# EE First

[NPM version](https://npmjs.org/package/ee-first) [Build status](https://travis-ci.org/jonathanong/ee-first) [Test coverage](https://coveralls.io/r/jonathanong/ee-first?branch=master) [License](http://license.md) [Downloads](https://npmjs.org/package/ee-first) [Gittip](https://www.gittip.com/jonathanong/)

Get the first event in a set of event emitters and event pairs, then clean up after itself.

## Install

$ npm install ee-first

## API

var first = require('ee-first')

### first(arr, listener)

Invoke listener on the first event from the list specified in arr. arr is an array of arrays, with each array in the format [ee, ...event]. listener will be called only once, the first time any of the given events are emitted. If error is one of the listened events, then if that fires first, the listener will be given the err argument.

The listener is invoked as listener(err, ee, event, args), where err is the first argument emitted from an error event, if applicable; ee is the event emitter that fired; event is the string event name that fired; and args is an array of the arguments that were emitted on the event.

var ee1 = new EventEmitter()

var ee2 = new EventEmitter()

first([

[ee1, 'close', 'end', 'error'],

[ee2, 'error']

], function (err, ee, event, args) {

// listener invoked

})

#### .cancel()

The group of listeners can be cancelled before being invoked and have all the event listeners removed from the underlying event emitters.

var thunk = first([

[ee1, 'close', 'end', 'error'],

[ee2, 'error']

], function (err, ee, event, args) {

// listener invoked

})

// cancel and clean up

thunk.cancel()