In [1]:
```
# imports
import pandas as pd
import agrilution_aws
import logging
import boto3
from datetime import datetime
import sys
from boto3.dynamodb.conditions import Key, Attr
import time
from agrilution_aws import DynamoDbApi
from matplotlib.pyplot import figure
from matplotlib import pyplot as plt
import seaborn as sns
import plotly.express as px
import dask.dataframe as dd
import plotly.graph_objects as go
from plotly.subplots import make_subplots
from plotly import tools
import warnings
warnings.filterwarnings('ignore')
pd.options.mode.chained_assignment = None
import dask.dataframe as dd
import datetime
```

In [2]:
```
# globals

# dynamoDB API
dynamo = DynamoDbApi(logging.getLogger(), table_name = 'archive')
# timestamp in ms marking 6th of july 2022
timestamp2 = 1657065600000
```

In [3]:
```
# list of all lab cubes
plantcubes = {
    'A1_lower': 'd6472f5d-94f9-4a31-9a8e-ddc6744023d6',
    'A1_upper': 'bf6b3065-a5ad-49f0-96e3-f1ed22e55e18',
    'A2_lower': '07b17561-3b04-4094-a8ab-2f67315adfdd',
    'A2_upper': '2ba34bbe-1611-4c9b-8a5e-1c802ff77768',
    'A3_lower': '26b03d30-3a9d-4460-a0cb-7ef5c1d5dec8',
    'A3_upper': '955605fe-8666-449b-96b4-e973b1e197da',
    'A4_lower': '09ef2ce0-2f99-45cf-8cb5-99550fca494f',
    'A4_upper': 'b637f6a6-b6e2-486c-86db-cc431d0b2a58',
    'B1_lower': '5a9039ae-957b-42b2-9d09-3baf73cf0020',
    'B1_upper': '0b66fd54-465b-409f-838f-ca5e494e68fb',
    'B2_lower': 'd9dd3086-fe92-4cab-b235-be2b283c4999',
    'B2_upper': '2853d150-f30a-4f35-a4fc-5985b35876dc',
    'B3_lower': 'a27588d5-bc01-44ab-b96d-cad7f86402b0',
    'B3_upper': 'd22ff6af-211b-4743-a5ae-5fd89ffbe446',
    'B4_lower': '11c45cd6-8d1f-4140-a545-0db886918e3b',
    'B4_upper': '510d7df1-234c-46f8-a153-ec792edc93b1',
    'C1_lower': '0427a2fa-8a50-4d00-ad56-6246c03ef9d0',
    'C1_upper': 'eac52b39-02c0-4a7a-a9e5-010709ee15c8',
    'C2_lower': 'ab713fff-4bd2-4a72-afdd-603e31b57689',
    'C2_upper': '09aefdec-f638-4e2d-91d2-375094a3d881',
    'C3_lower': '8cb8a481-a70d-4988-b419-d905d06ca65d',
    'C3_upper': '7d53b428-7777-47f0-9605-01ac8bda96f4',
    'C4_lower': '1acd7d04-fb3b-4983-abbf-24053e3a1499',
    'C4_upper': '5b23e086-1365-48a2-af39-defa77768aa5',
    'D1_lower': '5ae3a1b3-5354-4b23-ab83-aa9f3029098d',
    'D1_upper': '820b0870-b586-45b8-9a1e-fdd41a842f5d',
    'D2_lower': 'd183f2bd-d1df-4f83-a34d-6c72601b97f2',
    'D2_upper': '69a5e2a3-624c-4522-b0ee-ee28846fc700',
    'D3_lower': 'f598f96e-b0f4-4009-85e1-e621e8306c36',
    'D3_upper': '9788f724-0b7a-47ae-8e95-2c35152e20b8',
    'E1' : '2933af4a-51d4-4894-aa60-753219ca1918',
    'E2' : 'f29ffb36-be56-46e1-9e9d-d05e44e9a1a0',
    'E3': 'b2d1811e-dbff-4fcb-a219-468adfb045ea',
    'E4' : '422d6453-a501-4ed9-bd4d-02b510a6e6d7',
    'E5' : '2b9c5df5-e286-4f0a-ab87-2271535677b6',
    'E6' : '12652341-6356-4c7f-9a60-eb5d82b16a57',
    'E7' : '52fdc759-32a3-43da-8207-3e4b89bafaae',
    'E8' : '424b5b0a-724f-4ab6-9688-f3fb2fab1cef',
    'E9' : 'dfde8871-522f-4ee5-a572-82049fe112cd',
    'E10' : 'c9299fd6-a636-4487-a252-837399139e8e',
```

```
}
```

In [4]:
```
# list of data frames for each single cube
frames = []
# iterate through plantcubes
for cube in plantcubes.values():
    # query_table of agrutils package already queries until no next token is given anymore
    resp = dynamo.query_table(
        KeyConditionExpression = (
            Key('plantcube').eq(cube) &
            Key('timestamp').gt(timestamp2)
        )
    )
    # convert to pandas data frame
    df = pd.DataFrame(resp)
    # attach to list of all data frames
    frames.append(df)
# create one big data frame for all cubes
all_cubes= pd.concat(frames)
```

In [6]:
```
all_cubes.to_csv('jul6-now')
```

In [ ]:
```
all_cubes = pd.read_csv('jul6-now')
```

In [31]:
```
df2 = all_cubes.copy()
```

In [ ]:
```
#converting it into pandas dataframe
#df2 = df2.compute()
```

In [32]:
```
df2.columns
```

Out[32]:
```
Index(['temp_b', 'plantcube', 'timestamp', 'temp_a', 'rssi', 'temp_led_b',
       'temp_tank', 'cooling', 'fan_a', 'fan_a_tacho', 'humid_b', 'temp_led_a',
       'fan_led_a', 'fan_led_a_tacho', 'fan_led_b', 'fan_led_b_tacho',
       'wifi_level', 'light_b', 'led_a', 'led_b', 'light_a', 'door',
       'recipe_id', 'user_button', 'signal_led', 'mode', 'fan_b', 'pump',
       'valve', 'fan_b_tacho', 'tank_level_raw', 'ec', 'humid_a', 'connected',
       'firmware_ncu', 'total_offset', 'tank_level', 'last_cleaning_done',
       'next_cleaning_due', 'virtual_cube_mode', 'firmware_mcu',
       'verbose_reporting', 'led_a_board_state', 'currently_cleaning',
       'led_b_board_state', 'stage', 'nutrient_deficit', 'owner',
       'user_offset'],
      dtype='object')
```

In [33]:
```
#extracting the required columns
cols = [0,1,2,3,22,25,33]
df2 = df2[df2.columns[cols]]
```

