



## Curso de Webpack

### ¡No te rindas!

Necesitas una **calificación mínima de 9.0** para aprobar.

Vuelve a intentarlo en 05 horas, 46 minutos, 40 segundos

# 7.24

Calificación

# 21 / 29

Aciertos

#### 1. ¿Qué es Webpack?

Es un set de herramientas para JavaScript que nos permite crear aplicaciones c [REPASAR CLASE](#)

#### 2. ¿Cuáles loaders usamos para trabajar en Webpack con CSS?

css-loader y style-loader



#### 3. ¿Debemos usar expresiones regulares para definir la extensión de los archivos que afectarán nuestros loaders?

Verdadero



4. ¿Cuál plugin nos permite minificar nuestros estilos CSS?

CssMinimizerPlugin

REPASAR CLASE

5. ¿Cuál es la función del source-map?

Crear un mapa para encontrar las piezas de nuestro código por separado y hacer más fácilmente debugging.



6. ¿Para qué sirven los Alias en Webpack?

Para identificar más fácilmente el "path" de los archivos con los que trabajamos en el proyecto.



7. ¿Cuál es la función de las variables de entorno?

Crear un espacio seguro para la configuración de nuestro proyecto que no queremos exponer en nuestro código.



8. ¿Cuáles loaders usamos para trabajar en Webpack con fuentes?

css-loader y font-loader

REPASAR CLASE

9. ¿Cuál es el comando de webpack-cli para ejecutar nuestro proyecto en modo de desarrollo?

webpack --mode development



10. ¿Cuál es el comando de webpack-cli para ejecutar nuestro proyecto en modo de producción?

webpack --production

REPASAR CLASE

11. ¿Cuál es la expresión regular para que Webpack transpile nuestros archivos .css y .styl con su loader correspondiente?

/\.css|.styl\$/



12. ¿Cuál es la propiedad del archivo de configuración de Webpack que guarda las opciones de optimización?

```
module.exports = { [...] optimization: { minimize: true, minimizer: [ new  
CssMinimizerPlugin(), new TerserPlugin() ] } }
```



13. ¿Cuál plugin nos ayuda a mover archivos a la carpeta de distribution?

TransferWebpack

REPASAR CLASE

14. Después de hacer un import de las imágenes, ¿cómo las llamamos?

Como variables.



15. ¿En dónde establecemos la configuración, loaders y plugins que usamos en Webpack?

webpack.config.js



16. ¿Cómo conseguimos que nuestro código de JavaScript sea compatible con todos los navegadores?

Integrando Babel a nuestro proyecto y babel-loader a la configuración de Webpack para archivos .js.



17. ¿Para qué sirven los loaders y plugins en Webpack?

Agregar funcionalidades o configuraciones para trabajar con nuestros recursos.



18. ¿Qué propiedad agregamos al archivo de configuración para indicarle a Webpack que trabajaremos en modo desarrollo?

mode: "dev"

**REPASAR CLASE**

19. ¿A qué nos ayuda clean-webpack-plugin?

A limpiar nuestra carpeta del build cada vez que volvemos a transpilar el proyecto.



20. ¿Con cuál comando creamos nuestro package.json?

npm init -y



21. ¿Qué dependencia adicional agregamos a nuestro proyecto para entender la sintaxis de React y JSX?

@babel/preset-react



22. ¿Con qué comando puedes instalar Webpack Dev Server?

`npm install webpack --save-dev`

REPASAR CLASE

23. ¿En cuál archivo habilitamos el watch en nuestro proyecto?

`webpack.config.js`



24. ¿Con qué comando puedes instalar Webpack en tu proyecto como dependencia de desarrollo?

`npm install webpack -D`



25. ¿Con qué repositorios te puedes conectar a Netlify?

Todas las respuestas son correctas



26. Netlify es la única plataforma en la que se puede desplegar un proyecto hecho con React.js. Esta afirmación es:

Falsa



27. La propiedad `template` en nuestro plugin `HTMLWebpackPlugin` se refiere al archivo FINAL que se ejecuta como resultado de las configuraciones que le aplicamos al HTML de nuestro proyecto:

Verdadero

REPASAR CLASE

28. ¿Cuál es el script adecuado que nos permitirá habilitar nuestro proyecto en un entorno de desarrollo local y no de producción?

"start" : "webpack serve"



---

29. Son los loaders necesarios para ejecutar nuestro proyecto de webpack con SASS

style-loader, css-loader, sass-loader



---

**REGRESAR**