Are comments necessary for a programming language? [closed]

Asked 16 years ago Modified 3 years, 10 months ago Viewed 5k times

▲

25

▼

Closed. This question is opinion-based. It is not currently accepting answers.

Want to improve this question? Update the question so it can be answered with facts and citations by editing this post.

Closed 3 years ago.

Improve this question

After some stupid musings about Klingon languages, that came from this post I began a silly hobby project creating a Klingon programming language that compiles to Lua byte-code. During the initial language design phase I looked up information about Klingon programmers, and found out about this Klingon programming rule:

> **A TRUE Klingon Warrior does not comment his code!**

So I decided my language would **not support commenting**, as any good Klingon would never use them.

Now many of the Klingon ways don't seem reasonable to us Human programmers, however while dabbling with the design and implementation of my hobby language I came to realize that this Klingon rule about commenting is indeed very reasonable, if not great.

Removing the ability to comment from a programming language meant I **HAVE** to write **literate code**, no exceptions.

So it got me wondering if there are any languages out there that don't support comments?

Is there are any really good arguments to not remove commenting from a language?

Edit: Any good examples of comments required?

P.S.> My hobby language above is partially silly anyways, so don't focus too much on my implementation, as much as the concept of comments required in general

language-agnostic **comments**

Share

Improve this question

Follow

edited May 23, 2017 at 10:29

community wiki

5 revs, 2 users 64%
Robert Gould

Sometimes my English needs comments. I guess I am no Klingon. – Michael Burr Dec 24, 2008 at 5:51

1 Why do you say it should be closed? Seems entirely programming-related to me and it seems unlikely to be a duplicate of an existing question. – Dave Sherohman Dec 24, 2008 at 6:28

I didn't find any duplicates while entering the question atleast – Robert Gould Dec 24, 2008 at 6:45

2 Hahaha, that made me laugh. (The link, that is.) :D – Robert K Dec 24, 2008 at 18:08

32 Answers

Sorted by: Highest score (default)

1 2 Next

▲

24

▼

Do not comment WHAT you are doing, but WHY you are doing it.

The WHAT is taken care of by clean, readable and simple code with proper choice of variable names to support it. Comments show a higher level structure to the code that can't be (or is hard to) show by the code itself.

Share Improve this answer Follow

answered Dec 24, 2008 at 5:27

community wiki

cschol

May I also suggest to keep comments current when Refactoring. I hope that goes without saying.
– Yzmir Ramirez Nov 14, 2011 at 4:52

▲

23

▼

✓

I am not sure I agree with the "Have" in the statement "Removing the ability to comment from a programming language meant I HAVE to write literate code, no exceptions", since it is not as if all code is documented. My guess is that most people would write unreadable code.

More to the point, I personally do not believe in the reality of the self-explanatory program or API in the practical world.

My experience from manually analyzing the documentation of entire APIs for my dissertation suggests that all too often you would have to carry more information than you could convey in the signature alone. If you eliminate interface comments from your language, what are the alternatives? No documentation is not an option. External documentation is less likely to be read.

As for internal documentation, I can see your point in wanting to reduce documentation to convince people to write better. However, comments serve many collaboration and coordination purposes and are meant to raise awareness of things. By banishing these details to extenral locations, you are reducing the chances that they come to a future reader's awareness, unless your tooling is great.

Share Improve this answer Follow

answered Dec 24, 2008 at 5:19

community wiki
Uri

+1 because having to externalize the documentation dependency is an important point in favor of comments. – Robert Gould Dec 24, 2008 at 5:24

Thanks. It is generally a problem: people don't look for things if they are not aware they exist. A lot of my research is focused on the question of whether people investigate method docs to see if they convey anything important - they don't, which leads to serious problems. – Uri Dec 24, 2008 at 5:37

13

Ugh, not being able to quickly comment out a line (or lines) during testing sounds annoying to me, especially when scripting.

Share Improve this answer Follow

answered Dec 24, 2008 at 6:00

community wiki
J Cooper

Python handles that quite nicely. Just wrap your "dead code" block in triple-quotes, and the interpreter interprets it as a string literal that does not get used. Even C and C++ handle it with the preprocessor. Comments aren't necessary to temporarily suppress code. – Tom Dec 24, 2008 at 20:49

Why would I want to type 6 chars instead of two? – Matthew Flaschen Dec 25, 2008 at 18:47

9

In general comments are a wart that indicates poor design, especially long rambling comments where its clear the developer didn't have a clue what the heck they where doing and tried to make up for it by writing a comment.