In [34]:
```
df2.head()
```

Out[34]:

	temp_b	plantcube	timestamp	temp_a	recipe_id	mode	connected
0	21.9	d6472f5d-94f9-4a31-9a8e-ddc6744023d6	1657065620650	NaN	NaN	NaN	NaN
1	NaN	d6472f5d-94f9-4a31-9a8e-ddc6744023d6	1657065631644	19.97	NaN	NaN	NaN
2	NaN	d6472f5d-94f9-4a31-9a8e-ddc6744023d6	1657065708650	20.07	NaN	NaN	NaN
3	NaN	d6472f5d-94f9-4a31-9a8e-ddc6744023d6	1657065721203	NaN	NaN	NaN	NaN
4	NaN	d6472f5d-94f9-4a31-9a8e-ddc6744023d6	1657065731200	NaN	NaN	NaN	NaN

In [35]:
```
df2['mode'].unique()
```

Out[35]:
```
array([nan, Decimal('3'), Decimal('0'), Decimal('1'), Decimal('4'),
       Decimal('2')], dtype=object)
```

In [37]:
```
#Number of Null values in each column
df2.isnull().sum()
```

Out[37]:
```
temp_b       17690038
plantcube           0
timestamp           0
temp_a       17568719
recipe_id    18573304
mode         18574163
connected    18565890
dtype: int64
```

In [38]:
```
#dropping the row if all the values in the given columns are NA
df2.dropna(subset=['connected','recipe_id','temp_b','temp_a','mode'], how='all', inplace=True)
```

In [39]:
```
df2.head()
```

Out[39]:

	temp_b	plantcube	timestamp	temp_a	recipe_id	mode	connected
0	21.9	d6472f5d-94f9-4a31-9a8e-ddc6744023d6	1657065620650	NaN	NaN	NaN	NaN
1	NaN	d6472f5d-94f9-4a31-9a8e-ddc6744023d6	1657065631644	19.97	NaN	NaN	NaN
2	NaN	d6472f5d-94f9-4a31-9a8e-ddc6744023d6	1657065708650	20.07	NaN	NaN	NaN
6	NaN	d6472f5d-94f9-4a31-9a8e-ddc6744023d6	1657065782664	20.17	NaN	NaN	NaN
7	NaN	d6472f5d-94f9-4a31-9a8e-ddc6744023d6	1657065865626	20.27	NaN	NaN	NaN

In [40]:
```
#converting timestamp to datetime format
df2['timestamp'] = df2['timestamp'].astype('int64')
df2['timestamp'] = pd.to_datetime(df2['timestamp'], unit='ms')
```

In [41]:
```
#replacing the plantcube name with their alias names
dict1 = {v : k for k, v in plantcubes.items()}
df2.plantcube = df2.plantcube.replace(dict1)
df2.head()
```

Out[41]:

	temp_b	plantcube	timestamp	temp_a	recipe_id	mode	connected
0	21.9	A1_lower	2022-07-06 00:00:20.650	NaN	NaN	NaN	NaN
1	NaN	A1_lower	2022-07-06 00:00:31.644	19.97	NaN	NaN	NaN
2	NaN	A1_lower	2022-07-06 00:01:48.650	20.07	NaN	NaN	NaN
6	NaN	A1_lower	2022-07-06 00:03:02.664	20.17	NaN	NaN	NaN
7	NaN	A1_lower	2022-07-06 00:04:25.626	20.27	NaN	NaN	NaN

In [42]:
```
df2 = df2.reset_index()
```

In [43]:
```
#applying ffill for the columns connected and recipe id
df2['connected'] = df2.groupby('plantcube')['connected'].apply(lambda x:x.fillna(method='ffill'))
df2['recipe_id'] = df2.groupby('plantcube')['recipe_id'].apply(lambda x:x.fillna(method='ffill'))
df2['mode'] = df2.groupby('plantcube')['mode'].apply(lambda x:x.fillna(method='ffill'))
```

In [44]:
```
df2.isnull().sum()
```

Out[44]:
```
index            0
temp_b       959859
plantcube         0
timestamp         0
temp_a       838540
recipe_id    250168
mode         331694
connected     92907
dtype: int64
```

In [45]:
```
#after forward filling the columns 'connected' and 'recipe id' in the above step. Remove the rows if any of these column
#values are null.
df2.dropna(subset=['connected','recipe_id'], how='any', inplace=True)
```

In [46]:
```
df2 = df2.drop('index', axis=1)
```

In [47]:
```
#changing the datatype of the temperature columns
df2['temp_a'] = df2['temp_a'].astype(float)
df2['temp_b'] = df2['temp_b'].astype(float)
```

In [48]:
```
#copying dataframe df2 to idata
idata = df2.copy()
```

In [49]:
```
#dropping the rows which contains the mode(debug)
t1 = idata[idata['mode'] == 1]
```

In [52]:
```
#total records in debug mode
t1.shape[0]
```

Out[52]: 23670

In [53]:
```
#remove the records in debug mode
idata = idata[idata['mode'] != 1]
```

In [54]:
```
idata.head()
```

Out[54]:

	temp_b	plantcube	timestamp	temp_a	recipe_id	mode	connected
7255	NaN	A1_lower	2022-07-12 01:05:55.946	NaN	1657281884	0	False
7256	NaN	A1_lower	2022-07-12 01:41:33.614	NaN	1657281884	0	True
7257	23.9	A1_lower	2022-07-12 01:41:55.004	23.37	1657281884	0	True
7258	NaN	A1_lower	2022-07-12 01:42:03.537	NaN	1657281884	0	True
7259	NaN	A1_lower	2022-07-12 01:45:23.790	23.47	1657281884	0	True

In [55]:
```
#adding recipe along with it.
rdf = pd.read_csv('Recipe_table_sApril')
```

