Tutors
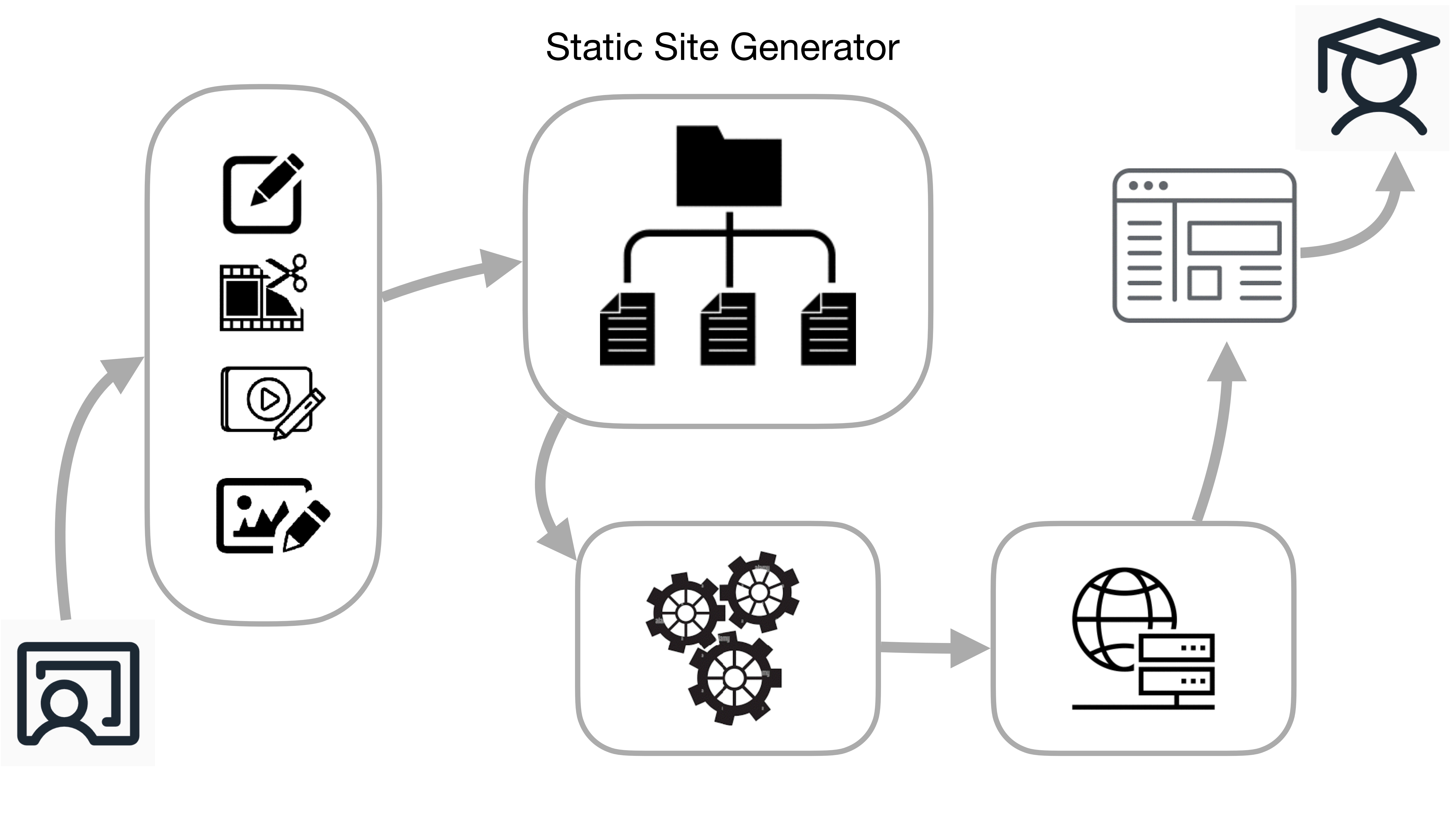

Open Web Learning Toolkit

The Educator Experience

The **Educator Experience** prioritises the creation of a **guided paths** through a curriculum via the creation of learning materials