webpack.config.js

```js
var path = require("path");
module.exports = {
    // mode: "development || "production",
    entry: {
        // The entry points for the pages
        // They also contains router
        pageA: ["./aEntry", "./router"],
        pageB: ["./bEntry", "./router"]
    },
    output: {
        path: path.join(__dirname, "dist"),
        publicPath: "js/",
        filename: "[name].bundle.js",
        chunkFilename: "[name].chunk.js"
    },
    optimization: {
        // Extract common modules from initial chunks too
        // This is optional, but good for performance.
        splitChunks: {
            chunks: "all",
            minSize: 0 // This example is too small
        },
        chunkIds: "named" // To keep filename consistent between different modes
(for example building only)
    }
};
```

aEntry.js

```js
// Just show the page "a"
var render = require("./render");
render(require("./aPage"));
```

`bEntry.js` is similar. You may want to use a loader to generate this file.

aPage.js

```js
module.exports = function() {
    return "This is page A.";
};
```

`bEntry.js` is similar.

router.js

```javascript
var render = require("./render");

// Event when another page should be opened
// Maybe hook click on links, hashchange or popstate
window.onLinkToPage = function onLinkToPage(name) { // name is "a" or "b"
    // require the page with a dynamic require

    // It's important that this require only matches the pages
    //  otherwise there is blood in the bundle. Here this is done with a
    //  specific file prefix. It's also possible to use a directory,
    //  overwriting the RegExp with the ContextReplacementPlugin, or
    //  using the require.context method.

    // This line may throw a exception on runtime if the page wasn't found.
    import(/* webpackChunkName: "[request]" */`./${name}Page`).then(page => {;
        render(page.default);
    });
}
```

pageA.html

```html
<html>
    <head></head>
    <body>
        <script async src="dist/pageA~pageB.chunk.js" charset="utf-8"></script>
        <script async src="dist/aPage.chunk.js" charset="utf-8"></script>
        <script async src="dist/pageA.bundle.js" charset="utf-8"></script>
    </body>
</html>
```

dist/router_js.bundle.js

```javascript
(self["webpackChunk"] = self["webpackChunk"] || []).push([["router_js"],[
/* 0 */,
/* 1 */
/*!********************!*\
  !*** ./render.js ***!
  \********************/
/*! unknown exports (runtime-defined) */
/*! runtime requirements: module */
/*! CommonJS bailout: module.exports is used directly at 1:0-14 */
/***/ ((module) => {
```

```
module.exports = function(page) {
    console.log(page());
};


/***/ }),
/* 2 */,
/* 3 */
/*!********************!*\
  !*** ./router.js ***!
  \********************/
/*! unknown exports (runtime-defined) */
/*! runtime requirements: __webpack_require__ */
/***/ ((__unused_webpack_module, __unused_webpack_exports, __webpack_require__) => {

var render = __webpack_require__(/*! ./render */ 1);

// Event when another page should be opened
// Maybe hook click on links, hashchange or popstate
window.onLinkToPage = function onLinkToPage(name) { // name is "a" or "b"
    // require the page with a dynamic require

    // It's important that this require only matches the pages
    //  otherwise there is blood in the bundle. Here this is done with a
    //  specific file prefix. It's also possible to use a directory,
    //  overwriting the RegExp with the ContextReplacementPlugin, or
    //  using the require.context method.

    // This line may throw a exception on runtime if the page wasn't found.
    __webpack_require__(4)(`./${name}Page`).then(page => {;
        render(page.default);
    });
}


/***/ }),
/* 4 */
/*!*********************************************************************!*\
  !*** ./ lazy ^\.\/.*Page$ chunkName: [request] namespace object ***!
  \*********************************************************************/
/*! default exports */
/*! exports [not provided] [no usage info] */
/*! runtime requirements: module, __webpack_require__.o, __webpack_require__,
__webpack_require__.e, __webpack_require__.t, __webpack_require__.* */
/***/ ((module, __unused_webpack_exports, __webpack_require__) => {

var map = {
    "./aPage": [
        2,
        "aPage"
    ],
    "./bPage": [
        6,
```

```
        "bPage"
    ]
};
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__.t(id, 7 | 16);
    });
}
webpackAsyncContext.keys = () => (Object.keys(map));
webpackAsyncContext.id = 4;
module.exports = webpackAsyncContext;

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

dist/pageA.bundle.js

