[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

PROCESSING **LIST** [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx)

A list is an EE variable object that represents an ordered sequence of objects any type. Lists can be processed by using operations of the types listed below, which vary according to the nature of that processing. Each operation name is linked to a separate page describing that operation.

**CREATING** LISTS [ee.List](#List)  [ee.List.repeat](#repeat) [ee.List.sequence](#sequence)

**EDITING** LISTS

BY **RESETTING** VALUES [list.set](#set)  [list.replace](#replace) [list.replaceAll](#replaceAll)

BY **ADDING** ELEMENTS [list.add](#add) [list.insert](#insert)

BY **COMBINING** THEM [list.cat](#cat) [list.splice](#splice) [list.zip](#zip)

BY **REORGANIZING** THEM [list.reverse](#reverse) [list.rotate](#rotate) [list.sort](#sort) [list.swap](#swap) [list.flatten](#flatten)

**QUERYING** LISTS

FOR **VALUES** [list.get](#get)

FOR **SUBLISTS** [list.slice](#slice) [list.remove](#remove) [list.removeAll](#removeAll)

**COMPARING** LISTS

TO **LISTS** [ee.Algorithms.IsEqual(list)](#IsEqual) [list.equals](#equals) [list.containsAll](#containsAll)

[list.indexOfSublist](#indexOfSublist) [list.lastIndexOfSubList](#lastIndex)

TO **LIST ELEMENTS**  [list.indexOf](#indexOf) [list.frequency](#frequency)  [list.contains](#contains)

**PARALLEL PROCESSING** LISTS [list.map](#map)

**TALLYING** LISTS [list.iterate](#iterate)

**MEASURING** LISTS [list.length](#length_size) [list.size](#length_size)

**DOCUMENTING** LISTS [list.getInfo](#Describe_getInfo) [ee.Algorithms.Describe(list)](#Describe_getInfo)

[list.toString](#toString_serialize) [list.serialize](#toString_serialize)

**PRESENTING** LISTS

IN **PRINT** [print(list)](#print_console) [console.log(list)](#print_console)

[alert(list)](#alert_confirm) [confirm(list)](#alert_confirm)

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**CREATING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx)

ee.List creates a new (Earth Engine) list a specified (JavaScript) list.

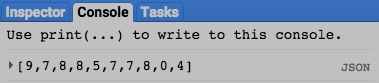
newList = ee.List ( oldList )

The specified list

The new list

var TheLIST = ee.List( [9,7,8,8,5,7,7,8,0,4] );

print( TheLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**CREATING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx)

ee.List.repeat creates a new list containing a specified number of elements that are all set to the same specified value.

newArray = ee.List.repeat ( elementValue, numberOfElements )

The specified number of elements, given as an integer. Default: 0

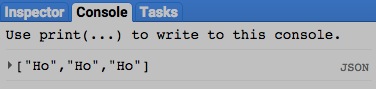
The specified value, given as an object

The specified array

The new list

var TheLIST = ee.List.repeat( 'Ho', 3 );

print( TheLIST );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**CREATING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx)

ee.List.sequence creates a new list containing a sequence of numbers ranging from a specified starting value to a specified stopping value.

These numbers can be generated in intervals of a specified range or in a specified number of equal-range intervals.

newList = ee.List.sequence ( startingValue*, stoppingValue, intervalSize, numberOfListedNumbers*)

A specified number of list items,

which must be specified if

***stoppingValue*** is not.

The new list

The specified starting value

A specified stopping value,

which must be specified if

***numberOfIntervals*** is not.

A specified

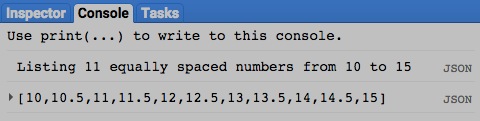
interval range.