Places where comments are useful:

- Leaving a ticket number next to a fix so future programmers can understand business requirements
- Explaining a particularly tricky hack
- Commentary on business logic for a piece of code
- Terse descriptions in API docs so a third-party can use your API

In all circumstances programmers should endeavor to write code that is descriptive and NOT write comments that describe poorly written code. That being said I think there are plenty of valid reasons that languages should and must support comments.

Share Improve this answer Follow

answered Dec 24, 2008 at 5:21

community wiki
mcrute

+1 for real examples, especially for the ticket (and in relation TODOs) – Robert Gould Dec 24, 2008 at 5:29

tickets and TODOs can be incorporated into the language. In fact that would be awesome. – Mark Cidade Dec 24, 2008 at 5:48

Your right Marxidad, TODO and tickets could be done similar to Deprecatation, great idea will incorporate that into my design – Robert Gould Dec 24, 2008 at 5:54

Another point to add to your list is external references such as "Numerical Recipes in C, chapter 4" or "example.com/awesome_algorithm.html" – Tom Dec 24, 2008 at 20:47

5 I fundamentally disagree that comments are generally indicative of poor design. Certainly, comments don't substitute for good code, but they are an important complement. You acknowledge this later in your post, because "Commentary on business logic for a piece of code" fits much useful commenting. – Matthew Flaschen Dec 26, 2008 at 11:07

7

Your code has two distinct audiences:

- The compiler
- Human beings like us

If you choose to remove comments altogether, the assumption you are taking is that you will be catering only to the compiler, and to nothing else.

Of course you, being Klingon, may not need comments because you are not human. Perhaps you could clearly demonstrate to us your ability by speaking in IL instead?

Share Improve this answer Follow

edited Dec 24, 2008 at 5:36

community wiki
2 revs
Jon Limjap

5

You don't *need* a single assertion in your code because, in release mode, they're all gone. But when C++ didn't have assertions built-in, someone wrote the assert macro to replace it.

Of course you don't *need* comments, either, for more or less the same reason. But if you design a language without comments, people will start doing things like:

```
HelperFunctionDoesNothing("This is a comment! Blah
Blah Blah...");
```

Share Improve this answer Follow

answered Dec 24, 2008 at 7:43

community wiki
Daniel Daranas

4

I'm curious. How do you stop someone from declaring a static string containing a comment and then ignoring the variable for the rest of the func/method/procedure/battle/whatever?

```
var useless_comment = "Can we destroy our
enemies?"
if (phasers on full) return Qapla'
```

Share Improve this answer Follow

answered Dec 24, 2008 at 8:26

community wiki
jmucchiello

can't stop 'em ;) But then again we can't stop lots of stuff – Robert Gould Dec 24, 2008 at 8:31

3

Languages need comments. At least 95% of comments can be replaced by clearer code but there are still assumptions you need to document and you absolutely need to document if there's some external problem you are working around.

I never write a comment without first considering if I can change the code to eliminate the need for it but sometimes you can't.

Share Improve this answer Follow

answered Dec 24, 2008 at 5:17

community wiki
Loren Pechtel

3

While all source code is copyrighted by default. It is often nice to:

1. remind the person reading the source code that it is subject to copyright
2. tell people what the licensing terms are for that source code file
3. tell them whether or not they are looking at a protected trade secret

Unfortunately, without comments, it is difficult to do this.

Share Improve this answer Follow

answered Dec 24, 2008 at 7:30

community wiki
Juan

Am I the only one who comments out a couple of lines **code** for a number of purposes?

3

Share Improve this answer Follow

answered Dec 24, 2008 at 9:33

community wiki
Tamas Czinege

Do Klingons not use version control either? :-) – richq Dec 24, 2008 at 11:36

It's going to be harder than you think to make a language where comments are impossible.

3

```
if (false) {
    print("This is a comment. Chew on that, Klingons!")
}
```

Share Improve this answer Follow

answered Jul 6, 2009 at 22:27

community wiki
ojrac

You really don't EVER want to print out such a rude comment to an innocent user. I would add an aditional check: if (false && true) { "Hey Klingons, all your bases are belong to us!"; } – Stephane Gosselin Aug 5, 2020 at 6:56

2

While it's true that humans need to be able to comment code, it is not absolutely necessary that the language directly support commenting: for most languages, it would be trivial to write a script that deletes one line comments (for example, all lines beginning with '#' or some other character) then runs the compiler.