In [56]:
```
rdf.drop('Unnamed: 0', axis=1, inplace=True)
```

In [57]:
```
rdf.head(30)
```

Out[57]:

	layers	plantcube	recipe_id
0	[[{'periods': [{'duration': Decimal('86400'), ...	A1_lower	1649666148
1	[[{'periods': [{'duration': Decimal('86400'), ...	A1_lower	1649854144
2	[[{'periods': [{'duration': Decimal('86400'), ...	A1_lower	1649934487
3	[[{'periods': [{'duration': Decimal('86400'), ...	A1_lower	1650446765
4	[[{'periods': [{'duration': Decimal('86400'), ...	A1_lower	1650462922
5	[[{'periods': [{'duration': Decimal('86400'), ...	A1_lower	1653983191
6	[[{'periods': [{'duration': Decimal('86400'), ...	A1_lower	1653983554
7	[[{'periods': [{'duration': Decimal('86400'), ...	A1_lower	1653985982
8	[[{'periods': [{'duration': Decimal('86400'), ...	A1_lower	1654078825
9	[[{'periods': [{'duration': Decimal('86400'), ...	A1_lower	1657093174
10	[[{'periods': [{'duration': Decimal('86400'), ...	A1_lower	1657281884
11	[[{'periods': [{'duration': Decimal('86400'), ...	A1_upper	1652711767
12	[[{'periods': [{'duration': Decimal('86400'), ...	A1_upper	1652711832
13	[[{'periods': [{'duration': Decimal('86400'), ...	A1_upper	1652711870
14	[[{'periods': [{'duration': Decimal('86400'), ...	A1_upper	1653900414
15	[[{'periods': [{'duration': Decimal('86400'), ...	A1_upper	1654078815
16	[[{'periods': [{'duration': Decimal('86400'), ...	A1_upper	1659961675
17	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1649666159
18	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1649754540
19	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1652084306
20	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1652184211
21	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1652863133
22	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1653910382
23	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1654078855
24	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1654078884

	layers	plantcube	recipe_id
25	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1657015628
26	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1657281891
27	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1662463213
28	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1662463214
29	[[{'periods': [{'duration': Decimal('86400'), ...	A2_lower	1662463215

In [58]:
```
#joining the recipe and archive table based on the attributes plantcube and recipe_id
jdf = pd.merge(idata, rdf, on=['plantcube','recipe_id'], how="left",indicator=True)
```

In [59]:
```
jdf.head()
```

Out[59]:

	temp_b	plantcube	timestamp	temp_a	recipe_id	mode	connected	layers	_merge
0	NaN	A1_lower	2022-07-12 01:05:55.946	NaN	1657281884	0	False	[[{'periods': [{'duration': Decimal('86400'), ...	both
1	NaN	A1_lower	2022-07-12 01:41:33.614	NaN	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...	both
2	23.9	A1_lower	2022-07-12 01:41:55.004	23.37	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...	both
3	NaN	A1_lower	2022-07-12 01:42:03.537	NaN	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...	both
4	NaN	A1_lower	2022-07-12 01:45:23.790	23.47	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...	both

In [60]:
```
#recipe_id - 1 is the default recipe. It is just used to reset the cube. It doesn't mean anything.
jdf.loc[jdf.recipe_id == 1, 'layers'] = "default recipe"
jdf.head()
```

Out[60]:

	temp_b	plantcube	timestamp	temp_a	recipe_id	mode	connected	layers	_merge
0	NaN	A1_lower	2022-07-12 01:05:55.946	NaN	1657281884	0	False	[[{'periods': [{'duration': Decimal('86400'), ...	both
1	NaN	A1_lower	2022-07-12 01:41:33.614	NaN	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...	both
2	23.9	A1_lower	2022-07-12 01:41:55.004	23.37	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...	both
3	NaN	A1_lower	2022-07-12 01:42:03.537	NaN	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...	both
4	NaN	A1_lower	2022-07-12 01:45:23.790	23.47	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...	both

In [61]:
```
jdf.drop('_merge', axis=1, inplace=True)
```

In [62]:
```
jdf.head()
```

Out[62]:

	temp_b	plantcube	timestamp	temp_a	recipe_id	mode	connected	layers
0	NaN	A1_lower	2022-07-12 01:05:55.946	NaN	1657281884	0	False	[[{'periods': [{'duration': Decimal('86400'), ...
1	NaN	A1_lower	2022-07-12 01:41:33.614	NaN	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...
2	23.9	A1_lower	2022-07-12 01:41:55.004	23.37	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...
3	NaN	A1_lower	2022-07-12 01:42:03.537	NaN	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...
4	NaN	A1_lower	2022-07-12 01:45:23.790	23.47	1657281884	0	True	[[{'periods': [{'duration': Decimal('86400'), ...

In [63]:
```
jdf['layers'] = jdf.groupby('plantcube')['layers'].apply(lambda x:x.fillna(method='ffill'))
```

In [64]:
```
jdf['recipe']= jdf['layers'].map(str)
#changing the datatype of column 'recipe' to category
jdf['recipe']= jdf['recipe'].astype('category')
#converting the values in the column 'recipe' to numerical codes
jdf['recipe'] = jdf['recipe'].cat.codes
```

In [65]:
```
idata1 = jdf.copy()
```

In [66]:
```
idata1.recipe.unique()
```

Out[66]:
```
array([ 0, 10,  3,  8,  5,  9,  6,  7,  2,  4,  1], dtype=int8)
```

In [67]:
```
idata1.to_csv("Jul6-preprocessed")
```

In [ ]:
```
idata1 = pd.read_csv('Jul6-preprocessed')
```

In [68]:
```
dataframes =['A1_lower','A1_upper','A2_lower','A2_upper','A3_lower','A3_upper','A4_lower','A4_upper','B1_lower','B1_upper','B2_lower','B2_upper','B3_
```

In [69]:
```
val = ""
def my_func(ndf):
    val = ndf
    # creating a dataframe to store the plantcube
    df = idata1[idata1.plantcube == val]
    df.head(10)

    #set the timestamp as index
    df['timestamp'] = pd.to_datetime(df['timestamp'])
    df = df.set_index('timestamp')

    #interpolating the temperature values based on linear interpolation method after resampling it by date.
    df['temp_a'] = df.resample('D')['temp_a'].apply(lambda x:x.interpolate(method="time",limit_direction = "forward"))
    df['temp_b'] = df.resample('D')['temp_b'].apply(lambda x:x.interpolate(method="time",limit_direction = "forward"))

    #converting connected as category type
    df['connected']= df['connected'].astype('category')

    #instead of true and false, converting it into 0's and 1's
    df['connected'] = df['connected'].cat.codes

    return df

res = []
for dataframe in dataframes:
    data = my_func(dataframe)
    res.append(data)
res1= pd.concat(res)
```

In [70]:
```
res1
```

Out[70]:

timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe
2022-07-12 01:05:55.946	NaN	A1_lower	NaN	1657281884	0	0	[[{'periods': [{'duration': Decimal('86400'), ...	0
2022-07-12 01:41:33.614	NaN	A1_lower	NaN	1657281884	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	0
2022-07-12 01:41:55.004	23.900000	A1_lower	23.370000	1657281884	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	0
2022-07-12 01:42:03.537	23.898462	A1_lower	23.374087	1657281884	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	0
2022-07-12 01:45:23.790	23.862365	A1_lower	23.470000	1657281884	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	0
...	...	...	...	...	...	...	...	...
2022-10-12 15:41:04.543	22.950330	E10	22.710000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
2022-10-12 15:47:40.519	22.990000	E10	22.757822	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
2022-10-12 15:54:52.564	23.029422	E10	22.810000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
2022-10-12 16:05:56.464	23.090000	E10	22.891271	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
2022-10-12 16:08:29.464	23.090000	E10	22.910000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1

