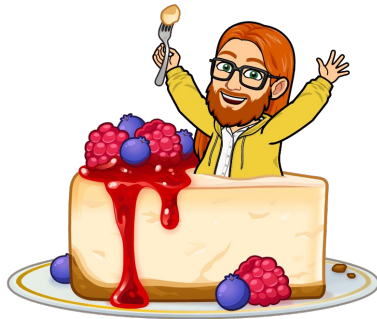
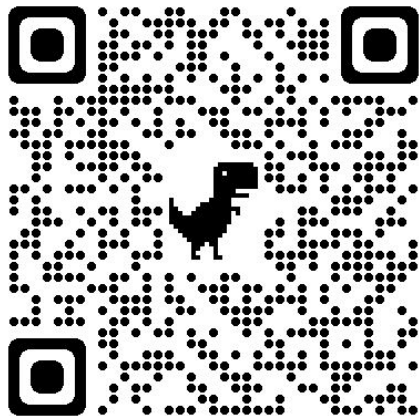


Learning to love technical interview exercises

With Ben Downen, the Full Snack Tester



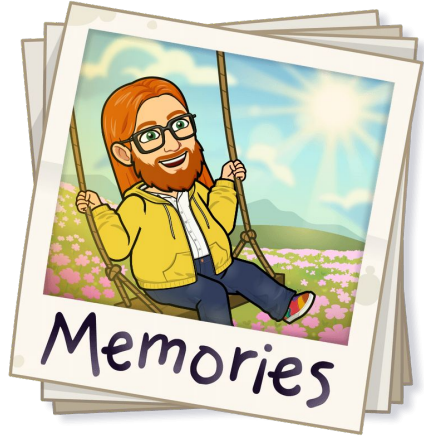
Slides



<https://github.com/dowenb/slides>



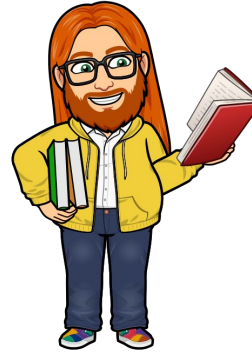
Who is Ben?



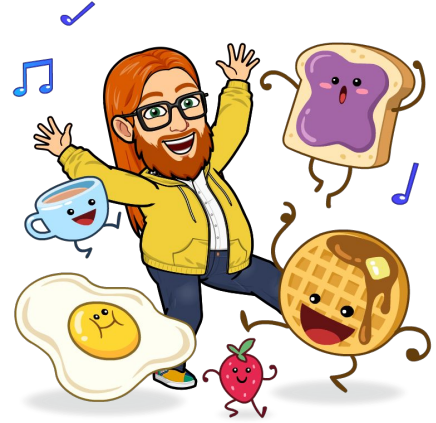
11+ Years



Love testing



Here to learn



Follow me

The Adventure




Once upon a time, I worked at Citrix



And then...



Control F1!

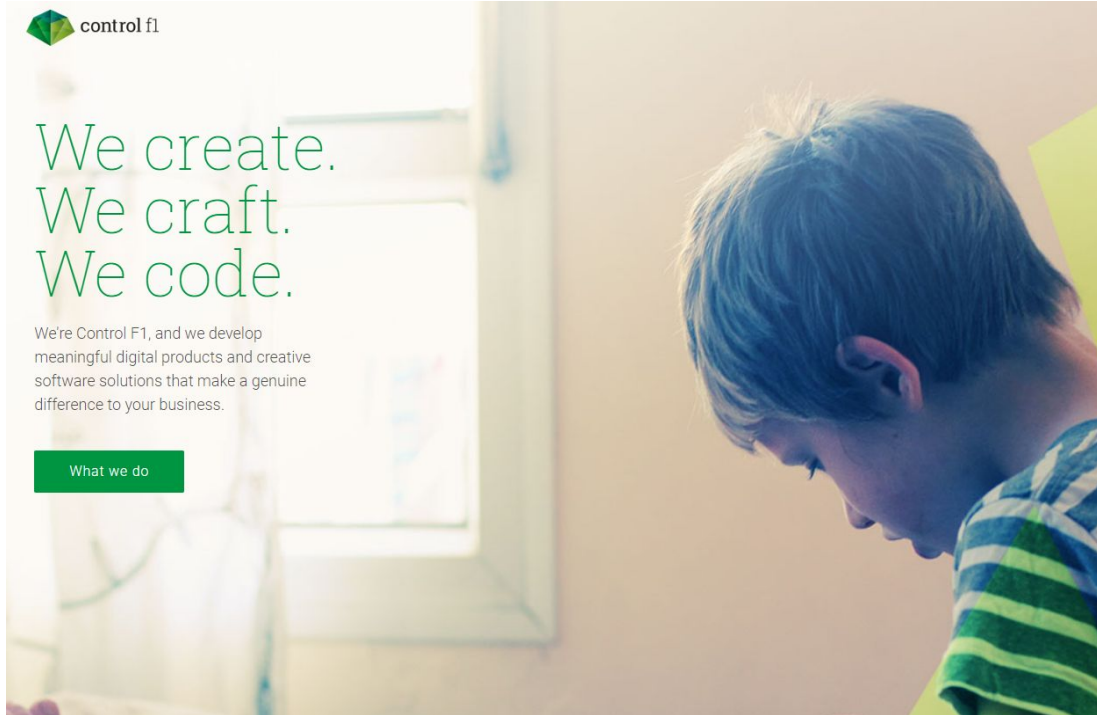


control f1

We create.
We craft.
We code.

