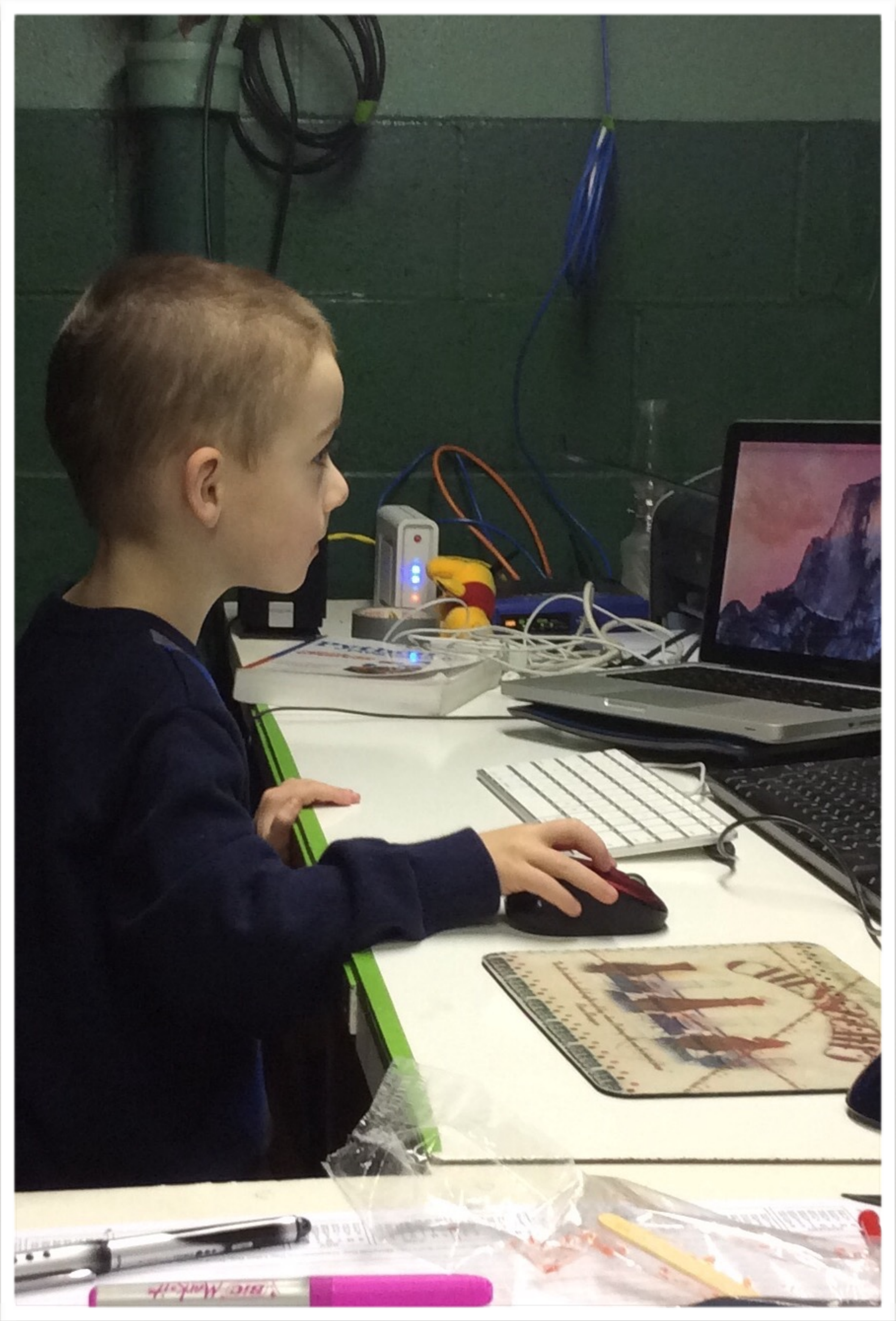
Teaching Youth to Code

Caleb Salt
Pyowa

March 07, 2017

Computerification

“Ain’t too young”



Challenges

Vocabulary

variable

class

polymorphism

P vs NP

cryptography

parallelism

instance

agile

api

semantic versioning

hash table

abend

boolean

instantiate

waterfall

parameter

concurrency

brute force

recursion

stack

big O

abstract base class

heap

algorithm

pure virtual function

allocate

root kit

Programming Environment

IDE? Editor?

