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

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

import { implementation as URLSearchParamsImpl } from "./URLSearchParams-impl";

/\*\*

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

\* provided by this package.

\*/

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

/\*\*

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

\* provided by this package.

\*/

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

/\*\*

\* Converts the `URLSearchParams` wrapper into a `URLSearchParamsImpl` object.

\*

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

\*/

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

/\*\*

\* Creates a new `URLSearchParams` instance.

\*

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

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

\* in the WebIDL2JS constructor registry.

\*/

export function create(

globalObject: object,

constructorArgs?: readonly [

init:

| ReadonlyArray<[name: string, value: string]>

| { readonly [name: string]: string }

| string,

],

privateData?: { doNotStripQMark?: boolean | undefined },

): URLSearchParams;

/\*\*

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

\*

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

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

\* in the WebIDL2JS constructor registry.

\*/

export function createImpl(

globalObject: object,

constructorArgs?: readonly [

init:

| ReadonlyArray<[name: string, value: string]>

| { readonly [name: string]: string }

| string,

],

privateData?: { doNotStripQMark?: boolean | undefined },

): URLSearchParamsImpl;

/\*\*

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

\*

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

\*/

export function setup<T extends URLSearchParams>(

obj: T,

globalObject: object,

constructorArgs?: readonly [

init:

| ReadonlyArray<[name: string, value: string]>

| { readonly [name: string]: string }

| string,

],

privateData?: { doNotStripQMark?: boolean | undefined },

): T;

/\*\*

\* Creates a new `URLSearchParams` 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): URLSearchParamsImpl;

export { \_new as new };

/\*\*

\* Installs the `URLSearchParams` 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;