# destroy

[NPM version](https://npmjs.org/package/destroy) [Build Status](https://github.com/stream-utils/destroy/actions/workflows/ci.yml) [Test coverage](https://coveralls.io/r/stream-utils/destroy?branch=master) [License](http://license.md) [Downloads](https://npmjs.org/package/destroy)

Destroy a stream.

This module is meant to ensure a stream gets destroyed, handling different APIs and Node.js bugs.

## API

var destroy = require('destroy')

### destroy(stream [, suppress])

Destroy the given stream, and optionally suppress any future error events.

In most cases, this is identical to a simple stream.destroy() call. The rules are as follows for a given stream:

1. If the stream is an instance of ReadStream, then call stream.destroy() and add a listener to the open event to call stream.close() if it is fired. This is for a Node.js bug that will leak a file descriptor if .destroy() is called before open.
2. If the stream is an instance of a zlib stream, then call stream.destroy() and close the underlying zlib handle if open, otherwise call stream.close(). This is for consistency across Node.js versions and a Node.js bug that will leak a native zlib handle.
3. If the stream is not an instance of Stream, then nothing happens.
4. If the stream has a .destroy() method, then call it.

The function returns the stream passed in as the argument.

## Example

var destroy = require('destroy')

var fs = require('fs')

var stream = fs.createReadStream('package.json')

// ... and later

destroy(stream)