- syntax highlighting
- auto indent
- autocomplete
- debugging
- linting
- code execution
- project organization
- snippets

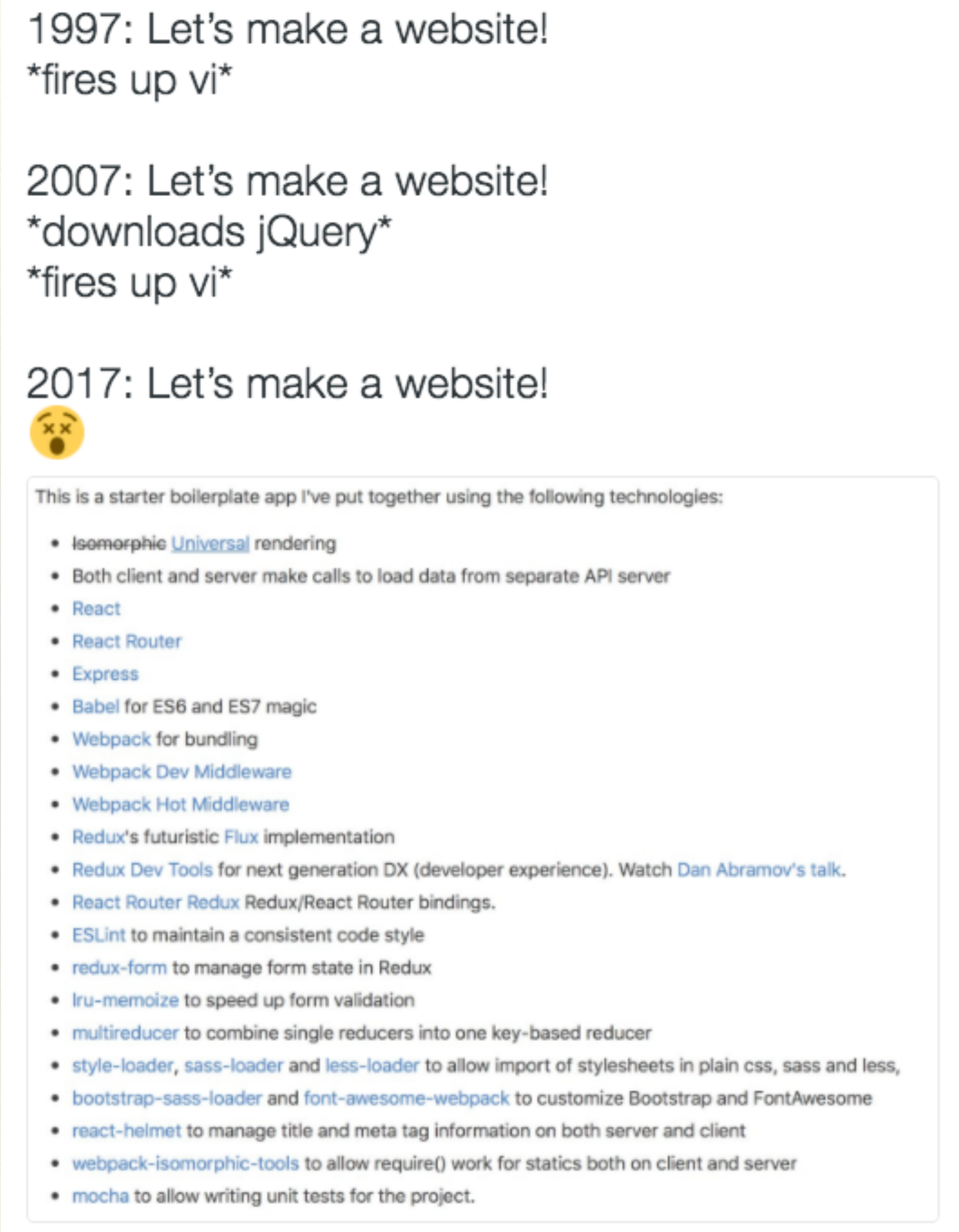
PC
eclipse
Notepad++
gedit
XCODE
S

Or You’re

WRONG

Scope

- Language
- Design Patterns
- Algorithms
- All the other stuff



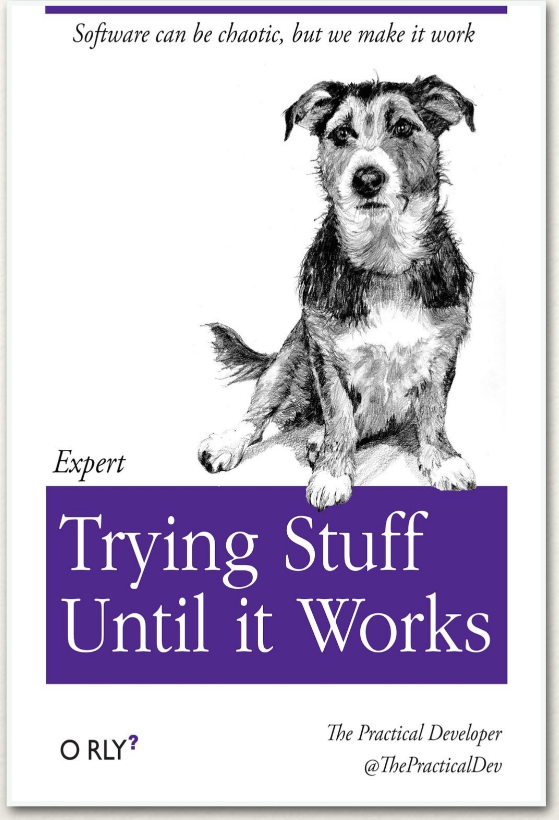
1997: Let's make a website!
fires up vi

2007: Let's make a website!
downloads jQuery
fires up vi

2017: Let's make a website!
😵

This is a starter boilerplate app I've put together using the following technologies:

- ~~Isomorphic~~ Universal rendering
- Both client and server make calls to load data from separate API server
- React
- React Router
- Express
- Babel for ES6 and ES7 magic
- Webpack for bundling
- Webpack Dev Middleware
- Webpack Hot Middleware
- Redux's futuristic Flux implementation
- Redux Dev Tools for next generation DX (developer experience). Watch Dan Abramov's talk.
- React Router Redux Redux/React Router bindings.
- ESLint to maintain a consistent code style
- redux-form to manage form state in Redux
- lru-memoize to speed up form validation
- multireducer to combine single reducers into one key-based reducer
- style-loader, sass-loader and less-loader to allow import of stylesheets in plain css, sass and less,
- bootstrap-sass-loader and font-awesome-webpack to customize Bootstrap and FontAwesome
- react-helmet to manage title and meta tag information on both server and client
- webpack-isomorphic-tools to allow require() work for statics both on client and server
- mocha to allow writing unit tests for the project.



