# example.js

```javascript
document.body.innerHTML = `
    <pre id="history"></pre>
    <form>
    <input id="message" type="text">
    <button id="send">Send Message</button>
    </form>
    <p>Computing fibonacci without worker:</p>
    <input id="fib1" type="number">
    <pre id="output1"></pre>
    <p>Computing fibonacci with worker:</p>
    <input id="fib2" type="number">
    <pre id="output2"></pre>
`;

const history = document.getElementById("history");
const message = document.getElementById("message");
const send = document.getElementById("send");
const fib1 = document.getElementById("fib1");
const output1 = document.getElementById("output1");
const fib2 = document.getElementById("fib2");
const output2 = document.getElementById("output2");

/// CHAT with shared worker ///

const chatWorker = new SharedWorker(
    new URL("./chat-worker.js", import.meta.url),
    {
        name: "chat",
        type: "module"
    }
);

let historyTimeout;
const scheduleUpdateHistory = () => {
    clearTimeout(historyTimeout);
    historyTimeout = setTimeout(() => {
        chatWorker.port.postMessage({ type: "history" });
    }, 1000);
};
scheduleUpdateHistory();

const from = `User ${Math.floor(Math.random() * 10000)}`;

send.addEventListener("click", e => {
```

```javascript
        chatWorker.port.postMessage({
            type: "message",
            content: message.value,
            from
        });
        message.value = "";
        message.focus();
        e.preventDefault();
});

chatWorker.port.onmessage = event => {
    const msg = event.data;
    switch (msg.type) {
        case "history":
            history.innerText = msg.history.join("\n");
            scheduleUpdateHistory();
            break;
    }
};

/// FIBONACCI without worker ///

fib1.addEventListener("change", async () => {
    try {
        const value = parseInt(fib1.value, 10);
        const { fibonacci } = await import("./fibonacci");
        const result = fibonacci(value);
        output1.innerText = `fib(${value}) = ${result}`;
    } catch (e) {
        output1.innerText = e.message;
    }
});

/// FIBONACCI with worker ///

const fibWorker = new Worker(new URL("./fib-worker.js", import.meta.url), {
    name: "fibonacci",
    type: "module"
    /* webpackEntryOptions: { filename: "workers/[name].js" } */
});

fib2.addEventListener("change", () => {
    try {
        const value = parseInt(fib2.value, 10);
        fibWorker.postMessage(`${value}`);
    } catch (e) {
```

```
        output2.innerText = e.message;
    }
});

fibWorker.onmessage = event => {
    output2.innerText = event.data;
};
```

## fib-worker.js

```
onmessage = async event => {
    const { fibonacci } = await import("./fibonacci");
    const value = JSON.parse(event.data);
    postMessage(`fib(${value}) = ${fibonacci(value)}`);
};
```

## fibonacci.js

```
export function fibonacci(n) {
    return n < 1 ? 0 : n <= 2 ? 1 : fibonacci(n - 1) + fibonacci(n - 2);
}
```

## chat-worker.js

```
onconnect = function (e) {
    for (const port of e.ports) {
        port.onmessage = async event => {
            const msg = event.data;
            switch (msg.type) {
                case "message":
                    const { add } = await import("./chat-module");
                    add(msg.content, msg.from);
                // fallthrough
                case "history":
                    const { history } = await import("./chat-module");
                    port.postMessage({
                        type: "history",
                        history
                    });
                    break;
            }
        };
    }
};
```

## chat-module.js

```javascript
export const history = [];

export const add = (content, from) => {
    if (history.length > 10) history.shift();
    history.push(`${from}: ${content}`);
};
```

