This example illustrates a very simple case of Code Splitting with `require.ensure`.

- `a` and `b` are required normally via CommonJS
- `c` is made available(,but doesn't get execute) through the `require.ensure` array.
    - webpack will load it on demand
- `b` and `d` are required via CommonJs in the `require.ensure` callback
    - webpack detects that these are in the on-demand-callback and
    - will load them on demand
    - webpack's optimizer can optimize `b` away
        - as it is already available through the parent chunks

You can see that webpack outputs two files/chunks:

- `output.js` is the entry chunk and contains
    - the module system
    - chunk loading logic
    - the entry point `example.js`
    - module `a`
    - module `b`
- `1.output.js` is an additional chunk (on-demand loaded) and contains
    - module `c`
    - module `d`

You can see that chunks are loaded via JSONP. The additional chunks are pretty small and minimize well.

# example.js

```js
var a = require("a");
var b = require("b");
require.ensure(["c"], function(require) {
    require("b").xyz();
    var d = require("d");
});
```

# dist/output.js

```js
/******/ (() => { // webpackBootstrap
/******/     var __webpack_modules__ = ([
/* 0 */,
/* 1 */
/*!************************!*\
  !*** ./node_modules/a.js ***!
  \************************/
/*! unknown exports (runtime-defined) */
/*! runtime requirements:  */
/***/ (() => {
```

```
// module a

/***/ }),
/* 2 */
/*!***************************!*\
  !*** ./node_modules/b.js ***!
  \***************************/
/*! unknown exports (runtime-defined) */
/*! runtime requirements:  */
/***/ (() => {

// module b

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

▶ /* 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__, __webpack_require__.e,
__webpack_require__.* */
var a = __webpack_require__(/*! a */ 1);
var b = __webpack_require__(/*! b */ 2);
__webpack_require__.e(/*! require.ensure */ 796).then((function(require) {
    __webpack_require__(/*! b */ 2).xyz();
    var d = __webpack_require__(/*! d */ 4);
}).bind(null, __webpack_require__)).catch(__webpack_require__.oe);
})();


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

## dist/796.output.js

```
(self["webpackChunk"] = self["webpackChunk"] || []).push([[796],[
/* 0 */,
/* 1 */,
/* 2 */,
/* 3 */
/*!***************************!*\
  !*** ./node_modules/c.js ***!
```

```
   \****************************/
 /*! unknown exports (runtime-defined) */
 /*! runtime requirements:  */
 /***/ (() => {

 // module c

 /***/ }),
 /* 4 */
 /*!****************************!*\
   !*** ./node_modules/d.js ***!
   \****************************/
 /*! unknown exports (runtime-defined) */
 /*! runtime requirements:  */
 /***/ (() => {

 // module d

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

Minimized

```
 (self.webpackChunk=self.webpackChunk||[]).push([[796],{286:()=>{},882:()=>{}}]);
```

# Info

## Unoptimized

```
asset output.js 9.49 KiB [emitted] (name: main)
asset 796.output.js 528 bytes [emitted]
chunk (runtime: main) output.js (main) 161 bytes (javascript) 4.98 KiB (runtime)
[entry] [rendered]
  > ./example.js main
  runtime modules 4.98 KiB 6 modules
  dependent modules 22 bytes [dependent] 2 modules
  ./example.js 139 bytes [built] [code generated]
    [used exports unknown]
    entry ./example.js main
chunk (runtime: main) 796.output.js 22 bytes [rendered]
  > ./example.js 3:0-6:2
  ./node_modules/c.js 11 bytes [built] [code generated]
    [used exports unknown]
    require.ensure item c ./example.js 3:0-6:2
  ./node_modules/d.js 11 bytes [built] [code generated]
    [used exports unknown]
    cjs require d ./example.js 5:12-24
webpack 5.51.1 compiled successfully
```

## Production mode

```
asset output.js 1.74 KiB [emitted] [minimized] (name: main)
asset 796.output.js 80 bytes [emitted] [minimized]
chunk (runtime: main) output.js (main) 161 bytes (javascript) 4.98 KiB (runtime)
[entry] [rendered]
  > ./example.js main
  runtime modules 4.98 KiB 6 modules
  dependent modules 22 bytes [dependent] 2 modules
  ./example.js 139 bytes [built] [code generated]
    [no exports used]
    entry ./example.js main
chunk (runtime: main) 796.output.js 22 bytes [rendered]
  > ./example.js 3:0-6:2
  ./node_modules/c.js 11 bytes [built] [code generated]
    [used exports unknown]
    require.ensure item c ./example.js 3:0-6:2
  ./node_modules/d.js 11 bytes [built] [code generated]
    [used exports unknown]
    cjs require d ./example.js 5:12-24
webpack 5.51.1 compiled successfully
```