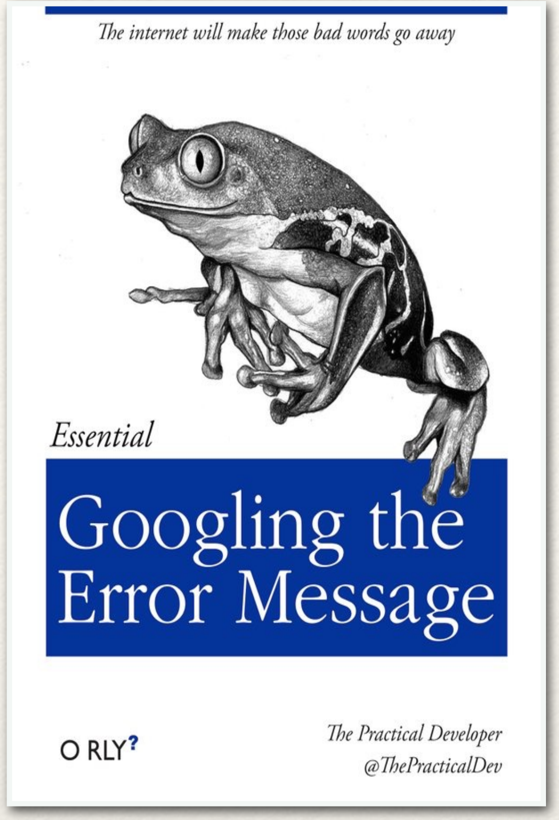
Software can be chaotic, but we make it work

Expert

Trying Stuff Until it Works

O RLY?

The Practical Developer
@ThePracticalDev



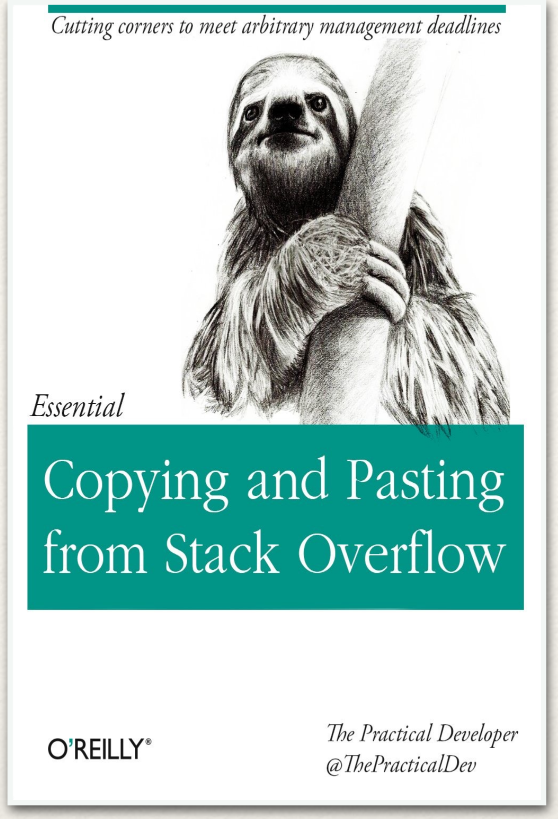
The internet will make those bad words go away

Essential

Googling the Error Message

O RLY?

The Practical Developer
@ThePracticalDev



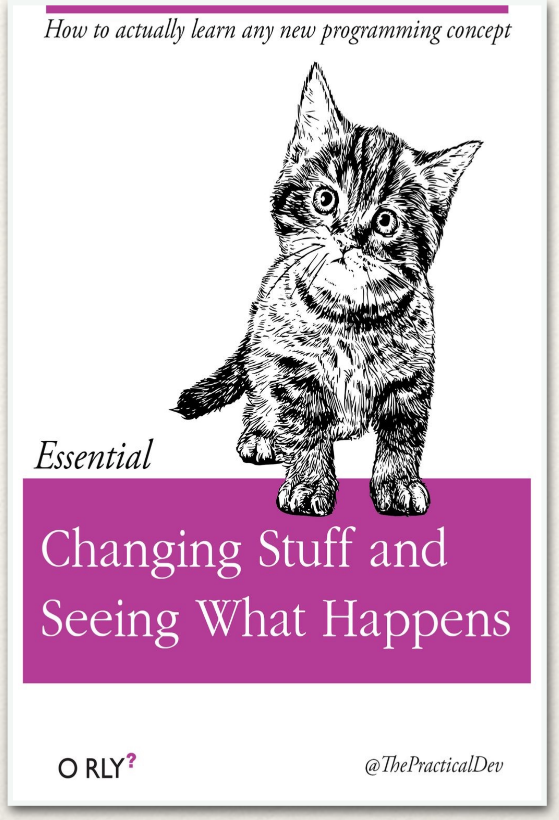
Cutting corners to meet arbitrary management deadlines