that are **autonomous**, **structurally aligned**, **composable**, **auditable**, **extensible**, **versioned** and **independent**.
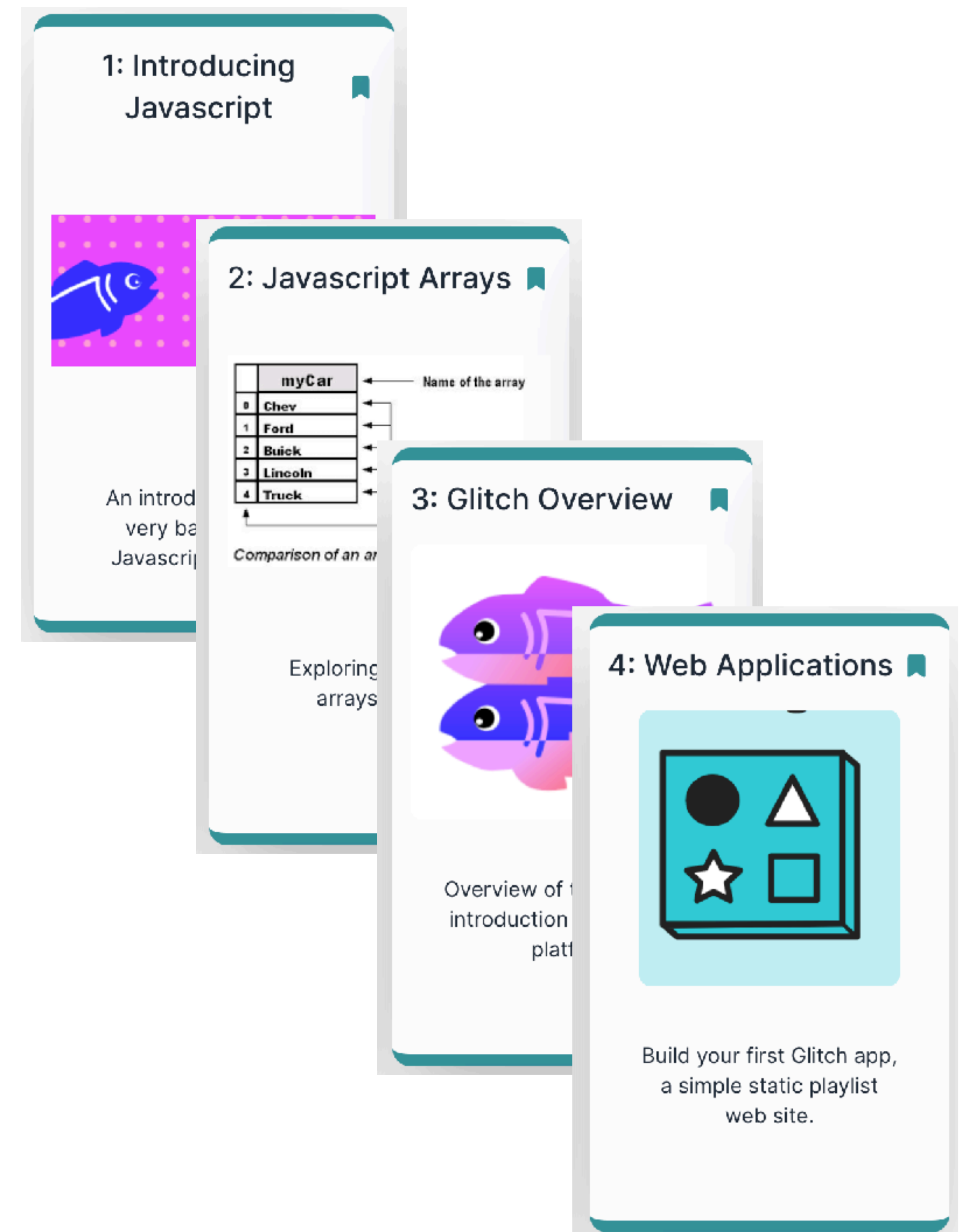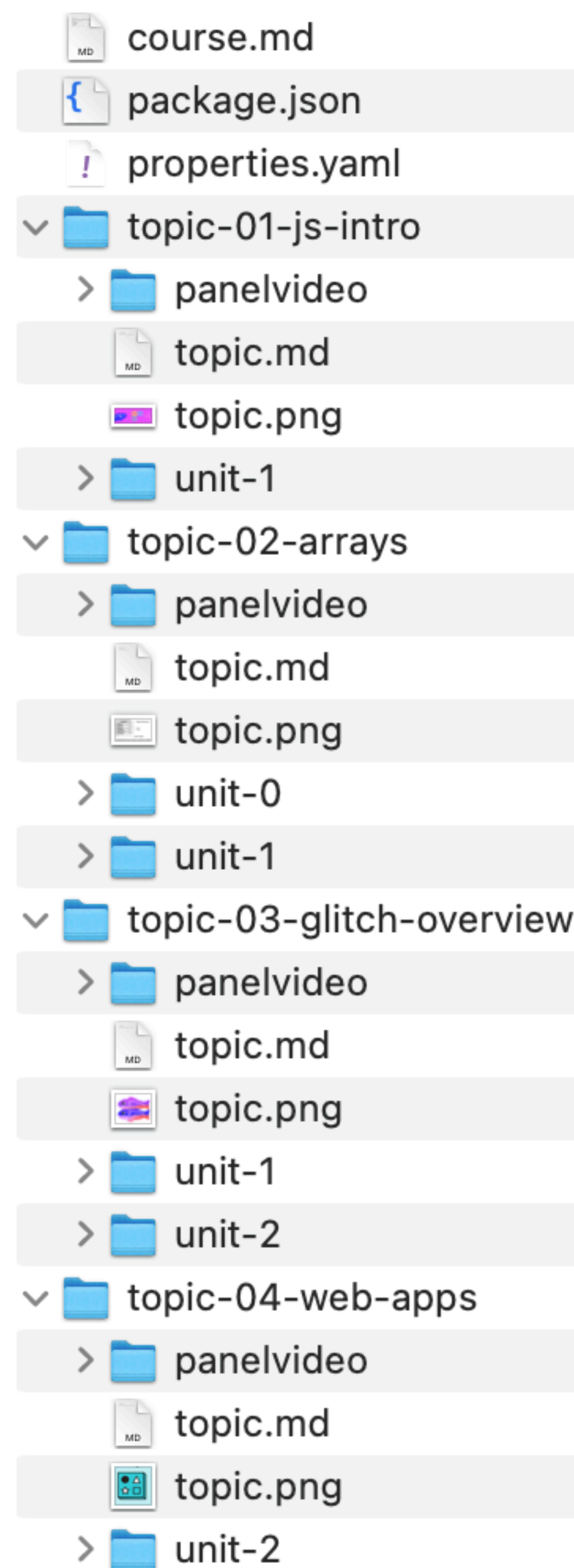
Static Site Generator