1532878 rows × 8 columns

In [71]:
```
res1.recipe.unique()
```

Out[71]:
```
array([ 0, 10,  3,  8,  5,  9,  6,  7,  2,  4,  1], dtype=int8)
```

In [72]:
```
recipe1 = res1[res1.recipe == 1]
recipe1.layers.values[0]
```

Out[72]:
```
"[[{'periods': [{'duration': Decimal('86400'), 'temp': Decimal('23'), 'watering': [Decimal('60'), Decimal('17940')], 'light': [Decimal('0'), Decimal
('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('1')}, {'periods': [{'duration': Decimal('57600'), 'temp': Decimal('23'), 'watering': [Decim
al('0'), Decimal('57600')], 'light': [Decimal('0'), Decimal('0'), Decimal('0'), Decimal('0')]}, {'duration': Decimal('28800'), 'temp': Decimal('2
1'), 'watering': [Decimal('0'), Decimal('28800')], 'light': [Decimal('0'), Decimal('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('60')}],
[{'periods': [{'duration': Decimal('86400'), 'temp': Decimal('23'), 'watering': [Decimal('60'), Decimal('17940')], 'light': [Decimal('0'), Decimal
('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('1')}, {'periods': [{'duration': Decimal('57600'), 'temp': Decimal('23'), 'watering': [Decim
al('0'), Decimal('57600')], 'light': [Decimal('0'), Decimal('0'), Decimal('0'), Decimal('0')]}, {'duration': Decimal('28800'), 'temp': Decimal('2
1'), 'watering': [Decimal('0'), Decimal('28800')], 'light': [Decimal('0'), Decimal('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('60')}]]"
```

In [73]:
```
recipe2 = res1[res1.recipe == 2]
recipe2.layers.values[0]
```

Out[73]:
```
"[[{'periods': [{'duration': Decimal('86400'), 'temp': Decimal('23'), 'watering': [Decimal('60'), Decimal('17940')], 'light': [Decimal('0'), Decimal
('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('1')}, {'periods': [{'duration': Decimal('57600'), 'temp': Decimal('23'), 'watering': [Decim
al('70'), Decimal('28983')], 'light': [Decimal('61'), Decimal('39'), Decimal('33'), Decimal('10')]}, {'duration': Decimal('28800'), 'temp': Decimal
('21'), 'watering': [Decimal('0'), Decimal('28800')], 'light': [Decimal('0'), Decimal('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('1')}],
[{'periods': [{'duration': Decimal('86400'), 'temp': Decimal('23'), 'watering': [Decimal('60'), Decimal('17940')], 'light': [Decimal('0'), Decimal
('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('1')}, {'periods': [{'duration': Decimal('50400'), 'temp': Decimal('23'), 'watering': [Decim
al('40'), Decimal('3540')], 'light': [Decimal('50'), Decimal('60'), Decimal('30'), Decimal('90')]}, {'duration': Decimal('36000'), 'temp': Decimal
('19'), 'watering': [Decimal('40'), Decimal('3540')], 'light': [Decimal('0'), Decimal('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal
('3')}]]"
```

In [74]:
```
recipe3 = res1[res1.recipe == 3]
recipe3.layers.values[0]
```

Out[74]:
```
"[[{'periods': [{'duration': Decimal('86400'), 'temp': Decimal('23'), 'watering': [Decimal('60'), Decimal('17940')], 'light': [Decimal('0'), Decimal
('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('1')}, {'periods': [{'duration': Decimal('57600'), 'temp': Decimal('23'), 'watering': [Decim
al('70'), Decimal('28983')], 'light': [Decimal('61'), Decimal('39'), Decimal('33'), Decimal('10')]}, {'duration': Decimal('28800'), 'temp': Decimal
('21'), 'watering': [Decimal('0'), Decimal('28800')], 'light': [Decimal('0'), Decimal('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('1')}],
[{'periods': [{'duration': Decimal('86400'), 'temp': Decimal('23'), 'watering': [Decimal('60'), Decimal('17940')], 'light': [Decimal('0'), Decimal
('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('1')}, {'periods': [{'duration': Decimal('57600'), 'temp': Decimal('23'), 'watering': [Decim
al('70'), Decimal('28983')], 'light': [Decimal('61'), Decimal('39'), Decimal('33'), Decimal('10')]}, {'duration': Decimal('28800'), 'temp': Decimal
('21'), 'watering': [Decimal('0'), Decimal('28800')], 'light': [Decimal('0'), Decimal('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal
('1')}]]"
```

In [75]:
```
recipe4 = res1[res1.recipe == 4]
recipe4.layers.values[0]
```

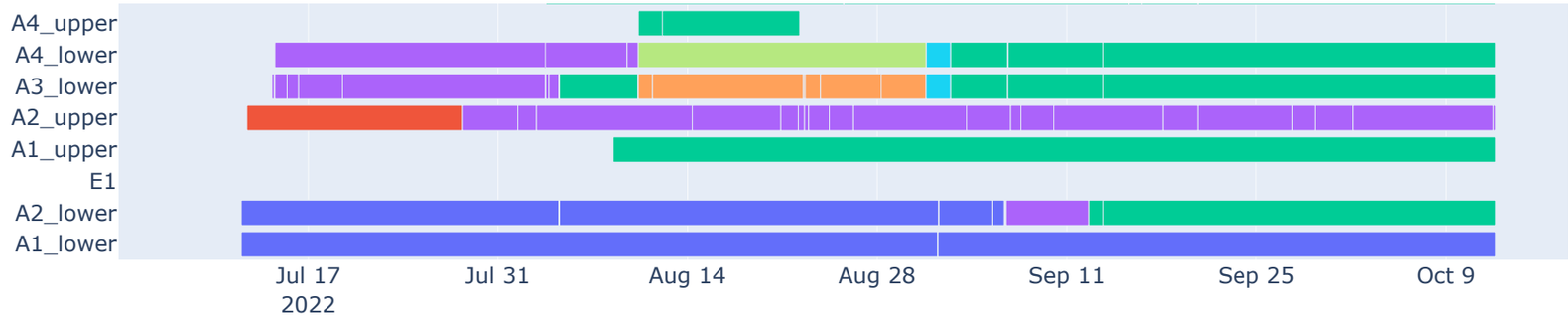
Out[75]:
```
"[[{'periods': [{'duration': Decimal('86400'), 'temp': Decimal('23'), 'watering': [Decimal('60'), Decimal('17940')], 'light': [Decimal('0'), Decimal
('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('1')}, {'periods': [{'duration': Decimal('57600'), 'temp': Decimal('23'), 'watering': [Decim
al('70'), Decimal('57530')], 'light': [Decimal('61'), Decimal('39'), Decimal('33'), Decimal('10')]}, {'duration': Decimal('28800'), 'temp': Decimal
('21'), 'watering': [Decimal('0'), Decimal('28800')], 'light': [Decimal('0'), Decimal('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('6
0')}], [{'periods': [{'duration': Decimal('86400'), 'temp': Decimal('23'), 'watering': [Decimal('60'), Decimal('17940')], 'light': [Decimal('0'), De
cimal('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('1')}, {'periods': [{'duration': Decimal('57600'), 'temp': Decimal('23'), 'watering':
[Decimal('70'), Decimal('57530')], 'light': [Decimal('61'), Decimal('39'), Decimal('33'), Decimal('10')]}, {'duration': Decimal('28800'), 'temp': De
cimal('21'), 'watering': [Decimal('0'), Decimal('28800')], 'light': [Decimal('0'), Decimal('0'), Decimal('0'), Decimal('0')]}], 'cycles': Decimal('6
0')}]]"
```