Essential

Copying and Pasting from Stack Overflow

O'REILLY®

The Practical Developer
@ThePracticalDev



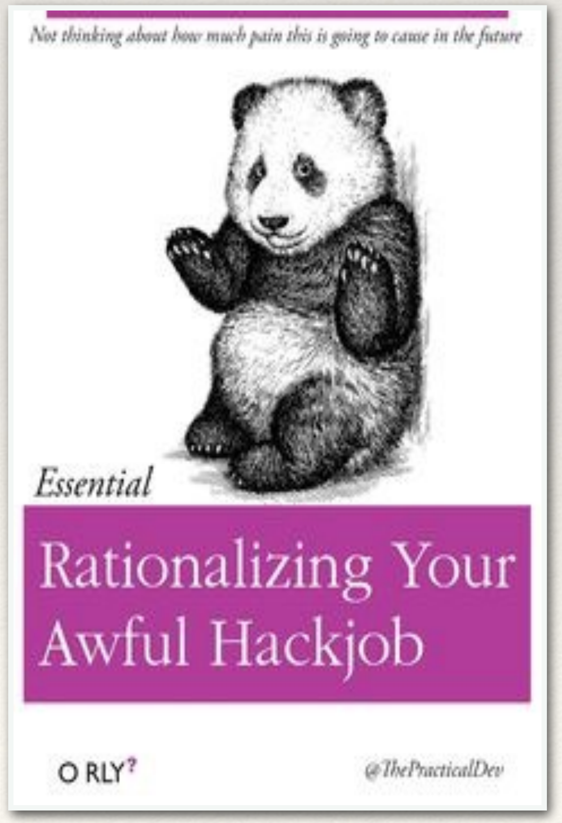
How to actually learn any new programming concept

Essential

Changing Stuff and Seeing What Happens

O RLY?

@ThePracticalDev



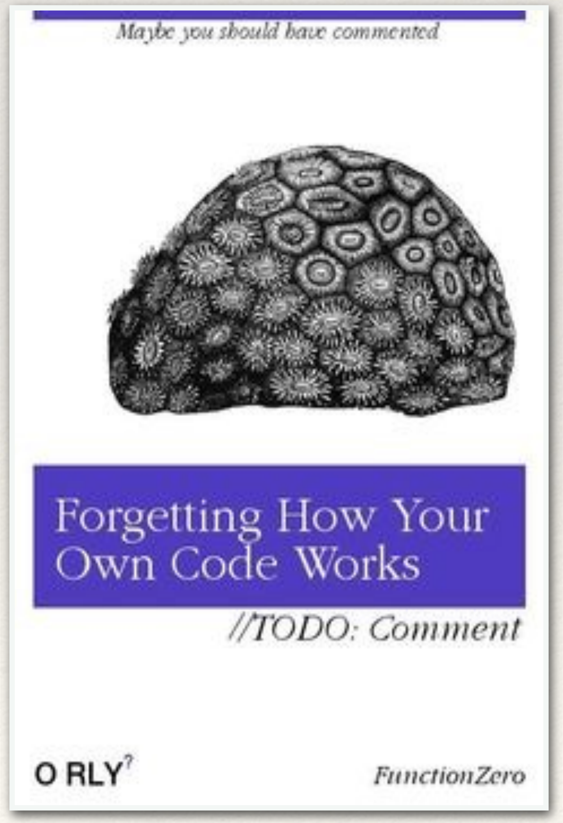
Not thinking about how much pain this is going to cause in the future

Essential

Rationalizing Your Awful Hackjob

O RLY?

