

Iterator

<i>Pot.Iter</i> new Pot.Iter ()
<i>Pot.Iter</i> Pot.tolter (x)
<i>Array</i> Pot.range ((begin = 0,] end [, step = 1])
<i>Number String</i> Pot.indexOf (object, subject [, from = 0])
<i>Number String</i> Pot.lastIndexOf (object, subject [, from])
Synchronous Iterator ↳ Core.Iterator
Asynchronous Iterator ↳ Deferred.Iterator
Asynchronous Iterator on Chain ↳ Deferred.prototype.Iterator

* Pot.debug (msg)
<i>Object</i> Pot.update (target, obj1 [, obj2 [, ...]])
<i>String</i> Pot.stringify (x [, ignoreBoolean = false])
<i>Array</i> Pot.arrayize (x [, index])
<i>Number</i> Pot.now ()

<i>Boolean</i> Pot.some (object, callback [, context])
<i>Array</i> Pot.items (object [, callback [, context]])
<i>Array</i> Pot.zip (object [, callback [, context]])

Deferred

prototype **Basic**

<i>Pot.Deferred</i> new Pot.Deferred ([options])
<i>Pot.Deferred</i> Pot.Deferred.prototype.then (callback [, errback])
<i>Pot.Deferred</i> Pot.Deferred.prototype.rescue (errback)
<i>Pot.Deferred</i> Pot.Deferred.prototype.ensure (callback)
<i>Pot.Deferred</i> Pot.Deferred.prototype.begin ([...args])
<i>Pot.Deferred</i> Pot.Deferred.prototype.raise ([...args])
<i>Pot.Deferred</i> Pot.Deferred.prototype.end ()
<i>Pot.Deferred</i> Pot.Deferred.prototype.cancel ()
<i>Pot.Deferred Array</i> Pot.Deferred.prototype.canceller (func)
<i>Pot.Deferred Number *</i> Pot.Deferred.prototype.speed (speeds)
<i>Pot.Deferred Boolean</i> Pot.Deferred.prototype.async (isAsync)
<i>Pot.Deferred</i> Pot.Deferred.prototype.wait (seconds [, value])
<i>Pot.Deferred</i> Pot.Deferred.prototype.till (cond)
<i>Pot.Deferred *</i> Pot.Deferred.prototype.args ([...args])
<i>Pot.Deferred *</i> Pot.Deferred.prototype.data ({[key [, value]])

prototype **Iterator**

Iterator
<i>Object Array</i> Pot.forEach (object, callback [, context])
<i>Void</i> Pot.forEver (callback [, context])
<i>Void</i> Pot.repeat (max, callback [, context])
* Pot.iterate (iter [, callback [, context]])
<i>Object Array</i> Pot.filter (object, callback [, context])
<i>Object Array</i> Pot.map (object, callback [, context])
* Pot.reduce (object, callback [, initial [, context]])
<i>Boolean</i> Pot.every (object, callback [, context])

<i>Pot.Deferred</i> Pot.Deferred.prototype.reduce (callback [, initial [, context]])
<i>Pot.Deferred</i> Pot.Deferred.prototype.every (callback [, context])
<i>Pot.Deferred</i> Pot.Deferred.prototype.some (callback [, context])
<i>Pot.Deferred</i> Pot.Deferred.prototype.items ([callback [, context]])
<i>Pot.Deferred</i> Pot.Deferred.prototype.zip ([callback [, context]])

Static

<i>Pot.Deferred</i> Pot.Deferred.begin (callback)
<i>Pot.Deferred</i> Pot.Deferred.flush (callback)
<i>Pot.Deferred</i> Pot.Deferred.callLater (seconds, callback)
<i>Pot.Deferred</i> Pot.Deferred.callLazy (callback)
<i>Pot.Deferred</i> Pot.Deferred.wait (seconds [, value])
<i>Pot.Deferred</i> Pot.Deferred.till (cond)
<i>Pot.Deferred</i> Pot.Deferred.succeed ([...args])
<i>Pot.Deferred</i> Pot.Deferred.fail ([...args])
<i>Pot.Deferred</i> Pot.Deferred.maybeDeferred (x)
<i>Pot.Deferred</i> Pot.Deferred.parallel (deferredList [, deferredList2 [, ...]])
<i>Pot.Deferred</i> Pot.Deferred.chain (args [, args2 [, ...]])
<i>Function</i> Pot.Deferred.deferize (object [, func])
<i>Function</i> Pot.Deferred.deferred (object [, func])
<i>Boolean</i> Pot.Deferred.isFired (deferred)
* Pot.Deferred.lastResult (deferred [, value])
* Pot.Deferred.lastError (deferred [, value])
<i>Number</i> Pot.Deferred.register (name [, func])
<i>Number</i> Pot.Deferred.unregister (name)

