[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 **DATERANGE** [PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)

A dateRange is an Earth Engine parameter object that represents the span of time from the beginning of “start” date onward until (but not including) the beginning of an “end” date. DateRanges 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** DATERANGES [ee.DateRange](#DateRange) [ee.DateRange.unbounded](#isUnbounded)

**TRANSFORMING** DATERANGES [dateRange.union](#union) [dateRange.intersection](#intersection)

**QUERYING** DATERANGES [dateRange.start](#start) [dateRange.end](#end)

**DESCRIBING** DATERANGES [dateRange.intersects](#intersects) [dateRange.contains](#contains)

[dateRange.isEmpty](#isEmpty) [dateRange.isUnbounded](#isUnbounded)

**DOCUMENTING** DATERANGES dateRange.getInfo ee.Algorithms.Describe(dateRange)

dateRange.toString dateRange.serialize

**PRESENTING** DATERANGES

IN **PRINT** print(dateRange) console.log(dateRange)

alert(dateRange) confirm(dateRange)

[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

ee.DateRange creates a DateRange object extending from a specified starting date to a specified stopping date and including both of those dates.

newDateRange = ee.DateRange( startingDate*, stoppingDate, timeZone* )

A time zone to be assumed when either startingDate

or stoppingDate is specified as a string. Default: UTC

The specified starting date, given as a Date object, a number

(interpreted as milliseconds since the beginning of January 1,

1970), or a string (in YYYY-MM-DD, YYYY-DDD, or

YYYY-MM-DD-TTTTTT format such as '1996-01-01' or

'1996-001' or '1996-01-01T08:00').

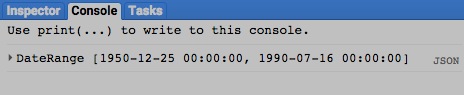
The new

DateRange

A specified stopping date, given as a Date object, a number (interpreted as milliseconds since the beginning of January 1, 1970), or a string (in YYYY-MM-DD, YYYY-DDD, or YYYY-MM-DD-TTTTTT format such as '1996-01-01' or '1996-001' or '1996-01-01T08:00'). Default: one millisecond after **startingDate**

var TheDATERANGE = ee.DateRange( '1950-12-25', '1990-07-16' );

print( TheDATERANGE );



[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

ee.DateRange.unbounded creates a dateRange that encompasses all possible dates.

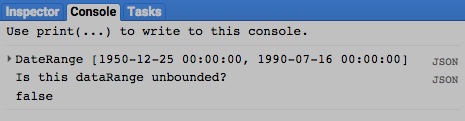
newDateRange = ee.DateRange.unbounded( )

The new dateRange

var TheDATERANGE = ee.DateRange( '1950-12-25', '1990-07-16' );

var TheBOOLEAN = TheDATERANGE.isUnbounded( );

print( TheDATERANGE, "Is this dataRange unbounded?", TheBOOLEAN );



[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

dateRange.union creates a new dateRange encompassing the union of two specified dateRanges.

newDateRange = 1stDateRange.union( 2ndDateRange )

The first specified dateRange

The first specified dateRange

The new dateRange

var MozartDATERANGE = ee.DateRange( '1756-1-27', '1791-12-5' );

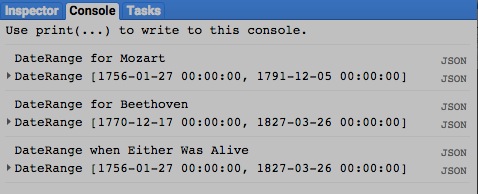
var BeethovenDATERANGE = ee.DateRange( '1770-12-17', '1827-3-26' );

var TheirTotalDATERANGE = MozartDATERANGE.union( BeethovenDATERANGE );

print( 'DateRange for Mozart', MozartDATERANGE );

print( 'DateRange for Beethoven', BeethovenDATERANGE );

print( 'DateRange when Either Was Alive', TheirTotalDATERANGE );



[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

dateRange.intersection creates a new dateRange encompassing the intersection of two specified dateRanges.

newDateRange = 1stDateRange.intersection( 2ndDateRange )

The first specified dateRange

The first specified dateRange

The new dateRange

var MozartDATERANGE = ee.DateRange( '1756-1-27', '1791-12-5' );

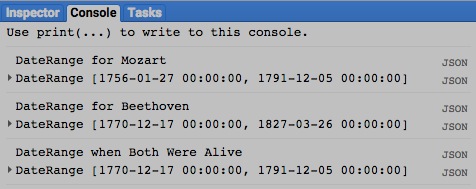
var BeethovenDATERANGE = ee.DateRange( '1770-12-17', '1827-3-26' );

var TheirSharedDATERANGE = MozartDATERANGE.intersection( BeethovenDATERANGE );

print( 'DateRange for Mozart', MozartDATERANGE );

print( 'DateRange for Beethoven', BeethovenDATERANGE );

print( 'DateRange when Both Were Alive', TheirSharedDATERANGE );



[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

dateRange.start creates a new date set to the (inclusive) start of a specified dateRange.

newDate = oldDateRange.start( )

The specified dateRange

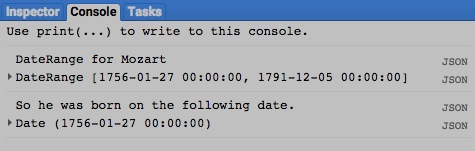
The new Date

var MozartDATERANGE = ee.DateRange( '1756-1-27', '1791-12-5' );

var MozartBirthDATE = MozartDATERANGE.start( );

print( 'DateRange for Mozart', MozartDATERANGE );

print( 'So he was born on the following date.', MozartBirthDATE );



