import { EventTarget } from "event-target-shim"

type Events = {

abort: any

}

type EventAttributes = {

onabort: any

}

/\*\*

\* The signal class.

\* @see https://dom.spec.whatwg.org/#abortsignal

\*/

declare class AbortSignal extends EventTarget<Events, EventAttributes> {

/\*\*

\* AbortSignal cannot be constructed directly.

\*/

constructor()

/\*\*

\* Returns `true` if this `AbortSignal`"s `AbortController` has signaled to abort, and `false` otherwise.

\*/

readonly aborted: boolean

}

/\*\*

\* The AbortController.

\* @see https://dom.spec.whatwg.org/#abortcontroller

\*/

declare class AbortController {

/\*\*

\* Initialize this controller.

\*/

constructor()

/\*\*

\* Returns the `AbortSignal` object associated with this object.

\*/

readonly signal: AbortSignal

/\*\*

\* Abort and signal to any observers that the associated activity is to be aborted.

\*/

abort(): void

}

export default AbortController

export { AbortController, AbortSignal }