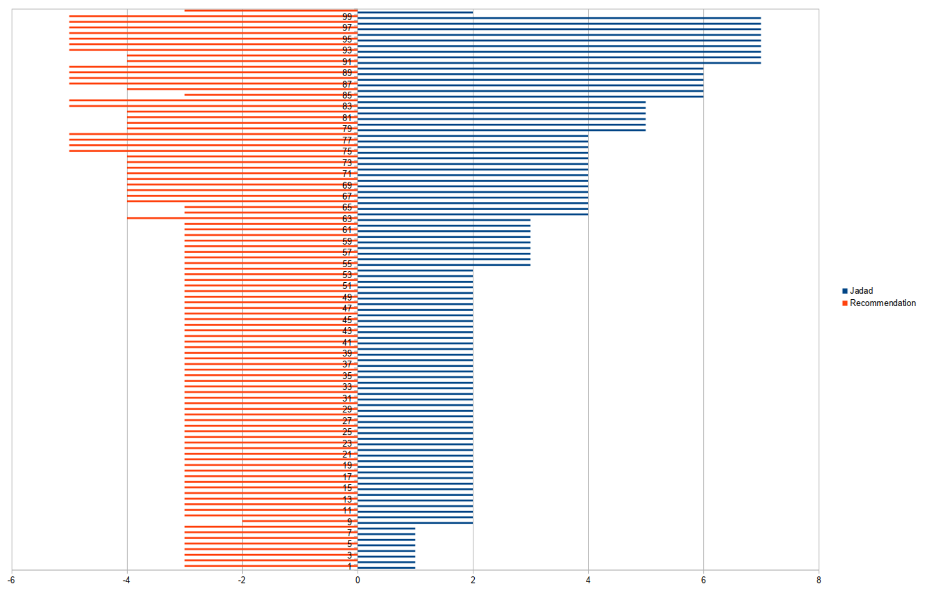

Fig 1. Singleton - Results

Each line represents an article and the left (orange) bar for each article is the recommednation going from 5: Avoid this practice at all costs (Far left) to 1: Favour this practice over alternatives.

The right (blue) bar for each article is the Jadad style score measuring academic rigour. A score of seven means the article describes the practice, provide code examples, discusses alternative approaches, provides like-for-like code samples, discusses the pros/cons of each approach and makes a recommendation of which approach should be used.

Article 1 has a recommendation score of 3 and a Jadad style score of 1. It does not go into detail and it's recommendation is neutral; it doesn't suggest either avoiding or favouring use of the Singleton Pattern.

Article 99 on the other hand has a strongly recommends against using the Singleton Pattern and has an Jadad stlye score of 7, it compares the singleton against alternatives in detail and concludes by strongly recommending against its use (recommendation score of 5).

Raw data is available as appendix 1.

As hypothesised, articles with a high academic rigour are considerably more like to suggest avoiding the singleton pattern.

Table 1. Singleton Pattern recommendation score

Recommendation	Number of articles making recommendation
1: Always favour this practice over alternatives	0
2: Favour this practice over alternaitves except in specific circumstances	1
3: Neutral/no recommedation	65
4: Favour alternative approaches except in specific circumstances	16
5: Always favour alternative approaches	18

If a simple tally was used, the singleton pattern would appear to have a mostly neutral recommendation score. 65% of articles do not recommend for or against its use.

5.1.1 Key findings - Singleton Pattern

- The mode recommendation is neutral. If a developer looked through articles about the singleton pattern, 65% of the articles they read would not recommend against using the Singleton Pattern.
- The mean recommendation score is is 3.5. From this alone it could be inferred that the singleton pattern is generally considered to be neutral, slightly discouraged but not widely avoided.
- When the Jadad style score is taken into account, every article which makes a recommendation recommends against using the singleton pattern (recommendation score of 4 or 5).
- Only 22% of articles about the singleton pattern even mention alternative approaches that can be used to solve the same problem
- Of those that recommend against using the pattern, over half say it should be a voided at all cost.
- 55 of the 65 articles which make a neutral recommendation are manual type pages (Jadad style score of 2) which show how to use the pattern but do not weigh in on when, where or if it should be used and do not compare the pattern to alternatives.
- No articles which make a recommendation recommend using the singleton pattern instead of alternative approaches