## dist/main.js

```javascript
/******/ var __webpack_modules__ = ({});
```

/* webpack runtime code */

```javascript
/************************************************************************/
/******/ // 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(exp
/******/                  Object.defineProperty(exports, key, { enumerable: true, get: defini
/******/              }
/******/          }
/******/      };
/******/ })();
/******/
/******/ /* 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, key
/******/              __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 not based on template
/******/          if (chunkId === 631) return "workers/fibonacci.js";
/******/          // return url for filenames based on template
/******/          return "" + (chunkId === 348 ? "chat" : chunkId) + ".js";
/******/      };
/******/ })();
/******/
/******/ /* webpack/runtime/hasOwnProperty shorthand */
/******/ (() => {
/******/      __webpack_require__.o = (obj, prop) => (Object.prototype.hasOwnProperty.call(ob
/******/ })();
/******/
/******/ /* 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/import chunk loading */
/******/ (() => {
/******/      __webpack_require__.b = new URL("./", import.meta.url);
/******/
/******/      // 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
/******/      };
/******/
/******/      var installChunk = (data) => {
/******/          var {ids, modules, runtime} = data;
/******/          // add "modules" to the modules object,
/******/          // then flag all "ids" as loaded and fire callback
/******/          var moduleId, chunkId, i = 0;
/******/          for(moduleId in modules) {
/******/              if(__webpack_require__.o(modules, moduleId)) {
/******/                  __webpack_require__.m[moduleId] = modules[moduleId];
/******/              }
/******/          }
/******/          if(runtime) runtime(__webpack_require__);
/******/          for(;i < ids.length; i++) {
/******/              chunkId = ids[i];
/******/              if(__webpack_require__.o(installedChunks, chunkId) && installedChunks[c
/******/                  installedChunks[chunkId][0]();
/******/              }
/******/              installedChunks[ids[i]] = 0;
/******/          }
/******/
/******/      }
/******/
/******/      __webpack_require__.f.j = (chunkId, promises) => {
/******/              // import() chunk loading for javascript
/******/              var installedChunkData = __webpack_require__.o(installedChunks, chunkId
/******/              if(installedChunkData !== 0) { // 0 means "already installed".
/******/
```

6

```
/******/                   // a Promise means "currently loading".
/******/                   if(installedChunkData) {
/******/                       promises.push(installedChunkData[1]);
/******/                   } else {
/******/                       if(true) { // all chunks have JS
/******/                           // setup Promise in chunk cache
/******/                           var promise = import("./" + __webpack_require__.u(chunkId))
/******/                               if(installedChunks[chunkId] !== 0) installedChunks[chun
/******/                               throw e;
/******/                           });
/******/                           var promise = Promise.race([promise, new Promise((resolve)
/******/                           promises.push(installedChunkData[1] = promise);
/******/                       } else installedChunks[chunkId] = 0;
/******/                   }
/******/               }
/******/       };
/******/
/******/       // no external install chunk
/******/
/******/       // no on chunks loaded
/******/ })();
/******/
/****************************************************************************/

var __webpack_exports__ = {};
/*!*********************!*\
  !*** ./example.js ***!
  \*********************/
/*! unknown exports (runtime-defined) */
/*! runtime requirements: __webpack_require__.p, __webpack_require__.b, __webpack_require__
document.body.innerHTML = `
    <pre id="history"></pre>
    <form>
    <input id="message" type="text">
    <button id="send">Send Message</button>
    </form>
    <p>Computing fibonacci without worker:</p>
    <input id="fib1" type="number">
    <pre id="output1"></pre>
    <p>Computing fibonacci with worker:</p>
    <input id="fib2" type="number">
    <pre id="output2"></pre>
`;

const history = document.getElementById("history");
const message = document.getElementById("message");
```

```javascript
const send = document.getElementById("send");
const fib1 = document.getElementById("fib1");
const output1 = document.getElementById("output1");
const fib2 = document.getElementById("fib2");
const output2 = document.getElementById("output2");

/// CHAT with shared worker ///

const chatWorker = new SharedWorker(
    new URL(/* worker import */ __webpack_require__.p + __webpack_require__.u(348), __webpa
    {
        name: "chat",
        type: "module"
    }
);

let historyTimeout;
const scheduleUpdateHistory = () => {
    clearTimeout(historyTimeout);
    historyTimeout = setTimeout(() => {
        chatWorker.port.postMessage({ type: "history" });
    }, 1000);
};
scheduleUpdateHistory();

const from = `User ${Math.floor(Math.random() * 10000)}`;

send.addEventListener("click", e => {
    chatWorker.port.postMessage({
        type: "message",
        content: message.value,
        from
    });
    message.value = "";
    message.focus();
    e.preventDefault();
});

chatWorker.port.onmessage = event => {
    const msg = event.data;
    switch (msg.type) {
        case "history":
            history.innerText = msg.history.join("\n");
            scheduleUpdateHistory();
            break;
    }
```

```javascript
};

/// FIBONACCI without worker ///

fib1.addEventListener("change", async () => {
    try {
        const value = parseInt(fib1.value, 10);
        const { fibonacci } = await __webpack_require__.e(/*! import() */ 129).then(__webpa
        const result = fibonacci(value);
        output1.innerText = `fib(${value}) = ${result}`;
    } catch (e) {
        output1.innerText = e.message;
    }
});

/// FIBONACCI with worker ///

const fibWorker = new Worker(new URL(/* worker import */ __webpack_require__.p + __webpack_
    name: "fibonacci",
    type: "module"
    /* webpackEntryOptions: { filename: "workers/[name].js" } */
});

fib2.addEventListener("change", () => {
    try {
        const value = parseInt(fib2.value, 10);
        fibWorker.postMessage(`${value}`);
    } catch (e) {
        output2.innerText = e.message;
    }
});

fibWorker.onmessage = event => {
    output2.innerText = event.data;
};
```

