# Front-end Development with aemfed

Objectives:

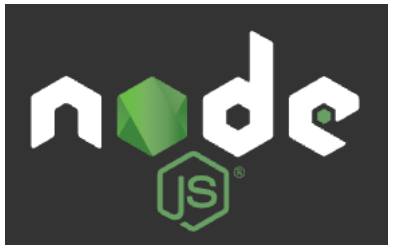
- Describe benefits of aemfed
- Install aemfed
- Develop using aemfed

ADOBE COPYRIGHT PROTECTED

# Node.js® and NPM

## Node.js

- Asynchronous event driven JavaScript runtime
- Built on Chrome V8 JavaScript engine




## Node Package Manager (NPM)

- CLI package manager
- Manages an online repository of open source node.js packages
- Manages dependencies for Node.js packages

ADOBE COPYRIGHT PROTECTED

© 2021 Adobe. All Rights Reserved

# aemfed

Open-source, command-line tool used to speed up front-end development

- URL to the open source project - https://aemfed.io/

It is powered by

- aemsync
- Browsersync
- Sling Log Tracer

ADOBE COPYRIGHT PROTECTED

© 2021 Adobe. All Rights Reserved

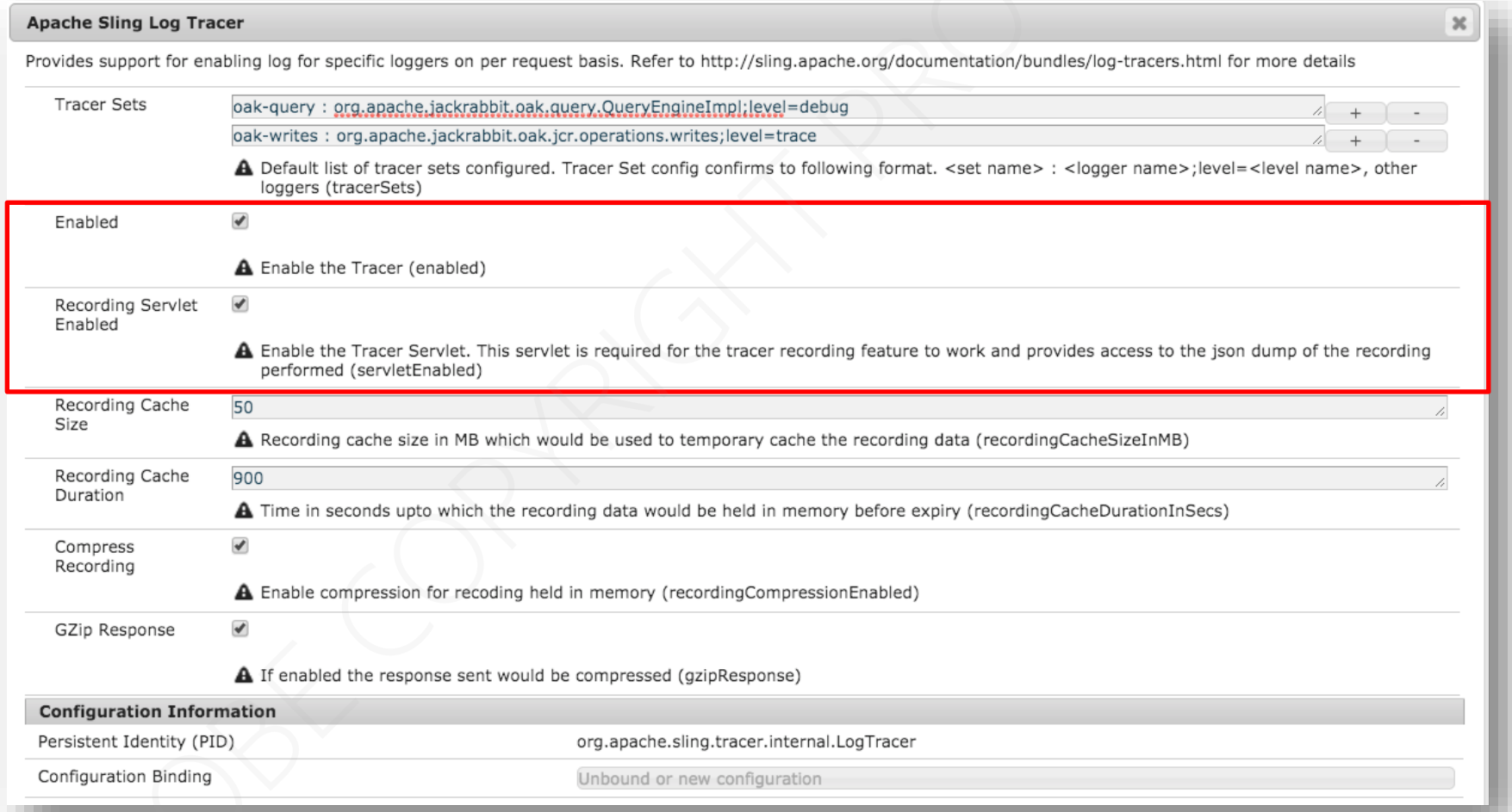
# Why aemfed?