Default: 1

var TheLIST = ee.List.sequence( 10, 15, null, 11 );

print( 'Listing 11 equally spaced numbers from 10 to 15' );

print( TheLIST );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **RESETTING** VALUES

list.set creates a new list by replacing the element at a specified ordinal position in a specified list with a specified object.

newList = oldList.set ( positionNumber, oldObject )

The specified object

The specified

list

The specified position, given as an integer referring to a list index starting with 0

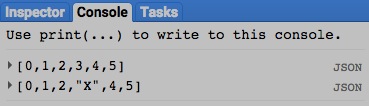
The new

list

var OldLIST = ee.List( [0,1,2,3,4,5] );

var NewLIST = OldLIST.set( 3,'X' );

print( OldLIST, NewLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **RESETTING** VALUES

list.replace creates a new list by replacing the first occurrence of a specified value in a specified list with another specified value.

newList = oldList.replace ( oldValue, newValue)

The specified value to be substituted

The specified value to be replaced

The specified

list

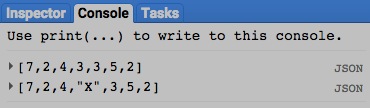
The new

list

var OldLIST = ee.List( [7,2,4,3,3,5,2] );

var NewLIST = OldLIST.replace( 3,'X' );

print( OldLIST, NewLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **RESETTING** VALUES

list.replaceAll creates a new list by replacing all occurrences of a specified value in a specified list with another specified value.

newList = oldList.replace ( oldValue, newValue)

The specified value to be substituted

The specified value to be replaced

The specified

list

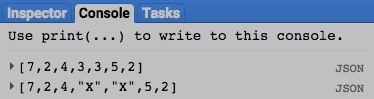
The new

list

var OldLIST = ee.List( [7,2,4,3,3,5,2] );

var NewLIST = OldLIST.replaceAll( 3,'X' );

print( OldLIST, NewLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **ADDING** ELEMENTS

list.add creates a new list by appending a specified object to the end of a specified list.

newList = oldList.add ( oldObject )

The specified list

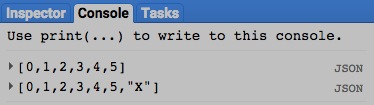
The new list

The specified object

var OldLIST = ee.List( [0,1,2,3,4,5] );

var NewLIST = OldLIST.add( 'X' );

print( OldLIST, NewLIST );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **ADDING** ELEMENTS

list.insert creates a new list by inserting a specified object at a specified position in a specified list.

newList = oldList.insert ( positionNumber, oldObject )

The specified object

The specified position, given as an integer referring to a list index starting with 0

The specified

list

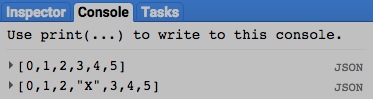
The new

list

var OldLIST = ee.List( [0,1,2,3,4,5] );

var NewLIST = OldLIST.insert( 3,'X' );

print( OldLIST, NewLIST );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **COMBINING** THEM

list.cat creates a new array by appending the contents of one specified list to the initial axis of another specified list.

newList = 1stList.cat ( 2ndList)

The specified lists to be combined such that **2ndList** is appended to **1stList**

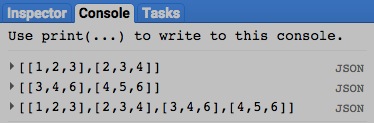
The new list

var FirstLIST = ee.List( [ [1,2,3], [2,3,4] ] );

var SecondLIST = ee.List( [ [3,4,6], [4,5,6] ] );

var NewLIST = FirstLIST.cat( SecondLIST );

print( FirstLIST, SecondLIST, NewLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **COMBINING** THEM

list.splice creates a new list by removing a specified number of elements from specified positions

in one specified list and optionally replacing them with another specified list.

newList = 1stList.swap ( startingPosition, numberOfElements, 2ndList )

The new list

The specified position of the first element to be

removed, given as an integer referring to a list index beginning with 0. Negative indices are counted backward from the end of the list.

The specified list from which elements are to be removed

The specified list to be inserted

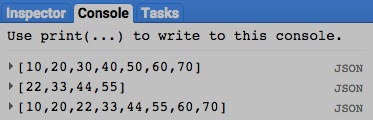
The specified number of elements to be removed

var FirstLIST = ee.List( [10,20,30,40,50,60,70] );

var SecondLIST = ee.List( [22,33,44,55] );

var NewLIST = FirstLIST.splice( 2, 3, SecondLIST );

print( FirstLIST, SecondLIST, NewLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **COMBINING** THEM

list.zip creates a new list by replacing each element in a specified list with a sub-list containing that element and the corresponding element of another specified list.

