Kai Röder

Working for AEB

Storybook maintainer

@kairoeder kroeder

The Benefits of a Design System

How Storybook can improve your daily work

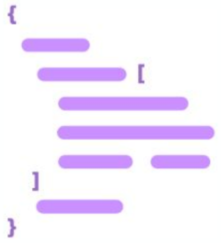
Before we start, a couple of questions...

- Who has ever made a form?
- Who made a multi-step form?
- Who had some difficulty developing a component that was buried deep in their app?

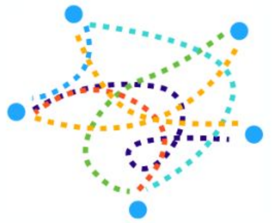
Before we start, a couple of questions...

- Who has ever made a form? I did
- Who made a multi-step form? I did
- Who had some difficulty developing a component that was buried deep in their app? I had

Component development is complicated



Flaky data in your app



Tangled business logic

Unfinished APIs

Component development is complicated

- A component might depend on a particular (server) state
  Show a <vip-offer> component only to VIP users

- A component might only appear for a short amount of time
  Show an info message popup after submitting a form

- A component might be hard to get to
  A component that is used on step 3 in a form requiring you to fill out the first two steps

How can we solve this problem?

Isolated Component Development

Develop your components in a distraction-free environment

Developing in isolation means

- **State can be mocked**
  Test your components in all different states in one place

- **Focus on the component API**
  No other components are involved

- **Put them together**
  Build a component library and raise awareness of existing components

Downsides of an isolated environment

You need an application that renders your components

This includes

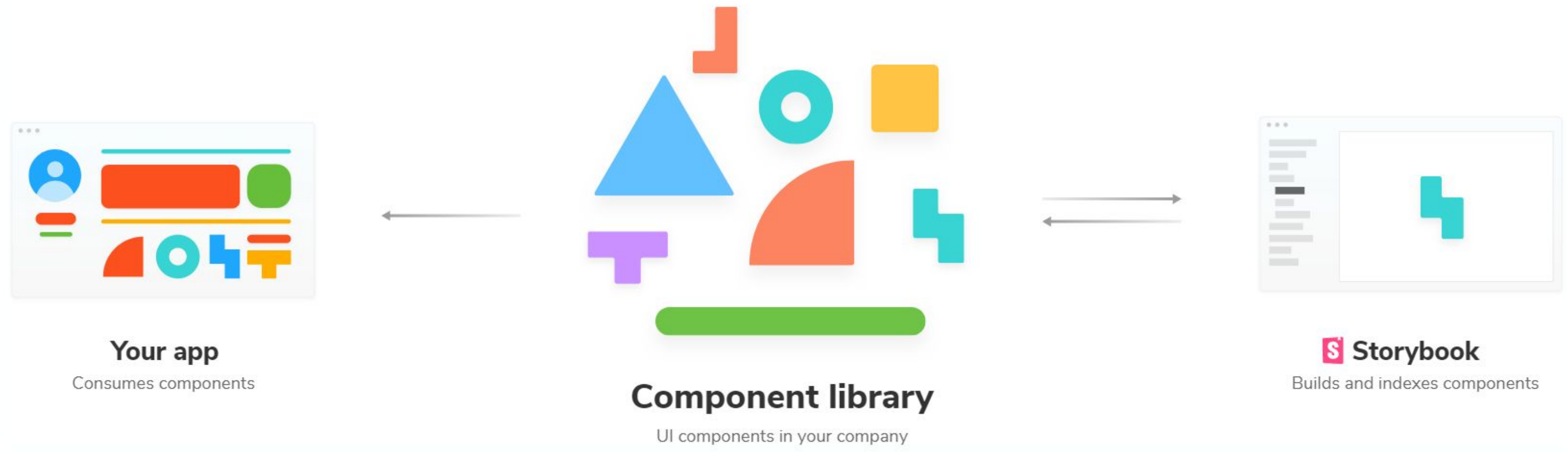
- Handling routing
- Creating a layout
- Maintenance effort for new components is huge