Actually, though, I am surprised and disappointed to learn that even my favorite esoteric programming languages support comments: Brainf**k and Whitespace. These languages are meant to be hard to read, so it seems like they shouldn't support commenting. (As opposed to my other favorite esoteric language: LOLCode, which is meant to be self-documenting, in lolcats-speech)

I would dissent from the other answerers on this point: I say, be true to your vision of a Klingon programming language, and do not support comments!

Share Improve this answer Follow

edited Dec 24, 2008 at 5:32

community wiki
2 revs
Colin

I totally agree, both boast of their ability to write comments freely in code :(– Robert Gould Dec 24, 2008 at 5:31

2

A point against comments is that they tend to often fall out of date with the code. Any time you add a redundancy, you're risking this sort of inconsistency.

There's actually some interesting research that I've seen when a group used NLP to analyze locking comments in some large system and then compare them to the results of static analysis and were able to fix a few bugs that way.

Share Improve this answer Follow

answered Dec 24, 2008 at 5:39

community wiki
Uri

1 Got a URL for that, perchance? – TraumaPony Dec 24, 2008 at 9:03

And what's NLP? Neuro-Linguistic Progrmming? – innaM Dec 24, 2008 at 10:16

NLP is Natural Language Processing. Any technique that tries to automatically read natural language. I'll have to dig up the reference and post it. – Uri Dec 24, 2008 at 18:10

2

Isn't literate programming as much comment as it is code? Certainly, much of what I've seen of literate programming has as much explanation as code, if not more comment.

Share Improve this answer

answered Dec 24, 2008 at 6:27

Follow

community wiki
Jonathan Leffler

▲

1

▼

You might think that developers writing in your language will make an extra effort to write clear code but the onus will actually be on **you** to design a language that's *so* expressive that it doesn't need to be commented. Hell, not even English is like that (we still parenthesize!). If your language isn't so designed it may very well be as usable as Brainfuck and enjoy the popularity and respect of Brainfuck.

Should I add links or are links considered commentlike?

Besides, people will find ways to add comments if they need to by highjacking strings and misusing variable names (that do nothing other than stand in for comments). Have you read *Godel Escher Bach*?

Share Improve this answer

Follow

edited Dec 24, 2008 at 5:22

community wiki
2 revs
Mark Cidade

---

well respectability isn't the main issue here, but besides that I can't think of any language feature that requires expressivity in comments, do you have any good examples? – Robert Gould Dec 24, 2008 at 5:14

AFAIK, comments are ignored by all languages parsers, since they're used for writing in natural languages, which are expressive as we can get but are too context-dependant for writing deterministic parsers for. – Mark Cidade Dec 24, 2008 at 5:19

Haven't read GED, but seems like a good read. Gonna order it :) – Robert Gould Dec 24, 2008 at 6:58

1

It will be a bad idea to remove the commenting facility altogether. Surely developers must learn to write code with minimum comments i.e. to write self documenting code but there a lot of cases where one has to explain why something is being done the way it is. Consider the following cases:

- a new developer might start maintaining the code and the original dev has left/ out of the project
- a change in specification or market requirement leads to something that is counter intuitive
- copy right notice especially if open source (some open source libs require you to do this)

It is also my experience that new programmers tend to comment more and as they develop expertise their code tends to become self documenting and concise. In general comments should be about WHY and not HOW or WHAT.

Share Improve this answer

Follow

answered Dec 24, 2008 at 5:29

community wiki
Sesh

1

NO -- there is not a single programming language out there that requires comments.

The language is for the computer. The comments are for the humans. You can write a program with 0% comments. It'll execute, rightly or wrongly. You can't write a program with 100% comments. It'll either not compile -- no main(), etc. -- or, for scripting languages, do exactly nothing.

And, besides, real programmers don't comment their code. Just like Klingons.

Share Improve this answer

Follow

answered Dec 24, 2008 at 5:38

community wiki
Jim Nelson

1

While I agree with Uri's responses, I too have made a language with no comments. (ichbins.) The language was to be as simple as possible while still being able to express its own compiler cleanly; since you can do that without comments, they got jettisoned.

I'm working off and on on a revision that does support commentary, but a bit differently: literate-programming style with code nested in text instead of comments embedded in code. It might also get examples/test-cases later as a first-class language feature.