If one of the specified lists is longer than the other, its unpaired elements are ignored.

newList = 1stList.zip ( 2ndList)

The first specified list

The first specified list

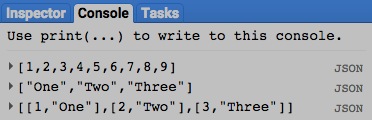
The new list

var FirstLIST = ee.List( [1,2,3,4,5,6,7,8,9] );

var SecondLIST = ee.List( ['One', 'Two', 'Three'] );

var NewLIST = FirstLIST.zip( SecondLIST );

print( FirstLIST, SecondLIST, NewLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **REORGANIZING** THEM

list.reverse creates a new list by reversing the order of elements in a specified list.

newList = oldList.reverse ()

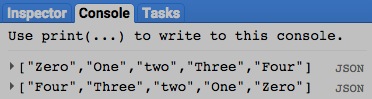
The specified list

The new list

var OldLIST = ee.List( [ 'Zero', 'One', 'two', 'Three', 'Four' ] );

var NewLIST = OldLIST.reverse( );

print( OldLIST, NewLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **REORGANIZING** THEM

list.rotate creates a new list that replicates the elements of a specified list by starting at a specified position,

continuing to the list’s end, and restarting from its beginning.

newList = oldList.rotate ( startingPosition )

The specified position, given as a negative integer referring to list indices

starting at 0. Positive integers are counted backwards from the list’s end.

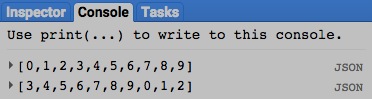
The specified list

The new list

var OldLIST = ee.List( [ 0,1,2,3,4,5,6,7,8,9 ] );

var NewLIST = OldLIST.rotate( -3 );

print( OldLIST, NewLIST );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **REORGANIZING** THEM

list.sort creates a new list created by sorting the elements of a specified list in ascending order.

newList = oldList.sort ()

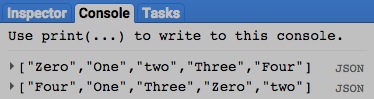
The specified list, whose elements must all be of the same type

The new list

var OldLIST = ee.List( [ 'Zero', 'One', 'two', 'Three', 'Four' ] );

var NewLIST = OldLIST.sort( );

print( OldLIST, NewLIST );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **REORGANIZING** THEM

list.swap creates a new list by swapping the elements at two specified positions in a specified list.

newList = oldList.swap ( 1stPosition, 2ndPosition )

The two specified positions, each given as an integer referring to a list index beginning with 0. Negative indices are counted backward from the end of the list.

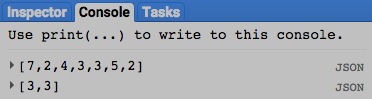
The specified list

The new list

var OldLIST = ee.List( [7,2,4,3,3,5,2] );

var NewLIST = OldLIST.slice( 3,5 );

print( OldLIST, NewLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**EDITING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) BY **REORGANIZING** THEM

list.flatten creates a new one-dimensional list that replicates the elements of a specified multi-dimensional list in the order of its axes.

newList = oldList.flatten ( )

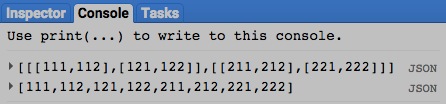
The specified list

The new list

var OldLIST = ee.List( [ [ [111,112], [121,122] ], [ [211,212], [221,222] ] ] );

var NewLIST = OldLIST.flatten( );

print( OldLIST, NewLIST );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**QUERYING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) FOR **VALUES**

list.get creates a new list element by copying the element at a specified ordinal position (starting with 0) in a specified list.

newElement = oldList.get ( position )

The specified ordinal position (i.e. index), given as an integer with negative indices counted backwards from the list’s end