In [ ]:
```
dataframes =[A1_lower,A1_upper,A2_lower,A2_upper,A3_lower,A3_upper,A4_lower,A4_upper,B1_lower,B1_upper,B2_lower,B2_upper,B3_upper,B3_lower,B4_lower,B
```

In [109]:
```
#function to find out when the plantcube starts and ends the recipe
def my_func(dfn):
    val = dfn
    dfn1 = idata1[idata1.plantcube == val]
    current_recipe = None
    start_plantcube = None
    start_time = None
    time = None
    result = []
    for recipe, time ,plantcube in zip(dfn1['recipe'], dfn1['timestamp'],dfn1['plantcube']):
        if recipe != current_recipe:
            if current_recipe is not None and start_plantcube is not None and start_time is not None and time is not None:
                result.append([start_plantcube, current_recipe, start_time, time])
            current_recipe, start_time,start_plantcube = recipe, time,plantcube

    result.append([start_plantcube, current_recipe, start_time, time])
    ddata1 = pd.DataFrame(result, columns=['plantcube','recipe','RecipeStartTime','RecipeEndTime'])
    return(ddata1)

resn = []
for dataframe in dataframes:
    dataframe = my_func(dataframe)
    resn.append(dataframe)
resn1= pd.concat(resn)
print(resn1)
```

```
   plantcube recipe             RecipeStartTime               RecipeEndTime
0   A1_lower      0  2022-07-12 01:05:55.946000  2022-09-01 11:18:14.765000
1   A1_lower     10  2022-09-01 11:18:14.765000  2022-09-01 11:18:14.888000
2   A1_lower      0  2022-09-01 11:18:14.888000  2022-10-12 15:14:37.321000
0   A1_upper      3  2022-08-08 12:12:01.877000  2022-08-08 12:12:02.187000
1   A1_upper     10  2022-08-08 12:12:02.187000  2022-08-08 12:12:02.269000
..       ...    ...                         ...                         ...
29       E10      1  2022-09-06 07:14:59.417000  2022-09-13 15:33:24.321000
30       E10     10  2022-09-13 15:33:24.321000  2022-09-13 15:33:24.449000
31       E10      1  2022-09-13 15:33:24.449000  2022-09-29 10:01:07.757000
32       E10     10  2022-09-29 10:01:07.757000  2022-09-29 10:01:07.842000
33       E10      1  2022-09-29 10:01:07.842000  2022-10-12 16:08:29.464000

[743 rows x 4 columns]
```

In [110]:
```
resn1['recipe'] = resn1['recipe'].astype(str)
fig = px.timeline(
    resn1, x_start="RecipeStartTime", x_end="RecipeEndTime", y="plantcube",
    color='recipe', height=800, width=1000
)
fig.show()
```



In [ ]:
```
#recipe 1 - 21,23
#recipe 2 - 21,23,19
#recipe 3 - 21,23
#recipe 4 - 21,23
#recipe 6 - 28,25
```

In [ ]:
```
#uncommon recipes(5,7,8,9 - runs for 24 hrs|10-default recipe|0 - different duration|6-different temp target)
#common recipes(1,3,4 - runs day for 16 night for 8 hrs with same day and night target)
#so picking only the recipes 1,3,4 for further evaluation in res2
```

In [76]:
```
res2 = res1[(res1.recipe != 7)&(res1.recipe != 8)&(res1.recipe != 9)&(res1.recipe != 10)&(res1.recipe != 0)&(res1.recipe != 5)&(res1.recipe != 6)&(re
```

In [77]:
```
#res2 contains only the common recipes
res2
```

Out[77]:

timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe
2022-08-08 12:12:01.877	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
2022-08-08 12:12:02.269	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
2022-08-08 12:12:30.469	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
2022-08-08 12:12:31.183	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
2022-08-08 12:14:27.892	23.540000	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
...	...	...	...	...	...	...	...	...
2022-10-12 15:41:04.543	22.950330	E10	22.710000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
2022-10-12 15:47:40.519	22.990000	E10	22.757822	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
2022-10-12 15:54:52.564	23.029422	E10	22.810000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
2022-10-12 16:05:56.464	23.090000	E10	22.891271	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
2022-10-12 16:08:29.464	23.090000	E10	22.910000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1

1286114 rows × 8 columns

In [78]:
```
res2.plantcube.unique()
```

Out[78]:
```
array(['A1_upper', 'A2_lower', 'A3_lower', 'A4_lower', 'A4_upper',
       'B1_lower', 'B1_upper', 'B2_lower', 'B2_upper', 'B3_upper',
       'B3_lower', 'B4_lower', 'B4_upper', 'C1_lower', 'C1_upper',
       'C2_upper', 'D1_lower', 'D1_upper', 'D2_lower', 'D2_upper',
       'D3_lower', 'D3_upper', 'E2', 'E3', 'E4', 'E5', 'E6', 'E7', 'E8',
       'E9', 'E10'], dtype=object)
```

In [79]:
```
res2 = res2.reset_index()
```

In [80]:
```
#user offset 64800
r1 = res2[(res2['plantcube'] == 'A1_upper')]
str(datetime.timedelta(seconds = 64800))
```

Out[80]: '18:00:00'

In [81]:
```
r1
```

Out[81]:

	timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe
0	2022-08-08 12:12:01.877	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
1	2022-08-08 12:12:02.269	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
2	2022-08-08 12:12:30.469	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
3	2022-08-08 12:12:31.183	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
4	2022-08-08 12:14:27.892	23.540000	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
...	...	...	...	...	...	...	...	...	...
71896	2022-10-12 15:07:42.830	21.540000	A1_upper	21.134459	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
71897	2022-10-12 15:09:24.821	21.571098	A1_upper	21.180000	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
71898	2022-10-12 15:12:56.061	21.635506	A1_upper	21.280000	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
71899	2022-10-12 15:13:10.801	21.640000	A1_upper	21.286023	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
71900	2022-10-12 15:17:00.786	21.640000	A1_upper	21.380000	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3