[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

dateRange.end creates a new date set to the (exclusive) end of a specified dateRange.

newDate = oldDateRange.end( )

The specified dateRange

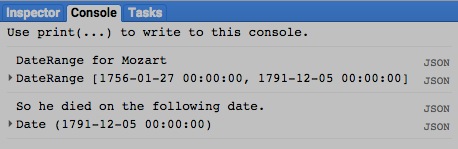
The new Date

var MozartDATERANGE = ee.DateRange( '1756-1-27', '1791-12-5' );

var MozartDeathDATE = MozartDATERANGE.end( );

print( 'DateRange for Mozart', MozartDATERANGE );

print( 'So he died on the following date.', MozartDeathDATE );



[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

dateRange.intersects creates a new Boolean set to true (only) if one specified dateRange overlaps at all with another specified dateRange.

newBoolean = 1stDateRange.intersects( 2ndDateRange )

The new Boolean

The first specified dateRange

The first specified dateRange

var MozartDATERANGE = ee.DateRange( '1756-1-27', '1791-12-5' );

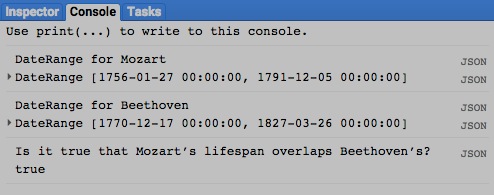
var BeethovenDATERANGE = ee.DateRange( '1770-12-17', '1827-3-26' );

var TheBOOLEAN = MozartDATERANGE.intersects( BeethovenDATERANGE );

print( 'DateRange for Mozart', MozartDATERANGE );

print( 'DateRange for Beethoven', BeethovenDATERANGE );

print( 'Is it true that Mozart’s lifespan overlaps Beethoven’s?', TheBOOLEAN );

****

[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

dateRange.contains creates a new Boolean set to true (only) if a specified dateRange contains another specified date or dateRange.

newBoolean = oldDateRange.contains( anotherDateOrDateRange )

The specified “containee” date or dateRange

The specified “container” dateRange

The new Boolean

var MozartDATERANGE = ee.DateRange( '1756-1-27', '1791-12-5' );

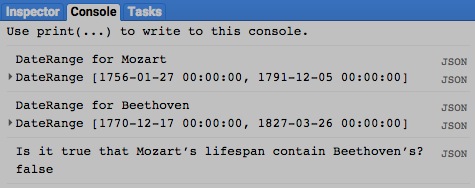
var BeethovenDATERANGE = ee.DateRange( '1770-12-17', '1827-3-26' );

var TheBOOLEAN = MozartDATERANGE.contains( BeethovenDATERANGE );

print( 'DateRange for Mozart', MozartDATERANGE );

print( 'DateRange for Beethoven', BeethovenDATERANGE );

print( 'Is it true that Mozart’s lifespan contain Beethoven’s?', TheBOOLEAN );



[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

dateRange.isEmpty creates a new Boolean set to true (only) if a specified dateRange contains no dates (because its ending date precedes its starting date).

newBoolean = oldDateRange.isEmpty( )

The specified dateRange

The new Boolean

var TheDATERANGE = ee.DateRange( '1990-07-16', '1950-12-25');

var TheBOOLEAN = TheDATERANGE.isEmpty( );

print( TheDATERANGE );

print( 'Is it true that this dateRange is empty?', TheBOOLEAN );



[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

dateRange.isUnbounded creates a new Boolean set to true (only) if a specified dateRange contains all possible dates.

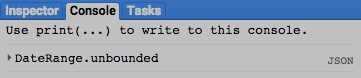
newBoolean = oldDateRange.isUnbounded( )

The specified dateRange

The new Boolean

var TheDATERANGE = ee.DateRange.unbounded( );

print( TheDATERANGE );



[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

ee.Algorithms.Describe and dateRange.getInfo

each create a JSON-compatible text object representing a specified dateRange.

newObject = ee.Algorithms.Describe( oldDateRange )

and oldDateRange.getInfo( )

The specified dateRange

The new object

var TheDATERANGE = ee.DateRange( '1990-07-16', '1950-12-25');

print( 'From print:', TheDATERANGE );

print( 'From ee.Algorithms.Describe( ):', ee.Algorithms.Describe( TheDATERANGE ) );

print( 'From getInfo( ):', TheDATERANGE.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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx)

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

newString = oldDateRange.toString ( )

and oldDateRange.serialize( )

The specified dateRange

The new string

var TheDATERANGE = ee.DateRange( '1990-07-16', '1950-12-25');

print( 'From print:', TheDATERANGE );

print( 'From toString( ):', TheDATERANGE.toString( ) );

print( 'From serialize( ):', TheDATERANGE.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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx) IN **PRINT**

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

print( oldDateRange ) or console.log( oldDateRange )

The specified dateRange

var TheDATERANGE = ee.DateRange( '1990-07-16', '1950-12-25');

print( 'From print:', TheDATERANGE );

console.log( 'From console.log:', TheDATERANGE );

[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** [DATERANGE](#_top) [[PARAMETERS](EE26%20%20%20%20%20%20%20%20%20Parameters.docx)](EE25%20%20%20%20%20%20%20%20%20Parameters.docx) IN **PRINT**

alert ( dateRange ) and confirm ( dateRange ) present JSON-formatted text renditions of a specified

dateRange in a pop-up message box.

alert( oldDateRange ) or confirm( oldDateRange )

The specified dateRange

var TheDATERANGE = ee.DateRange( '1990-07-16', '1950-12-25');

alert( TheDATERANGE );

confirm( TheDATERANGE );