Rollup is a module bundler for JavaScript which compiles small pieces of code into something larger and more complex, such as a library or application. It uses the standardized ES module format for code, instead of previous idiosyncratic solutions such as CommonJS and AMD.

INSTALL:

```
#Global
npm install --global rollup
#Local
npm install --save-dev rollup
```

Rollup can be used either through a command line interface with an optional configuration file, or else through its JavaScript API. Run rollup --help to see the available options and parameters.

COMMANDS

These commands assume the entry point to your application is named main.js, and that you'd like all imports compiled into a single file named bundle.js.

FOR BROWSERS:

```
# compile to a script containing a self-executing function
rollup main.js --format iife --name "Bundle" --file
bundle.js
```

FOR NODE.JS:

```
# compile to a CommonJS module
rollup main.js --format cjs --file bundle.js
```

FOR BOTH BROWSERS AND NODE.JS:

```
# UMD format requires a bundle name
rollup main.js --format umd --name "Bundle" --file
bundle.js
```

ROLLUP.CONFIG.JS

```
# To Build: rollup-c
export default {
        entry: 'src/scripts/main.js',
        dest: 'bild/js/main.min.js',
        format: 'iife',
        sourceMap: 'inline'
};
```

TREE SHAKING

In addition to enabling the use of ES modules, Rollup also statically analyzes and optimizes the code you are importing, and will exclude anything that isn't actually used. This allows you to build on top of existing tools and modules without adding extra dependencies or bloating the size of your project.

Conclusion: Rollup JS is a tool to generate output JS files that are easy for human to read, the use of treeshake remove unused scripts and other pluggins help developers to catch errors and write better LOC.

ROLLUP JS:

https://rollupjs.org/guide/en/