71901 rows × 9 columns

In [82]:
```
#user offset 32422
r2 = res2[(res2['plantcube'] == 'E1')]
str(datetime.timedelta(seconds = 32422))
```

Out[82]: '9:00:22'

In [83]:
```
r2
```

Out[83]:

timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe

In [84]:
```
#user offset 32400
r3 = res2[(res2['plantcube'] != 'E1') & (res2['plantcube'] != 'A1_upper')]
#r3 = res2.query("plantcube not in ['E1', 'A1_upper']")
str(datetime.timedelta(seconds = 32400))
```

Out[84]: '9:00:00'

In [85]:
```
r3
```

Out[85]:

	timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe
71901	2022-09-12 14:28:12.493	23.309375	A2_lower	NaN	1662992890	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
71902	2022-09-12 14:28:29.499	23.310000	A2_lower	NaN	1662992890	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
71903	2022-09-12 14:29:46.409	23.410000	A2_lower	NaN	1662992890	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
71904	2022-09-12 14:31:34.861	23.510000	A2_lower	NaN	1662992890	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
71905	2022-09-12 14:32:56.826	23.410000	A2_lower	NaN	1662992890	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3
...	...	...	...	...	...	...	...	...	...
1286109	2022-10-12 15:41:04.543	22.950330	E10	22.710000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
1286110	2022-10-12 15:47:40.519	22.990000	E10	22.757822	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
1286111	2022-10-12 15:54:52.564	23.029422	E10	22.810000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
1286112	2022-10-12 16:05:56.464	23.090000	E10	22.891271	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1
1286113	2022-10-12 16:08:29.464	23.090000	E10	22.910000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1

1214213 rows × 9 columns

In [86]:
```
#plantcubes with user offset 32400(9:00) - r3
# 9a.m to 1a.m - day(16 hr)
# 1a.m to 9a.m - night(8 hr)
#If the timestamp is from 10 a.m to 12 a.m, then the dayflag will be set to True
r3['dayflag'] = ((r3.timestamp.dt.hour >= 10 ) & (r3.timestamp.dt.hour <= 23))
#If the timestamp is from 2 a.m to 8 a.m, then the nightflag will be set to True
r3['nightflag'] = ((r3.timestamp.dt.hour >= 2 ) & (r3.timestamp.dt.hour <= 8))
```

In [87]:
```
r3
```

Out[87]:

	timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe	dayflag	nightflag
71901	2022-09-12 14:28:12.493	23.309375	A2_lower	NaN	1662992890	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
71902	2022-09-12 14:28:29.499	23.310000	A2_lower	NaN	1662992890	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
71903	2022-09-12 14:29:46.409	23.410000	A2_lower	NaN	1662992890	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
71904	2022-09-12 14:31:34.861	23.510000	A2_lower	NaN	1662992890	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
71905	2022-09-12 14:32:56.826	23.410000	A2_lower	NaN	1662992890	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
...	...	...	...	...	...	...	...	...	...	...	...
1286109	2022-10-12 15:41:04.543	22.950330	E10	22.710000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286110	2022-10-12 15:47:40.519	22.990000	E10	22.757822	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286111	2022-10-12 15:54:52.564	23.029422	E10	22.810000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286112	2022-10-12 16:05:56.464	23.090000	E10	22.891271	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286113	2022-10-12 16:08:29.464	23.090000	E10	22.910000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False

1214213 rows × 11 columns

In [88]:
```
#plantcubes with user offset 64800(18:00) - r1
# 6p.m to 10a.m - day(16 hr)
# 10a.m to 6p.m - night(8 hr)
#If the timestamp is from 7 p.m to 9 a.m, then the dayflag will be set to True
r1['dayflag'] = ((r1.timestamp.dt.hour >= 19 ) & (r1.timestamp.dt.hour <= 23)) | (r1.timestamp.dt.hour == 0 ) | ((r1.timestamp.dt.hour >= 1 ) & (r1.t
#If the timestamp is from 11 a.m to 17 p.m, then the dayflag will be set to True
r1['nightflag'] = ((r1.timestamp.dt.hour >= 11 ) & (r1.timestamp.dt.hour <= 16))
```

In [89]:
```
r1
```

Out[89]:

	timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe	dayflag	nightflag
0	2022-08-08 12:12:01.877	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
1	2022-08-08 12:12:02.269	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
2	2022-08-08 12:12:30.469	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
3	2022-08-08 12:12:31.183	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
4	2022-08-08 12:14:27.892	23.540000	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
...	...	...	...	...	...	...	...	...	...	...	...
71896	2022-10-12 15:07:42.830	21.540000	A1_upper	21.134459	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
71897	2022-10-12 15:09:24.821	21.571098	A1_upper	21.180000	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
71898	2022-10-12 15:12:56.061	21.635506	A1_upper	21.280000	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
71899	2022-10-12 15:13:10.801	21.640000	A1_upper	21.286023	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
71900	2022-10-12 15:17:00.786	21.640000	A1_upper	21.380000	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True

71901 rows × 11 columns

In [90]:
```
final_r =  r1.append([r3])
final_r
```

Out[90]:

	timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe	dayflag	nightflag
0	2022-08-08 12:12:01.877	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
1	2022-08-08 12:12:02.269	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
2	2022-08-08 12:12:30.469	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
3	2022-08-08 12:12:31.183	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
4	2022-08-08 12:14:27.892	23.540000	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
...	...	...	...	...	...	...	...	...	...	...	...
1286109	2022-10-12 15:41:04.543	22.950330	E10	22.710000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286110	2022-10-12 15:47:40.519	22.990000	E10	22.757822	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286111	2022-10-12 15:54:52.564	23.029422	E10	22.810000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286112	2022-10-12 16:05:56.464	23.090000	E10	22.891271	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286113	2022-10-12 16:08:29.464	23.090000	E10	22.910000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False

1286114 rows × 11 columns

In [91]:
```
daynight = final_r[(final_r.nightflag == True)|(final_r.dayflag == True)]
```

In [92]:
```
daynight
```

Out[92]:

	timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe	dayflag	nightflag
0	2022-08-08 12:12:01.877	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
1	2022-08-08 12:12:02.269	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
2	2022-08-08 12:12:30.469	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
3	2022-08-08 12:12:31.183	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
4	2022-08-08 12:14:27.892	23.540000	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
...	...	...	...	...	...	...	...	...	...	...	...
1286109	2022-10-12 15:41:04.543	22.950330	E10	22.710000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286110	2022-10-12 15:47:40.519	22.990000	E10	22.757822	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286111	2022-10-12 15:54:52.564	23.029422	E10	22.810000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286112	2022-10-12 16:05:56.464	23.090000	E10	22.891271	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286113	2022-10-12 16:08:29.464	23.090000	E10	22.910000	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False

