// TypeScript Version: 3.2

/// <reference types="node" lib="esnext" />

import \* as fs from 'fs';

import { Readable } from 'stream';

declare namespace readdir {

interface EntryInfo {

path: string;

fullPath: string;

basename: string;

stats?: fs.Stats;

dirent?: fs.Dirent;

}

interface ReaddirpOptions {

root?: string;

fileFilter?: string | string[] | ((entry: EntryInfo) => boolean);

directoryFilter?: string | string[] | ((entry: EntryInfo) => boolean);

type?: 'files' | 'directories' | 'files\_directories' | 'all';

lstat?: boolean;

depth?: number;

alwaysStat?: boolean;

}

interface ReaddirpStream extends Readable, AsyncIterable<EntryInfo> {

read(): EntryInfo;

[Symbol.asyncIterator](): AsyncIterableIterator<EntryInfo>;

}

function promise(

root: string,

options?: ReaddirpOptions

): Promise<EntryInfo[]>;

}

declare function readdir(

root: string,

options?: readdir.ReaddirpOptions

): readdir.ReaddirpStream;

export = readdir;