Storybook

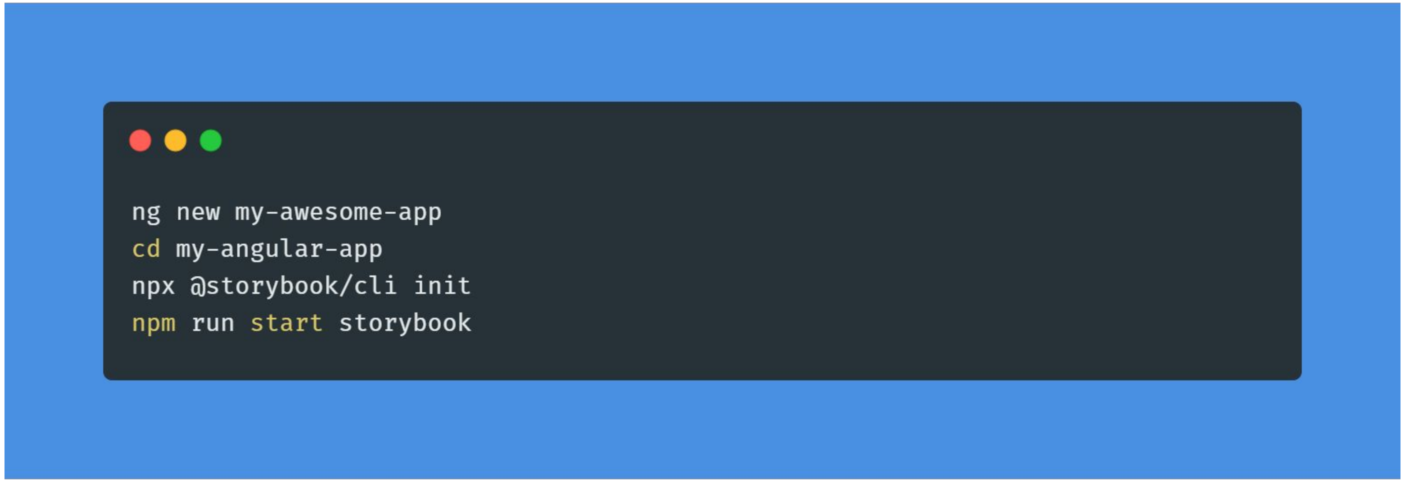
RIOT

HTML

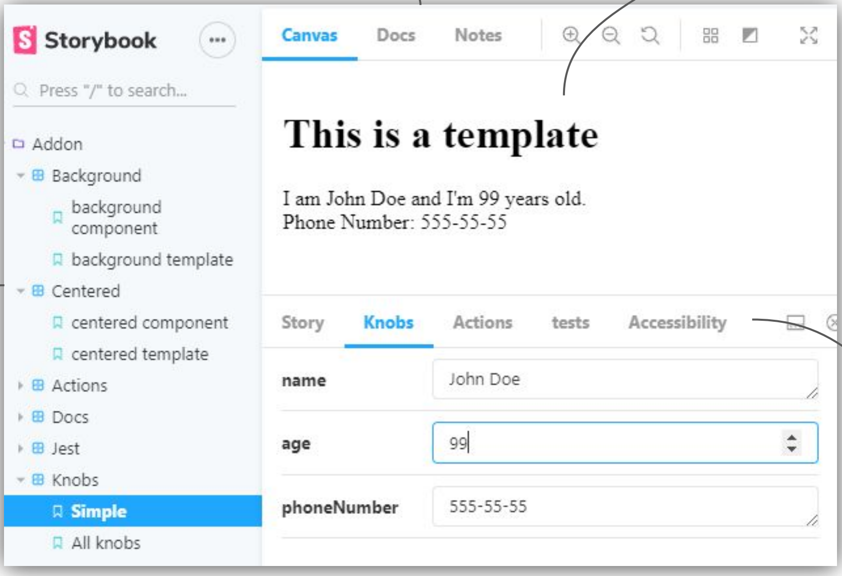
Storybook lives alongside your app



Setup Storybook in just a few steps



```
ng new my-awesome-app
cd my-angular-app
npx @storybook/cli init
npm run start storybook
```



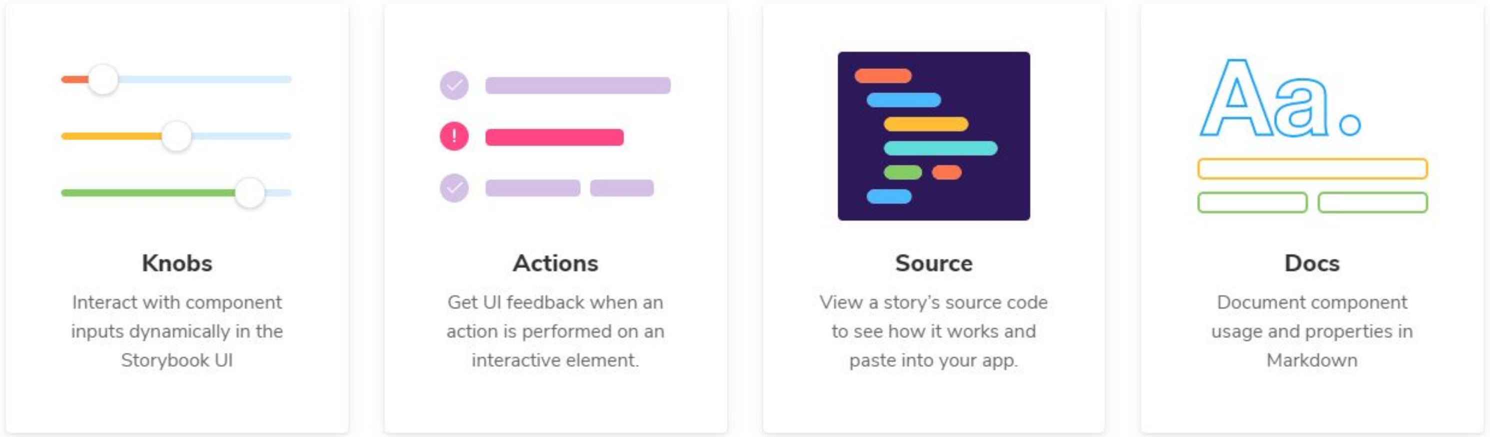
This is a template

I am John Doe and I'm 99 years old.
Phone Number: 555-55-55

name	John Doe
age	99
phoneNumber	555-55-55

https://storybookjs-next.now.sh/angular-cli

Personalize Storybook with official and community addons



Knobs

Interact with component inputs dynamically in the Storybook UI

Actions

Get UI feedback when an action is performed on an interactive element.

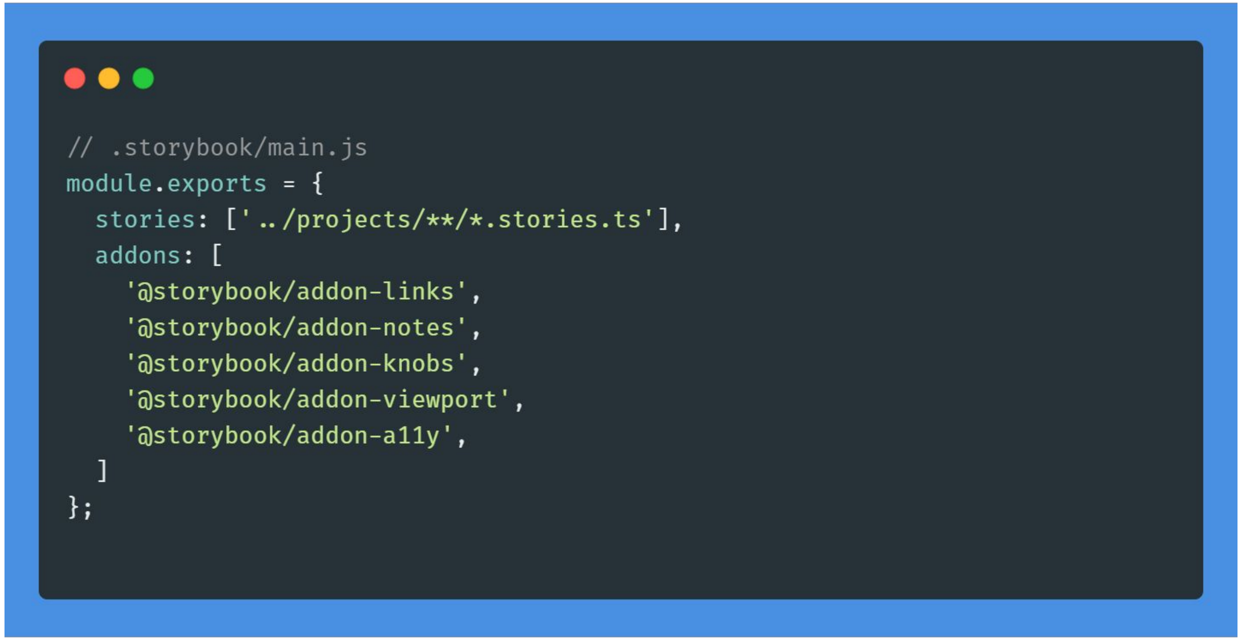
Source

View a story's source code to see how it works and paste into your app.