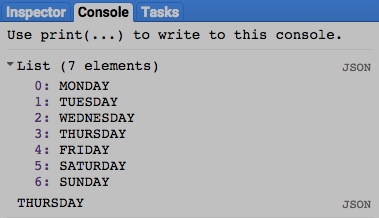
The specified list

The new element

var OldLIST = ee.List( ['MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY', 'FRIDAY', 'SATURDAY', 'SUNDAY'] );

var NewELEMENT = OldLIST.get( 3 );

print( OldLIST, NewELEMENT );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**QUERYING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) FOR **SUBLISTS**

list.slice creates a new list by retaining only that sub-list of a specified list that that begins and ends at specified ordinal positions.

newList = oldList.slice ( startingPosition, stoppingPosition )

The new list

The specified list

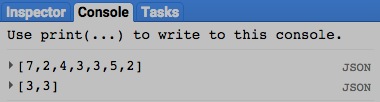
The specified positions at which 1) the substring is to start and 2) just after which it is to stop,

each given as an integer starting at 0 with negative indices counted backwards from the list’s end.

var OldLIST = ee.List( [7,2,4,3,3,5,2] );

var NewLIST = OldLIST.slice( 3,5 );

print( OldLIST, NewLIST );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**QUERYING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) FOR **SUBLISTS**

list.remove creates a new list by removing the first occurrence of a specified element, if any, from a specified list.

newList = oldList.remove ( oldObject )

The specified

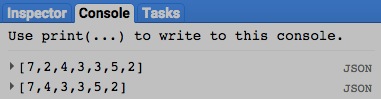
The specified list

The new list

var OldLIST = ee.List( [7,2,4,3,3,5,2] );

var NewLIST = OldLIST.remove( 2 );

print( OldLIST, NewLIST );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**QUERYING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) FOR **SUBLISTS**

list.removeAll creates a new list by removing, from the first of two specified lists, whatever elements also occur within the second of those lists.

newList = 1stList.removeAll ( 2ndList )

The second specified list, given as either an Earth Engine list or a JavaScript list

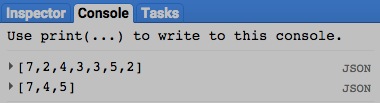
The first specified list

The new list

var OldLIST = ee.List( [7,2,4,3,3,5,2] );

var NewLIST = OldLIST.removeAll( [2,3] );

print( OldLIST, NewLIST );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**COMPARING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) TO **LISTS**

ee.Algorithms.IsEqual creates a new Boolean set to True (only) if the first of two specified lists is identical to the second in both structure and content.

newBoolean = ee.Algorithms.IsEqual ( 1stList, 2ndList )

The first specified list

The second specified list

The new Boolean

var FirstLIST = ee.List( [0,1,2,3,2,1] );

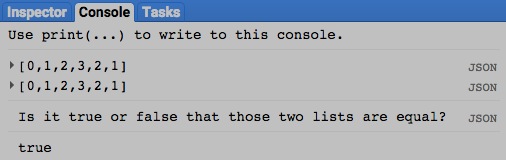
var SecondLIST = ee.List( [0,1,2,3,2,1.0] );

var TrueOrFalse = ee.Algorithms.IsEqual( FirstLIST, SecondLIST );

print( FirstLIST, SecondLIST );

print( 'Is it true or false that those two lists are equal?' );

print( TrueOrFalse );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**COMPARING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) TO **LISTS**

list.equals creates a new Boolean set to True (only) if one specified list contains all (and only) the elements of a second specified list, presented in the same order.

newBoolean = 1stList.equals ( 2ndList )

The second specified list

The first specified list

The new integer

var FirstLIST = ee.List( [0,1,2,3,2,1] );

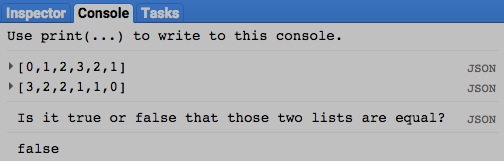
var SecondLIST = ee.List( [3,2,2,1,1,0] );

var TrueOrFalse = FirstLIST.equals( SecondLIST );

print( FirstLIST, SecondLIST );

print( 'Is it true or false that those two lists are equal?' );

print( TrueOrFalse );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**COMPARING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) TO **LISTS**

list.containsAll creates a new Boolean set to true (only) if one specified list contains all (and only) the elements of a second specified list, presented in any order.

newBoolean = 1stList.equals ( 2ndList )

