THE ROLE AND IMPORTANCE OF PROGRAMMING (PYTHON) IN MODERN CARTOGRAPHY

A.Kh. Abdullaev[1], U.P. Islamov[2], O.Z. Abdullaev[3], A.S. Avazov[4]
Deputy Director of the Cadastre Agency, Director of the State Enterprise "Geoinnovation"[1]
PhD, Associate Professor, National Research University "Tashkent Institute of Irrigation and Agricultural Mechanization Engineers"[2]
Doctoral Student, National Research University "Tashkent Institute of Irrigation and Agricultural Mechanization Engineers"[3]
Director of Tashkent College of Geodesy and Cartography[4]
https://doi.org/10.5281/zenodo.14760729

Abstract*. The given article discusses in detail the role and importance of Python, one of the programming languages in modern cartography, using the IMRAD method. The convenience of Python, the richness of its open-source libraries, and its effectiveness in automating cartographic processes will be discussed. The results of the study show that Python is an important technological tool in the development of modern cartography and the creation of digital maps.*
Keywords*: modern cartography, Python, code libraries, digital maps and GIS technology.*

It is important to suggest that cartography is an ancient field of science that aims to describe, analyze, and present to the public knowledge about geographic locations and objects. Since prehistoric times, people have created maps to understand and manage the environment. Maps, originally hand-drawn and printed on paper, have acquired a new look due to the development of modern technology, especially computer technology. At present time digital cartography has completely replaced traditional cartography, and this process is constantly evolving.

Digital cartography is based on various software tools and technologies for collecting, analyzing and visualizing data. These technologies are based on geographic information systems (GIS) and programming languages that provide automation and innovative approaches in the field of cartography. In particular, the widespread use of GIS technologies has made it possible to process, analyze and effectively use large volumes of geographic data. In this context, the importance of programming languages in cartography is growing. The Python programming language is a leader in this field and is widely used for efficient management and automation of modern cartographic processes. Python is a programming language popular all over the world due to its simplicity, flexibility and open-source libraries.

However, its libraries and tools provide unique opportunities for creating digital maps, analyzing geographic data and creating interactive maps. In addition, the open-source Python ecosystem creates broad opportunities for research and development in the field of cartography.

Besides that, modern cartography processes such as automatic processing of location and state data of geographic objects, conducting territorial analysis and visualizing GIS data can be performed using Python, which significantly saves not only time but also resources. In particular, powerful Python libraries such as GeoPandas, Folium, Matplotlib, Shapely, and Basemap provide great convenience in the field of cartography. These libraries greatly facilitate working with geographic data and its presentation in various visual tools. At the same time, modern cartographic

needs require speed, the ability to manage large amounts of data and make reliable decisions based on them. Although it is difficult to perform such complex tasks using traditional programs, the Python programming language is becoming one of the main technological tools for solving such problems.

It is important to suggest that the mentioned article analyzes in detail the role and significance of Python programming language in modern cartography, as well as the methods of processing, analyzing and creating digital maps using it. It also discusses the competitiveness of Python tools and their contribution to automation in the field of cartography. In the current digital era, Python's contribution to cartographic processes takes modern cartography to a whole new level, and it is expected to be integrated even more deeply into this field in the future. This study used the IMRAD (Introduction, Methodology, Results and Discussion) method to analyze the role and significance of Python programming language in modern cartography. In the methodological section, a multi-stage approach was used to achieve the objectives of the study. Each stage aims to evaluate the capabilities of Python in the field of cartography through a deeper analysis of the research process and practical approaches.

Moreover, the initial phase of the study included reviewing scientific articles, technical documentation, and practical guides on the use of the Python programming language in cartography, in particular the functionality of open-source libraries and their advantages over other programming languages. The literature review yielded the following information:

The use of Python in geographic information systems (GIS) and cartography.

The role of libraries such as GeoPandas, Folium, Matplotlib, Basemap, and Shapely in cartographic processes. Aspects of Python compared to other programming languages.

The literature review is a key step in defining the scientific basis and practical approaches to the study.

The second phase of the study involved practical experiments. At this stage, several practical examples were developed for analyzing geographic data, visualizing, and creating digital maps using Python. The following practical processes were implemented at this stage:

Loading and analyzing geographic data: GeoPandas was used to load data in Shapefile, GeoJSON, and other formats, and perform basic analysis.

Creating static maps: Different visualizations were created based on geographic data using Matplotlib and GeoPandas.

Interactive Map Development: Interactive maps were developed using the Folium library with markers, regional boundaries, and other elements.

Process Automation: Python programs were created to process map data and automate repetitive processes. Hands-on experiments allowed us to identify Python’s strengths, weaknesses, and technical limitations.

The results of the hands-on experiments were compared with other programming languages and tools to assess Python’s competitiveness. At this stage, the following aspects were considered:

Advantages and disadvantages of Python mapping libraries compared to other software tools (e.g., R, JavaScript, or ArcGIS). Efficiency of processing and automating geographic data using Python. Integrating the library's capabilities with other GIS software (e.g. QGIS, PostGIS, others).

This step is important to analyze how Python has become a universal tool in cartography.

The study also explored Python's potential in modern technologies, particularly machine learning and big data processing. This approach helped to determine what new opportunities Python can create in the field of cartography.

