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

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

# dist/output.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 */
/*****/
          // The module cache
/*****/
          var __webpack_module_cache__ = {};
/*****/
/*****/
          // The require function
/*****/
          function __webpack_require__(moduleId) {
/*****/
              // Check if module is in cache
/*****/
              var cachedModule = __webpack_module_cache__[moduleId];
/*****/
              if (cachedModule !== undefined) {
/*****/
                  return cachedModule.exports;
/*****/
/*****/
              // Create a new module (and put it into the cache)
/*****/
              var module = __webpack_module_cache__[moduleId] = {
/*****/
                 // no module.id needed
/*****/
                  // no module.loaded needed
/*****/
                  exports: {}
/*****/
              };
/*****/
/*****/
              // Execute the module function
/*****/
              __webpack_modules__[moduleId] (module, module.exports, __webpack_require__);
/*****/
/*****/
              // Return the exports of the module
/*****/
              return module.exports;
/*****/
          }
/*****/
```

```
/*****/
            // expose the modules object (__webpack_modules__)
/*****/
            __webpack_require__.m = __webpack_modules__;
/*****/
/*****
             **********************
/*****/
            /* webpack/runtime/ensure chunk */
/*****/
            (() => {
               __webpack_require__.f = {};
/*****/
/*****/
               // This file contains only the entry chunk.
/*****/
               // The chunk loading function for additional chunks
/*****/
               __webpack_require__.e = (chunkId) => {
/*****/
                   return Promise.all(Object.keys(__webpack_require__.f).reduce((promises,
/*****/
                       __webpack_require__.f[key](chunkId, promises);
/*****/
                       return promises;
/*****/
                   }, []));
/*****/
               };
/*****/
            })();
/*****/
/*****/
            /* webpack/runtime/get javascript chunk filename */
/*****/
            (() => {
/*****/
               // This function allow to reference async chunks
/*****/
                __webpack_require__.u = (chunkId) => {
/*****/
                    // return url for filenames based on template
/*****/
                   return "" + chunkId + ".output.js";
/*****/
               };
/*****/
            })();
/*****/
/*****/
            /* webpack/runtime/hasOwnProperty shorthand */
/*****/
/*****/
               __webpack_require__.o = (obj, prop) => (Object.prototype.hasOwnProperty.cal
/*****/
            })();
/*****/
/*****/
            /* webpack/runtime/load script */
/*****/
            (() => \{
/*****/
               var inProgress = {};
/*****/
               // data-webpack is not used as build has no uniqueName
/*****/
               // loadScript function to load a script via script tag
/*****/
                __webpack_require__.l = (url, done, key, chunkId) => {
/*****/
                   if(inProgress[url]) { inProgress[url].push(done); return; }
/*****/
                   var script, needAttach;
/*****/
                   if(key !== undefined) {
/*****/
                       var scripts = document.getElementsByTagName("script");
/*****/
                       for(var i = 0; i < scripts.length; i++) {</pre>
/*****/
                           var s = scripts[i];
/*****/
                           if(s.getAttribute("src") == url) { script = s; break; }
/*****/
                       }
/*****/
                   }
```

```
/*****/
                    if(!script) {
/*****/
                        needAttach = true;
/*****/
                        script = document.createElement('script');
/*****/
/*****/
                        script.charset = 'utf-8';
/*****/
                        script.timeout = 120;
/*****/
                        if (_webpack_require__.nc) {
/*****/
                            script.setAttribute("nonce", __webpack_require__.nc);
/*****/
/*****/
/*****/
                        script.src = url;
/*****/
                    }
/*****/
                    inProgress[url] = [done];
/*****/
                    var onScriptComplete = (prev, event) => {
/*****/
                        // avoid mem leaks in IE.
/*****/
                        script.onerror = script.onload = null;
/*****/
                        clearTimeout(timeout);
/*****/
                        var doneFns = inProgress[url];
/*****/
                        delete inProgress[url];
/*****/
                        script.parentNode && script.parentNode.removeChild(script);
/*****/
                        doneFns && doneFns.forEach((fn) => (fn(event)));
/*****/
                        if(prev) return prev(event);
/*****/
                    }
/*****/
/*****/
                    var timeout = setTimeout(onScriptComplete.bind(null, undefined, { type:
/*****/
                    script.onerror = onScriptComplete.bind(null, script.onerror);
/*****/
                    script.onload = onScriptComplete.bind(null, script.onload);
/*****/
                    needAttach && document.head.appendChild(script);
/*****/
                };
/*****/
            })();
/*****/
/*****/
            /* webpack/runtime/publicPath */
/*****/
            (() => \{
/*****/
                __webpack_require__.p = "dist/";
/*****/
            })();
/*****/
/*****/
            /* webpack/runtime/jsonp chunk loading */
/*****/
            (() => {
/*****/
               // no baseURI
/*****/
/*****/
                // object to store loaded and loading chunks
/*****/
                // undefined = chunk not loaded, null = chunk preloaded/prefetched
/*****/
                // [resolve, reject, Promise] = chunk loading, 0 = chunk loaded
/*****/
                var installedChunks = {
/*****/
                    179: 0
/*****/
                };
```