Semantic Naming

panelvideoXXX ->
Video

unit-XXX ->
related resources

topicXXX ->
topic

bookXXX ->
lab

talkXXX ->
pdf/slides

- course.md
- package.json
- properties.yaml
- ⌄ 📁 topic-01-js-intro
  - > 📁 panelvideo
  - topic.md
  - topic.png
  - > 📁 unit-1
- ⌄ 📁 topic-02-arrays
  - > 📁 panelvideo
  - topic.md
  - topic.png
  - > 📁 unit-0
  - > 📁 unit-1
- ⌄ 📁 topic-03-glitch-overview
  - > 📁 panelvideo
  - topic.md
  - topic.png
  - > 📁 unit-1
  - > 📁 unit-2
- ⌄ 📁 topic-04-web-apps
  - > 📁 panelvideo
  - topic.md
  - topic.png
  - > 📁 unit-2

1: Introducing Javascript

An introd...
very ba...
Javascrip...

2: Javascript Arrays

Comparison of an ar...

Exploring...
arrays...

3: Glitch Overview

Overview of t...
introduction...
platf...

4: Web Applications

Build your first Glitch app,
a simple static playlist
web site.

2: Javascript Arrays 🔖

Comparison of an array to a column of data

Exploring Javascript arrays in detail

topic-02-arrays
  panelvideo
    panelvideo.md
    videoid
  topic.md
  topic.png
  unit-0
    talk-e-variables-objects
    talk-f-methods-arrays
    topic.md
  unit-1
    book-a-js-arrays
    talk-a-arrays-basics
    talk-b-array-methods
    talk-c-array-iteration
    talk-d-arrays-of-objects
  topic.md

02: Javascript Arrays

Javascript Variables, Objects & Methods

Variables & Objects Review

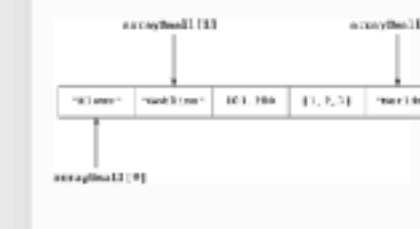A concise tour of the structure of variables & objects in Javascript

Methods & Arrays Review

A concise look at methods & Arrays

Javascript Arrays

Arrays: Basics
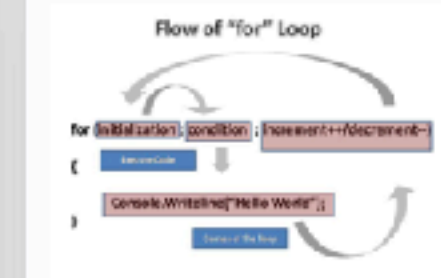
Creating, accessing, adding to and removing from arrays.

