**My Gulp Job Example**

The one line definition of [Gulp](https://gulpjs.com/) would be that it is a build tool that automate application builds. It is usually used for front end build, but you can also use it for backend application build (e.g. automating .NET application build with gulp-msbuild).

When it comes to front end build automation, gulp is the most flexible and powerful build tool. The tasks can be written in JavaScript. The heavy lifting like compiling and minifying SASS, JavaScript or TypeScript can be handled by using existing gulp modules. It can put vendor scripts and your custom scripts into a single main file. The performance is good with JavaScript asynch executions and the ability to execute tasks in parallel.

Gulp is easy to learn. It is the tool that has extremely high effort-rewards ratio. When it comes to gulp, you only needs to know four functions.

Gulp.task defines the task. You can simply pass the call back function that defines the actual task.

Gulp.src is the folder where the task is executed.

Gulp.des is the folder where the build product goes.

Gulp.watch enables hot loading. It triggers build jobs whenever the file changes.

That is it. Once you know them, all you need to do is to use gulp modules that do all the magic.

Now, let me show you the gulp task I made. It used to compile front end code in this repo ([Sakura Translation Frontend example](https://github.com/mydatahack/sakura-translation-frontend-example)). It copies images, compiles javascript and sass. You can use this as a boilerplate or expand your existing gulp job.

This is my gulp job!

<pre class=" toolbar:2 lang:js font-size:12px decode:true" >

</pre>

If you have your gulp job example, please feel free to post the link in the comment!