```
/******/ (() => { // webpackBootstrap
/******/         var __webpack_modules__ = ([
/* 0 */
/*!*******************!*\
  !*** ./aEntry.js ***!
  \*******************/
/*! unknown exports (runtime-defined) */
/*! runtime requirements: __webpack_require__ */
/***/ ((__unused_webpack_module, __unused_webpack_exports, __webpack_require__) => {

// Just show the page "a"
var render = __webpack_require__(/*! ./render */ 1);
render(__webpack_require__(/*! ./aPage */ 2));

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

▶ /* webpack runtime code */

```
/******/
/******/         // startup
/******/         // Load entry module and return exports
/******/         // This entry module depends on other loaded chunks and execution
```

```
    need to be delayed
/******/            __webpack_require__.O(undefined, ["router_js","aPage"], () =>
(__webpack_require__(0)))
/******/            var __webpack_exports__ = __webpack_require__.O(undefined,
["router_js","aPage"], () => (__webpack_require__(3)))
/******/            __webpack_exports__ = __webpack_require__.O(__webpack_exports__);
/******/
/******/ })()
;
```

dist/aPage.bundle.js

```
(self["webpackChunk"] = self["webpackChunk"] || []).push([["aPage"],{

/***/ 2:
/*!*******************!*\
  !*** ./aPage.js ***!
  \*******************/
/*! unknown exports (runtime-defined) */
/*! runtime requirements: module */
/*! CommonJS bailout: module.exports is used directly at 1:0-14 */
/***/ ((module) => {

module.exports = function() {
    return "This is page A.";
};


/***/ })

}]);
```

Info

Unoptimized

```
asset pageB.bundle.js 13 KiB [emitted] (name: pageB)
asset pageA.bundle.js 13 KiB [emitted] (name: pageA)
asset router_js.bundle.js 2.45 KiB [emitted]
asset aPage.bundle.js 392 bytes [emitted] (name: aPage)
asset bPage.bundle.js 392 bytes [emitted] (name: bPage)
Entrypoint pageA 15.9 KiB = router_js.bundle.js 2.45 KiB aPage.bundle.js 392 bytes
pageA.bundle.js 13 KiB
Entrypoint pageB 15.9 KiB = router_js.bundle.js 2.45 KiB bPage.bundle.js 392 bytes
pageB.bundle.js 13 KiB
chunk (runtime: pageA, pageB) aPage.bundle.js (aPage) 59 bytes [initial] [rendered]
reused as split chunk (cache group: default)
  > ./aPage ././ lazy ^\.\/.*Page$ chunkName: [request] namespace object ./aPage
```

```
    > ./aEntry pageA
    > ./router pageA
   ./aPage.js 59 bytes [built] [code generated]
     [used exports unknown]
     cjs require ./aPage ./aEntry.js 3:7-25
     cjs self exports reference ./aPage.js 1:0-14
     import() context element ./aPage ././ lazy ^\.\/.*Page$ chunkName: [request]
namespace object ./aPage
chunk (runtime: pageA, pageB) bPage.bundle.js (bPage) 59 bytes [initial] [rendered]
reused as split chunk (cache group: default)
    > ./bPage ././ lazy ^\.\/.*Page$ chunkName: [request] namespace object ./bPage
    > ./bEntry pageB
    > ./router pageB
   ./bPage.js 59 bytes [built] [code generated]
     [used exports unknown]
     cjs require ./bPage ./bEntry.js 3:7-25
     cjs self exports reference ./bPage.js 1:0-14
     import() context element ./bPage ././ lazy ^\.\/.*Page$ chunkName: [request]
namespace object ./bPage
chunk (runtime: pageA) pageA.bundle.js (pageA) 87 bytes (javascript) 7.61 KiB
(runtime) [entry] [rendered]
    > ./aEntry pageA
    > ./router pageA
   runtime modules 7.61 KiB 10 modules
   ./aEntry.js 87 bytes [built] [code generated]
     [used exports unknown]
     entry ./aEntry pageA
chunk (runtime: pageB) pageB.bundle.js (pageB) 87 bytes (javascript) 7.61 KiB
(runtime) [entry] [rendered]
    > ./bEntry pageB
    > ./router pageB
   runtime modules 7.61 KiB 10 modules
   ./bEntry.js 87 bytes [built] [code generated]
     [used exports unknown]
     entry ./bEntry pageB
chunk (runtime: pageA, pageB) router_js.bundle.js 951 bytes [initial] [rendered] split
chunk (cache group: default)
    > ./aEntry pageA
    > ./router pageA
    > ./bEntry pageB
    > ./router pageB
   dependent modules 218 bytes [dependent] 2 modules
   ./router.js 733 bytes [built] [code generated]
     [used exports unknown]
     entry ./router pageA
     entry ./router pageB
webpack 5.51.1 compiled successfully
```

