

| | | | |
|--------------------|--|---------------------|----------------------|
| Date: | 06/06/2020 | Name: | Abhishek M Shastry K |
| Course: | The Python Mega Course: Build 10 Real World Applications | USN: | 4AL17EC002 |
| Topic: | 1] Application 10: Project Exercise on Building a Geocoder Web Service | Semester & Section: | 6 th 'A' |
| Github Repository: | AbhishekShastry-Courses | | |

SESSION DETAILS

Image of session

```

1 from flask import Flask, render_template, request, send_file
2 from geopy.geocoders import ArcGIS
3 import pandas
4 import datetime
5
6 app=Flask(__name__)
7
8 @app.route("/")
9 def index():
10     return render_template("index.html")
11
12 @app.route('/success-table', methods=['POST'])
13 def success_table():
14     global filename
15     if request.method=="POST":
16         file=request.files['file_name']
17         try:
18             df=pandas.read_csv(file)
19             gc=ArcGIS(scheme='http')
20             df['coordinates']=df['Address'].apply(lambda x: gc.geocode(x).latitude if x != None else "No details")
21             df['longitude']=df['coordinates'].apply(lambda x: gc.geocode(x).longitude if x != None else "No details")
22             df=df.drop("coordinates",1)
23             filename=datetime.datetime.now().strftime("sample_files/%Y-%m-%d-%H-%M-%S-%f"+".csv")
24             df.to_csv(filename,index=None)
25             return render_template("index.html", text=df.to_html(), btn='download.html')
26

```

data_address.csv - Excel

File Home Insert Page Layout Formulas Data Review View Help Tell me what you want to do

Clipboard: Cut, Copy, Paste, Format Painter

Font: Calibri, 11, Bold, Italic, Underline, Text Color, Fill Color

Alignment: Wrap Text, Merge & Center

Number: General, Percentage, Decimal, Fraction

Conditional Formatting, Format as Table, Cell Styles

Insert, Delete, Format Cells

AutoSum, Fill, Clear, Sort & Filter, Find & Select

Address

| ID | Address | Name |
|----|----------------------------------|------|
| 1 | Udupi Karnataka 576101 INDIA | AAA |
| 2 | Mangalore Karnataka 575001 INDIA | BBB |
| 3 | Bangalore Karnataka 560001 INDIA | CCC |
| 4 | Hyderabad Telangana 500001 INDIA | DDD |
| 5 | New Delhi Delhi 110001 INDIA | EEE |

data_address

Super Geocoder

localhost:5000/success-table

Quick Geocoder

Please upload your CSV file. The values containing addresses should be in a column named *address* or *Address*

Browse... Submit

| ID | Address | Name | Latitude | Longitude |
|-----|----------------------------------|------|-----------|-----------|
| 0 1 | Udupi Karnataka 576101 INDIA | AAA | 13.331835 | 74.754564 |
| 1 2 | Mangalore Karnataka 575001 INDIA | BBB | 12.874880 | 74.836373 |
| 2 3 | Bangalore Karnataka 560001 INDIA | CCC | 12.979185 | 77.606623 |
| 3 4 | Hyderabad Telangana 500001 INDIA | DDD | 17.396225 | 78.468764 |
| 4 5 | New Delhi Delhi 110001 INDIA | EEE | 28.623410 | 77.222558 |

Download

What do you want to do with yourfile.csv (411 bytes)?
From: localhost

Open Save Cancel

Report

Application 10: Project Exercise on Building a Geocoder Web Service

- The web application allows the user to upload a **csv file** containing address of different locations and the application processes the data using **geopy** library and produces the downloadable file containing latitude and longitude positions of the given address.
- For uploading and downloading of files **pandas** and **flask** library is used and for web page **flask** library is used.
- **geopy** is a Python 2 and 3 clients for several popular geocoding web services.
- **geopy** makes it easy for Python developers to locate the coordinates of addresses, cities, countries, and landmarks across the globe using third-party geocoders and other data sources.
- **geopy** includes geocoder classes for the OpenStreetMap Nominatim, Google Geocoding API (V3), and many other geocoding services. The full list is available on the Geocoders doc section. Geocoder classes are located in **geopy.geocoders**.
- The **ArcGIS** API for Python is a powerful, modern and easy to use Pythonic library to perform GIS visualization and analysis, spatial data management and GIS system administration tasks that can run both in an interactive fashion, as well as using scripts.
- It enables power users, system administrators and developers to leverage the rich SciPy ecosystem for automating their workflows and performing repetitive tasks using scripts. It integrates well with the **Jupyter** Notebook and enables academics, data scientists, GIS analysts and visualization enthusiasts to share geo-enriched literate programs and reproducible research with others.
- **Flask** is a lightweight WSGI web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around **Werkzeug** and **Jinja** and has become one of the most popular Python web application frameworks.
- The **send_file ()** function under flask library sends the contents of a file to the client. This will use the most efficient method available and configured.