# dist/chat.js

```javascript
/******/ var __webpack_modules__ = ({});
/* 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(exp
/******/                 Object.defineProperty(exports, key, { enumerable: true, get: defini
/******/             }
/******/         }
/******/     };
/******/ })();
/******/
/******/ /* 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, key
/******/             __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 + ".js";
/******/    };
/******/ })();
/******/
/******/ /* webpack/runtime/hasOwnProperty shorthand */
/******/ (() => {
/******/    __webpack_require__.o = (obj, prop) => (Object.prototype.hasOwnProperty.call(ob
/******/ })();
/******/
/******/ /* 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/import 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 = {
/******/        348: 0
/******/    };
/******/
/******/    var installChunk = (data) => {
/******/        var {ids, modules, runtime} = data;
/******/        // add "modules" to the modules object,
/******/        // then flag all "ids" as loaded and fire callback
/******/        var moduleId, chunkId, i = 0;
/******/        for(moduleId in modules) {
```

11

```
/******/                    if(__webpack_require__.o(modules, moduleId)) {
/******/                        __webpack_require__.m[moduleId] = modules[moduleId];
/******/                    }
/******/                }
/******/                if(runtime) runtime(__webpack_require__);
/******/                for(;i < ids.length; i++) {
/******/                    chunkId = ids[i];
/******/                    if(__webpack_require__.o(installedChunks, chunkId) && installedChunks[c
/******/                        installedChunks[chunkId][0]();
/******/                    }
/******/                    installedChunks[ids[i]] = 0;
/******/                }
/******/
/******/        }
/******/
/******/        __webpack_require__.f.j = (chunkId, promises) => {
/******/                // import() chunk loading for javascript
/******/                var installedChunkData = __webpack_require__.o(installedChunks, chunkId
/******/                if(installedChunkData !== 0) { // 0 means "already installed".
/******/
/******/                    // a Promise means "currently loading".
/******/                    if(installedChunkData) {
/******/                        promises.push(installedChunkData[1]);
/******/                    } else {
/******/                        if(true) { // all chunks have JS
/******/                            // setup Promise in chunk cache
/******/                            var promise = import("./" + __webpack_require__.u(chunkId))
/******/                                if(installedChunks[chunkId] !== 0) installedChunks[chun
/******/                                throw e;
/******/                            });
/******/                            var promise = Promise.race([promise, new Promise((resolve)
/******/                            promises.push(installedChunkData[1] = promise);
/******/                        } else installedChunks[chunkId] = 0;
/******/                    }
/******/                }
/******/        };
/******/
/******/        // no external install chunk
/******/
/******/        // no on chunks loaded
/******/ })();
/******/
/******************************************************************/

var __webpack_exports__ = {};
/*!************************!*\
```

```
   !*** ./chat-worker.js ***!
  \************************/
/*! unknown exports (runtime-defined) */
/*! runtime requirements: __webpack_require__.e, __webpack_require__, __webpack_require__.*
onconnect = function (e) {
    for (const port of e.ports) {
        port.onmessage = async event => {
            const msg = event.data;
            switch (msg.type) {
                case "message":
                    const { add } = await __webpack_require__.e(/*! import() */ 192).then(_
                    add(msg.content, msg.from);
                // fallthrough
                case "history":
                    const { history } = await __webpack_require__.e(/*! import() */ 192).th
                    port.postMessage({
                        type: "history",
                        history
                    });
                    break;
            }
        };
    }
};

var e,o,t={},r={};function n(e){var o=r[e];if(void 0!==o)return o.exports;var s=r[e]={export
```