The second specified list

The first specified list

The new integer

var FirstLIST = ee.List( [0,1,2,3,2,1] );

var SecondLIST = ee.List( [3,2,2,1,1,0] );

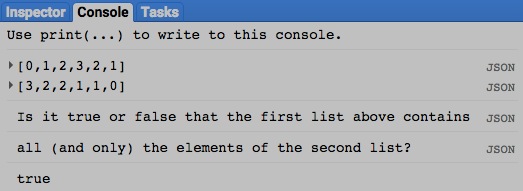
var TrueOrFalse = FirstLIST.containsAll( SecondLIST );

print( FirstLIST, SecondLIST );

print( 'Is it true or false that the first list above contains' );

print( 'all (and only) the elements of the second list?' );

print( TrueOrFalse );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**COMPARING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) TO **LISTS**

list.indexOfSublist creates a new integer indicating the ordinal position of the first element of a specified sublist within a specified list (set to -1 if none).

newInteger = 1stList.indexOf ( 2ndList )

The specified sublist, given as an EE or JavaScript list

The new integer

The specified list

var FirstLIST = ee.List( [9,7,8,8,5,7,7,8,0,4] );

var SecondLIST = ee.List( [7,8] );

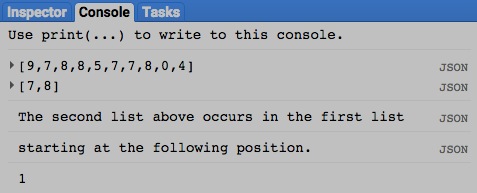
var TheNUMBER = FirstLIST.indexOfSublist( SecondLIST );

print( FirstLIST, SecondLIST );

print( 'The second list above occurs in the first list' );

print( 'starting at the following position.' );

print( TheNUMBER );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**COMPARING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) TO **LISTS**

list.lastIndexOfSubList\* creates a new integer indicating the ordinal position of the first element in the final occurrence

of a specified sublist within a specified list (set to -1 if none).

newInteger = 1stOldList.indexOf ( 2ndOldList ) \* Note the uppercase **L** in **lastIndexOfSubList**

The specified sublist, given as an EE or JavaScript list

The new integer

The specified list

var FirstLIST = ee.List( [2,0,7,9,9,7,3,7,6,8] );

var SecondLIST = ee.List( [9,9,7] );

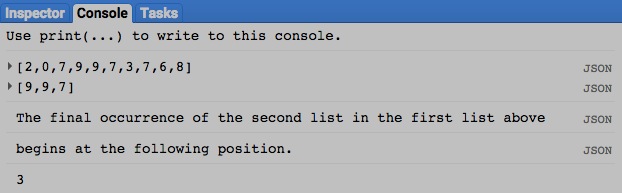
var TheNUMBER = FirstLIST.lastIndexOfSubList( SecondLIST );

print( FirstLIST, SecondLIST );

print( 'The final occurrence of the second list in the first list above' );

print( 'begins at the following position.' );

print( TheNUMBER );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**COMPARING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) TO **LISTS**

list.contains creates a new Boolean set to true (only) if a specified list contains a specified element.

newBoolean = oldList.contains ( element )

The specified element

The new integer

The specified list

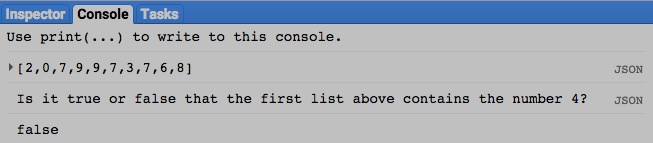
var TheLIST = ee.List( [2,0,7,9,9,7,3,7,6,8] );

var TrueOrFalse = TheLIST.contains( 4 );

print( TheLIST );

print( 'Is it true or false that the first list above contains the number 4?' );

print( TrueOrFalse );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**COMPARING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) TO **LISTS**

list.frequency creates a new integer indicating the number of elements in a specified list that match a specified element.

newInteger = oldList.frequency ( element )