Docs

Document component usage and properties in Markdown

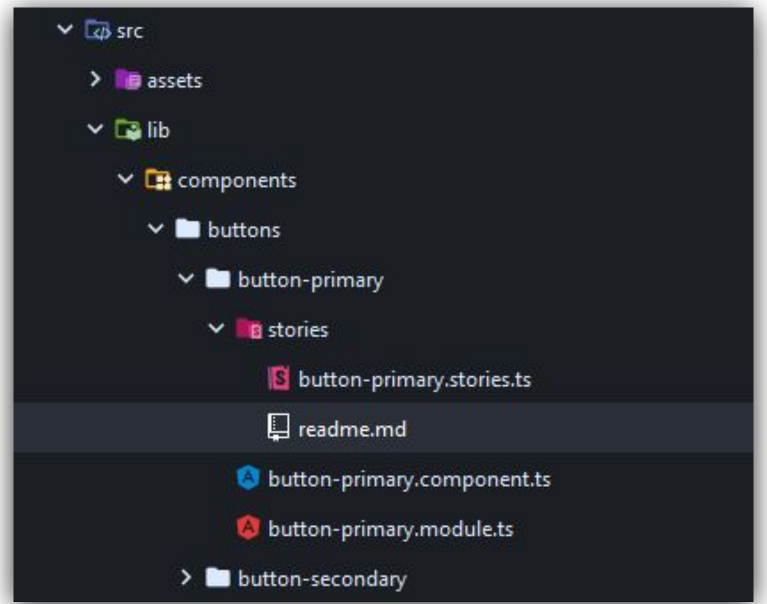
Storybook configuration



```
// .storybook/main.js
module.exports = {
  stories: ['../projects/**/*.stories.ts'],
  addons: [
    '@storybook/addon-links',
    '@storybook/addon-notes',
    '@storybook/addon-knobs',
    '@storybook/addon-viewport',
    '@storybook/addon-a11y',
  ]
};
```