Array Methods

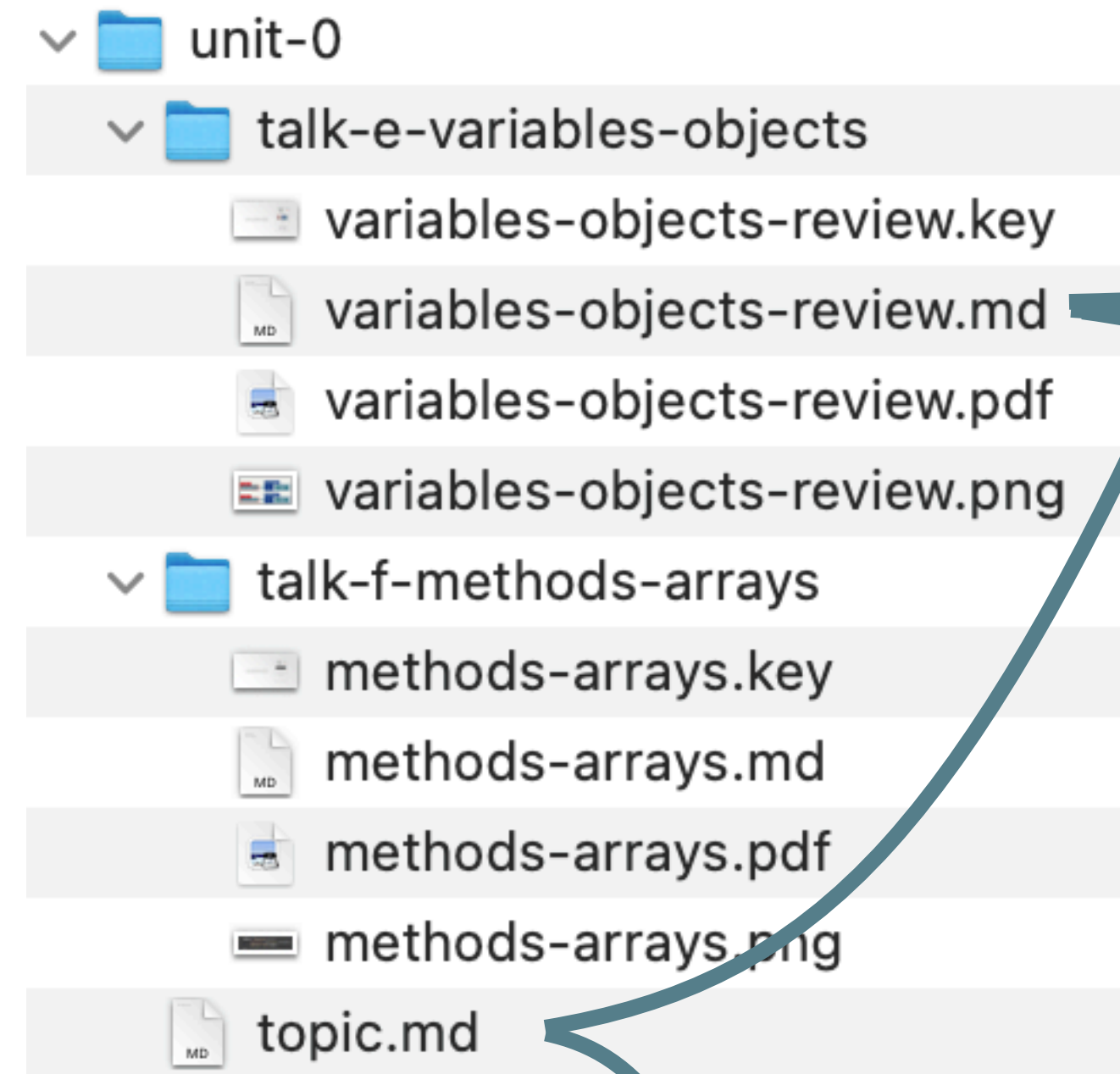Exploring length, slice, concat, join, indexOf, lastIndexOf

Array Iteration

Using for, while and do-while to iterate over an array

Arrays of Objects

Lab-02 JS Arrays

unit-0
- talk-e-variables-objects
  - variables-objects-review.key
  - variables-objects-review.md
  - variables-objects-review.pdf
  - variables-objects-review.png
- talk-f-methods-arrays
  - methods-arrays.key
  - methods-arrays.md
  - methods-arrays.pdf
  - methods-arrays.png
- topic.md

Javascript Variables, Objects & Methods

Variables & Objects Review

```
addr: 0x440892
x
ref:0x440892        a        100

addr: 0x3c980c
y
ref:0x3c980c        a        100
```

A concise tour of the structure of variables & objects in Javascript

Methods & Arrays Review

```
meow: function () {
    console.log(this.sound);
    return this.age;
},
```