- Listens for file changes and, using *aemsync*, automatically syncs to an AEM instance
- Determines which clientlibs are affect by the uploaded changes
  - *Browsersync* communicates with all tabs of your browser that are open to AEM pages, without any browser plugins
  - Styling changes are immediately injected without a reload, maintaining the state of the page
  - All other changes, result in a page reload/refresh
- Works with the *Sling Log tracer* to automatically display any front-end related, server-side errors directly in the terminal window

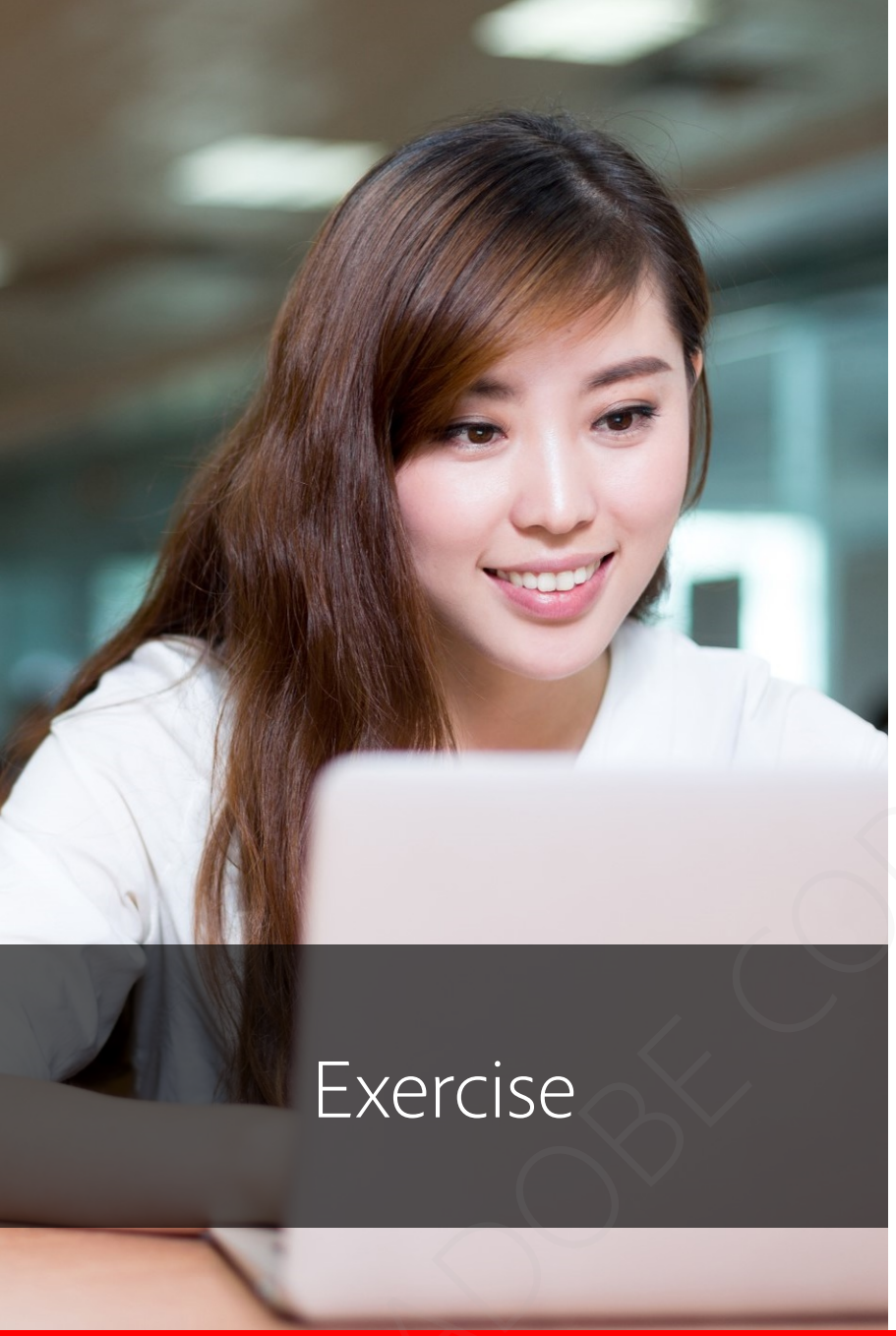
ADOBE COPYRIGHT PROTECTED

© 2021 Adobe. All Rights Reserved

# Enabling the Apache Sling Log Tracer




© 2021 Adobe. All Rights Reserved



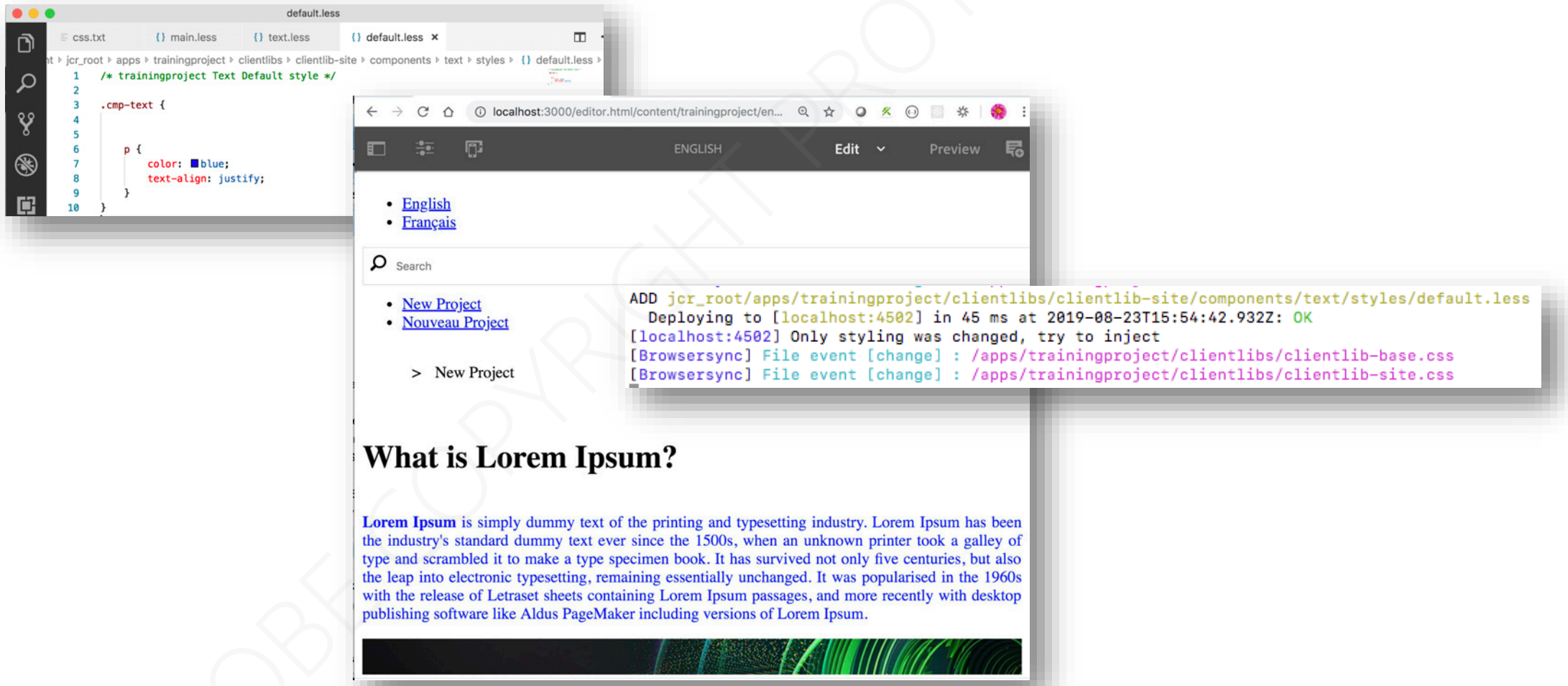
# Exercise

**Exercise 1 - Install Node.js and NPM**
**Exercise 2 – Configure and run aemfed**
**Exercise 3 – Install content locally (optional)**

ADOBE COPYRIGHT PROTECTED

© 2021 Adobe. All Rights Reserved. Adobe Confidential.

