

آموزش Webpack قسمت ۶: Debug کردن فایل های JavaScript (نسخه PDF)

در ادامه مباحث مربوط به آموزش Webpack با نحوه Debug کردن فایل های JavaScript آشنا میشیم. همونطور که قبلاً دیدیم، زمان build کردن پروژه ها توسط Webpack، یکسری کد به صورت خودکار توسط Webpack به کدهای نوشته شده ما اضافه میشه. وقتی که عملیات build در حالت production انجام میشه، فایل js به صورت bundle شده و minify شده تبدیل میشه، یعنی اگر ما ۱۰ فایل مختلف داشته باشیم تمامی این فایل ها به یک فایل با نام مثلاً application.js تبدیل میشه و این باعث میشه که عملیات خطایابی و Debug کردن کدها کار سختی بشه یا عملاً غیر ممکن بشه.

برای رفع این مشکل از قابلیتی به نام Source Map استفاده می کنیم. کاری که Source Map انجام میده کدهای Bundle شده رو به فایل سورس اصلی Map میکنه و همیشه متوجه شد که یک خطا در کدوم فایل افتاده. خود Webpack ابزاری برای اینکار در اختیار ما قرار میده. اگر به این لینک مراجعه کنید جدولی رو مشاهده می کنید که مربوط به devtool در webpack هست، بخشی از این جدول قابلیت های مربوط به Source Map رو شامل میشه که راهکار های مختلف برای ایجاد Source Map رو توضیح میده. شما می تونید هر کدوم از این قابلیت ها که مناسب شماست رو انتخاب کنید یا حتی می تونید از Plugin هایی که برای اینکار ایجاد شدن استفاده کنید، اما در این مطلب ما با یکی از این ابزارها که نامش cheap-module-eval-source-map هست آشنا میشیم.

اگر توضیحات مربوطه رو مطالعه کنید، گفته شده که این ابزار تنها شماره خط و نام فایلی که خطا در اون افتاده رو برای ما نمایش میده که در حال حاضر برای ما کافی هست.

برای فعال سازی این قابلیت، در فایل webpack.config.js بخشی به نام devtool رو به صورت زیر اضافه کنید:

```
module.exports = {
  watch: true,
  mode: "development",
  devtool: "cheap-module-eval-source-map",
  entry: "./src/index.js",
```

حالا باید ببینیم که قابلیت اضافه شده چه کاری برای ما انجام میده. فایل greeting.js رو باز میکنیم و دستور console.log زیر رو بهش اضافه میکنیم:

```
function sayHello(){
  let website = 'TOSINSO';
  alert(`Hello Webpack! This is from ${website}. I'm watching you!`);
  console.log("This message added for debugging!");
}

export { sayHello };
```

تغییرات فایل رو ذخیره می کنیم و دستور npm run build رو اجرا می کنیم. بعد از عملیات build فایل index.html رو داخل مرورگر باز میکنیم، کلید F12 رو برای نمایش Developer Tools میزنیم و وارد تب Console میشیم، پیام ما نمایش داده شده و جلوی پیام نمایش داده شده اطلاعات مربوط به فایل و خطی که دستور console.log داخلش نوشته شده نمایش داده میشه:

اگر روی شماره خط و نام فایل کلیک کنیم، کد داخل فایل greeting.js برای ما نمایش داده میشه:

اگر از قابلیت Source Map استفاده نمی‌کردیم، اطلاعات نمایش داده شده به فایل application.js اشاره میکرد که عملاً کمکی به ما برای پیدا کردن بخش مورد نظر نمیکرد.

سورس این بخش به صورت ضمیمه به این مطلب اضافه شده است که از بالای صفحه قابل دریافت است. برای کم شدن فایل پوشه node_modules حذف شده که برای resolve کردن dependency ها می تونید دستور npm install رو برای دانلود dependency ها اجرا کنید.

نویسنده: حسین احمدی

منبع: جزیره طراحی و توسعه وب توسینسو

هر گونه نشر و کپی برداری با ذکر نام نویسنده و منبع بلامانع است.

[مطلب اصلی](#)