@ThePracticalDev



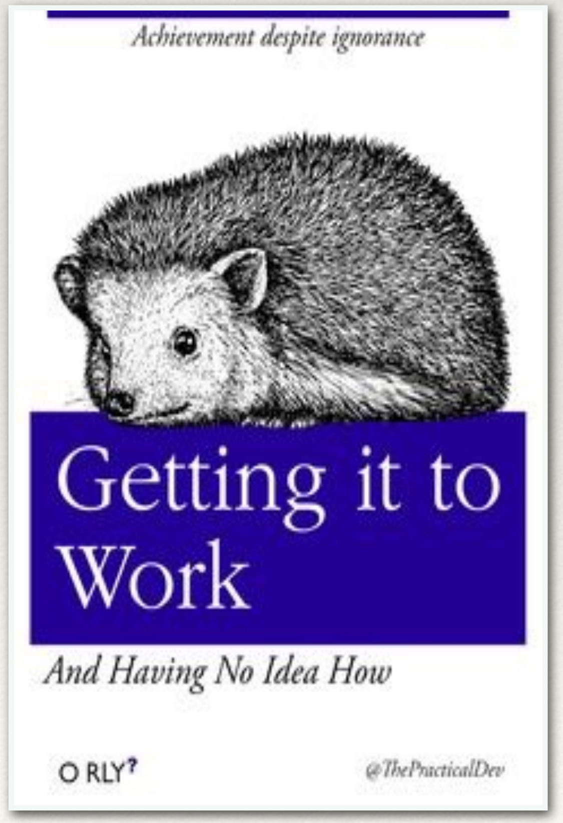
Maybe you should have commented

Forgetting How Your Own Code Works

//TODO: Comment

O RLY?

FunctionZero



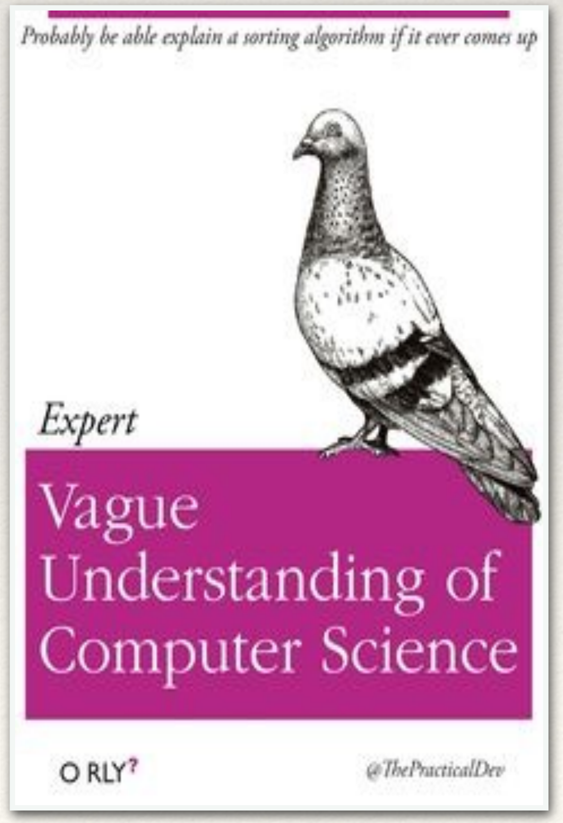
Achievement despite ignorance

Getting it to Work

And Having No Idea How

O RLY?

@ThePracticalDev



Probably be able explain a sorting algorithm if it ever comes up

Expert

Vague Understanding of Computer Science

O RLY?