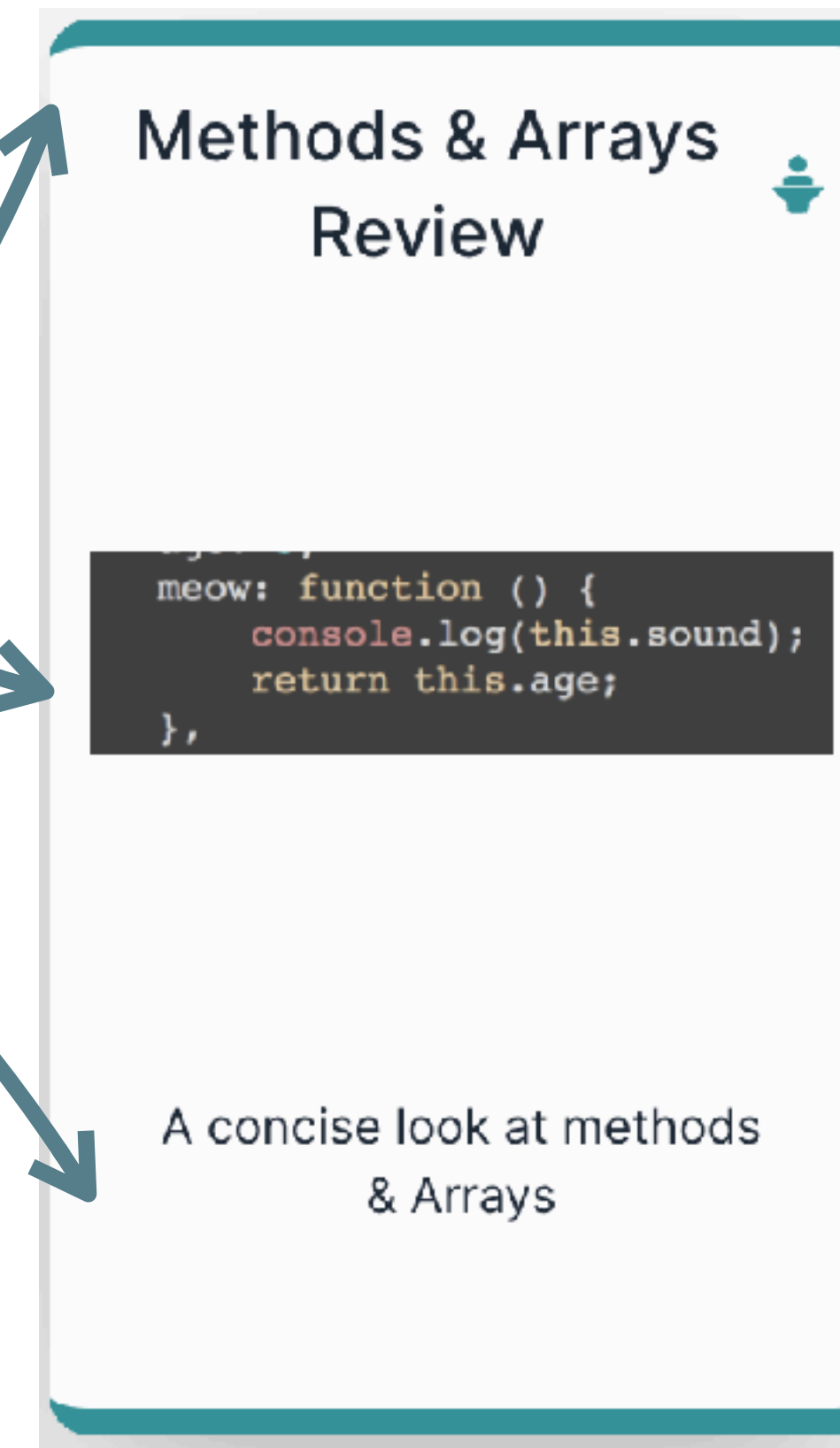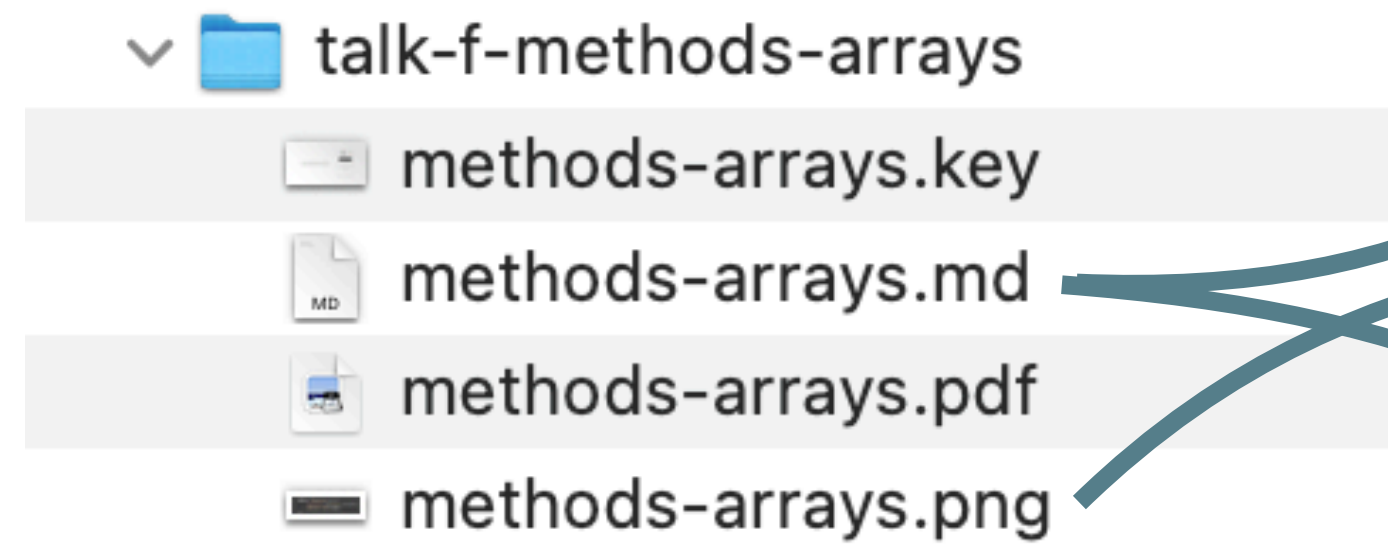A concise look at methods & Arrays

Variables & Objects Review

A concise tour of the structure of variables & objects in Javascript

variables-objects-review.md

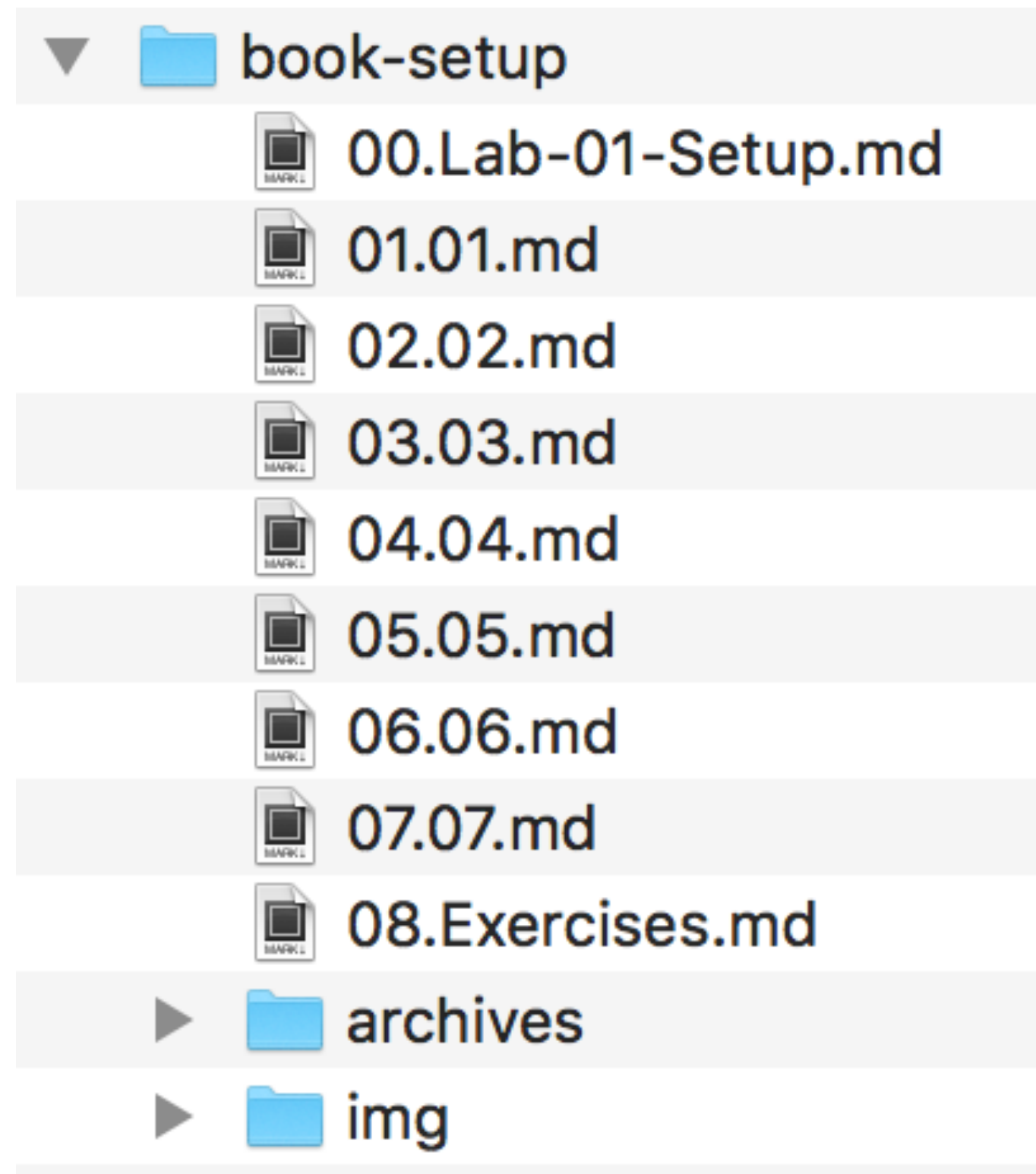Javascript Variables, Objects & Methods

topic.md

Talk structure


```
∨ 📁 talk-f-methods-arrays
    methods-arrays.key
    methods-arrays.md
    methods-arrays.pdf
    methods-arrays.png
```

Methods & Arrays
Review

```
meow: function () {
    console.log(this.sound);
    return this.age;
},
```

A concise look at methods
& Arrays

all files
same name

- image
- pdf
- markdown

Methods & Arrays Review

A concise look at methods &
Arrays

Lab structure

book-setup

- 00.Lab-01-Setup.md
- 01.01.md
- 02.02.md
- 03.03.md
- 04.04.md
- 05.05.md
- 06.06.md
- 07.07.md
- 08.Exercises.md
- ▶ archives
- ▶ img

img
- images used in lab

archives
- zipped archive linked to in labs

One *md* file per step

Objectives

Running Javascript

Variables

Primitive Data Types

Boolean Logic

Const, Let & Simple Objects

Objects with Functions

Chrome Debugger

Exercises

Solutions

Const, Let & Simple Objects

Const & Let

The variable statement declares a variable, optionally initializing it to a value.

var

```javascript
// String
var greeting = "hello";
// Number
var favoriteNum = 33;
```

const

The `const` statement is similar to the var statement, however, the value cannot be reassigned.

```javascript
// String
const greeting = 'hello';

// Number
const favoriteNum = 33;
```

So if we try this:

```javascript
favoriteNum = 23;
```

We will get an error. Try the above in the chrome console now and see what happens. You will see something like this:

```
Console
top                              Preserve log  Show all messages
> const favoriteNum = 33;
  favoriteNum = 23;
⊗ ► Uncaught SyntaxError: Identifier 'favoriteNum' has already been declared
        at <anonymous>:1:1
> |
```

