import { Options as WebIDLConversionOptions } from "webidl-conversions";

import { URL } from "../index";

import { implementation as URLImpl } from "./URL-impl";

/\*\*

\* Checks whether `obj` is a `URL` object with an implementation

\* provided by this package.

\*/

export function is(obj: unknown): obj is URL;

/\*\*

\* Checks whether `obj` is a `URLImpl` WebIDL2JS implementation object

\* provided by this package.

\*/

export function isImpl(obj: unknown): obj is URLImpl;

/\*\*

\* Converts the `URL` wrapper into a `URLImpl` object.

\*

\* @throws {TypeError} If `obj` is not a `URL` wrapper instance provided by this package.

\*/

export function convert(obj: unknown, options?: WebIDLConversionOptions): URLImpl;

/\*\*

\* Creates a new `URL` instance.

\*

\* @throws {Error} If the `globalObject` doesn't have a WebIDL2JS constructor

\* registry or a `URL` constructor provided by this package

\* in the WebIDL2JS constructor registry.

\*/

export function create(

globalObject: object,

[url, base]: readonly [string, string?],

privateData?: {},

): URL;

/\*\*

\* Calls `create()` and returns the internal `URLImpl`.

\*

\* @throws {Error} If the `globalObject` doesn't have a WebIDL2JS constructor

\* registry or a `URL` constructor provided by this package

\* in the WebIDL2JS constructor registry.

\*/

export function createImpl(

globalObject: object,

[url, base]: readonly [string, string?],

privateData?: {},

): URLImpl;

/\*\*

\* Initializes the `URL` instance, called by `create()`.

\*

\* Useful when manually sub-classing a non-constructable wrapper object.

\*/

export function setup<T extends URL>(

obj: T,

globalObject: object,

[url, base]: readonly [string, string?],

privateData?: {},

): T;

/\*\*

\* Creates a new `URL` object without runing the constructor steps.

\*

\* Useful when implementing specifications that initialize objects

\* in different ways than their constructors do.

\*/

declare function \_new(globalObject: object): URLImpl;

export { \_new as new };

/\*\*

\* Installs the `URL` constructor onto the `globalObject`.

\*

\* @throws {Error} If the target `globalObject` doesn't have an `Error` constructor.

\*/

export function install(globalObject: object, globalNames: readonly string[]): void;