@ThePracticalDev

Methodology

Computer Sciencey

- Lecture
- Topical
- Textbook
- Problem sets

Projects

Assistance

Books

- inventwithpython.com
  - Automate the Boring Stuff With Python
  - Scratch Programming Playground
  - Invent Your own Computer Games with Python
  - Hacking Secret Ciphers with Python
- Learn Python the Hard Way
- Python Programming for the Absolute Beginner

Practice

coursera

CheckiO

Learning Python
Through Python Koans
https://github.com/gregmalcolm/python_koans

RSI
Runestone Interactive

The
Hitchhiker's
Guide
to
Python

Podcasts

- Talk Python to Me (https://talkpython.fm/)
  - #8 Teaching Python at Grok Learning and Classrooms
  - #19 Automate the Boring Stuff with Python
  - #20 Interactive Python and Teaching Python: Beyond Text Books
  - #27 Four Years of Python for High Schoolers
- CodeNewbie (http://www.codenewbie.org/podcast)
- The Changelog (https://changelog.com/)
  - #195 Free Code Camp

Where to go from here?

Hour of Code

Community Teaching Day

“Tell me and I forget.
Teach me and I remember.
Involve me and I learn.”

–Benjamin Franklin

TE
ONAPARTE

Editors

Pycharm https://www.jetbrains.com/pycharm-edu/

Komodo http://www.activestate.com/komodo-ide

Notepad++ https://notepad-plus-plus.org/

Xcode https://developer.apple.com/xcode/

Sublime Text https://www.sublimetext.com/3

Atom https://atom.io/

Visual Studio Code https://code.visualstudio.com/

Eclipse https://eclipse.org/

gedit https://wiki.gnome.org/Apps/Gedit

TextWrangler http://www.barebones.com/products/textwrangler/

IDLE

IDLE https://docs.python.org/3.6/library/idle.html

Quotes

- ## Daniel Procida https://github.com/evildmp

  “The Programmer’s Body” DjangoCon Europe 2014
  Programmers are now building the world. Anyone in the world will live in the world we build. The systems that govern us. The systems that govern our relations. The way we are educated. The way we are formed. Programmers are quite important.
  https://www.youtube.com/watch?v=jjtV4yw48PE

  Also watch ‘All You need is L***’ https://www.youtube.com/watch?v=0mACXAmaL1M

- ## Jessica McKellar

  “Breaking The Rules” PyBay2016
  “Learning how to program changes the way you think about, debug, and interact with the world”
  “You learn a set of rules to build software, … then you learn that you can change the rules.”
  “Programmers master a system they know they can change.”
  “This comes from the tenets of free software.”
  “We take for granted that changing something to make it better is just a thing you do when you need to.”
  “This can and should carry over to the rest of your life.”
  https://www.youtube.com/watch?v=9UnMZYMaosw

  Also see her “A hands on Introduction to Python for Beginning Programmers” PyCon 2014 https://www.youtube.com/watch?v=MirG-vJOg04

Other Things Mentioned

- REPL (Read Evaluate Print Loop) https://docs.python.org/3.6/tutorial/interpreter.html
- PythonAnywhere https://www.pythonanywhere.com/
- Luciano Ramalho ‘Fluent Python’ http://shop.oreilly.com/product/0636920032519.do https://github.com/fluentpython
- O’Reilly HeadFirst Books http://headfirstlabs.com/

Coursera https://www.coursera.org/

CheckIO https://checkio.org/

RuneStone Interactive http://runestoneinteractive.org/

The Hitchhikers Guide to Python http://docs.python-guide.org/en/latest/

Code Combat https://codecombat.com/

Python Koans https://github.com/gregmalcolm/python_koans/wiki

Grok Learning https://groklearning.com/

Lightbot https://lightbot.com/flash.html

MIT Open Courseware https://ocw.mit.edu/index.htm

PythonBytes (podcast) https://pythonbytes.fm/

PyVideo http://pyvideo.org/

"The Young Coder: Let's Learn Python" on PyVideo: http://pyvideo.org/video/2570/the-young-coder-lets-learn-python

PyCon Youth Coding Workshop http://www.letslearnpython.com/ https://github.com/mechanicalgirl/young-coders-tutorial

Organizations in DSM

- Tech Journey http://www.techjourney.org/
- HyperStream http://changetheequation.org/hyperstream