## dist/workers/fibonacci.js

```
/******/ var __webpack_modules__ = ({});

/* 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(exp
/******/                  Object.defineProperty(exports, key, { enumerable: true, get: defini
/******/              }
/******/          }
/******/      };
/******/ })();
/******/
/******/ /* 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, key
/******/              __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 + ".js";
/******/        };
/******/ })();
/******/
/******/ /* webpack/runtime/hasOwnProperty shorthand */
/******/ (() => {
/******/        __webpack_require__.o = (obj, prop) => (Object.prototype.hasOwnProperty.call(ob
/******/ })();
/******/
/******/ /* 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/import 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 = {
/******/            631: 0
/******/        };
/******/
/******/        var installChunk = (data) => {
/******/            var {ids, modules, runtime} = data;
/******/            // add "modules" to the modules object,
/******/            // then flag all "ids" as loaded and fire callback
/******/            var moduleId, chunkId, i = 0;
/******/            for(moduleId in modules) {
/******/                if(__webpack_require__.o(modules, moduleId)) {
/******/                    __webpack_require__.m[moduleId] = modules[moduleId];
/******/                }
/******/            }
/******/            if(runtime) runtime(__webpack_require__);
/******/            for(;i < ids.length; i++) {
/******/                chunkId = ids[i];
/******/                if(__webpack_require__.o(installedChunks, chunkId) && installedChunks[c
/******/                    installedChunks[chunkId][0]();
```

15

```
/******/                }
/******/                installedChunks[ids[i]] = 0;
/******/            }
/******/
/******/        }
/******/
/******/        __webpack_require__.f.j = (chunkId, promises) => {
/******/                // import() chunk loading for javascript
/******/                var installedChunkData = __webpack_require__.o(installedChunks, chunkId
/******/                if(installedChunkData !== 0) { // 0 means "already installed".
/******/
/******/                    // a Promise means "currently loading".
/******/                    if(installedChunkData) {
/******/                        promises.push(installedChunkData[1]);
/******/                    } else {
/******/                        if(true) { // all chunks have JS
/******/                            // setup Promise in chunk cache
/******/                            var promise = import("../" + __webpack_require__.u(chunkId)
/******/                                if(installedChunks[chunkId] !== 0) installedChunks[chun
/******/                                throw e;
/******/                            });
/******/                            var promise = Promise.race([promise, new Promise((resolve)
/******/                            promises.push(installedChunkData[1] = promise);
/******/                        } else installedChunks[chunkId] = 0;
/******/                    }
/******/                }
/******/        };
/******/
/******/        // no external install chunk
/******/
/******/        // no on chunks loaded
/******/ })();
/******/
/************************************************************************/

var __webpack_exports__ = {};
/*!************************!*\
  !*** ./fib-worker.js ***!
  \************************/
/*! unknown exports (runtime-defined) */
/*! runtime requirements: __webpack_require__.e, __webpack_require__, __webpack_require__.*
onmessage = async event => {
    const { fibonacci } = await __webpack_require__.e(/*! import() */ 129).then(__webpack_r
    const value = JSON.parse(event.data);
    postMessage(`fib(${value}) = ${fibonacci(value)}`);
};
```

```
var e,o,r={},t={};function i(e){var o=t[e];if(void 0!==o)return o.exports;var a=t[e]={export
```

## dist/129.js

```
export const id = 129;
export const ids = [129];
export const modules = {

/***/ 2:
/*!********************!*\
  !*** ./fibonacci.js ***!
  \********************/
/*! namespace exports */
/*! export fibonacci [provided] [no usage info] [missing usage info prevents renaming] */
/*! other exports [not provided] [no usage info] */
/*! runtime requirements: __webpack_require__.r, __webpack_exports__, __webpack_require__.d
/***/ ((__unused_webpack_module, __webpack_exports__, __webpack_require__) => {

__webpack_require__.r(__webpack_exports__);
/* harmony export */ __webpack_require__.d(__webpack_exports__, {
/* harmony export */   "fibonacci": () => (/* binding */ fibonacci)
/* harmony export */ });
function fibonacci(n) {
    return n < 1 ? 0 : n <= 2 ? 1 : fibonacci(n - 1) + fibonacci(n - 2);
}


/***/ })

};
```