Each Lab written in Markdown

Const, Let & Simple Objects

Const & Let

The variable statement declares a variable, optionally initializing it to a value.

var

~~~javascript
// String
var greeting = "hello";
// Number
var favoriteNum = 33;
~~~

const

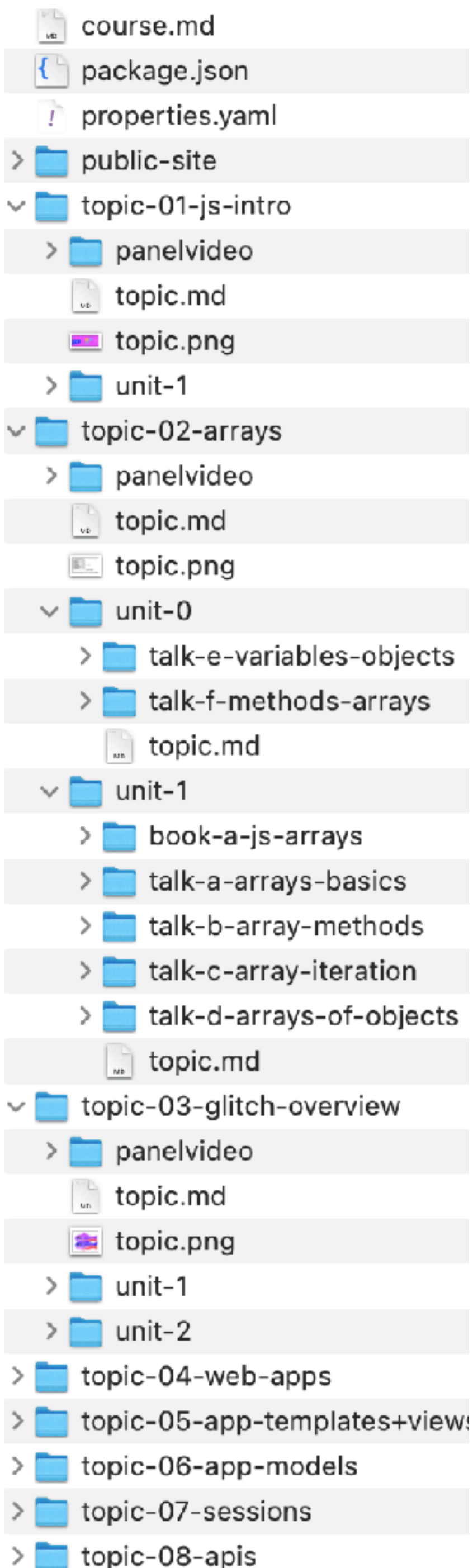The `const` statement is similar to the var statement, however, the value cannot be redeclared or reassigned.

~~~javascript
// String
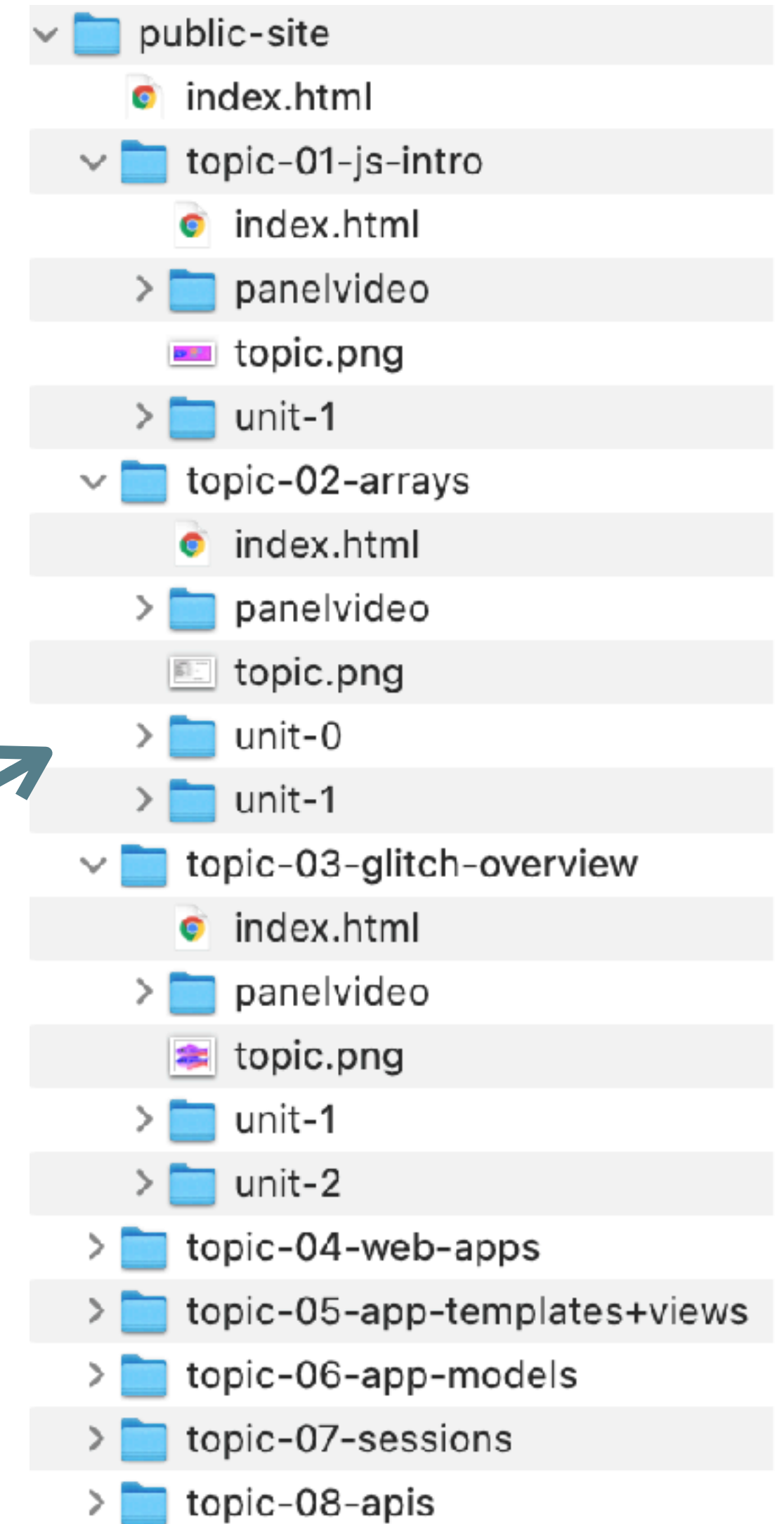const greeting = 'hello';

// Number
const favoriteNum = 33;
~~~

'npx tutors-html' command generates to 'public-site' folder

Left file tree:
- course.md
- package.json
- properties.yaml
- public-site
- topic-01-js-intro
  - panelvideo
  - topic.md
  - topic.png
  - unit-1
- topic-02-arrays
  - panelvideo
  - topic.md
  - topic.png
  - unit-0
    - talk-e-variables-objects
    - talk-f-methods-arrays
    - topic.md
  - unit-1
    - book-a-js-arrays
    - talk-a-arrays-basics
    - talk-b-array-methods
    - talk-c-array-iteration
    - talk-d-arrays-of-objects
    - topic.md
- topic-03-glitch-overview
  - panelvideo
  - topic.md
  - topic.png
  - unit-1
  - unit-2
- topic-04-web-apps
- topic-05-app-templates+views
- topic-06-app-models
- topic-07-sessions
- topic-08-apis

Terminal window (-bash):

```
Eamonns-Mac-mini:web-development-2-2022 edeleastar$ npx tutors-html
Static course generator tutors-json 0.6.7 (tutors-lib: ^0.6.5)
::    Web Development II
  --> 1: Introducing Javascript (topic)
:: 1: Introducing Javascript
  --> Introducing Javascript (unit)
:: Introducing Javascript
  --> Module Overview (talk)
  --> JS Introduction (talk)
  --> Variables & Logic (talk)
  --> Const, Let & Objects (talk)
  --> Lab-01 JS Intro(lab)
  --> 1: Introducing Javascript (panelvideo)
  --> 2: Javascript Arrays (topic)
:: 2: Javascript Arrays
  --> Javascript Variables, Objects & Methods (unit)
:: Javascript Variables, Objects & Methods
  --> Variables & Objects Review (talk)
  --> Methods & Arrays Review (talk)
  --> Javascript Arrays (unit)
:: Javascript Arrays
  --> Arrays: Basics (talk)
  --> Array Methods (talk)
  --> Array Iteration (talk)
  --> Arrays of Objects (talk)
  --> Lab-02 JS Arrays(lab)
  --> 02: Javascript Arrays (panelvideo)
  --> 3: Glitch Overview (topic)
:: 3: Glitch Overview
```

Right file tree:
- public-site
  - index.html
  - topic-01-js-intro
    - index.html
    - panelvideo
    - topic.png
    - unit-1
  - topic-02-arrays
    - index.html
    - panelvideo
    - topic.png
    - unit-0
    - unit-1
  - topic-03-glitch-overview
    - index.html
    - panelvideo
    - topic.png
    - unit-1
    - unit-2
  - topic-04-web-apps
  - topic-05-app-templates+views
  - topic-06-app-models
  - topic-07-sessions
  - topic-08-apis

https://tutors.dev/

Tutors: An Open Learning Web Toolkit

A collection of open source components & services supporting the creation of transformative learning experiences using open web standards.

EXPLORE

The project supports three complimentary perspectives

The Learner Experience

The *Learner Experience* prioritises web interactions that are **engaging**, **contextual**, **linkable**, **searchable**, **accessible** and **responsive**. In addition the experience should foster a sense of **community** and **connection** among fellow learners.

The Educator Experience

The *Educator Experience* prioritises the creation of a **guided paths** through a curriculum via the creation of learning materials that are **autonomous**, **structurally aligned**, **composable**, **auditable**, **extensible**, **versioned** and **independent**.

The Developer Experience

The *Developer Experience* prioritises the specification and implementation of **robust**, **well documented**, **loosely coupled components & services**, integrated into a **coherent toolkit** open to contributions from **diverse skill sets**.