1162138 rows × 11 columns

In [93]:
```
#day temperature
daydf = daynight[daynight.dayflag == True]
nightdf = daynight[daynight.nightflag == True]
```

In [94]:
```
daydf = daydf.round(3)
daydf
```

Out[94]:

	timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe	dayflag	nightflag
258	2022-08-08 19:00:11.016	23.822	A1_upper	22.990	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
259	2022-08-08 19:01:20.009	23.832	A1_upper	23.090	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
260	2022-08-08 19:02:09.003	23.840	A1_upper	23.165	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
261	2022-08-08 19:02:25.007	23.850	A1_upper	23.190	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
262	2022-08-08 19:03:28.995	23.891	A1_upper	23.290	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
...	...	...	...	...	...	...	...	...	...	...	...
1286109	2022-10-12 15:41:04.543	22.950	E10	22.710	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286110	2022-10-12 15:47:40.519	22.990	E10	22.758	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286111	2022-10-12 15:54:52.564	23.029	E10	22.810	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286112	2022-10-12 16:05:56.464	23.090	E10	22.891	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286113	2022-10-12 16:08:29.464	23.090	E10	22.910	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False

913715 rows × 11 columns

In [95]:
```
#for night dataframe- target should be 21
```

In [96]:
```
nightdf = nightdf.round(3)
nightdf
```

Out[96]:

	timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe	dayflag	nightflag
0	2022-08-08 12:12:01.877	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
1	2022-08-08 12:12:02.269	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
2	2022-08-08 12:12:30.469	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
3	2022-08-08 12:12:31.183	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
4	2022-08-08 12:14:27.892	23.540	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
...	...	...	...	...	...	...	...	...	...	...	...
1286029	2022-10-12 08:30:56.055	22.190	E10	22.008	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	False	True
1286030	2022-10-12 08:31:08.056	22.191	E10	22.010	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	False	True
1286031	2022-10-12 08:42:39.018	22.277	E10	22.110	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	False	True
1286032	2022-10-12 08:44:27.014	22.290	E10	22.123	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	False	True
1286033	2022-10-12 08:56:38.045	22.390	E10	22.210	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	False	True

248423 rows × 11 columns

In [97]:
```
#day dataframes target should be 23
```

In [98]:
```
#Target temperature
#Daythreshold = 23
#Nightthreshold = 21
```

In [99]:
```
daydf
```

Out[99]:

	timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe	dayflag	nightflag
258	2022-08-08 19:00:11.016	23.822	A1_upper	22.990	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
259	2022-08-08 19:01:20.009	23.832	A1_upper	23.090	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
260	2022-08-08 19:02:09.003	23.840	A1_upper	23.165	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
261	2022-08-08 19:02:25.007	23.850	A1_upper	23.190	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
262	2022-08-08 19:03:28.995	23.891	A1_upper	23.290	1659961675	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	True	False
...	...	...	...	...	...	...	...	...	...	...	...
1286109	2022-10-12 15:41:04.543	22.950	E10	22.710	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286110	2022-10-12 15:47:40.519	22.990	E10	22.758	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286111	2022-10-12 15:54:52.564	23.029	E10	22.810	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286112	2022-10-12 16:05:56.464	23.090	E10	22.891	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False
1286113	2022-10-12 16:08:29.464	23.090	E10	22.910	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	True	False

913715 rows × 11 columns

In [100]:
```
nightdf
```

Out[100]:

	timestamp	temp_b	plantcube	temp_a	recipe_id	mode	connected	layers	recipe	dayflag	nightflag
0	2022-08-08 12:12:01.877	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
1	2022-08-08 12:12:02.269	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
2	2022-08-08 12:12:30.469	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
3	2022-08-08 12:12:31.183	NaN	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
4	2022-08-08 12:14:27.892	23.540	A1_upper	NaN	1654078815	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	3	False	True
...	...	...	...	...	...	...	...	...	...	...	...
1286029	2022-10-12 08:30:56.055	22.190	E10	22.008	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	False	True
1286030	2022-10-12 08:31:08.056	22.191	E10	22.010	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	False	True
1286031	2022-10-12 08:42:39.018	22.277	E10	22.110	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	False	True
1286032	2022-10-12 08:44:27.014	22.290	E10	22.123	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	False	True
1286033	2022-10-12 08:56:38.045	22.390	E10	22.210	1652357469	0	1	[[{'periods': [{'duration': Decimal('86400'), ...	1	False	True

248423 rows × 11 columns

In [101]:
```
#day temperature average
dayavg = daydf.groupby('plantcube')['plantcube','temp_a','temp_b'].mean(numeric_only = True)
dayavg['deviation_temp_a'] = abs(dayavg['temp_a'] - 23)
dayavg['deviation_temp_b'] = abs(dayavg['temp_b'] - 23)
dayavg
```

Out[101]:

plantcube	temp_a	temp_b	deviation_temp_a	deviation_temp_b
A1_upper	22.933614	23.576361	0.066386	0.576361
A2_lower	NaN	22.925879	NaN	0.074121
A3_lower	23.048853	23.505283	0.048853	0.505283
A4_lower	22.919684	23.105956	0.080316	0.105956
A4_upper	22.884935	23.641495	0.115065	0.641495
B1_lower	22.887987	23.351094	0.112013	0.351094
B1_upper	22.781456	23.606939	0.218544	0.606939
B2_lower	22.830233	NaN	0.169767	NaN
B2_upper	22.806571	23.664461	0.193429	0.664461
B3_lower	23.268113	22.857249	0.268113	0.142751
B3_upper	22.610341	23.464782	0.389659	0.464782
B4_lower	22.951471	23.467516	0.048529	0.467516
B4_upper	22.989072	22.861015	0.010928	0.138985
C1_lower	22.691417	22.780483	0.308583	0.219517
C1_upper	22.897268	23.167060	0.102732	0.167060
C2_upper	21.729656	23.804718	1.270344	0.804718
D1_lower	23.140000	NaN	0.140000	NaN
D1_upper	22.800051	22.958656	0.199949	0.041344
D2_lower	22.789852	23.505557	0.210148	0.505557
D2_upper	22.841881	23.379461	0.158119	0.379461
D3_lower	22.751976	23.169239	0.248024	0.169239

plantcube	temp_a	temp_b	deviation_temp_a	deviation_temp_b
D3_upper	NaN	NaN	NaN	NaN
E10	22.178766	22.652501	0.821234	0.347499
E2	NaN	23.380000	NaN	0.380000
E3	22.645783	23.783231	0.354217	0.783231
E4	22.858623	22.591987	0.141377	0.408013
E5	22.944433	23.194353	0.055567	0.194353
E6	23.023767	23.812422	0.023767	0.812422
E7	22.872850	23.739649	0.127150	0.739649
E8	22.764879	23.437595	0.235121	0.437595
E9	22.974596	23.025042	0.025404	0.025042

