LAMPIRAN

Pengolahan Data dengan Google Earth Engine (GEE)

```
//#1 Fungsi Cloud Masking var
maskL8 = function(image) {
var qa = image.select('BQA');
return image.updateMask(qa.bitwiseAnd(1 << 4).eq(0));};
//#2 Filter Data Citra var
composite = imageCollection
.filterDate('2021-01-01', '2021-
12-31') //format tanggal YYYY-
MM-DD
.map(maskL8) //Membuat cloud masking pada data komposit citra
.median() //Mengambil median dari populasi piksel pada periode tersebut
.clip(table); //Memotong data citra menggunakan polygon
//#3 Memunculkan komposit pada Map
Map.addLayer(composite,{bands:['B5','B4','B3'],min:0,max:0.3},'Komposit 543')
Map.addLayer(composite,{bands:['B4','B3','B2'],min:0,max:0.3},'Komposit 432')
//#4 Membuat NDVI
//var ndvi = composite.normalizedDifference(['B5','B4']).rename('NDVI'); //format
Map (data,{visualisasi},"nama layer")
//#5 Memunculkan komposit pada Map
//Map.addLayer(ndvi,{palette:["blue","red","yellow","green",
"darkgreen"],min:0,max:1},"NDVI")
//#6 Membuat LST
//cloud mask function
maskL8sr(col) {
// Bits 3 and 5 are cloud shadow and cloud, respectively.
```

```
var cloudShadowBitMask = (1 << 3);   var
cloudsBitMask = (1 << 5);
// Get the pixel QA band. var
qa= col.select('pixel_qa');
// Both flags should be set to zero, indicating clear conditions.
var mask = qa.bitwiseAnd(cloudShadowBitMask).eq(0)
.and(qa.bitwiseAnd(cloudsBitMask).eq(0));   return
col.updateMask(mask);}
//vis params var
vizParams = { bands:
['B5', 'B6', 'B4'], min:
0, max: 4000, gamma:
[1, 0.9, 1.1]}; var
vizParams2 = { bands:
['B4', 'B3', 'B2'], min:
0, max: 3000, gamma:
1.4,};
//load the collection:
{var col = ee.ImageCollection('LANDSAT/LC08/C01/T1_SR')
.map(maskL8sr)
.filterDate('2021-01-01', '2021-12-31')
.filterBounds(table);}
print(col, 'coleccion');
//imagen reduction {var
image = col.median();
print(image, 'image');
Map.addLayer(image, vizParams2);}
//median
```

```
{var ndvi = image.normalizedDifference(['B5',
'B4']).rename('NDVI');
var ndviParams = {min: -1, max: 1, palette: ["blue","red","yellow","green",
"darkgreen"]}; print(ndvi,'ndvi');
Map.addLayer(ndvi, ndviParams, 'ndvi');}
//select thermal band 10(with brightness tempereature), no calculation  var
thermal= image.select('B10').multiply(0.1);
var b10Params = {min: 291.918, max: 302.382, palette: ['blue',
'white', 'green']};
Map.addLayer(thermal, b10Params, 'thermal');
// find the min and max of NDVI {var min
= ee.Number(ndvi.reduceRegion({ reducer:
ee.Reducer.min(), geometry: table, scale:30,
maxPixels: 1e9}).values().get(0)); print(min,
'min');
var max = ee.Number(ndvi.reduceRegion({ reducer:
ee.Reducer.max(),
geometry: table,scale: 30, maxPixels: 1e9}).values().get(0)); print(max,
'max')}
//fractional vegetation
{var fv
=(ndvi.subtract(min).divide(max.subtract(min))).pow(ee.Number(2)).rename('FV');
print(fv, 'fv');
Map.addLayer(fv);}
//Emissivity var a=
ee.Number(0.004); var b=
ee.Number(0.986);
var EM=fv.multiply(a).add(b).rename('EMM');
var imageVisParam3 = {min: 0.9865619146722164, max:0.989699971371314};
```

```
Map.addLayer(EM, imageVisParam3,'EMM');
//LST in Celsius Degree bring -273.15
//NB: In Kelvin don't bring -273.15 var
LST = thermal.expression(
'(Tb/(1 + (0.00115* (Tb / 1.438))*log(Ep)))-273.15', {
 'Tb': thermal.select('B10'),
'Ep': EM.select('EMM')
}).rename('LST');
Map.addLayer(LST, {min: 20.569706944223423, max:29.328077233404645, palette:
['040274', '040281', '0502a3', '0502b8', '0502ce', '0502e6',
'0602ff', '235cb1', '307ef3', '269db1', '30c8e2', '32d3ef',
'3be285', '3ff38f', '86e26f', '3ae237', 'b5e22e', 'd6e21f',
'fff705', 'ffd611', 'ffb613', 'ff8b13', 'ff6e08', 'ff500d',
'ff0000', 'de0101', 'c21301', 'a71001', '911003' ]},'LST');
Export.image.toDrive({ image: LST.clip(table),
description: 'LST_2021', folder: 'GEE_jaksel_T',
fileNamePrefix: 'LST__2021', region: table,
fileFormat: 'GEOTIFF', scale: 30});
 Export.image.toDrive({ image:
ndvi.clip(table), description:
'NDVI_2021', folder:
'GEE_jaksel_T',
fileNamePrefix:
'NDVI__2021', region: table,
fileFormat: 'GEOTIFF', scale:
30});
```