

0x00. Webpack

Front-end My JavaScript ES6



Weight: 1

Projects(/projects/current)



Project over - took place from Jul 22, 2024 6:00 AM to Jul 23, 2024 6:00 AM



An auto-generated review can make corrections(to_review)



Evaluation quizzes(/dashboards/my_current_evaluation_quizzes)

BRACE YOURSELVES

Curriculums(/dashboards/my_curriculums)



Concepts(/concepts)



Conference rooms(/dashboards/video_rooms)



Servers(/servers)



Sandboxes(/user_containers/current)



Tools(/dashboards/my_tools)



Video on demand(/dashboards/videos)

WEBPACK IS COMING

Resources

Peer / Self Peers)

Read or watch:

Discord(<https://discord.com/app>)

- Webpack documentation (/rltoken/XEFTUAcZ_9sKurp1Bui7ug)
- Webpack beginner guide (/rltoken/6ngQzrV7xeKJjcRwdmrYAQ)
- npm-package.json (/rltoken/P00rJM5qCeaf33hsPuhgog)



My Profile(/users/my_profile)

Learning Objectives

At the end of this project, you are expected to be able to explain to anyone (/token/ME_HaE8ycyyO7RaAoOyclw), **without the help of Google**.



- How to setup Webpack for a basic project
- Entry points, output, and loaders
- How to add plugins



- How to splitting (bundling) the chunks
- How to setup a dev server



Projects(/projects/current)

Requirements



- All of your code will be executed on Ubuntu 18.04 LTS using Node 12.x.x
- Allowed editors: vi, vim, emacs, Visual Studio Code
- All of your files should end with a new line



- A README.md file at the root of the folder of the project is mandatory



Tasks

Curriculums(/dashboards/my_curriculums)



1. Basic setup

mandatory



Create and run Webpack using a basic installation.

Conference rooms(/dashboards/video_rooms)

- Create a folder named task_0.
- Install webpack and webpack-cli as developer dependencies within the folder using npm.
- Install jQuery as a regular dependency using npm.
- Make sure that webpack and webpack-cli are listed under the devDependencies key along with jQuery, being listed under the dependencies key within the package.json
- Create a src directory with a index.js in it.
- The file should import jQuery and add three different paragraphs to the page body: (refer to this link (/token/POV-DBrnVx50XBkYKp_69Q) under the Babel section for importing jQuery)



Tools(/dashboards/my_tools)



Holberton Dashboard

Video on demand(/dashboards/videos)

Dashboard data for the students

Copyright - Holberton School



Peers(/users/peers)

- Create a dist/index.html. Import your main.js in the body.
- You must use jQuery to add the elements to the body of the page.
- When running webpack, your javascript and html files should be generated in a dist folder.
- You should not use a custom webpack config file.
- Opening your main file should not generate any error in the console.
- Your html code should only import one Javascript script (the one generated by webpack).
- Do NOT push your dist/main.js if you have one.

My Profile(/users/my_profile)



Repo:



GitHub repository: alx-react

Directory: 0x00-webpack



File: task_0/package.json, task_0/src/index.js, task_0/dist/index.html
Home(/)



Check submission

My Planning(/planning/me)

1. Learning how to use Webpack with a config file

mandatory



Projects(/projects/current)

Installing packages



- Create a folder named task_1 and go into it, and create a package.json using `npm init -y`
- Install webpack (dev dependency), jQuery (dependency), and Lodash (dependency) within the folder using `npm`.



- Modify your package.json to add a build script that runs webpack to create a production build. This lets you execute `npm run build` on the command line.

Using jQuery



- Create a js directory with a javascript file named dashboard_main.js in it. The file should import jquery and add the below elements in the following order:



- Paragraph element: Holberton Dashboard
- Paragraph element: Dashboard data for the students
- Button element with the text Click here to get started



Add a paragraph element as <video id='count'></p>

- Add another paragraph: Copyright - Holberton School



Event spammers
Servers(servers)

- Write a function called `updateCounter()` that tracks the number of times the button element has been clicked.



- Each time it's called, update the count by 1 and set the content of the paragraph element with `id='count'` to `{count}` clicks on the button



- Bind the debounce function in Lodash to the click event on the button you created previously.

Requirements:



- When running Webpack, your javascript should be generated in a `public` folder. You should also place your `index.html` in this folder.

- The javascript file generated by Webpack should be named `bundle.js`

- Set Webpack config mode to production



- Opening your html file should not generate any error in the console
- Your html code should only import one Javascript script (the one generated by webpack)



Resources:

Discord(<https://discord.com/app>)

- Debounce documentation on Lodash ([/rltoken/fuAiFYn08OQLb7SM8cYdYA](https://lodash.com/docs/4.17.15#debounce))



Repo:

- GitHub repository: alx-react
My Profile(users/my_profile)
- Directory: 0x00-webpack

- File: task_1/js/dashboard_main.js, task_1/package.json, task_1/webpack.config.js, task_1/public/index.html (/)

Check submission

Home(/)

2. Adding CSS & Images

mandatory



My Planning(/planning/me)

Create a specific configuration for Webpack.