At the final stage of the study, conclusions were made on the role of Python in modern cartography based on the experiments and analyses. Opinions were formed on the advantages, practical approaches and prospects of Python in this area. At the same time, the disadvantages and technical limitations of Python were noted.

This methodology created ample opportunities for the scientific substantiation of the study, a clearer definition of the role of Python in modern cartography and the study of practical aspects. With the help of practical experiments and analyses, the actual effectiveness of Python tools was tested and their suitability for cartographic tasks was determined. Open-source Python libraries are widely used in modern cartography. The main libraries are listed below:

- GeoPandas: This library is specially designed for working with geographic data. It allows you to analyze geographic objects and display them on maps.
- Matplotlib: a convenient and flexible graphical tool for cartographic visualization.
- Folium: a tool for creating dynamic and interactive maps. Supports OpenStreetMap and other sources.
- Basemap: used to create beautiful maps based on geodata.
- Shapely: designed to analyze and manipulate the geometry of geographic objects.

Definitely, with these libraries, users can automate a wide range of cartographic processes, manage and analyze complex data sets. Automating cartographic processes using the Python programming language saves a lot of labor and time. For example, large amounts of GIS (geographical information systems) data can be processed and visualized with just a few lines of code using the GeoPandas library. This eliminates the need for manual processing of geographic data.

```
import geopandas as gpd
import matplotlib.pyplot as plt
# Load data
data = gpd.read_file('world_countries.shp')
# Create a map
data.plot(color='lightblue', edgecolor='black')
plt.title("World Map")
plt.show()
```

This code can quickly generate a world map using just a few lines of code. Creating interactive maps using libraries like Folium is an integral part of modern cartographic practice. Below is a simple example:

```
import folium
# Create a center point and map
map = folium.Map(location=[41.2995, 69.2401], zoom_start=10)
# Add a symbol
folium.Marker([41.2995, 69.2401], popup="Tashkent").add to(map)
# Save the map
map.save("map.html")
```

As a result, the user will be able to use the interactive map and open it in various programs. According to the research results, Python has the following advantages in the field of modern cartography: Convenience: Python's unique, user-friendly syntax makes it easy to learn programming. Open-source tools: Python's rich library repository provides a wide range of possibilities in the field of cartography. Flexibility and automation: allows you to process and automate large amounts of data. However, Python also has its drawbacks. For example, sometimes working with large amounts of GIS data can be relatively slow. In such situations, there is a need for integration with other languages or technologies, such as C++ or Java.

By summarizing it should be mentioned that the Python programming language is considered not only an important tool, but also an indispensable tool in modern cartography. Open-source libraries based on it increase the efficiency of managing, analyzing, and visualizing geographic data. Python has found its place as a key programming tool in interactive mapping, automation, and precise computing processes. In the future, the possibilities of using Python more widely in the areas of cartographic analysis and visualization may be explored. In particular, new Python approaches in the areas of machine learning and big data management will be explored.

REFERENCES

1. McKinney, W. (2012). *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. O'Reilly Media.
2. Harman, A., & Nikolov, D. (2019). *Geospatial Data Analysis with Python*. Packt Publishing.
3. Westra, E. (2013). *Python Geospatial Analysis Cookbook*. Packt Publishing.
4. Fu, P., & Sun, J. (2011). *Web GIS: Principles and Applications*. Esri Press.
5. Python Software Foundation. (2023). *Python Documentation*. https://www.python.org/doc/
6. GeoPandas Documentation. (2023). *GeoPandas: Python tools for geographic data*. https://geopandas.org
7. Folium Documentation. (2023). *Folium: Python Data, Leaflet.js Maps*. https://python-visualization.github.io/folium/
8. Hunter, J. D. (2007). "Matplotlib: A 2D Graphics Environment." *Computing in Science & Engineering*, 9(3), 90-95.
9. PostGIS Development Team. (2023). *PostGIS Manual*. https://postgis.net/
10. Esri. (2023). *What is GIS?*. https://www.esri.com/en-us/what-is-gis/overview
11. Shapely Documentation. (2023). *Shapely: Geometric objects, predicates, and operations*. https://shapely.readthedocs.io/
12. Hegedus, T. (2020). "Automating GIS Processes with Python." *GIScience Journal*, 22(1), 56-73.
13. Ramsay, A. (2015). *Learning Geospatial Analysis with Python*. Packt Publishing.
14. Toms, R. (2019). "The Role of Python in Modern Geospatial Development." *Cartography and Geographic Information Science*, 46(3), 220-229.
15. OpenStreetMap. (2023). *About OpenStreetMap*. https://www.openstreetmap.org/about
16. PyProj Documentation. (2023). *PyProj: A Python interface to PROJ*. https://pyproj4.github.io/pyproj/stable/
17. Van Rossum, G., & Drake, F. L. (2009). *The Python Language Reference Manual*. Network Theory Ltd.

18. Basemap Documentation. (2023). *Basemap Toolkit for Matplotlib*. https://matplotlib.org/basemap/
19. QGIS Development Team. (2023). *QGIS Documentation*. https://qgis.org
20. Goldberg, D. W. (2015). "Advances in Geocoding Research with Python." *International Journal of Geographical Information Science*, 29(4), 753-768.