In [102]:
```
#night temperature average
nightavg = nightdf.groupby('plantcube')['plantcube','temp_a','temp_b'].mean(numeric_only = True)
nightavg['deviation_temp_a'] = abs(nightavg['temp_a'] - 21)
nightavg['deviation_temp_b'] = abs(nightavg['temp_b'] - 21)
nightavg
```

Out[102]:

plantcube	temp_a	temp_b	deviation_temp_a	deviation_temp_b
A1_upper	21.191034	21.644148	0.191034	0.644148
A2_lower	NaN	21.742154	NaN	0.742154
A3_lower	21.895634	18.898622	0.895634	2.101378
A4_lower	21.552416	21.618612	0.552416	0.618612
A4_upper	21.488378	22.010268	0.488378	1.010268
B1_lower	21.386690	21.953447	0.386690	0.953447
B1_upper	21.398518	21.897604	0.398518	0.897604
B2_lower	21.341271	NaN	0.341271	NaN
B2_upper	21.433509	21.750651	0.433509	0.750651
B3_lower	21.451142	20.795669	0.451142	0.204331
B3_upper	21.297435	21.806363	0.297435	0.806363
B4_lower	21.459729	21.460263	0.459729	0.460263
B4_upper	21.460809	21.417745	0.460809	0.417745
C1_lower	21.642039	21.360280	0.642039	0.360280
C1_upper	21.408385	21.571500	0.408385	0.571500
C2_upper	20.363487	22.304668	0.636513	1.304668
D1_upper	21.350953	21.560337	0.350953	0.560337
D2_lower	21.580573	22.041036	0.580573	1.041036
D2_upper	21.362487	21.655920	0.362487	0.655920
D3_lower	16.642048	21.842917	4.357952	0.842917
D3_upper	NaN	NaN	NaN	NaN

plantcube	temp_a	temp_b	deviation_temp_a	deviation_temp_b
E10	20.850475	21.312555	0.149525	0.312555
E3	21.508410	22.048800	0.508410	1.048800
E4	21.410647	21.294046	0.410647	0.294046
E5	21.486218	21.705656	0.486218	0.705656
E6	21.510542	22.169719	0.510542	1.169719
E7	21.483778	22.184285	0.483778	1.184285
E8	21.404653	21.712514	0.404653	0.712514
E9	22.836543	22.798327	1.836543	1.798327

In [103]:
```
#day target - both sensors - above 0.8
higher_deviation_day = dayavg[(dayavg['deviation_temp_a']> 0.8) & (dayavg['deviation_temp_b']> 0.8)]
higher_deviation_day
```

Out[103]:

plantcube	temp_a	temp_b	deviation_temp_a	deviation_temp_b
C2_upper	21.729656	23.804718	1.270344	0.804718

In [104]:
```
#night temperature - both sensors - above 0.8
higher_deviation_night = nightavg[(nightavg['deviation_temp_a']> 0.8) & (nightavg['deviation_temp_b']> 0.8)]
higher_deviation_night
```

Out[104]:

plantcube	temp_a	temp_b	deviation_temp_a	deviation_temp_b
A3_lower	21.895634	18.898622	0.895634	2.101378
D3_lower	16.642048	21.842917	4.357952	0.842917
E9	22.836543	22.798327	1.836543	1.798327

In [105]:
```
#save the dataframe as image
import pandas as pd
import dataframe_image as dfi
dfi.export(dayavg, 'dayavg1.png')
```

In [107]:
```
#standard deviation
s1 = daydf.groupby('plantcube')['temp_a','temp_b'].std()
s1
```

Out[107]:

plantcube	temp_a	temp_b
A1_upper	0.557507	0.434074
A2_lower	NaN	0.418818
A3_lower	0.469504	0.697424
A4_lower	0.549632	0.424339
A4_upper	0.470653	0.449632
B1_lower	0.483978	0.568192
B1_upper	0.657308	0.438537
B2_lower	4.356286	NaN
B2_upper	0.516825	0.467195
B3_lower	0.355612	0.615042
B3_upper	0.652409	0.637177
B4_lower	0.451698	0.432223
B4_upper	0.332188	0.467866
C1_lower	0.582636	0.662925
C1_upper	3.447102	0.454760
C2_upper	0.740058	0.551239
D1_lower	0.216025	NaN
D1_upper	0.536123	0.620360
D2_lower	0.520078	0.330090
D2_upper	0.517018	0.549583
D3_lower	3.762720	0.466970
D3_upper	NaN	NaN

plantcube	temp_a	temp_b
E10	0.962967	0.793747
E2	NaN	0.000000
E3	0.553428	0.504996
E4	0.502546	0.646016
E5	0.454225	0.504398
E6	0.459054	0.538336
E7	0.486765	0.533245
E8	3.459463	0.521841
E9	0.416595	0.460180

In [108]:
```
s2 = daydf.groupby('plantcube')['temp_a','temp_b'].std()
s2
```

Out[108]:

plantcube	temp_a	temp_b
A1_upper	0.557507	0.434074
A2_lower	NaN	0.418818
A3_lower	0.469504	0.697424
A4_lower	0.549632	0.424339
A4_upper	0.470653	0.449632
B1_lower	0.483978	0.568192
B1_upper	0.657308	0.438537
B2_lower	4.356286	NaN
B2_upper	0.516825	0.467195
B3_lower	0.355612	0.615042
B3_upper	0.652409	0.637177
B4_lower	0.451698	0.432223
B4_upper	0.332188	0.467866
C1_lower	0.582636	0.662925
C1_upper	3.447102	0.454760
C2_upper	0.740058	0.551239
D1_lower	0.216025	NaN
D1_upper	0.536123	0.620360
D2_lower	0.520078	0.330090
D2_upper	0.517018	0.549583
D3_lower	3.762720	0.466970
D3_upper	NaN	NaN
E10	0.962967	0.793747

plantcube	temp_a	temp_b
E2	NaN	0.000000
E3	0.553428	0.504996
E4	0.502546	0.646016
E5	0.454225	0.504398
E6	0.459054	0.538336
E7	0.486765	0.533245
E8	3.459463	0.521841
E9	0.416595	0.460180

In [ ]: