example.js

This example illustrates how to leverage the `import()` syntax to create ContextModules which are separated into separate chunks for each module in the `./templates` folder.

```js
async function getTemplate(templateName) {
    try {
        let template = await import(`./templates/${templateName}`);
        console.log(template);
    } catch(err) {
        console.error("template error");
        return new Error(err);
    }
}

getTemplate("foo");
getTemplate("bar");
getTemplate("baz");
```

templates/

- foo.js
- baz.js
- bar.js

All templates are of this pattern:

```js
var foo = "foo";

export default foo;
```

dist/output.js

```js
/******/ (() => { // webpackBootstrap
/******/        var __webpack_modules__ = ([
/* 0 */,
/* 1 */
/*!*****************************************************!*\
  !*** ./templates/ lazy ^\.\/.*$ namespace object ***!
  \*****************************************************/
/*! default exports */
/*! exports [not provided] [no usage info] */
/*! runtime requirements: module, __webpack_require__.o, __webpack_require__,
__webpack_require__.e, __webpack_require__.* */
/***/ ((module, __unused_webpack_exports, __webpack_require__) => {

var map = {
```

```
    "./bar": [
        2,
        398
    ],
    "./bar.js": [
        2,
        398
    ],
    "./baz": [
        3,
        544
    ],
    "./baz.js": [
        3,
        544
    ],
    "./foo": [
        4,
        718
    ],
    "./foo.js": [
        4,
        718
    ]
};
function webpackAsyncContext(req) {
    if(!__webpack_require__.o(map, req)) {
        return Promise.resolve().then(() => {
            var e = new Error("Cannot find module '" + req + "'");
            e.code = 'MODULE_NOT_FOUND';
            throw e;
        });
    }

    var ids = map[req], id = ids[0];
    return __webpack_require__.e(ids[1]).then(() => {
        return __webpack_require__(id);
    });
}
webpackAsyncContext.keys = () => (Object.keys(map));
webpackAsyncContext.id = 1;
module.exports = webpackAsyncContext;

/***/ })
/******/        ]);
```

▶ /* webpack runtime code */

```
var __webpack_exports__ = {};
// This entry need to be wrapped in an IIFE because it need to be isolated against
other modules in the chunk.
```

```
(() => {
/*!********************!*\
  !*** ./example.js ***!
  \********************/
/*! unknown exports (runtime-defined) */
/*! runtime requirements: __webpack_require__ */
async function getTemplate(templateName) {
    try {
        let template = await __webpack_require__(1)(`./${templateName}`);
        console.log(template);
    } catch(err) {
        console.error("template error");
        return new Error(err);
    }
}

getTemplate("foo");
getTemplate("bar");
getTemplate("baz");




})();

/******/ })()
;
```

Info

Unoptimized

```
asset output.js 11 KiB [emitted] (name: main)
asset 398.output.js 858 bytes [emitted]
asset 544.output.js 858 bytes [emitted]
asset 718.output.js 858 bytes [emitted]
chunk (runtime: main) output.js (main) 441 bytes (javascript) 5.54 KiB (runtime)
[entry] [rendered]
  > ./example.js main
  runtime modules 5.54 KiB 8 modules
  dependent modules 160 bytes [dependent] 1 module
  ./example.js 281 bytes [built] [code generated]
    [used exports unknown]
    entry ./example.js main
chunk (runtime: main) 398.output.js 38 bytes [rendered]
  > ./bar ./templates/ lazy ^\.\/.*$ namespace object ./bar
  > ./bar.js ./templates/ lazy ^\.\/.*$ namespace object ./bar.js
  ./templates/bar.js 38 bytes [optional] [built] [code generated]
    [exports: default]
    [used exports unknown]
```

```
    import() context element ./bar ./templates/ lazy ^\.\/.*$ namespace object ./bar
    import() context element ./bar.js ./templates/ lazy ^\.\/.*$ namespace object
./bar.js
chunk (runtime: main) 544.output.js 38 bytes [rendered]
  > ./baz ./templates/ lazy ^\.\/.*$ namespace object ./baz
  > ./baz.js ./templates/ lazy ^\.\/.*$ namespace object ./baz.js
  ./templates/baz.js 38 bytes [optional] [built] [code generated]
    [exports: default]
    [used exports unknown]
    import() context element ./baz ./templates/ lazy ^\.\/.*$ namespace object ./baz
    import() context element ./baz.js ./templates/ lazy ^\.\/.*$ namespace object
./baz.js
chunk (runtime: main) 718.output.js 38 bytes [rendered]
  > ./foo ./templates/ lazy ^\.\/.*$ namespace object ./foo
  > ./foo.js ./templates/ lazy ^\.\/.*$ namespace object ./foo.js
  ./templates/foo.js 38 bytes [optional] [built] [code generated]
    [exports: default]
    [used exports unknown]
    import() context element ./foo ./templates/ lazy ^\.\/.*$ namespace object ./foo
    import() context element ./foo.js ./templates/ lazy ^\.\/.*$ namespace object
./foo.js
webpack 5.51.1 compiled successfully
```

Production mode

```
asset output.js 2.44 KiB [emitted] [minimized] (name: main)
asset 398.output.js 130 bytes [emitted] [minimized]
asset 544.output.js 130 bytes [emitted] [minimized]
asset 718.output.js 130 bytes [emitted] [minimized]
chunk (runtime: main) output.js (main) 441 bytes (javascript) 5.54 KiB (runtime)
[entry] [rendered]
  > ./example.js main
  runtime modules 5.54 KiB 8 modules
  dependent modules 160 bytes [dependent] 1 module
  ./example.js 281 bytes [built] [code generated]
    [no exports used]
    entry ./example.js main
chunk (runtime: main) 398.output.js 38 bytes [rendered]
  > ./bar ./templates/ lazy ^\.\/.*$ namespace object ./bar
  > ./bar.js ./templates/ lazy ^\.\/.*$ namespace object ./bar.js
  ./templates/bar.js 38 bytes [optional] [built] [code generated]
    [exports: default]
    import() context element ./bar ./templates/ lazy ^\.\/.*$ namespace object ./bar
    import() context element ./bar.js ./templates/ lazy ^\.\/.*$ namespace object
./bar.js
chunk (runtime: main) 544.output.js 38 bytes [rendered]
  > ./baz ./templates/ lazy ^\.\/.*$ namespace object ./baz
  > ./baz.js ./templates/ lazy ^\.\/.*$ namespace object ./baz.js
  ./templates/baz.js 38 bytes [optional] [built] [code generated]
    [exports: default]
    import() context element ./baz ./templates/ lazy ^\.\/.*$ namespace object ./baz
```

```
    import() context element ./baz.js ./templates/ lazy ^\.\/.*$ namespace object
./baz.js
chunk (runtime: main) 718.output.js 38 bytes [rendered]
  > ./foo ./templates/ lazy ^\.\/.*$ namespace object ./foo
  > ./foo.js ./templates/ lazy ^\.\/.*$ namespace object ./foo.js
  ./templates/foo.js 38 bytes [optional] [built] [code generated]
    [exports: default]
    import() context element ./foo ./templates/ lazy ^\.\/.*$ namespace object ./foo
    import() context element ./foo.js ./templates/ lazy ^\.\/.*$ namespace object
./foo.js
webpack 5.51.1 compiled successfully
```