We're Control F1, and we develop meaningful digital products and creative software solutions that make a genuine difference to your business.

What we do



And then...



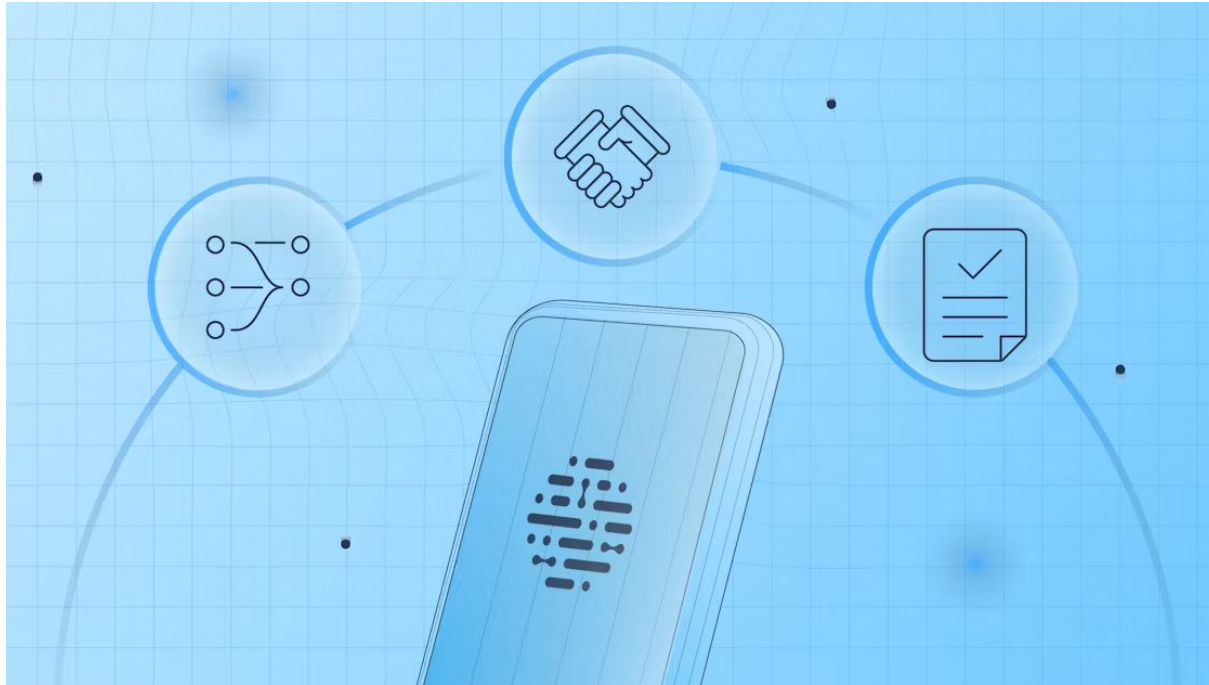
PA Media!



And then...



Ada health



And now ...



Every wants a unicorn

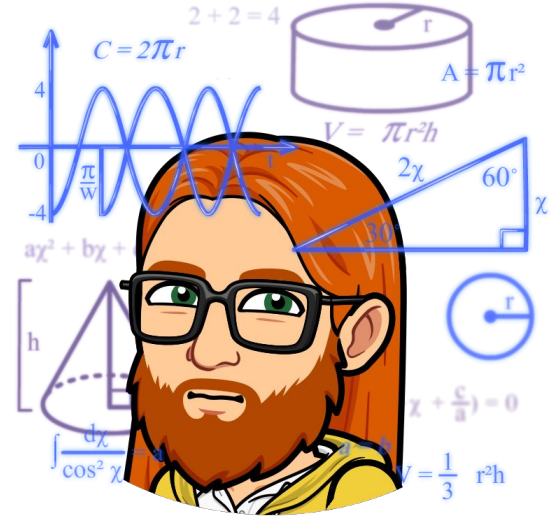


Technical Interview Exercises



Shapes and sizes

- Live or at home exercises
- Pairing or individual participation
- Panel discussion
- Writing code
- Exploratory testing
- Writing test plans or strategies
- And plenty more



You're sitting down to do that technical exercise for your test automation interview! What part do you wish you had more confidence in?

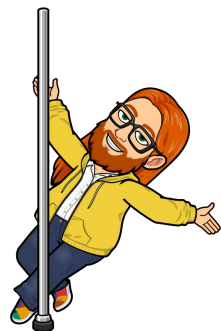
You can see how people vote. [Learn more](#)

Setting up development env 38%

Understanding what is expected 59%

Writing the test code 3%

29 votes • Poll closed



Where software lives



Creating a comfortable habitat for software

- Learn to really understand where code lives
- What is an “Environment” anyway?
- Where will the SUT run?
- Where will your automation run?
- What about supporting services?



Managing programming language versions

- Node Version Manager, NVM
- JENV, for managing Java Environments
- RBENV or RVM for Ruby
- Pyenv or many other choices for Python



Your IDE might be able to help

- JetBrains IDEs do a bunch of stuff for you
- VS Code and WSL2 is very powerful
- You might get to use your choice of IDE
- Learn your IDE's shortcuts
 - <https://code.visualstudio.com/shortcuts/keyboard-shortcuts-windows.pdf>
 - <https://code.visualstudio.com/shortcuts/keyboard-shortcuts-macos.pdf>
 - <https://www.jetbrains.com/help/idea/mastering-keyboard-shortcuts.html>



The task at hand



Break it down

- Use your critical thinking skills
- Create mind maps
- Ask clarifying questions



Implicit requirements

- Be thoughtful how you approach the task
- If you're asked to write code, make it easy to run
- Try and make your intentions explicit if you can
- Be honest if you've had help

THINKING...



There is never enough time

- Tasks might be time-boxed
- Make the most of your time
- Try not over-invest in a perfect response



Leave them wanting more

- Note down what you could do, given more time
- Highlight tech debt built up because you rushed
- Suggest future enhancements



Practice, practice, practice



Examples

<https://github.com/thestarsgroup/front-end-tech-test>

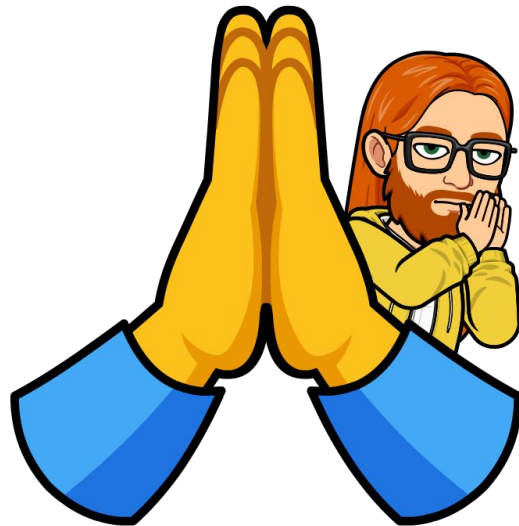
- My attempt at this is private

https://github.com/dowenb/cypress_metaweather

- This is public, I don't think they use it anymore

<https://github.com/dowenb/fast-quotes/pull/2>

- Don made simply amazing PR here!



Resources you use to practice

- <https://github.com/dowenb/fast-quotes/blob/main/exercise1.md>
- <https://www.eviltester.com/page/tools/apichallenges/>
- <https://automationpanda.com/2021/12/29/want-to-practice-test-automation-try-these-demo-sites/>



Document your journey and build a portfolio

Angie Jones

[10 PORTFOLIO PROJECTS FOR AUTOMATION ENGINEERS](#)

Beth Marshall

[5 Reasons You Should Consider Creating A Software Testing Portfolio](#)



Building practice apps



Project Snack Shop

Working with the amazing Make-Believe Collective...

... Vernon Richards, Lisi Hocke and myself

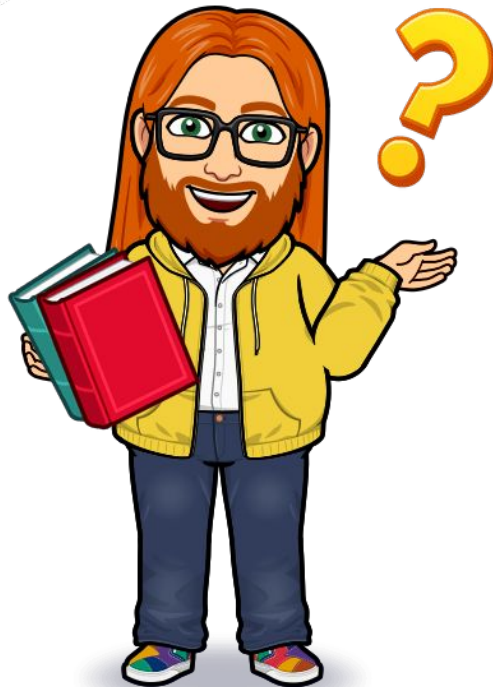
We have created this functional digital agency, and started Project Snack Shop, so we can build a practice app with some pretty unique qualities.

You will hear more soon, it's still very early days!

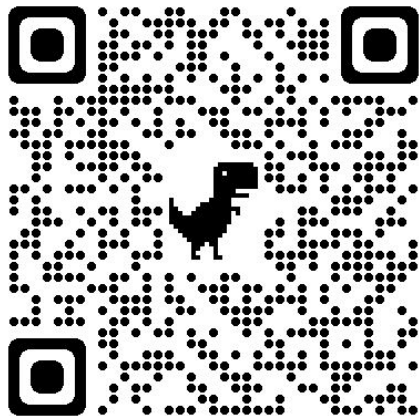




Any Questions?



Slides



<https://github.com/dowenb/slides>

