

```
In [ ]: import arcpy
import requests
import zipfile
from arcpy.sa import *
import os
```

Download Land Use and Extract the Files

```
In [ ]: #Paths to save land use files
landcoverpath = r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\LandUse\landcover.zip"
landusefolder = r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\LandUse"

#Request land use zipfile from commons
landcoverresp = requests.get(r"https://resources.gisdata.mn.gov/pub/gdrs/data/pub/us_mn_state_dnr/biota_landcover_nlcd_mn_2019/tif_biota_landcover_nlcd_mn_2019.zip")

#Write zipfile to disk
if landcoverresp.status_code == 200:
    with open(landcoverpath, 'wb') as file:
        file.write(landcoverresp.content)
#Extract files from zipfile
with zipfile.ZipFile(landcoverpath, 'r') as zip_ref:
    zip_ref.extractall(landusefolder)
```

Download Elevation

```
In [ ]: #Paths to Save DEMs
elevationpath = r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Elevation\elevation.zip"
elevationfolder = r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Elevation"

elevationresp = requests.get(r"https://resources.gisdata.mn.gov/pub/gdrs/data/pub/us_mn_state_dnr/elev_30m_digital_elevation_model/fgdb_elev_30m_digital_elevation_model.zip")

#Write zipfile to disk
if elevationresp.status_code == 200:
    with open(elevationpath, 'wb') as file:
        file.write(elevationresp.content)

#Extract files from zipfile
with zipfile.ZipFile(elevationpath, 'r') as zip_ref:
    zip_ref.extractall(elevationfolder)
```

Add Dory's House and State Park Points

```
In [ ]: aprx = arcpy.mp.ArcGISProject("CURRENT")
m = aprx.listMaps('Map')[0]
dorypoints = r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Lab2-2.gdb\Dory"
m.addDataFromPath(dorypoints)
```

Create Buffer Around Points

```
In [ ]: arcpy.analysis.Buffer(
    in_features="Dory",
    out_feature_class=r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Lab2-2.gdb\Dory_Buffer",
    buffer_distance_or_field="10 Miles",
    line_side="FULL",
    line_end_type="ROUND",
    dissolve_option="ALL",
    dissolve_field=None,
    method="PLANAR"
)
```

Clip Out Layer Using Points Buffer

```
In [ ]: # Define the input rasters and output location
landuse_raster = r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\LandUse\NLCD_2019_Land_Cover.tif"
output_folder = r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Clipped"
output_database = r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Lab2-2.gdb"
elevation_raster = r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Elevation\elev_30m_digital_elevation_model.gdb\digital_elevation_model_30m"

#Extract section of rasters overlapped by the Dory Buffer
landusebufferraster = arcpy.sa.ExtractByMask(landuse_raster, "Dory_Buffer")
elevationbufferraster = arcpy.sa.ExtractByMask(elevation_raster, "Dory_Buffer")

# Save the clipped raster to output folder
output_landuse = os.path.join(output_folder, "Clipped_LandUse.tif")
output_elevation = os.path.join(output_database, "Clipped_Elevation")
landusebufferraster.save(output_landuse)
elevationbufferraster.save(output_elevation)
```

Create Slope Layer

```
In [ ]: arcpy.ddd.Slope(
    in_raster="\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Lab2-2.gdb\Clipped_Elevation",
    out_raster=r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Lab2-2.gdb\SlopeLayer",
    output_measurement="DEGREE",
    z_factor=1,
    method="PLANAR",
    z_unit="METER",
)
```