The specified element

The new integer

The specified list

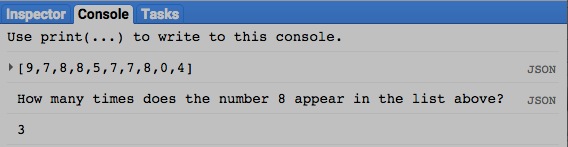
var TheLIST = ee.List( [9,7,8,8,5,7,7,8,0,4] );

var TheNUMBER = TheLIST.frequency( 8 );

print( TheLIST );

print( 'How many times does the number 8 appear in the list above?' );

print( TheNUMBER );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**COMPARING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) TO **LISTS**

list.indexOf creates a new integer indicating the ordinal position of the first occurrence of a specified element within a specified list (set to -1 if is none).

newInteger = oldList.indexOf ( element )

The specified element

The new integer

The specified list

var TheLIST = ee.List( [9,7,8,8,5,7,7,8,0,4] );

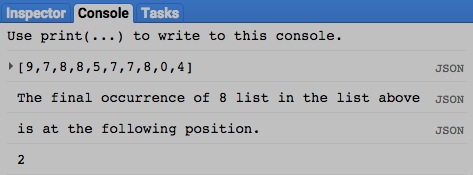
var TheNUMBER = TheLIST.indexOf( 8 );

print( FirstLIST );

print( 'The final occurrence of 8 list in the list above' );

print( 'is at the following position.' );

print( TheNUMBER );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**PARALLEL PROCESSING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx)

list.map creates a new list by applying a specified algorithm to each of the elements in a specified list.

newList = oldList.map ( algorithm)

The specified algorithm

The specified list

The new list

var OldLIST = ee.List( [3,2,1] );

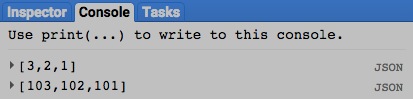
function TheFUNCTION( NextELEMENT )

{ return ee.Number( NextELEMENT ).add(100);

}

var NewLIST = OldLIST.map( TheFUNCTION );

print( OldLIST, NewLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**TALLYING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx)

list.iterate creates a new list by applying a specified algorithm to each element of a specified list in a cumulative manner. The algorithm is initially applied

to a specified starting object and the first element in the list. It is then applied to the output of that first iteration and the second element in the list,

then to that result and the third element, that result and the fourth element, and so on.

newList = oldList.map ( algorithm, startingObject)

The specified starting object

The specified list

The new list

The specified algorithm with two arguments: the first representing each new list element, and the second representing cumulative results

var OldLIST = ee.List( [3,2,1] );

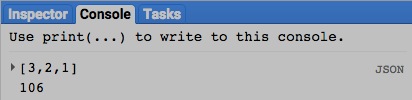
function TheFUNCTION( NextELEMENT, runningTOTAL )

{ return ee.Number(NextELEMENT).add(runningTOTAL);

}

var NewLIST = OldLIST.iterate( TheFUNCTION, 100 );

print( OldLIST, NewLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**MEASURING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx)

list.length and list.size create a new integer indicating the number of elements along the initial axis of a specified list.

newInteger = oldList.length ( ) or .size ( )

The new integer

The specified array

var TheLIST = ee.List( [ [ [111,112,113,114], [121,122,123,124], [131,132,133,134] ],

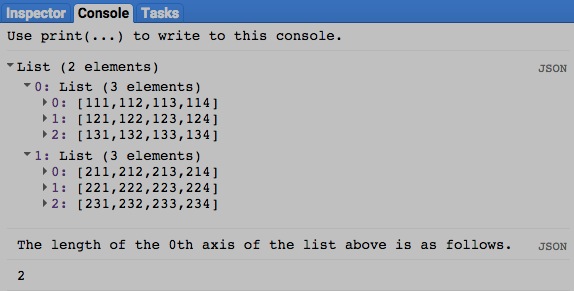
[ [211,212,213,214], [221,222,223,224], [231,232,233,234] ] ] );

var TheNUMBER = TheLIST.length( );

print( TheLIST );

print( 'The length of the 0th axis of the list above is as follows.' );

print( TheNUMBER );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**DOCUMENTING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx)

ee.Algorithms.Describe and list.getInfo

each creates a JSON-compatible text object representing a specified list.

newObject = ee.Algorithms.Describe( oldList )

and oldList.getInfo( )

