УДК 004.43

**В. В. Лендел**
**Л. В. Ібрагімова**

# PYTHON PROGRAMMING

Vinnytsia National Technical University

***Анотація*** *У статті дано визначення поняття python, наведено основні переваги та недоліки мови python, історія мови, особливості, фреймворки та вплив інших мов на python.*
**Ключові слова:** пайтон, фреймворк, особливості.

**Abstract** *The article defines the concept of python, presents the main advantages and disadvantages of python, language history, features, frameworks and the impact of other languages on python.*
**Keywords:** python, framework, features.

Python is an interpreted high-level object-oriented programming language with strict dynamic typing. It was developed by Guido van Rossum in 1990. High-level data structures together with dynamic semantics and dynamic linking make it attractive for rapid software development as well as a means of combining existing components. Python supports modules and module packages, which promotes modularity and code reuse. The Python interpreter and standard libraries are available in both compiled and source form on all major platforms. The Python programming language supports several programming paradigms including object-oriented, procedural, functional, and aspect-oriented. [1]

Python has many advantages:

- Pure syntax
- Portability of programs
- The standard distribution has a large number of useful modules
- The ability to use Python in dialog mode
- The standard distribution has a simple but at the same time quite powerful development environment called IDLE and written in Python
- Open source

Python has efficient, high-level data structures and a simple but effective approach to object-oriented programming. Python sleek syntax, dynamic type processing, and interpreted language make it ideal for scripting and rapid application development in many industries on most platforms. [2]

The development of the Python language was started in the late 1980s by Guido van Rossum of Dutch CWI Institute. Amoeba's distributed OS required an extensible scripting language, and Guido began writing Python at leisure borrowing some of the work for ABC. In February 1991, Guido published the source in the alt.sources newsgroup. The language began to spread freely over the Internet and other programmers liked it. Since 1991 Python has been entirely object-oriented. On December 3, 2008, after lengthy testing the first version of Python 3000 (or Python 3.0, also abbreviated Py3k) was released. Python 3000 fixes many architecture flaws with maximum (but not complete) compatibility with older versions. Python version 3 is currently supported. [3]

Python also has disadvantages such as:

1. **Low speed**. Python like many other interpreted languages that do not use, for example, JIT compilers has a common drawback – relatively low program execution speed.
2. **Lack of static typing**. The lack of static typing is not so much a defect of the interpreter as the choice of the language developer. The fact is that Python has adopted the so-called "duck typing". Because of this, the types of values passed are not available at compile time. Lack of static typing is also one of the main reasons for low performance.
3. **Global Interpreter Lock (GIL)**. GIL (Global Interpreter Lock) is a problem common to CPython, Stackless, and PyPy but absent in Jython and IronPython. While working the main Python interpreter constantly uses a large amount of thread-hazardous data. These are mostly dictionaries that store object attributes. To avoid the destruction of this data when co-modified from different threads before the execution

of several instructions (default 100), the interpreter thread captures the GIL, and then releases.[4]

Python has a large number of useful frameworks.

Framework is an infrastructure of software solutions that facilitates the development of complex systems. Simplified, this infrastructure can be considered as a kind of complex library but it has a number of limitations that set the rules for creating a project structure and writing code.

Software framework is a ready-to-use set of software solutions including the design, logic and basic functionality of the system or subsystem. Accordingly, the software framework may also include utilities, some code libraries, scripts, and generally anything that makes it easier to create and combine different components of large software or quickly create a ready-made and not voluminous software product. The construction of the final product is usually based on a single API.

There are 7 main frameworks in Python:

- Django
- CherryPy
- Flask
- TurboGears
- Pyramid
- Web2Py
- Bottle

The most popular and best is Django. Django is a high-level open Python framework (software framework) for web development. It was named after jazzman Django Reinhardt (according to the musical tastes of one of the founders of the project). The site on Django is built from one or more parts, which are recommended to be made modular. This is one of the significant architectural differences of this framework from some others. The initial development of Django as a tool for new resources has had a strong impact on its architecture: it provides a number of tools that help in the rapid development of informational websites. For example, the developer does not need to create controllers and pages for the administrative part of the site, Django has a built-in content management module that can be included in any site made on Django, and which can manage multiple sites on one server. The administrative module allows to create, modify and delete any objects filling the site, logging all actions as well as provides an interface for managing users and groups (with the assignment of rights). The Django distribution also includes programs for the comment system, RSS and Atom syndication, "static pages" (which can be managed without having to write controllers and displays), URL redirection and more.

Opportunities:

- Object-relational mapping (ORM)
- Automatic construction of the interface for administration
- Elegant URLs
- Convenient system of templates
- Flexible caching subsystem
- Simple internationalization

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТИ


1. Python [Електронний ресурс]. – Режим доступу: https://www.python.org/
2. Django [Електронний ресурс]. – Режим доступу: https://www.djangoproject.com/
3. Django in Python [Електронний ресурс]. – Режим доступу:https://tutorial.djangogirls.org/uk/django/



***Лендел Вадим Віталійович*** – ст. гр. 1ІСТ-20б, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м. Вінниця, e-mail: lendelvadim50@gmail.com.

***Науковий керівник – Ібрагімова Людмила Володимирівна***, старший викладач кафедри іноземних мов, Вінницький національний технічний університет, Вінниця, e-mail: milatvin@ukr.net

***Lendel Vadym*** – Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: lendelvadim50@gmail.com.

***Ibrahimova Liudmyla V.*** — Senior Lecture, Chair of Foreign Languages, Vinnytsia National Technical University, Vinnytsia, e-mail: milatvin@ukr.net