Iterator

<i>Pot.Deferred</i> Pot.Deferred.forEach (object, callback [, context])
<i>Pot.Deferred</i> Pot.Deferred.forEver (callback [, context])

<i>Pot.Deferred</i> Pot.Deferred.repeat (max, callback [, context])
<i>Pot.Deferred</i> Pot.Deferred.iterate (iter [, callback [, context]])
<i>Pot.Deferred</i> Pot.Deferred.filter (object, callback [, context])
<i>Pot.Deferred</i> Pot.Deferred.map (object, callback [, context])
<i>Pot.Deferred</i> Pot.Deferred.reduce (object, callback [, initial [, context]])
<i>Pot.Deferred</i> Pot.Deferred.every (object, callback [, context])
<i>Pot.Deferred</i> Pot.Deferred.some (object, callback [, context])
<i>Pot.Deferred</i> Pot.Deferred.items (object [, callback [, context]])
<i>Pot.Deferred</i> Pot.Deferred.zip (object [, callback [, context]])

Workeroid

<i>Pot.Workeroid</i> new Pot.Workeroid (script)
<i>Pot.Workeroid</i> Pot.Workeroid.prototype.postMessage (data)
<i>Pot.Workeroid</i> Pot.Workeroid.prototype.terminate ()
Pot.Workeroid.prototype.onmessage Function
Pot.Workeroid.prototype.onerror Function

Signal

<i>Object Array</i> Pot.attach (object, signalName, callback [, useCapture])
<i>Object Array</i> Pot.attachAfter (object, signalName, callback)
<i>Object Array</i> Pot.attachBefore (object, signalName, callback)
<i>Object Array</i> Pot.attachPropAfter (object, propName, callback)
<i>Object Array</i> Pot.attachPropBefore (object, propName, callback)
<i>Boolean</i> Pot.detach (object [, signalName [, callback [, useCapture]])
<i>Void</i> Pot.detachAll ([object [, signals]])
<i>Pot.Deferred</i> Pot.signal (object, signalName [, ...args])
<i>Boolean</i> Pot.cancelEvent (event)

Net

<i>Pot.Deferred</i> Pot.request (url [, options])
--

<i>Pot.Deferred</i> Pot.getJSON (url [, options])
<i>Pot.Deferred</i> Pot.jsonp (url [, options])
<i>Pot.Deferred</i> Pot.loadScript (url [, options])

URI

<i>String</i> Pot.urlEncode (string)
<i>String</i> Pot.urlDecode (string)

DropFile

<i>Pot.DropFile</i> new Pot.DropFile (target [, options])
<i>Void</i> Pot.DropFile.prototype.clearDropEvents ()
<i>Pot.Deferred</i> Pot.DropFile.prototype.upload (url [, options])

Serializer

<i>String</i> Pot.serializeToJSON (value)
* Pot.parseFromJSON (text)
<i>String</i> Pot.serializeToQueryString (params)
<i>Array Object</i> Pot.parseFromQueryString (queryString [, toObject = false])

XPCOM

<i>Void</i> Pot.throughout (cond)
* Pot.evalInSandbox (code, url)
<i>Window Object *</i> Pot.getChromeWindow (url)
<i>Window Object *</i> Pot.getMostRecentWindow ()

Plugin

<i>Boolean</i> Pot.addPlugin (name [, func [, force = false]])
<i>Boolean</i> Pot.removePlugin (name)
<i>Boolean</i> Pot.hasPlugin (name)
<i>Array</i> Pot.listPlugin ()

jQuery Plugin

<i>Void</i> Pot.deferizejQueryAjax ()