## Info

### Unoptimized

```
asset main.js 8.56 KiB [emitted] [javascript module] (name: main)
asset chat.js 6.34 KiB [emitted] [javascript module] (name: chat)
asset workers/fibonacci.js 5.99 KiB [emitted] [javascript module] (name: fibonacci)
asset 192.js 1.01 KiB [emitted] [javascript module]
asset 129.js 847 bytes [emitted] [javascript module]
chunk (runtime: 9a81d90cfd0dfd13d748, main) 129.js 103 bytes [rendered]
  > ./fibonacci ./example.js 70:30-51
  > ./fibonacci ./fib-worker.js 2:29-50
  ./fibonacci.js 103 bytes [built] [code generated]
    [exports: fibonacci]
```

```
      [used exports unknown]
      import() ./fibonacci ./example.js 70:30-51
      import() ./fibonacci ./fib-worker.js 2:29-50
chunk (runtime: main) main.js (main) 2.25 KiB (javascript) 3.07 KiB (runtime) [entry] [rende
  > ./example.js main
  runtime modules 3.07 KiB 7 modules
  ./example.js 2.25 KiB [built] [code generated]
      [used exports unknown]
      entry ./example.js main
chunk (runtime: 1fad8bf8de78b0a77bfd) 192.js 152 bytes [rendered]
  > ./chat-module ./chat-worker.js 11:31-54
  > ./chat-module ./chat-worker.js 7:27-50
  ./chat-module.js 152 bytes [built] [code generated]
      [exports: add, history]
      [used exports unknown]
      import() ./chat-module ./chat-worker.js 7:27-50
      import() ./chat-module ./chat-worker.js 11:31-54
chunk (runtime: 1fad8bf8de78b0a77bfd) chat.js (chat) 442 bytes (javascript) 2.86 KiB (runtim
  > ./example.js 25:19-31:1
  runtime modules 2.86 KiB 6 modules
  ./chat-worker.js 442 bytes [built] [code generated]
      [used exports unknown]
      new Worker() ./chat-worker.js ./example.js 25:19-31:1
chunk (runtime: 9a81d90cfd0dfd13d748) workers/fibonacci.js (fibonacci) 176 bytes (javascript
  > ./example.js 80:18-84:2
  runtime modules 2.87 KiB 6 modules
  ./fib-worker.js 176 bytes [built] [code generated]
      [used exports unknown]
      new Worker() ./fib-worker.js ./example.js 80:18-84:2
webpack 5.51.1 compiled successfully
```

## Production mode

```
asset main.js 2.5 KiB [emitted] [javascript module] [minimized] (name: main)
asset chat.js 1.19 KiB [emitted] [javascript module] [minimized] (name: chat)
asset workers/fibonacci.js 1.04 KiB [emitted] [javascript module] [minimized] (name: fibona
asset 192.js 187 bytes [emitted] [javascript module] [minimized]
asset 129.js 161 bytes [emitted] [javascript module] [minimized]
chunk (runtime: 9a81d90cfd0dfd13d748, main) 129.js 103 bytes [rendered]
  > ./fibonacci ./example.js 70:30-51
  > ./fibonacci ./fib-worker.js 2:29-50
  ./fibonacci.js 103 bytes [built] [code generated]
      [exports: fibonacci]
      import() ./fibonacci ./example.js 70:30-51
      import() ./fibonacci ./fib-worker.js 2:29-50
chunk (runtime: main) main.js (main) 2.25 KiB (javascript) 3.07 KiB (runtime) [entry] [rende
```

```
  > ./example.js main
  runtime modules 3.07 KiB 7 modules
  ./example.js 2.25 KiB [built] [code generated]
    [no exports used]
    entry ./example.js main
chunk (runtime: 1fad8bf8de78b0a77bfd) 192.js 152 bytes [rendered]
  > ./chat-module ./chat-worker.js 11:31-54
  > ./chat-module ./chat-worker.js 7:27-50
  ./chat-module.js 152 bytes [built] [code generated]
    [exports: add, history]
    import() ./chat-module ./chat-worker.js 7:27-50
    import() ./chat-module ./chat-worker.js 11:31-54
chunk (runtime: 1fad8bf8de78b0a77bfd) chat.js (chat) 442 bytes (javascript) 2.86 KiB (runtim
  > ./example.js 25:19-31:1
  runtime modules 2.86 KiB 6 modules
  ./chat-worker.js 442 bytes [built] [code generated]
    [no exports used]
    new Worker() ./chat-worker.js ./example.js 25:19-31:1
chunk (runtime: 9a81d90cfd0dfd13d748) workers/fibonacci.js (fibonacci) 176 bytes (javascript
  > ./example.js 80:18-84:2
  runtime modules 2.87 KiB 6 modules
  ./fib-worker.js 176 bytes [built] [code generated]
    [no exports used]
    new Worker() ./fib-worker.js ./example.js 80:18-84:2
webpack 5.51.1 compiled successfully
```