

Good challenges/tasks/exercises for learning or improving object oriented programming (OOP) skills [closed]

Asked 16 years, 3 months ago Modified 3 years, 2 months ago

Viewed 84k times



90



Closed. This question does not meet [Stack Overflow guidelines](#). It is not currently accepting answers.

💡 Questions asking us to **recommend or find a tool, library or favorite off-site resource** are off-topic for Stack Overflow as they tend to attract opinionated answers and spam. Instead, [describe the problem](#) and what has been done so far to solve it.

Closed 11 years ago.

[Improve this question](#)

What is a good challenge to improve your skills in object oriented programming?

The idea behind this poll is to provide an idea of which exercises are useful for learning OOP.

The challenge should be as language agnostic as possible, requiring either little or no use of specific libraries, or only the most common of libraries. Try to

include only one challenge per answer, so that a vote will correspond to the merits of that challenge alone. Would also be nice if a level of skill required was indicated, and also the rationale behind why it is a useful exercise.

Solutions to the challenges could then be posted as answers to a "How to..." question and linked to from here.

For example:

- Challenge - implement a last-in-first-out stack
- Skill level - beginner
- Rationale - gives experience of how to reference objects

oop

Share

Improve this question

Follow

asked Sep 12, 2008 at 22:45



Grundleleck

129k ● 25 ● 95 ● 112

My college had an exercise that was mandatory: Write a Java media player to play mp3 files. It was an assignment over multiple weeks, which was great for beginners. There were 8 chapters with sample solutions for each, also unit tests. ~70% of the code was always given and you had roughly a dozen quite specific tasks per chapter. Simple UI, file access, storing settings, using inheritance & polymorphism, adding your own classes, obviously loops and conditions. It had everything you wanted to start out. Sadly it is copy right

protected - ___ - I even added my own features after I was done. – [Heinzlmaen](#) Jun 10, 2020 at 12:37

8 Answers

Sorted by:

Highest score (default)



[Building Skills in Object-Oriented Design](#) is a free book that might be of use.

49

The description is as follows:



"The intent of this book is to help the beginning designer by giving them a sequence of interesting and moderately complex exercises in OO design. This book can also help managers develop a level of comfort with the process of OO software development. The applications we will build are a step above trivial, and will require some careful thought and design. Further, because the applications are largely recreational in nature, they are interesting and engaging. This book allows the reader to explore the processes and artifacts of OO design before project deadlines make good design seem impossible."



Share Improve this answer

Follow

edited Oct 8, 2021 at 3:39



[mhadidg](#)

1,804 ● 20 ● 30

answered May 13, 2009 at 9:38



[Craig Angus](#)

23.1k ● 18 ● 57 ● 64



18

Write a challenging program **from scratch**. Try to get some people (around five, that should be doable) to use it. Respond to their change requests.



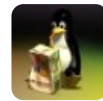
Adapt your program's design. Start small, then watch it grow. Manage this growth. **This is hard**. You will also have to fix bugs and maintain the thing over time, which for me was a very valuable lesson.



Share Improve this answer

answered Sep 17, 2008 at 19:40

Follow



[Paweł Hajdan](#)

18.5k ● 9 ● 52 ● 65

7 +1 **Respond to their change requests** – [JaDogg](#) Oct 22, 2014 at 6:40



9

Challenge: Write a wrapper for your web site/service API of choice in your language of choice, that doesn't already exist (ex. a [ZenDesk API](#) wrapper written in C#). Release the wrapper as open source for others to use.



Skill Level: Beginner to Intermediate



Rationale: To learn how to extrapolate a 3rd party web service API into a meaningful set of objects/classes, making the reuse of that API easier in your chosen language.

Share Improve this answer

answered Sep 12, 2008 at 23:07

Follow



busse

1,777 ● 14 ● 25



After you have learned the basics, study the "Gang of four" design patterns book.

5



http://www.amazon.com/Design-Patterns-Object-Oriented-Addison-Wesley-Professional/dp/0201633612/ref=pd_bbs_sr_1?ie=UTF8&s=books&qid=1221488916&sr=8-1



This is a classic, and a must read for any coder who wants to understand how to use OO to design elegant solutions to common coding problems.

Share Improve this answer

answered Sep 15, 2008 at 14:31

Follow



killdash10

716 ● 5 ● 12

7 With all due respect to the GOF, "Head first in design patterns" is a much more enjoyable read. – [CaptainHastings](#) Jul 24, 2009 at 20:58

2 GOF is the ultimate and we should not fear about it. Read it everyday before go to sleep. – [kta](#) Mar 24, 2014 at 4:45 ✎



3

Take a procedural-style written piece of code and try to transform it into OOP based solution. During the process, consult a book on refactoring and design patterns. A friend of mine was able to make a huge step forward in



understanding object oriented concepts exactly this way.
As with anything, this might not work for everyone.



Share Improve this answer

answered Sep 17, 2008 at 13:57

Follow



[petr k.](#)

8,100 ● 7 ● 43 ● 52



I have found [CRC cards](#) to be quite effective in learning, teaching and building good OO design.

2

Share Improve this answer

answered Sep 17, 2008 at 14:21

Follow



[swamy](#)

113 ● 2 ● 4





1



Certainly a good challenge, although less accessible than a "start from scratch" assignment, is to refactor some existing code that either doesn't use inheritance or doesn't use very much of it to make greater use of inheritance. The process of refactoring will expose a lot of the benefits and gotchas of oop, as it certainly has for me on my most recent project. It also pushed me to understand the concepts better than past projects have where I've created my own object oriented designs.

Share Improve this answer

answered Sep 12, 2008 at 22:54

Follow



Luke

19k ● 24 ● 88 ● 112



A given task has very little to do with being "OOP", it's more in how you grade it.

0



I would look at the Refactoring book, chapter 3, and make sure none of the bad code smells exist in the solution. Or, more importantly, go over ones that do apply.



Most importantly, watch for the existence of setters and getters (indicating that you are operating on values from a class and not asking the class to operate on it's own values)--or using "extends" without applying the Liskov Substitution Principle, stuff like that.

[Share](#) [Improve this answer](#)

answered Sep 16, 2008 at 21:06

[Follow](#)



[Bill K](#)

62.8k ● 18 ● 112 ● 158
