



FRONT-END DEVELOPMENT TOOLS

Copyright ©2017 Liferay, Inc.

All Rights Reserved.

No material may be reproduced electronically or in print,
distributed, copied, sold, resold, or otherwise exploited
for any commercial purpose without express written
consent of Liferay, Inc.

DEVELOPING IN LIFERAY

- ❖ Liferay is tool-agnostic.
- ❖ Anything from a command prompt or text editor to a full-blown IDE can be used to develop on Liferay.
- ❖ Liferay DXP introduces a number of new tools we can take advantage of to simplify development.
- ❖ Customizations and new features in Liferay are deployed in *modules*.
- ❖ Themes and Layout Templates are installed as modules.
- ❖ Previously, front-end developers needed to use the Plugins SDK.
- ❖ If you'd like to know how to set up Liferay with another development environment, our online Developer Guide covers this process:
https://dev.liferay.com/develop/reference/-/knowledge_base/7-0/development-reference

RIGHT TOOL FOR THE JOB

- ❖ Some tools will simplify creating, building, and testing *modules* in Liferay.
- ❖ There are tons of helpful tools for front-end development — you might have a few favorites already.
- ❖ We'll use a few standard tools for creating our projects:
 - ❖ *Node.js*
 - ❖ *gulp*
 - ❖ *Yeoman*
- ❖ You may already be familiar with these or similar tools.
- ❖ If not, don't worry: they're easy to learn and use.

TOOLS FOR CREATING

- ❖ We need a base *platform* for building and testing themes.
 - ❖ **Node.js** provides the platform.
- ❖ We want to easily *create* new projects.
 - ❖ **Yeoman** gives us that power.
- ❖ We want to *build* and *deploy* our projects to test them.
 - ❖ **gulp** makes those tasks a breeze.

NODE.JS AND NPM

- ❖ **Node.js** is a JavaScript runtime environment and will provide the platform we need to create and test our themes.
- ❖ With *Node.js*, we have access to a vast library of JavaScript packages.
- ❖ These libraries are accessed via *NPM*, the *Node.js* package manager.
- ❖ *NPM* helps users install and manage the packages and dependencies needed for the projects they are working on.
- ❖ This automation lends itself to a quicker, more efficient workflow.
- ❖ With *Node.js* and *NPM*, we'll have access to the tools needed to get our projects started.
 - ❖ If you'd like a refresher on *Node.js*, or want more details, check out <http://nodejs.org>.

CREATING WITH YEOMAN

- ❖ **Yeoman** is a powerful tool that runs on *Node.js*.
- ❖ Liferay modules are complex to create from scratch.
- ❖ We don't need to know the technical details; we just want to start building.
- ❖ *Yeoman* lets us create from templates using *generators*.
- ❖ A *generator* is a recipe for creating a new project from scratch — just add water!
- ❖ We'll look at installing a custom Liferay generator in Yeoman later.
- ❖ *Yeoman* makes sure projects are created the right way every time.

BUILDING WITH GULP

- ❖ Once we've created a project, we need to know how to:
 - ❖ Build
 - ❖ Package
 - ❖ Deploy
- ❖ **gulp** runs tasks with a simple command like:
`gulp build`
- ❖ All the details are handled behind the scenes.
 - ❖ You may already be familiar with other *Node.js*-based build tools like *Grunt*.
- ❖ *gulp* does all the hard work so we can *build* and *deploy* themes easily.

DEVELOPMENT TOOLS FOR MODULES

- ❖ So here's everything we need to style Liferay:
 1. **Java Development Kit (JDK) 8:** used to run Liferay and our installer. We'll also set the `JAVA_HOME` environment variable.
 2. **Node.js (packaged with npm):** used to install our Liferay Theme Generator and dependencies
 3. **Yeoman and gulp:** global dependencies, used to run the Liferay Theme Generator and `gulp liferay-theme` tasks
 4. **Liferay Theme Generator:** used to generate our themes and themelets
 5. **Brackets:** Our text editor of choice used to edit our theme files
 6. **Liferay Tomcat Bundle:** used to view development changes on our local machines

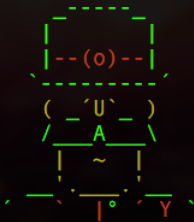
THEME BUILDER GRADLE PLUGIN

- ❖ We'll be using *Liferay Theme Generator* and the Node.js platform in exercises throughout the course.
- ❖ As an alternative Liferay also offers the Theme Builder Gradle plugin for theme development.
- ❖ For more information on using the Theme Builder Gradle plugin you can reference documentation at https://dev.liferay.com/de/develop/reference/-/knowledge_base/7-0/theme-builder-gradle-plugin

USING THE TOOLS WE'VE BEEN GIVEN

- ❖ Templates can be developed on the platform, as we'll explore later.
- ❖ Next, we'll set up our tools for developing Themes and Layouts.

```
MacBook-Pro-2:Desktop liferay$ yo liferay-theme
```



```
Welcome to the splendid  
Liferay Theme generator!
```

```
? What would you like to call your theme? (My Liferay Theme)
```

Notes: