

```

# import packages
import requests
import zipfile
import arcpy
import pprint
import json
import os
import csv
from io import BytesIO
import shutil
import arcpy.mp as mp

# call ndawn data

ndawn_url_2 = ""https://ndawn.ndsu.nodak.edu/table.csv?
station=78&station=111&station=98&station=162&station=174&station=142&
station=164&station=138&station=161&station=9&station=160&station=159&
station=10&station=118&station=56&station=165&station=11&station=12&st
ation=58&station=13&station=84&station=55&station=179&station=7&statio
n=186&station=87&station=14&station=15&station=96&station=191&station=
16&station=201&station=137&station=124&station=143&station=17&station=
85&station=140&station=134&station=18&station=136&station=65&station=1
04&station=99&station=192&station=19&station=129&station=20&station=10
1&station=166&station=178&station=81&station=21&station=97&station=22&
station=75&station=184&station=2&station=172&station=139&station=158&s
tation=23&station=157&station=62&station=86&station=24&station=89&stat
ion=126&station=167&station=93&station=183&station=90&station=25&stati
on=83&station=107&station=156&station=77&station=26&station=155&statio
n=70&station=127&station=144&station=27&station=173&station=132&statio
n=28&station=195&station=185&station=29&station=30&station=154&station
=31&station=187&station=102&station=32&station=119&station=4&station=8
0&station=33&station=59&station=153&station=105&station=82&station=34&
station=198&station=72&station=135&station=35&station=76&station=120&s
tation=141&station=109&station=36&station=79&station=193&station=71&st
ation=37&station=38&station=189&station=39&station=130&station=73&stat
ion=188&station=40&station=41&station=54&station=69&station=194&statio
n=145&station=113&station=128&station=42&station=43&station=103&statio
n=171&station=116&station=196&station=88&station=114&station=3&station
=163&station=200&station=64&station=115&station=168&station=67&station
=175&station=146&station=170&station=197&station=44&station=133&statio
n=106&station=100&station=121&station=45&station=46&station=61&station
=66&station=181&station=74&station=60&station=199&station=125&station=
176&station=177&station=8&station=180&station=204&station=47&station=1
22&station=108&station=5&station=152&station=48&station=151&station=14
7&station=68&station=169&station=49&station=50&station=91&station=182&
station=117&station=63&station=150&station=51&station=6&station=52&sta
tion=92&station=112&station=131&station=123&station=95&station=53&stat
ion=203&station=190&station=57&station=149&station=148&station=202&sta
tion=110&variable=ddavt&year=2023&dtype=daily&quick_pick=30_d&begin_da
te=2023-12-05&end_date=2023-12-05""

```

```

response = requests.get(ndawn_url_2)
rows = response.text.split('\r\n')
reader = csv.reader(rows, delimiter=',')

featureClassName = "stations"

coord_system = arcpy.SpatialReference(4326) # 4326 is the code for
WGS 1984
arcpy.management.CreateFeatureclass("", featureClassName, "POINT",
spatial_reference=coord_system)
columnNames = [x.replace(" ", "_") for x in ["SHAPE@XY"]
+rows[3].split(',')][:-1]
print(columnNames)
arcpy.management.AddField(featureClassName, columnNames[1], "STRING")
arcpy.management.AddField(featureClassName, columnNames[2], "DOUBLE")
arcpy.management.AddField(featureClassName, columnNames[3], "DOUBLE")

arcpy.management.AddField(featureClassName, columnNames[4], "DOUBLE")
arcpy.management.AddField(featureClassName, columnNames[5], "DOUBLE")
arcpy.management.AddField(featureClassName, columnNames[6], "DOUBLE")
arcpy.management.AddField(featureClassName, columnNames[7], "DOUBLE")
arcpy.management.AddField(featureClassName, columnNames[8], "DOUBLE")

with arcpy.da.InsertCursor(featureClassName, columnNames) as cursor:
    for i, line in enumerate(reader):
        if (i>5) and (len(line)>6):

cursor.insertRow([arcpy.Point(float(line[2]),float(line[1])), line[0],
line[1], line[2], line[3], line[4], line[5], line[6], line[7]])

['SHAPE@XY', 'Station_Name', 'Latitude', 'Longitude', 'Elevation',
'Year', 'Month', 'Day', 'Avg_Temp']

# idw interpolation

arcpy.ddd.Idw(
    in_point_features="stations",
    z_field="Avg_Temp",
    out_raster=r"C:\Users\18284\Documents\ArcGIS\Projects\
arc1lab3part2\arc1lab3part2.gdb\Idw_stations1",
    cell_size=0.0172421199999999,
    power=2,
    search_radius="VARIABLE 12",
    in_barrier_polyline_features=None
)

<Result 'C:\\Users\\18284\\Documents\\ArcGIS\\Projects\\
arc1lab3part2\\arc1lab3part2.gdb\\Idw_stations1'>

```

```
# kriging interpolation
```

```
arcpy.ddd.Kriging(  
    in_point_features="stations",  
    z_field="Avg_Temp",  
    out_surface_raster=r"C:\Users\18284\Documents\ArcGIS\Projects\  
arc1lab3part2\arc1lab3part2.gdb\Kriging_stat1",  
    semiVariogram_props="Spherical # # # #",  
    cell_size=0.0172421199999999,  
    search_radius="VARIABLE 12",  
    out_variance_prediction_raster=None  
)
```

```
<Result 'C:\\Users\\18284\\Documents\\ArcGIS\\Projects\\  
arc1lab3part2\\arc1lab3part2.gdb\\Kriging_stat1'>
```

```
# radial basis functions interpolation
```

```
arcpy.ga.RadialBasisFunctions(  
    in_features="stations",  
    z_field="Avg_Temp",  
    out_ga_layer=None,  
    out_raster=r"C:\Users\18284\Documents\ArcGIS\Projects\  
arc1lab3part2\Rbf_stations1",  
    cell_size=0.0172421199999999,  
    search_neighborhood="NBRTYPE=Standard S_MAJOR=3.1834293129079  
S_MINOR=3.1834293129079 ANGLE=0 NBR_MAX=15 NBR_MIN=10  
SECTOR_TYPE=ONE_SECTOR",  
    radial_basis_functions="COMPLETELY_REGULARIZED_SPLINE",  
    small_scale_parameter=None  
)
```

```
<Result ''>
```

```
arcpy.ddd.Kriging(  
    in_point_features="stations",  
    z_field="Avg_Temp",  
    out_surface_raster=r"C:\Users\18284\Documents\ArcGIS\Projects\  
arc1lab3part2\arc1lab3part2.gdb\Kriging_stat2",  
    semiVariogram_props="LinearDrift 0.017242 # # #",  
    cell_size=0.0172421199999999,  
    search_radius="VARIABLE 12",  
    out_variance_prediction_raster=None  
)
```

```
<Result 'C:\\Users\\18284\\Documents\\ArcGIS\\Projects\\  
arc1lab3part2\\arc1lab3part2.gdb\\Kriging_stat2'>
```