- Using the folder named task_2 . Projects(/projects/current)

- Reuse the code from task_1 .

- Modify the webpack config to support adding CSS to the bundle.



- Modify the webpack config to support adding images to the CSS.

- Create a folder named css

- In a file named main.css , change the position of the counter text to be displayed on the right of the button and in bold.



- Add an element at the top of document with the id #logo which has a width of 200px and height of 200px.



- Set the background of the element with the image in task_2/assets/holberton-logo.jpg (you can download this image from <https://s3.amazonaws.com/holbertonschool-webstack/581/holberton-logo.jpg>).



- Set the width and height of the logo background image to 200px by 200px.

- Make sure to configure Webpack to optimize images.

- When running Webpack, your javascript and html files should be generated in a public folder.



- The javascript file generated by Webpack should be named bundle.js .

- Set Webpack config mode to production.

- Opening your main file should not generate any error in the console.



- Your html code should only import one Javascript script (the one generated by webpack).

- When running Webpack, you should not see the warning WARNING in asset size limit: The following asset(s) exceed the recommended size limit



Sandboxes(/user_containers/current)

Repo:



- Tools(/dashboards/my_tools)

- GitHub repository: alx-react

- Directory: 0x00-Webpack



- Files: task_2/package.json, task_2/main.css, task_2/webpack.config.js, task_2/js/dashboard_main.js, task_2/public/index.html

Check submission

Peers(/users/peers)



Discord(<https://discord.com/app>)

3. Dev servers, modules, and tree shaking

mandatory

Using the folder named task_3 . **Set up a development server:**

- Reuse the code from my_profile (/users/my_profile)

- Modify the Webpack config to setup a development server running on the port 8564.

- Modify the Webpack config to set its mode to development .
- Add a script in the package.json to start the server and open the browser with npm run start- (/) dev .

Divide the code into modules:



- Home (/)
- Divide the main bundle into three modules.



- Import jQuery, and add the logo and the H1 title to the header.js file (with content Holberton My Planning (planning/me) Dashboard). Add a console.log printing Init header .
- Add the needed style to the header.css file.



- Projects (/projects/current)
- body should contain a body.css and body.js files.
- Import jQuery, Lodash and add the body code (button, counter) in the body.js file.



- Add the needed style to the body.css file.
- footer should contain a footer.css and footer.js files.
- Import jQuery, and append a paragraph the copyright text Copyright - Holberton School .



- Add the needed style to the footer.css file (for my_courses/evaluation_quizzes)
- Modify the Webpack configuration to support three different entry points (header , body , footer). Each entry point should generate a filename with the following format name_of_the_file.bundle.js .



- Do not push a task_3 to your repository
- Add a plugin to Webpack to automatically create an index.html HTML file



Improve development speed

- Modify the webpack config to support inline source mapping.
- Check that the console.log in the header.js now takes you to your javascript file instead of the main bundle.
- Add a plugin to Webpack to clean your build folder on each build.



Improve bundles size:

- You can see that the current modules generated with npm run build are quite large. They all contain jQuery and lodash. Modify the Webpack configuration to split the modules in chunks.

Requirements:



- Tools (/dashboards/my_tools)
- When running the dev server, your code should work on http://localhost:8564/ .
- When running Webpack, your javascript and html files should be generated in a public folder.



- Opening your main file should not generate any error in the console.

Repo:



- GitHub repository: alx-react
- Peers (/users/peers)
- Directory: 0x00-Webpack



- File: task_3/modules/body/body.css, task_3/modules/body/body.js, task_3/modules/footer/footer.css, task_3/modules/footer/footer.js, task_3/modules/header/header.css, task_3/modules/header/header.js, task_3/package.json, task_3/webpack.config.js



Check submission

My Profile (/users/my_profile)



(/)



Home(/)

Copyright © 2024 ALX, All rights reserved.



My Planning(/planning/me)



Projects(/projects/current)



QA Reviews I can make(/corrections/to_review)



Evaluation quizzes(/dashboards/my_current_evaluation_quizzes)



Curriculums(/dashboards/my_curriculums)



Concepts(/concepts)



Conference rooms(/dashboards/video_rooms)



Servers(/servers)



Sandboxes(/user_containers/current)



Tools(/dashboards/my_tools)



Video on demand(/dashboards/videos)



Peers(/users/peers)



Discord(<https://discord.com/app>)



My Profile(/users/my_profile)