# aemfed Watches File Changes




ADOBE COPYRIGHT PROTECTED

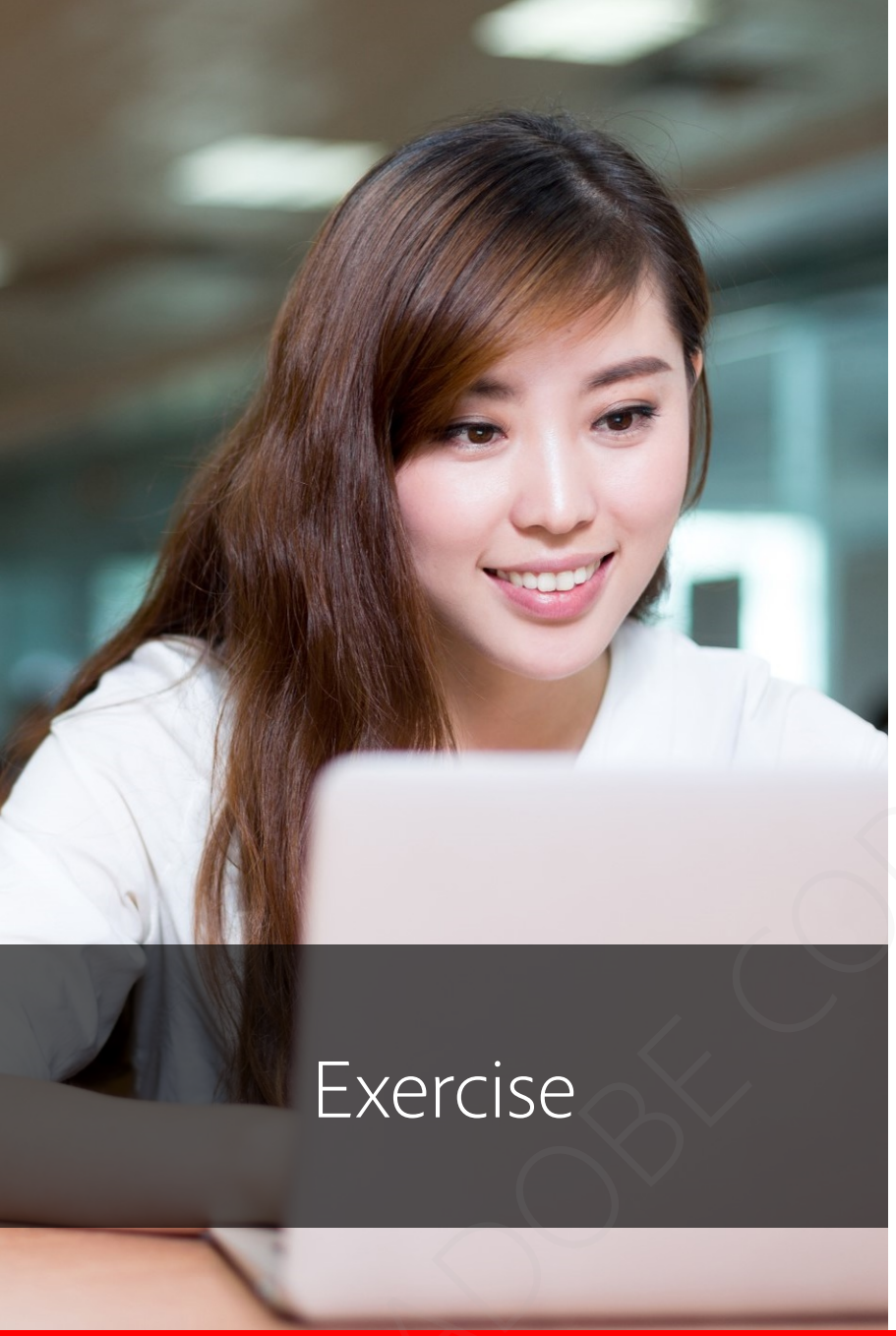
© 2021 Adobe. All Rights Reserved

# Apache Sling Log Tracer Displays Errors

```
[Browsersync] File event [change] : /apps/trainingproject/clientlibs/clientlib-base.css
[Browsersync] File event [change] : /apps/trainingproject/clientlibs/clientlib-site.css
[localhost:4502] Tracer output for [/etc.clientlibs/trainingproject/clientlibs/clientlib-base.css?browsersync=1566576729776] (daf7a936-ceca-43ac-ace6
-8a7b02436c93)
[ERROR] LessCompilerImpl: failed to compile less /apps/trainingproject/clientlibs/clientlib-site/main.less: ParseError: Unrecognised input in /apps/t
rainingproject/clientlibs/clientlib-site/components/text/styles/default.less on line 6, column 5:
5
6     p {
7         color: green

Local source: /Adobe/Archetype19/training/ui.apps/src/main/content/jcr_root/apps/trainingproject/clientlibs/clientlib-site/components/text/styles/def
ault.less:6:5
```

© 2021 Adobe. All Rights Reserved




# Exercise 4: Add a component style using aemfed

ADOBE COPYRIGHT PROTECTED

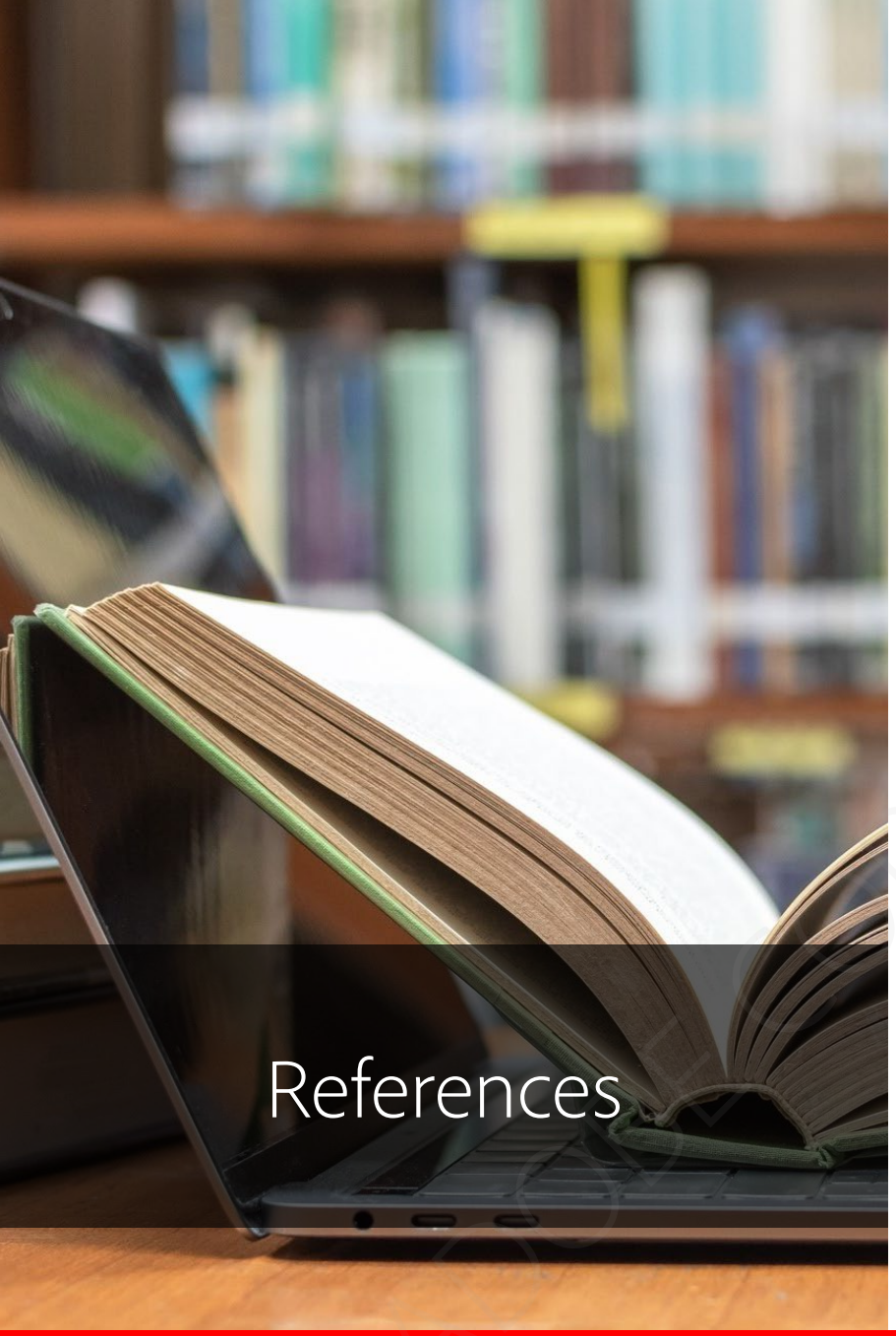
© 2021 Adobe. All Rights Reserved. Adobe Confidential.

## Key Takeaways:

- Node.js
  - Enables developers to use JavaScript to write command line (CLI) tools and for server-side scripting
- aemfed
  - Open-source, command-line tool used to speed up front-end development
  - Listens for file changes and, using *aemsync*, automatically syncs to an AEM instance
  - Works with the *Sling Log tracer* to automatically display any front-end related, server-side errors directly in the terminal window

ADOBE COPYRIGHT PROTECTED

© 2021 Adobe. All Rights Reserved. Adobe Confidential.



# References

## Resources:

- aemfed: https://aemfed.io/
- https://github.com/abmaonline/aemfed

ADOBE COPYRIGHT PROTECTED

© 2021 Adobe. All Rights Reserved. Adobe Confidential.