## 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.

```
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:

```
var foo = "foo";

export default foo;
```

## dist/output.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_requ
/***/ ((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 */

```
/************************************************************************/
/******/    // 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/define property getters */
/******/    (() => {
/******/        // define getter functions for harmony exports
/******/        __webpack_require__.d = (exports, definition) => {
/******/            for(var key in definition) {
/******/                if(__webpack_require__.o(definition, key) && !__webpack_require__.o
/******/                    Object.defineProperty(exports, key, { enumerable: true, get: de
/******/                }
/******/            }
/******/        };
/******/    })();
/******/
/******/    /* 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++) {
/******/                     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/make namespace object */
/******/     (() => {
/******/         // define __esModule on exports
/******/         __webpack_require__.r = (exports) => {
/******/             if(typeof Symbol !== 'undefined' && Symbol.toStringTag) {
/******/                 Object.defineProperty(exports, Symbol.toStringTag, { value: 'Module
/******/             }
/******/             Object.defineProperty(exports, '__esModule', { value: true });
/******/         };
/******/     })();
/******/
/******/     /* 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) { // 0 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 === 'l
/******/ 	                                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 HMR manifest
/******/
/******/ 		// 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++) {
/******/ 				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 mo
(() => {
/*!*********************!*\
  !*** ./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] [re
  > ./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] [re
  > ./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
```