The new object

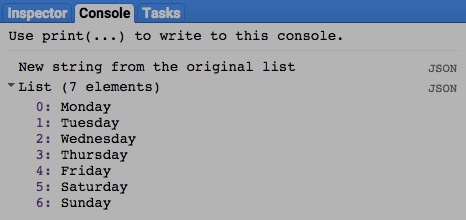
The specified list

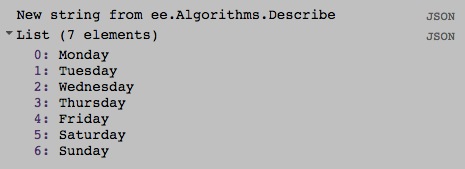
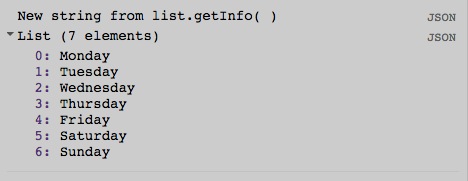
var OldLIST = ee.List( ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'] );

print( 'New string from the original list', OldLIST );

print( 'New string from ee.Algorithms.Describe', ee.Algorithms.Describe( OldLIST ) );

print( 'New string from list.getInfo( )', OldLIST.getInfo( ) );





[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**DOCUMENTING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx)

array.toString and .serialize each creates a new string presenting information on a specified list.

newString = oldList.toString ( )

and oldList.serialize( )

The specified list

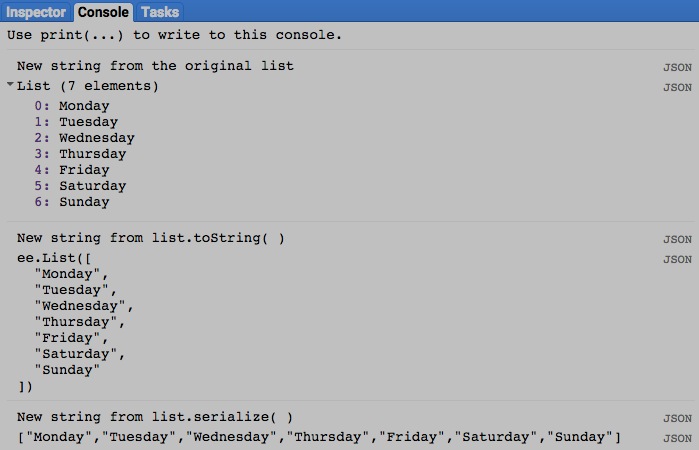
The new string

var OldLIST = ee.List( ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'] );

print( 'New string from the original list', OldLIST );

print( 'New string from list.toString( )', OldLIST.toString( ) );

print( 'New string from list.serialize( )', OldLIST.serialize( ) );

****

[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**PRESENTING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) IN **PRINT**

print ( list ) and console.log ( list ) present JSON-formatted text renditions of a specified list in the console.

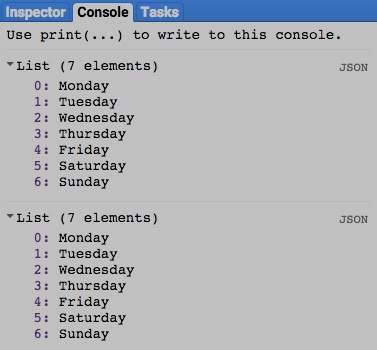
print( oldList ) or console.log( oldList )

The specified list

var TheLIST = ee.List( ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'] );

print( TheLIST );

console.log( TheLIST );



[GOOGLE EARTH ENGINE](EE01%20Earth%20Engine%20(EE).docx) [APPLICATION PROGRAMMING INTERFACE](EE05%20%20%20The%20EE%20API.docx) [CAPABILITIES](EE07%20%20%20%20%20%20API%20Capabilities.docx)

**PRESENTING** [LIST](#_top) [VARIABLES](EE13%20%20%20%20%20%20%20%20%20Variables.docx) IN **PRINT**

alert ( list ) and confirm( list ) present JSON-formatted text renditions of a specified list in a pop-up message box.

alert( oldList ) or confirm( oldList )

The specified list

var TheLIST = ee.List( ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'] );

alert( TheLIST );

confirm( TheLIST );