```
/*****/
/*****/
                __webpack_require__.f.j = (chunkId, promises) => {
/*****/
                        // JSONP chunk loading for javascript
/*****/
                        var installedChunkData = __webpack_require__.o(installedChunks, chu
/*****/
                        if(installedChunkData !== 0) { // O means "already installed".
/*****/
/*****/
                            // a Promise means "currently loading".
/*****/
                            if(installedChunkData) {
/*****/
                                promises.push(installedChunkData[2]);
/*****/
                            } else {
/*****/
                                if(true) { // all chunks have JS
/*****/
                                    // setup Promise in chunk cache
/*****/
                                    var promise = new Promise((resolve, reject) => (install
/*****/
                                    promises.push(installedChunkData[2] = promise);
/*****/
/*****/
                                    // start chunk loading
/*****/
                                    var url = __webpack_require__.p + __webpack_require__.u
/*****/
                                    // create error before stack unwound to get useful stac
/*****/
                                    var error = new Error();
/*****/
                                    var loadingEnded = (event) => {
/*****/
                                        if(__webpack_require__.o(installedChunks, chunkId))
/*****/
                                            installedChunkData = installedChunks[chunkId];
/*****/
                                            if(installedChunkData !== 0) installedChunks[ch
/*****/
                                            if(installedChunkData) {
/*****/
                                                var errorType = event && (event.type === '1
/*****/
                                                var realSrc = event && event.target && even
/*****/
                                                error.message = 'Loading chunk ' + chunkId
/*****/
                                                error.name = 'ChunkLoadError';
/*****/
                                                error.type = errorType;
/*****/
                                                error.request = realSrc;
/*****/
                                                installedChunkData[1](error);
/*****/
                                            }
/*****/
                                        }
/*****/
                                    __webpack_require__.l(url, loadingEnded, "chunk-" + chu
/*****/
/*****/
                                } else installedChunks[chunkId] = 0;
/*****/
                            }
/*****/
                        }
/*****/
                };
/*****/
/*****/
                // no prefetching
/*****/
/*****/
                // no preloaded
/*****/
                // no HMR
/*****/
```

/\*\*\*\*\*/

```
/*****/
/*****/
               // no on chunks loaded
/*****/
/*****/
               // install a JSONP callback for chunk loading
/*****/
               var webpackJsonpCallback = (parentChunkLoadingFunction, data) => {
/*****/
                  var [chunkIds, moreModules, runtime] = data;
/*****/
                  // add "moreModules" to the modules object,
/*****/
                  // then flag all "chunkIds" as loaded and fire callback
/*****/
                  var moduleId, chunkId, i = 0;
/*****/
                  if(chunkIds.some((id) => (installedChunks[id] !== 0))) {
/*****/
                      for(moduleId in moreModules) {
/*****/
                          if(__webpack_require__.o(moreModules, moduleId)) {
/*****/
                              __webpack_require__.m[moduleId] = moreModules[moduleId];
/*****/
                          }
/*****/
                      }
/*****/
                      if(runtime) var result = runtime(__webpack_require__);
/*****/
                  }
/*****/
                  if(parentChunkLoadingFunction) parentChunkLoadingFunction(data);
/*****/
                  for(;i < chunkIds.length; i++) {</pre>
/*****/
                      chunkId = chunkIds[i];
/*****/
                      if(__webpack_require__.o(installedChunks, chunkId) && installedChun
/*****/
                          installedChunks[chunkId][0]();
/*****/
                      }
/*****/
                      installedChunks[chunkIds[i]] = 0;
/*****/
                  }
/*****/
/*****/
               }
/*****/
/*****/
               var chunkLoadingGlobal = self["webpackChunk"] = self["webpackChunk"] || [];
/*****/
               chunkLoadingGlobal.forEach(webpackJsonpCallback.bind(null, 0));
/*****/
               chunkLoadingGlobal.push = webpackJsonpCallback.bind(null, chunkLoadingGloba
/*****/
           })();
/*****/
var __webpack_exports__ = {};
// This entry need to be wrapped in an IIFE because it need to be isolated against other more
(() => {
/*!*********************
 !*** ./example.js ***!
 /*! unknown exports (runtime-defined) */
/*! runtime requirements: __webpack_require__.*
var a = __webpack_require__(/*! a */ 1);
var b = __webpack_require__(/*! b */ 2);
```

/\*\*\*\*\*/

// no HMR manifest

```
__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] [re
  > ./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] [re
  > ./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
```