Good luck with the Klingon hacking. :-)

Share Improve this answer Follow

answered Dec 24, 2008 at 6:24

community wiki
Darius Bacon

1

I can't tell you how thankful I am for Javadoc - which is really simple to set up within comments. So that's at least one sense in which comments are useful.

Share Improve this answer Follow

answered Dec 24, 2008 at 6:28

community wiki
troyal

1

No, of course a language doesn't have to have commenting. But a (useful) program does have to have comments... I don't agree with your idea that literate code lacks comments. Some very good code is easily comprehensible with comments, but only with difficulty without.

Share Improve this answer Follow

answered Dec 24, 2008 at 7:03

community wiki
Matthew Flaschen

1

I think the comments are required in many situations.

For instance, think of the algorithmic ones. Suppose there is a function written in C which solves the Traveling Salesperson Problem, there are wide range of techniques that can be used to deal with this problem. And the codes are usually cryptic by its nature.

Without explicitly describing the parameters and the algorithm used, by using comments, it is almost impossible to reuse this piece of code.

Share Improve this answer Follow

answered Dec 24, 2008 at 9:32

community wiki
Gant

1

Can we live without comments on code? Sure, but that won't make live easier.

Share Improve this answer Follow

answered Dec 24, 2008 at 10:34

community wiki
Eldelshell

1

Comments are useful because they reassure the person reading your code - probably the "future you" - that you've thought about her welfare.

Share Improve this answer Follow

answered Jul 6, 2009 at 22:22

community wiki
Avi

Surely only a Human would think about the maintainers Welfare, Klingons wouldn't bother with that. But as a Human it does make sense :) – Robert Gould Jul 7, 2009 at 2:16

0

I think the question may become how self-contained would the language without comments be? If for example, it compiles down to DLLs that get used within other code, then how does one know anything beyond the function signature in terms of what it requires, changes and returns? I wouldn't want to have function names being dozens of characters to try to express what may be very easily done with comments above the function that can be used as documentation within something like the Object Browser within Visual Studio for example.

Share Improve this answer Follow

answered Dec 24, 2008 at 5:17

community wiki
JB King

0

Of course!!

The main reason is novice developers. Not everyone knows how to write literate code. Actually there are millions out there don't get a NullPointerException when they see one.

We all start at some point.

But if you're targeting to "expert" developers only, why bother in the language in first place. You should be using butterflies !!! That's what real developer use!

Comments is a must, try to make it harder if you wish (like using #//##/ sequence to create a comment or something like that) but don't leave it out.

:)

Share Improve this answer Follow

answered Dec 24, 2008 at 5:30

community wiki
OscarRyz

0

I agree with you that nicely written code does not need any comments as "Code is only good documentation available to programmer. However this is very ideal condition, not everyone writes good code all time. So to make poorly written code good in future comments are required.

Share Improve this answer Follow

answered Dec 24, 2008 at 6:27

community wiki
JRomio

0

I once wrote a VB app (a silly board game inspired by Monopoly) without *any* comments. But I did that just to piss off my teacher, who had told us comments were for "whatever we found *relevant*, so we could remember it later".

Share Improve this answer Follow

answered Dec 24, 2008 at 6:58

community wiki
isekaijin

0

Perfect code needs zero comments. It should be simple, and understandible by complete novices.

Share Improve this answer Follow

answered Dec 24, 2008 at 9:35

community wiki
Ali Afshar

0

Any code needs comments, I try to explain the reason for and workings of every function I write in 1 or 2 lines.

Code that explains itself only exists in a perfect world, there is always some weird hack or a reason to do something quick-n-dirty instead of the propper way. The best thing to remember is to comment WHY code does what it does, good code explains WHAT it does 99% of the time.

Write something simple, like a piece of code that can solve a Sudoku puzzle (3 reasonably simple while loops) and try reading that 3 months later. You will immidiatly find something that isn't exactly clear.

Share Improve this answer Follow

answered Dec 24, 2008 at 10:01

community wiki
user48864

0

Code is written once, but read many times over the course of its lifetime; thus it pays to optimize for readability. Clear and consistent naming of everything from constants to classes is necessary, but may or may not be sufficient to achieve this objective. If not, fill in the gaps with comments, and maintain them as you would the code.

Share Improve this answer Follow

answered Dec 26, 2008 at 6:11

community wiki
Dave

1 2 Next