Production mode

```
asset pageA.bundle.js 2.83 KiB [emitted] [minimized] (name: pageA)
asset pageB.bundle.js 2.83 KiB [emitted] [minimized] (name: pageB)
asset router_js.bundle.js 544 bytes [emitted] [minimized]
asset aPage.bundle.js 117 bytes [emitted] [minimized] (name: aPage)
asset bPage.bundle.js 117 bytes [emitted] [minimized] (name: bPage)
Entrypoint pageA 3.48 KiB = router_js.bundle.js 544 bytes aPage.bundle.js 117 bytes
pageA.bundle.js 2.83 KiB
Entrypoint pageB 3.48 KiB = router_js.bundle.js 544 bytes bPage.bundle.js 117 bytes
pageB.bundle.js 2.83 KiB
chunk (runtime: pageA, pageB) aPage.bundle.js (aPage) 59 bytes [initial] [rendered]
reused as split chunk (cache group: default)
  > ./aPage ././ lazy ^\.\/.*Page$ chunkName: [request] namespace object ./aPage
  > ./aEntry pageA
  > ./router pageA
  ./aPage.js 59 bytes [built] [code generated]
    [used exports unknown]
    cjs require ./aPage ./aEntry.js 3:7-25
    cjs self exports reference ./aPage.js 1:0-14
    import() context element ./aPage ././ lazy ^\.\/.*Page$ chunkName: [request]
namespace object ./aPage
chunk (runtime: pageA, pageB) bPage.bundle.js (bPage) 59 bytes [initial] [rendered]
reused as split chunk (cache group: default)
  > ./bPage ././ lazy ^\.\/.*Page$ chunkName: [request] namespace object ./bPage
  > ./bEntry pageB
  > ./router pageB
  ./bPage.js 59 bytes [built] [code generated]
    [used exports unknown]
    cjs require ./bPage ./bEntry.js 3:7-25
    cjs self exports reference ./bPage.js 1:0-14
    import() context element ./bPage ././ lazy ^\.\/.*Page$ chunkName: [request]
namespace object ./bPage
chunk (runtime: pageA) pageA.bundle.js (pageA) 87 bytes (javascript) 7.61 KiB
(runtime) [entry] [rendered]
  > ./aEntry pageA
  > ./router pageA
  runtime modules 7.61 KiB 10 modules
  ./aEntry.js 87 bytes [built] [code generated]
    [no exports used]
    entry ./aEntry pageA
chunk (runtime: pageB) pageB.bundle.js (pageB) 87 bytes (javascript) 7.61 KiB
(runtime) [entry] [rendered]
  > ./bEntry pageB
  > ./router pageB
  runtime modules 7.61 KiB 10 modules
  ./bEntry.js 87 bytes [built] [code generated]
    [no exports used]
    entry ./bEntry pageB
chunk (runtime: pageA, pageB) router_js.bundle.js 951 bytes [initial] [rendered] split
chunk (cache group: default)
  > ./aEntry pageA
  > ./router pageA
```

```
  > ./bEntry pageB
  > ./router pageB
  dependent modules 218 bytes [dependent] 2 modules
  ./router.js 733 bytes [built] [code generated]
    [no exports used]
    entry ./router pageA
    entry ./router pageB
webpack 5.51.1 compiled successfully
```