#Within 100-meter buffer from water sources waste dumping is not allowed. Identify possible waste dumping sites.

#\*Draw the flowchart

#\*Develop a python program to perform this task

#Add water sources layer

from qgis.core import QgsVectorLayer, QgsField

file\_name = "C:\spatial\_data\data\water\_sources.shp"

water\_sources = QgsVectorLayer(file\_name,"water\_sources", "ogr")

QgsProject.instance().addMapLayer(water\_sources)

#-------------------------------------------------------------------------------

#100m buffer for water sources

features = water\_sources.getFeatures()

print(type(features))

buffers = []

for feature in features:

buffers.append(feature.geometry().buffer(0.1,5))

union\_geometry = QgsGeometry.unaryUnion(buffers)

#Create a QgsVectorLayer from the union geometry

buffer\_layer = QgsVectorLayer("Polygon?crs=EPSG:4326","Union Geometry", "memory")

pr = buffer\_layer.dataProvider()

buffer\_layer.startEditing()

#Define the attributes (if needed)

fields = [QgsField("id", QVariant.Int)]

pr.addAttributes (fields)

#Add a feature with the union geometry

feature = QgsFeature()

feature.setGeometry(union\_geometry)

feature.setAttributes([1])

pr.addFeatures([feature])

buffer\_layer.commitChanges()

# Specify the path to save the shapefile (replace with your desired file path)

output\_shapefile\_path = "C:\spatial\_data\Result\water\_sources\_buffer.shp"

# Use QgsVectorFileWriter to save the layer as a shapefile

error = QgsVectorFileWriter.writeAsVectorFormat(buffer\_layer, "C:\spatial\_data\Result\water\_sources\_buffer.shp", "UTF-8", buffer\_layer.crs(), "ESRI Shapefile")

#Add water\_sources\_buffer

from qgis.core import QgsVectorLayer, QgsField

file\_name = "C:\spatial\_data\Result\water\_sources\_buffer.shp"

water\_sources\_buffer = QgsVectorLayer(file\_name,"water\_sources\_buffer", "ogr")

QgsProject.instance().addMapLayer(water\_sources\_buffer)

#---------------------------------------------------------------------------------

#Add waste\_dumping\_sites

from qgis.core import QgsVectorLayer, QgsField

file\_name = "C:\spatial\_data\data\waste\_dumping\_sites.shp"

waste\_dumping\_sites = QgsVectorLayer(file\_name,"waste\_dumping\_sites", "ogr")

QgsProject.instance().addMapLayer(waste\_dumping\_sites)

#-------------------------------------------------------------------------------

#file\_name = "C:\spatial\_data\data\waste\_dumping\_sites.shp"

#-------------------------------------------------------------------------------

#Like Union Geometry

waste\_dumping\_sites = QgsVectorLayer(file\_name,"possible waste dumping sites", "ogr")

is\_within = QgsField("is\_within", QVariant.String)

waste\_dumping\_sites.startEditing()

waste\_dumping\_sites.dataProvider().addAttributes([is\_within])

waste\_dumping\_sites.updateFields()

#Like within union geometry

waste\_dumping\_sites\_features = waste\_dumping\_sites.getFeatures()

for wf in waste\_dumping\_sites\_features:

contains = union\_geometry.contains(wf.geometry())

waste\_dumping\_sites.changeAttributeValue(wf.id(),waste\_dumping\_sites.fields().indexFromName("is\_within"), f"{str(contains)}")

# Commit the changes

waste\_dumping\_sites.commitChanges()

QgsProject.instance().addMapLayer(buffer\_layer)

QgsProject.instance().addMapLayer(waste\_dumping\_sites)

#-------------------------------------------------------------------------------