5.2 Dependency Injection

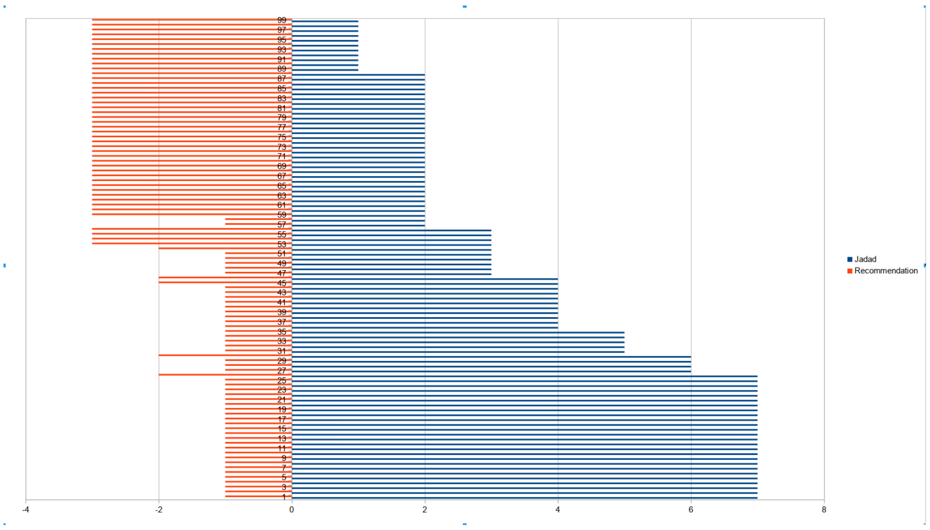

Fig 1. Dependency Injection - Results

Raw data is available in appendix 2.

As hypothesised, Dependency Injection is seen as overwhelmingly positive with zero articles discouraging its general use.

The breakdown of recommendation scores is as follows:

Table 2. Dependency Injection recommendation score

Recommendation	Number of articles making recommendation
1: Always favour this practice over alternatives	50
2: Favour this practice over alternaitves except in specific circumstances	5
3: Neutral/no recommedation	45
4: Favour alternative approaches except in specific circumstances	0
5: Always favour alternative approaches	0

In a tally of whether articles recommend using or avoiding the practice, 50% of articles recommending using the practice over alternatives.

5.2.1 Key findings - Dependency Injedction

- The mean score is 1.94 which shows that even using a simple tally, the overall recommendation is that Dependency Injection is a favourable pattern among developers.
- Every article with an academic rigour score of 4 or higer recommends using this practice instead of alternative.
- When the Jadad style score is taken into consideration, 47 of the 50 articles with a neutral recommendation are manual style pages which show how the pattern is used but do not discuss when, where or if it should be used.

- Discounting the manual pages, only two of the remaining 53 articles make a neutral recommendation and both of those have a *Jadad* style score 3.
- As the Jadad style score increases, the probability that an article will recommend using Dependency Injection over alternatives increases
- Only 5 of the 55 articles in favour of dependency injection suggest there some specific circumstances where alternatives should be used instead (*Jadad* style score of two).

5. Conclusion

By testing the methodology with practices that the outcome can be predicted for it was possible to validate this meta-analysis methodology.

The methodology produced the expected result. It was shown that if an author considered alternative approaches they were more likely to recommend against using the Singleton Pattern. The inverse was also true for Dependency Injection.

As these were the expected results, the methodology suggested can be shown to work as intended and provide an overview of the attitudes of developers about any given practice.

This meta-analysis methodology gives more insight into the overall opinion of programming practices than a simple tally of for/against/neutral by also accounting for academic rigour.

5.2 Additional findings

1. Although a small sample size of two practices were tested, in both cases roughly half of articles analyse do not make a recommendation on when/where the practice should be used. For the singleton pattern only 45% of analysed articles discussed whether the pattern should be used or avoided.
2. Any developer looking for information on a practice will find more information about *how* to use a practice than *when* or *where* the practice is applicable.

5.1 Problems Encountered

Data collection using Google became increasingly difficult after around 80 relevant results. The number of irrelevant articles appearing in search results begin to heavily outweigh the relevant articles and there was a significant issue with duplicated content. Articles had been posted on multiple websites, often without dates or author names, making it difficult to keep track of which articles had already been included in the meta-analysis.

Since Dependency Injection and the Singleton pattern are both widely known and discussed programming practices, finding 100 unique relevant results for lesser known practices may be difficult.

5.2 Future Research

This research could be continued by running the same meta-analysis on different search engines and comparing the results or looking into trends over time using article dates. For example, it may be observed that a practice is seen favourably in articles published in 1990s-2000s and then less favourably as time progresses.

This methodology could be abstracted to and used for a meta-analysis of any widely discussed topic by defining the scales for academic rigour and recommendation.

1. Hevery, M. (2008) *Flaw: Constructor Does Real Work* [online]. Available from: http://misko.hevery.com/code-reviewers-guide/flaw-constructor-does-real-work/
2. Cochrane, C. (n.d.) *Cochrane* [online]. Available from: http://www.cochrane.org/
3. Mathie, R., Frye, J., Fisher, P. (2015) Homeopathic Oscillococcinum® for preventing and treating influenza and influenza-like illness. *Cochrane Database System Rev 12* .
4. Goldacre, B. (2010) *Bad Science* ISBN: 978-0-00-724019-7. Fourth Estate.
5. Jadad, A., Moore, A., Carroll, D., Jenkinson, C. (1996) Assessing the quality of reports of randomized clinical trials: Is blinding necessary?. *Controlled Clinical Trials* 17(1), pp.1-12. ELSEVIER.
6. Olivo, S., Macedo, L., Caroline, I., Fuentes, J., Magee, D. (2008) Scales to assess the quality of randomized controlled trials: a systematic review.(Research Report). *Physical Therapy* 88(2), pp.156.
7. Buss, M. (2016) *Interfaces vs Inheritance in Swift* [online]. Available from: https://mikebuss.com/2016/01/10/interfaces-vs-inheritance/
8. Ericson, B. (1995) *Association vs Inheritance* [online]. Available from: http://ice-web.cc.gatech.edu/ce21/1/static/JavaReview-RU/OOBasics/ooAssocVsInherit.htm l
9. Bryson, B. (2010) *A short history of nearly everything* ISBN: 9780552997041. London : Black Swan.
10. Radford, M. (2003) Singleton - the anti-pattern. *Overload* 57. ACCU.
11. Hevery, M. (2008) *Singletons are Pathological Liars* [online]. Available from: http://misko.hevery.com/2008/08/17/singletons-are-pathological-liars/
12. Sayfan, M. (n.d.) *Avoid Global Variables, Environment Variables, and Singletons* [online]. Available from: https://sites.google.com/site/michaelsafyan/software-engineering/avoid-global-variables-env ironment-variables-and-singletons
13. Densmore, S. (2004) *Why Singletons Are Evil* [online]. Available from: http://blogs.msdn.com/b/scottdensmore/archive/2004/05/25/140827.aspx
14. Yegge, S. (2004) *Singleton Considered Stupid* [online]. Available from: https://sites.google.com/site/steveyegge2/singleton-considered-stupid
15. Ronacher, A. (2009) *Singletons and their problems in Python* [online]. Available from: http://lucumr.pocoo.org/2009/7/24/singletons-and-their-problems-in-python/
16. Brown, W. (2013) *Why Singletons are "Bad Patterns"* [online]. Available from: http://brollace.blogspot.co.uk/2013/04/why-singletons-are-bad-patterns.html
17. Kofler, P. (2012) *Why Singletons Are Evil* [online]. Available from: http://blog.code-cop.org/2012/01/why-singletons-are-evil.html
18. Weaver, R. (2010) *Static methods vs singletons: choose neither* [online]. Available from: http://www.phparch.com/2010/03/static-methods-vs-singletons-choose-neither/
19. Knack-Nielsen, T. (2008) *What's so bad about the Singleton?* [online]. Available from: http://www.sitepoint.com/whats-so-bad-about-the-singleton/
20. Badu, K. (2008) *What's so evil about Singleton?* [online]. Available from: http://www.sitepoint.com/forums/showthread.php?530917-What-s-so-evil-about-Singleton
21. Hart, S. (2011) *Why helper, singletons and utility classes are mostly bad* [online]. Available

from:
http://smart421.wordpress.com/2011/08/31/why-helper-singletons-and-utility-classes-are-mostly-bad-2/

22. Alipour, G., Sangar, A., Mogaddam, M. (2016) ASPECT ORIENTED IMPLEMENTATION OF DESIGN PATTERNS USING METADATA. *Journal of Fundamental and Applied Sciences* 57, pp.66-75.
23. Liu, H., Cai, C., Zu, C. (2011) An object-oriented serial implementation of a DSMC simulation package. *Journal of Fundamental and Applied Sciences* 8, pp.816-825.
24. Qwant, Q. (n.d.) *Qwant search engine* [online]. Available from: https://www.qwant.com/
25. Albert, A. (2013) *Why should we use dependency injection?* [online]. Available from: http://www.javacreed.com/why-should-we-use-dependency-injection/
26. Fowler, M. (2004) *Inversion of Control Containers and the Dependency Injection pattern* [online]. Available from: http://martinfowler.com/articles/injection.html

Appendix 1. Raw data: Singleton

URL	Describe	Example	Implications	Alternatives	Comparison code	Pros/Cons	Recommendation	Recommendation scale	Jadad	
http://www.fssnip.net/7p/title/Singleton-Pattern		1						3	1	-3
https://www.hackerrank.com/challenges/java-singleton/forum		1						3	1	-3
https://gist.github.com/mssola/6138155		1						3	1	-3
http://fortranwiki.org/fortran/show/Singleton+pattern	1							3	1	-3
http://www.adam-bien.com/roller/abien/entry/singleton_pattern_in_es6_and		1						3	1	-3
https://coderwall.com/p/iemfbg/objective-c-singleton-pattern-with-arc		1						3	1	-3
http://cruise.eecs.uottawa.ca/umple/SingletonPattern.html		1						3	1	-3
http://www.netobjectives.com/resources/books/design-patterns-explained/java-code-examples/chapter21		1						3	1	-3
https://www.dotnetperls.com/singleton-static		1					1	2	2	-2
https://www.tutorialspoint.com/design_pattern/singleton_pattern.htm	1	1						3	2	-3
https://msdn.microsoft.com/en-gb/library/ff650316.aspx	1	1						3	2	-3
https://www.javaworld.com/article/2073352/core-java/simply-singleton.html	1	1						3	2	-3
http://www.oodesign.com/singleton-pattern.html	1	1						3	2	-3
https://code.tutsplus.com/tutorials/android-design-patterns-the-singleton-pattern--cms-29153	1	1						3	2	-3
http://www.dofactory.com/javascript/singleton-design-pattern	1	1						3	2	-3
https://www.techopedia.com/definition/15830/singleton	1	1						3	2	-3
https://www.geeksforgeeks.org/singleton-design-pattern/	1	1						3	2	-3
https://dzone.com/articles/singleton-pattern-a-deep-dive	1	1						3	2	-3
https://www.javatpoint.com/singleton-design-pattern-in-java	1	1						3	2	-3
https://developer.salesforce.com/page/Apex_Design_Patterns_-_Singleton	1	1						3	2	-3
https://fullstack-developer.academy/singleton-pattern-in-typescript/	1	1						3	2	-3
http://jargon.js.org/_glossary/SINGLETON_PATTERN.md	1	1						3	2	-3
https://dalibornasevic.com/posts/9-ruby-singleton-pattern	1	1						3	2	-3
http://marcio.io/2015/07/singleton-pattern-in-go/	1	1						3	2	-3
https://basarat.gitbooks.io/typescript/docs/tips/singleton.html	1	1						3	2	-3
http://www.vogella.com/tutorials/DesignPatternSingleton/article.html	1	1						3	2	-3
https://www.nada.kth.se/kurser/kth/2D1359/01-02/contents/forelasningar/lecture10.ppt	1	1						3	2	-3
https://www.linkedin.com/pulse/singleton-pattern-eager-lazy-enum-ramasamy-kasiviswanathan	1	1						3	2	-3
https://www.avajava.com/tutorials/lessons/singleton-pattern.html	1	1						3	2	-3
http://ftp.mak.com/out/classdocs/vrforces4.4.1/classref/vrv_the_rooted_singleton_pattern.html	1	1						3	2	-3
http://www.galloway.me.uk/tutorials/singleton-classes/	1	1						3	2	-3
https://wiki.base22.com/btg/singleton-pattern-3459.html	1	1						3	2	-3
https://coffeescript-cookbook.github.io/chapters/design_patterns/singleton	1	1						3	2	-3
https://blogs.sap.com/2012/04/06/singleton-pattern-in-abap/	1	1						3	2	-3
http://docs.ros.org/indigo/api/typelib/html/group__singleton.html	1	1						3	2	-3
https://learnswiftwithbob.com/course/object-oriented-swift/singleton-pattern.html	1	1						3	2	-3
https://www.visual-paradigm.com/tutorials/singletonpattern.jsp	1	1						3	2	-3
https://medium.freecodecamp.org/lets-talk-about-you-and-the-singleton-design-pattern-bb2e160fa952	1	1						3	2	-3
https://www.htmlgoodies.com/beyond/javascript/implementing-the-singleton-design-pattern-in-javascript.html	1	1						3	2	-3
http://web.science.mq.edu.au/~mattr/courses/object_oriented_development_practices/6/notes.html	1	1						3	2	-3
http://www.jot.fm/issues/issue_2007_03/column2/	1	1						3	2	-3
https://itexico.com/blog/bid/99247/Software-Development-The-Singleton-Design-Pattern-and-other-Creational-Patterns	1	1						3	2	-3
http://code.activestate.com/recipes/52558-the-singleton-pattern-implemented-with-python/	1	1						3	2	-3
https://alvinalexander.com/scala/how-to-implement-singleton-pattern-in-scala-with-object	1	1						3	2	-3
http://wiki.unity3d.com/index.php/Singleton	1	1						3	2	-3
https://locklessinc.com/articles/singleton_pattern/	1	1						3	2	-3
http://www.tothenew.com/blog/singleton-pattern-with-javascript/	1	1						3	2	-3
https://sweetcode.io/singleton-design-pattern-using-java/	1	1						3	2	-3
http://elbenshira.com/blog/singleton-pattern-in-python/	1	1						3	2	-3
https://php.earth/docs/php/ref/oop/design-patterns/singleton	1	1						3	2	-3
https://www.ibm.com/developerworks/library/j-dcl/	1	1						3	2	-3
https://amir.rachum.com/blog/2012/04/26/implementing-the-singleton-pattern-in-python/	1	1						3	2	-3
https://krakendev.io/blog/the-right-way-to-write-a-singleton	1	1						3	2	-3
http://moddb.wikia.com/wiki/Singleton_Pattern	1	1						3	2	-3
http://csharpindepth.com/Articles/General/Singleton.aspx	1	1	1					3	3	-3
https://www.journaldev.com/1377/java-singleton-design-pattern-best-practices-examples	1	1	1					3	3	-3
http://best-practice-software-engineering.ifs.tuwien.ac.at/patterns/singleton.html	1	1	1					3	3	-3
https://refactoring.guru/design-patterns/singleton	1	1		1				3	3	-3
https://www.techrepublic.com/blog/software-engineer/using-the-singleton-pattern-in-java/	1	1	1					3	3	-3
https://www.gamasutra.com/blogs/MattChristian/20101013/88205/OOPsie_Patterns_The_Singleton_Pattern.php	1	1	1					3	3	-3
https://pdfs.semanticscholar.org/presentation/f1dc/667b86aac3f5db63c26a67d30d586d2244a6.pdf	1	1	1					3	3	-3
https://www.implementingquantlib.com/2017/09/odds-and-ends-singleton.html	1	1	1					3	3	-3
https://8thlight.com/blog/josh-cheek/2012/10/20/implementing-and-testing-the-singleton-pattern-in-ruby.html	1	1	1					4	3	-4
http://csc.columbusstate.edu/woolbright/java/singleton.html	1	1	1	1				3	4	-3
https://www.codeproject.com/Articles/307233/Singleton-Pattern-Positive-and-Negative-Aspects	1	1	1			1		3	4	-3
https://sourcemaking.com/design_patterns/singleton	1	1	1				1	4	4	-4
https://www.gofpatterns.com/design-patterns/module3/consequences-effects-singleton-pattern.php	1	1	1				1	4	4	-4
http://wiki.c2.com/?SingletonPattern	1	1	1				1	4	4	-4
https://medium.com/if-let-swift-programming/the-swift-singleton-pattern-442124479b19	1	1	1				1	4	4	-4
http://2ality.com/2011/04/singleton-pattern-in-javascript-not.html	1	1	1				1	4	4	-4
https://ieftimov.com/singleton-pattern	1	1	1				1	4	4	-4
https://addyosmani.com/resources/essentialjsdesignpatterns/book/	1	1				1	1	4	4	-4
https://phpenthusiast.com/blog/the-singleton-design-pattern-in-php	1	1	1				1	4	4	-4
https://www.perl.com/article/52/2013/12/11/Implementing-the-singleton-pattern-in-Perl/	1	1	1				1	4	4	-4
http://www.bogotobogo.com/DesignPatterns/singleton.php	1	1		1			1	5	4	-5
https://www.infoworld.com/article/3112025/application-development/design-patterns-that-i-often-avoid-singleton.html			1	1		1	1	5	4	-5
https://codeburst.io/design-patterns-for-modern-web-development-singletons-bf7bc06bd17d	1		1	1		1		5	4	-5
https://anasshekhamis.com/2017/07/27/the-singleton-design-pattern-es5-and-es2015/	1	1	1				1	5	4	-5
https://tommcfarlin.com/singleton-design-pattern-1/	1	1	1	1			1	4	5	-4
https://php.earth/docs/php/ref/oop/design-patterns/singleton	1	1	1	1			1	4	5	-4
https://torquemag.io/2016/11/singletons-wordpress-good-evil/	1	1	1	1			1	4	5	-4
http://adamschepis.com/2011/05/02/im-adam-and-im-a-recovering-singleton-addict.html	1		1	1		1	1	4	5	-4
http://www.phptherightway.com/pages/Design-Patterns.html	1		1	1		1	1	5	5	-5
http://beust.com/weblog/2011/03/10/rehabilitating-the-singleton-pattern/	1		1	1		1	1	5	5	-5
https://code.tutsplus.com/tutorials/design-patterns-the-singleton-pattern--cms-23073	1	1	1	1	1	1		3	6	-3
http://wiki.freepascal.org/Singleton_Pattern	1	1	1	1		1	1	4	6	-4
https://cocoacasts.com/are-singletons-bad/	1	1	1	1	1		1	5	6	-5
http://robdodson.me/javascript-design-patterns-singleton/	1	1	1	1		1	1	5	6	-5
https://www.vojtechruzicka.com/singleton-pattern-pitfalls/	1	1	1	1		1	1	5	6	-5
https://whydoesitsuck.com/why-the-singleton-pattern-sucks-and-you-should-avoid-it/	1	1	1	1		1	1	5	6	-5
https://theburningmonk.com/2013/09/dart-implementing-the-singleton-pattern-with-factory-constructors/	1	1	1	1	1	1	1	4	7	-4
http://enterprisecraftsmanship.com/2016/05/03/singleton-vs-dependency-injection/	1	1	1	1	1	1	1	4	7	-4
http://gameprogrammingpatterns.com/singleton.html	1	1	1	1	1	1	1	5	7	-5
https://blog.ndepend.com/singleton-pattern-costs/	1	1	1	1	1	1	1	5	7	-5
https://www.michaelsafyan.com/tech/design/patterns/singleton	1	1	1	1	1	1	1	5	7	-5
https://www.zaraffasoft.com/2016/10/14/singleton-pattern-the-light-or-the-dark-side-of-the-force/	1	1	1	1	1	1	1	5	7	-5
https://carlalexander.ca/singletons-in-wordpress/	1	1	1	1	1	1	1	5	7	-5
https://krakendev.io/blog/antipatterns-singletons	1	1	1	1	1	1	1	5	7	-5
https://www.objc.io/issues/13-architecture/singletons/	1	1	1	1	1	1	1	5	7	-5
http://rcardin.github.io/design/programming/2015/07/03/the-good-the-bad-and-the-singleton.html	1	1						3	2	-3

Appendix 2. Raw data: Dependency Injection

URL	Describe	Example	Implications	Alternatives	Comparison code	Pros/Cons	Recommendation	Recommendation scale	Jadad
https://msdn.microsoft.com/en-us/library/hh323705(v=vs.100).aspx	1	1	1	1	1	1	1	1	7
https://hackernoon.com/you-dont-need-to-know-dependency-injection-2e9d2ba1978a	1	1	1			1	1	1	5
https://www.theserverside.com/news/1321158/A-beginners-guide-to-Dependency-Injection	1	1	1	1		1	1	1	6
https://angular.io/guide/dependency-injection-pattern	1	1	1	1	1	1	1	1	7
https://martinfowler.com/articles/injection.html	1	1	1	1	1	1	1	1	7
https://medium.freecodecamp.org/demystifying-dependency-injection-49d4b6fe6536	1	1	1	1	1	1	1	1	7
https://documentation.magnolia-cms.com/display/DOCS56/Dependency+injection+and+inversion+of+control	1	1						1	2
http://www.tutorialsteacher.com/ioc/dependency-injection	1	1	1	1	1	1	1	1	7
https://www.codeguru.com/csharp/.net/net_asp/mvc/understanding-dependency-injection.htm	1	1	1	1	1	1	1	1	7
http://krasimirtsonev.com/blog/article/Dependency-injection-in-JavaScript	1	1	1	1	1		1	1	6
https://www.devbridge.com/articles/dependency-injection-in-javascript/#	1	1	1	1	1	1	1	1	7
https://codecraft.tv/courses/angular/dependency-injection-and-providers/overview/	1	1	1	1	1	1	1	1	7
https://aspnetboilerplate.com/Pages/Documents/Dependency-Injection	1	1	1	1	1	1	1	1	7
https://webdev.dartlang.org/angular/guide/dependency-injection	1	1	1	1	1	1	1	1	7
https://doc.nette.org/en/2.4/dependency-injection	1	1	1	1	1	1	1	1	7
https://medium.com/makingtuenti/dependency-injection-in-swift-part-1-236fddad144a	1	1					1	1	3
https://android.jlelse.eu/android-mvp-architecture-with-dependency-injection-dee43fe47af0	1	1	1	1	1	1	1	1	7
https://www.dotnettricks.com/learn/dependencyinjection/implementation-of-dependency-injection-pattern-in-csharp	1	1	1				1	1	4
https://nehalist.io/dependency-injection-in-typescript/	1	1	1	1	1	1	1	1	7
https://www.playframework.com/documentation/2.6.x/ScalaDependencyInjection	1	1	1	1			1	1	5
https://www.raywenderlich.com/171327/dependency-injection-android-dagger-2	1	1	1	1	1	1	1	1	7
https://blog.drewolson.org/dependency-injection-in-go/	1	1					1	1	3
https://docs.oracle.com/javaee/6/tutorial/doc/giwhl.html	1						1	1	2
https://stackify.com/dependency-injection/	1	1	1				1	1	4
https://www.swiftbysundell.com/posts/dependency-injection-using-factories-in-swift	1	1					1	1	3
https://matthiasnoback.nl/2018/06/road-to-dependency-injection/	1	1	1	1	1	1	1	1	7
https://codeburst.io/dependency-injection-with-vue-js-f6b44a0dae6d	1	1	1			1	1	1	5
https://www.getopensocial.com/blog/open-source-technology/dependency-injection-php	1	1	1	1	1	1	1	1	7
https://www.tutorialspoint.com/spring/spring_dependency_injection.htm	1	1	1	1	1	1	1	1	7
https://fsharpforfunandprofit.com/posts/dependency-injection-1/	1	1	1	1	1	1	1	1	7
http://fabien.potencier.org/what-is-dependency-injection.html	1	1	1	1	1	1	1	1	7
https://www.bignerdranch.com/blog/dependency-injection-ios/	1	1	1	1	1	1	1	1	7
https://code.tutsplus.com/tutorials/dependency-injection-in-php--net-28146	1	1	1	1	1	1	1	1	7
https://docs.sitefinity.com/use-constructor-dependency-injections-mvc	1		1				1	1	3
https://www.techyourchance.com/dependency-injection-android/	1	1	1				1	1	4
http://andrewembler.com/2018/03/concrete-guide-dependency-injection	1	1	1	1	1	1	1	1	7
https://www.tomasvotruba.cz/blog/2018/05/07/why-you-should-combine-symfony-console-and-dependency-injection/	1	1	1	1			1	1	5
https://codeshare.co.uk/blog/how-to-start-using-dependency-injection-in-mvc-and-umbraco/	1	1	1				1	1	4
https://samueleresca.net/2017/07/inversion-of-control-and-unit-testing-using-typescript/	1	1	1				1	1	4
https://php.earth/docs/php/ref/oop/design-patterns/dependency-injection	1	1	1	1	1	1	1	1	7
https://developer.telerik.com/featured/three-ds-web-development-3-dependency-injection/	1	1	1				1	1	4
https://learnappmaking.com/dependency-injection-swift/	1	1	1	1	1		1	1	6
https://www.guru99.com/angularjs-dependency-injection.html	1	1					1	1	3
https://appliedgo.net/di/	1	1	1	1	1	1	1	1	7
http://rubyblog.pro/2016/10/ruby-dependency-injection	1	1	1	1	1	1	1	1	7
https://vuejs.org/v2/guide/components-edge-cases.html	1	1					1	2	3
https://blog.gojekengineering.com/the-many-flavours-of-dependency-injection-in-go-25aa070d79a0	1	1	1				1	2	4
https://www.eclipse.org/che/docs/guice.html	1	1	1				1	2	4
http://flowframework.readthedocs.io/en/stable/TheDefinitiveGuide/PartIII/ObjectManagement.html	1	1	1	1	1		1	2	6
https://enterprisecraftsmanship.com/2016/05/03/singleton-vs-dependency-injection/	1	1	1	1	1	1	1	2	7
http://best-practice-software-engineering.ifs.tuwien.ac.at/patterns/dependency_injection.html	1	1	1					3	3
http://www.baeldung.com/inversion-control-and-dependency-injection-in-spring	1	1	1			1		3	4
http://www.vogella.com/tutorials/DependencyInjection/article.html	1	1	1					3	3
https://deviq.com/dependency-injection/	1							3	1
https://dotnet.github.io/orleans/Documentation/Core-Features/Dependency-Injection.html	1	1						3	2
https://aurelia.io/docs/fundamentals/dependency-injection/	1	1						3	2
https://symfony.com/doc/current/components/dependency_injection.html	1	1						3	2
https://guides.emberjs.com/release/applications/dependency-injection/	1	1						3	2
https://itnext.io/typescript-dependency-injection-setting-up-inversifyjs-ioc-for-a-ts-project-f25d48799d70	1	1						3	2
https://www.drupal.org/docs/8/api/services-and-dependency-injection/services-and-dependency-injection-in-drupal-8	1	1						3	2
https://devdocs.magento.com/guides/v2.2/extension-dev-guide/depend-inj.html	1	1						3	2
https://loopback.io/doc/en/lb4/Dependency-injection.html	1	1	1					3	3
https://blog.kotlin-academy.com/dependency-injection-the-pattern-without-the-framework-33cfa9d5f312	1	1						3	2
http://docs.automapper.org/en/stable/Dependency-injection.html	1	1						3	2
https://pimcore.com/docs/4.6.x/Development_Documentation/Extending_Pimcore/Dependency_Injection.html	1	1						3	2
https://www.mvvmcross.com/documentation/fundamentals/dependency-injection	1	1						3	2
https://blog.carbonfive.com/2018/03/19/lightweight-dependency-injection-in-elixir-without-the-tears/	1	1						3	2
https://learn-blazor.com/architecture/dependency-injection/	1	1						3	2
https://medium.com/scribd-data-science-engineering/weaver-a-painless-dependency-injection-framework-for-swift-7c4afad5ef6a	1	1						3	2
https://www.javatpoint.com/dependency-injection-in-spring	1	1						3	2
https://blog.angularindepth.com/angular-dependency-injection-and-tree-shakeable-tokens-4588a8f70d5d	1							3	1
https://www.yiiframework.com/doc/guide/2.0/en/concept-di-container	1	1						3	2
http://blog.thecodewhisperer.com/permalink/keep-dependency-injection-simple	1							3	1
https://docs.phalconphp.com/hr/3.3/di	1	1					1	3	3
https://docs.silverstripe.org/en/4/developer_guides/extending/injector/	1	1						3	2
https://struts.apache.org/core-developers/dependency-injection.html	1							3	1
https://exceptionnotfound.net/using-entity-framework-dbcontext-with-dependency-injection/	1	1	1	1				3	4
https://dzone.com/articles/optional-dependency-injection-with-spring		1						3	1
http://picocontainer.com/injection.html	1							3	1
https://www.pacoworks.com/2018/02/25/simple-dependency-injection-in-kotlin-part-1/	1	1						3	2
https://proandroiddev.com/better-dependency-injection-for-android-567b93353ad		1						3	1
https://getakka.net/articles/actors/dependency-injection.html		1						3	1
http://docs.drush.org/en/master/dependency-injection/	1	1						3	2
https://devdocs.magento.com/guides/v2.2/extension-dev-guide/depend-inj.html	1	1						3	2
https://doc.sitecore.net/sitecore_experience_platform/developing/developing_with_sitecore/dependency_injection	1	1						3	2
https://docs.litium.com/documentation/architecture/dependency-injection	1							3	1
https://arrow-kt.io/docs/patterns/dependency_injection/		1						3	1
http://www.dotnetcurry.com/aspnet-core/1426/dependency-injection-di-aspnet-core	1	1						3	2
https://framework.zend.com/manual/2.4/en/tutorials/quickstart.di.html	1	1						3	2
https://www.future-processing.pl/blog/introduction-to-dependency-injection/	1	1						3	2
https://docs.particular.net/nservicebus/dependency-injection/	1	1						3	2
https://www.oreilly.com/ideas/handling-dependency-injection-using-java9-modularity	1	1						3	2
https://blog.nrwl.io/essential-angular-dependency-injection-a6b9dcca1761	1	1						3	2
https://toddmotto.com/angular-dependency-injection	1	1						3	2
https://blog.mexia.com.au/dependency-injections-on-azure-functions-v2	1	1	1	1	1			3	1
http://tutorials.jenkov.com/dependency-injection/index.html	1	1	1	1				3	4
https://chromatichq.com/blog/dependency-injection-drupal-8-plugins	1	1						3	2
http://jbehave.org/reference/stable/dependency-injection.html		1						3	1
https://www.infoworld.com/article/2974298/application-architecture/exploring-the-dependency-injection-principle.html	1	1						3	2