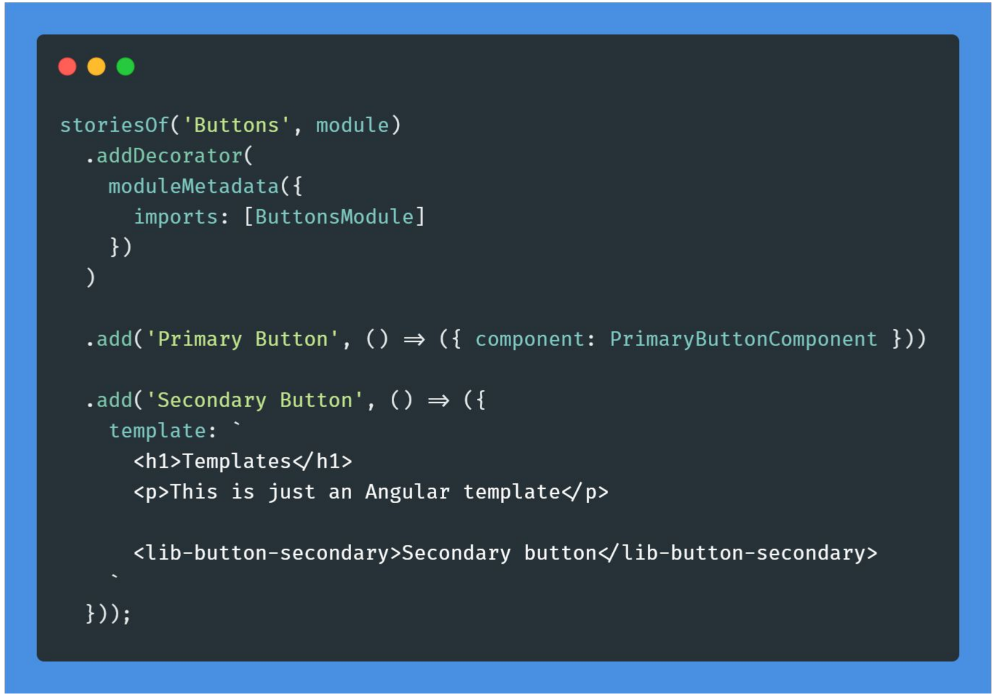
Writing stories

Writing stories



```
src
  assets
  lib
    components
      buttons
        button-primary
          stories
            button-primary.stories.ts
            readme.md
          button-primary.component.ts
          button-primary.module.ts
        button-secondary
```

Writing stories

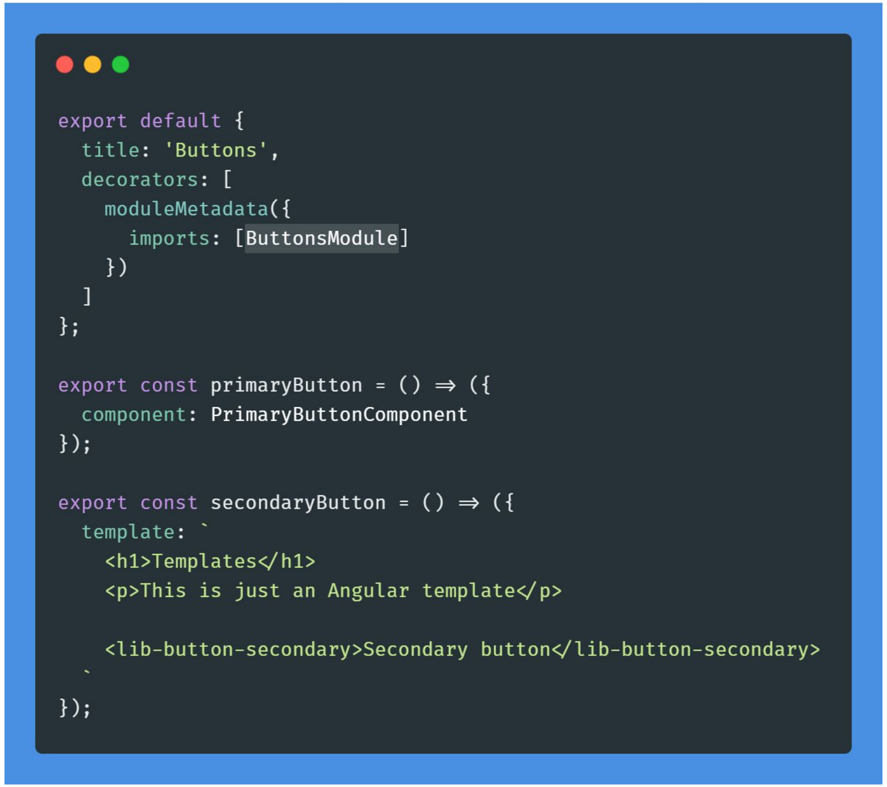


```
storiesOf('Buttons', module)
  .addDecorator(
    moduleMetadata({
      imports: [ButtonsModule]
    })
  )

  .add('Primary Button', () ⇒ ({ component: PrimaryButtonComponent }))

  .add('Secondary Button', () ⇒ ({
    template: `
      <h1>Templates</h1>
      <p>This is just an Angular template</p>

      <lib-button-secondary>Secondary button</lib-button-secondary>
    `
  }));
```

storiesOf API

- Standard story API until 2019
- Stories can reference **components** or ship their own **template**
- **storiesOf** is still supported but the recommended story format is the **“Component Story Format”** (CSF)

Writing stories



```
export default {
  title: 'Buttons',
  decorators: [
    moduleMetadata({
      imports: [ButtonsModule]
    })
  ]
};

export const primaryButton = () ⇒ ({
  component: PrimaryButtonComponent
});

export const secondaryButton = () ⇒ ({
  template: `
    <h1>Templates</h1>
    <p>This is just an Angular template</p>

    <lib-button-secondary>Secondary button</lib-button-secondary>
  `
});
```

