To run this example you need to install `webpack-dev-server` and run `webpack serve`.

example.js

```js
const libraries = {
    react: () => import("react"),
    acorn: () => import("acorn"),
    "core-js": () => import("core-js"),
    lodash: () => import("lodash"),
    xxhashjs: () => import("xxhashjs"),
    "all of them": () => import("./all")
};

document.body.style = "font-size: 16pt;";
const pre = document.createElement("pre");
pre.style = "height: 200px; overflow-y: auto";
pre.innerText =
    "Click on a button to load the library with import(). The first click triggers a lazy co
for (const key of Object.keys(libraries)) {
    const button = document.createElement("button");
    const loadFn = libraries[key];
    button.innerText = key;
    button.onclick = async () => {
        pre.innerText = "Loading " + key + "...";
        const result = await loadFn();
        pre.innerText = `${key} = {\n  ${Object.keys(result).join(",\n  ")}\n}`;
    };
    document.body.appendChild(button);
}
const button = document.createElement("button");
button.innerText = "Load more...";
button.onclick = async () => {
    pre.innerText = "Loading more...";
    await import("./more");
    pre.innerText = "More libraries available.";
};
document.body.appendChild(button);
document.body.appendChild(pre);
```

webpack.config.js

```js
const { HotModuleReplacementPlugin } = require("../../");
```

```javascript
module.exports = {
    mode: "development",
    entry: {
        main: "./example.js"
    },
    cache: {
        type: "filesystem",
        idleTimeout: 5000
    },
    experiments: {
        lazyCompilation: true
    },
    devServer: {
        hot: true,
        devMiddleware: {
            publicPath: "/dist/"
        }
    },
    plugins: [new HotModuleReplacementPlugin()]
};
```