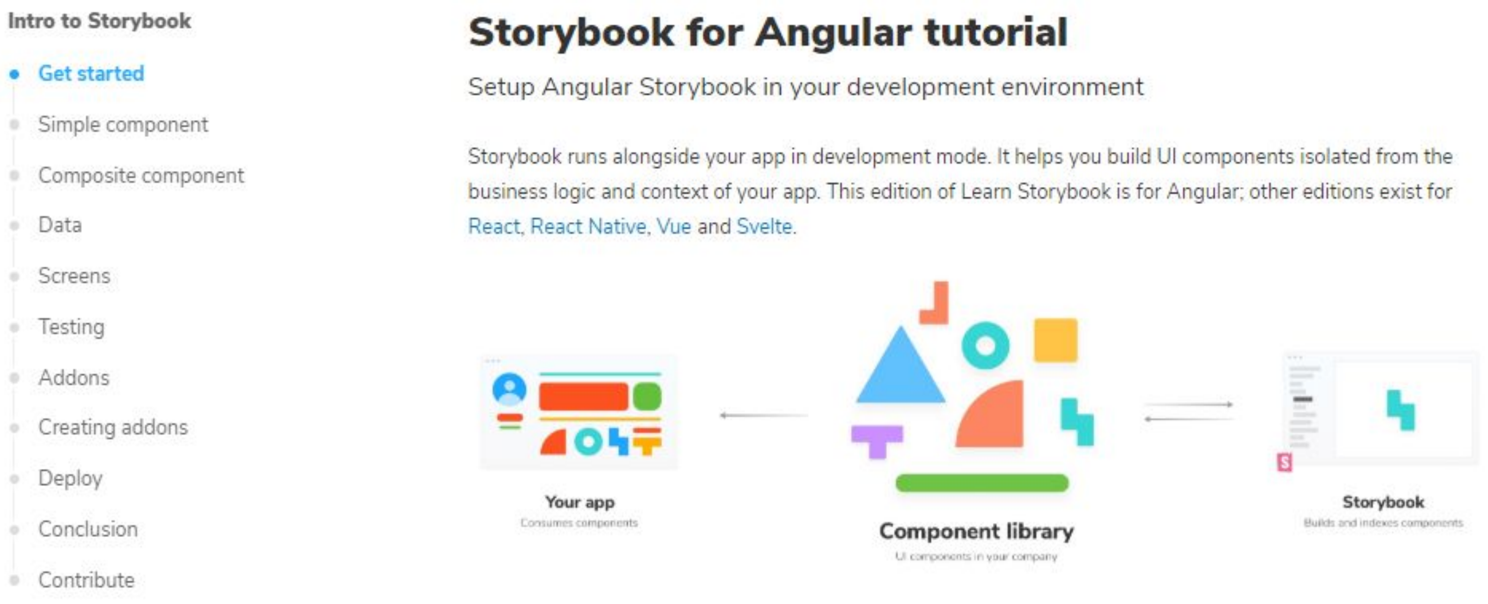
Component Story Format (CSF)

- **default export** describes your stories and dependencies
- Property names are the title of a story but can be overwritten
- **CSF** enables third-party tools to use your stories
- Re-use stories in test files

Summary

- **Component development is complicated**
  - Infrastructure, State, Business-logic and more are things that might hold you back from doing your task
- **Isolated Component Development**
  - It helps you to focus on one particular component and its inputs and outputs
  - It takes away the need of infrastructure and lets you develop components outside your real application
  - You still need an application that renders your isolated components
- **Storybook lives besides your application**
  - It helps you developing independently without the need of an actual application
  - API documentation or design system: It’s up to you how far you want to push your documentation

Storybook guides at learnstorybook.com



Intro to Storybook

- Get started
- Simple component
- Composite component
- Data
- Screens
- Testing
- Addons
- Creating addons
- Deploy
- Conclusion
- Contribute

Storybook for Angular tutorial

Setup Angular Storybook in your development environment

Storybook runs alongside your app in development mode. It helps you build UI components isolated from the business logic and context of your app. This edition of Learn Storybook is for Angular; other editions exist for React, React Native, Vue and Svelte.

Community

You can find all links and resources at storybook.js.org

github.com/storybookjs/storybook

Storybook discord

Get updates on medium.com/storybookjs

@storybookjs

Let’s have a drink!

@kairoeder kroeder