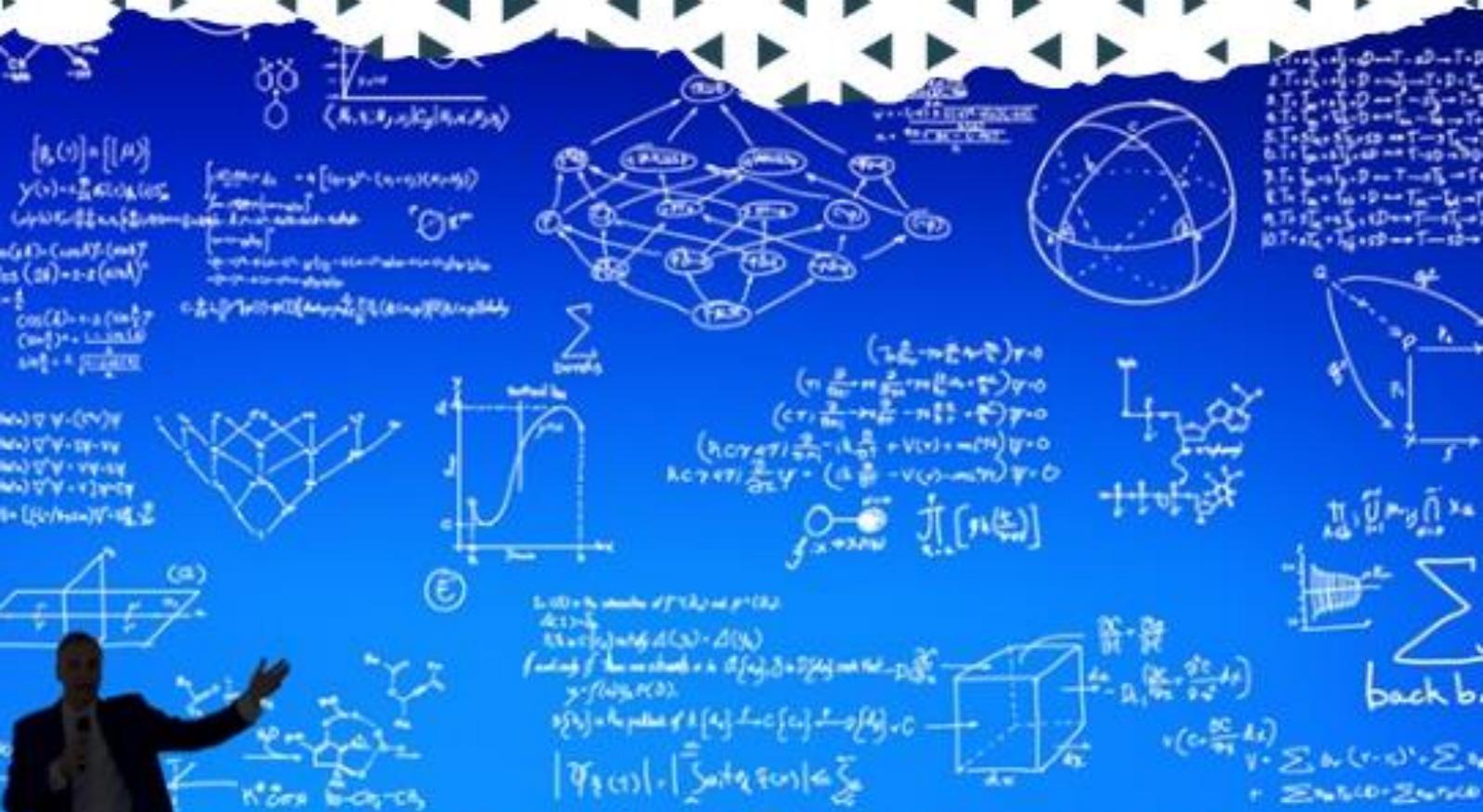




INNOVATIVE WORLD
Ilmiy tadqiqotlar markazi



ZAMONAVIY ILM-FAN VA TA'LIM: MUAMMO VA YECHIMLAR ILMIY-AMALIY KONFERENSIYA



Google Scholar doi zenodo OpenAIRE



+998335668868

<https://innoworld.net>

2025



**«INNOVATIVE WORLD» ILMIY TADQIQOTLARNI QO'LLAB-
QUVVATLASH MARKAZI**

**«ZAMONAVIY ILM-FAN VA TADQIQOTLAR: MUAMMO VA
YECHIMLAR» NOMLI 2025-YIL № 5-SONLI ILMIY, MASOFAVIY,
ONLAYN KONFERENSIYASI**

**ILMIY-ONLAYN KONFERENSIYA TO'PLAMI
СБОРНИК НАУЧНЫХ-ОНЛАЙН КОНФЕРЕНЦИЙ
SCIENTIFIC-ONLINE CONFERENCE COLLECTION**

Google Scholar



ResearchGate

zenodo



ADVANCED SCIENCE INDEX



Directory of Research Journals Indexing

www.innoworld.net

O'ZBEKISTON-2025

PYTHON DASTURLASH TILI: ASOSIY TUSHUNCHALALAR

Tojimatov Israil Nurmatovich

Farg'ona davlat universiteti amaliy matematika va informatika kafedrası
katta o'qituvchisi, israiltojimatov@gmail.com

Kamolova Rayhona Foziljon qizi

Farg'ona davlat universiteti Amaliy matematika yo'nalishi 3 – kurs talabasi
Kamolovarayhona14@gmail.com

Anotatsiya: Ushbu maqolada Python dasturlash tilining asosiy tushunchalari, uning sintaksis tuzilishi, ma'lumot turlari, operatorlar, funksiyalar va modullardan foydalanishning nazariy hamda amaliy jihatlari o'rganiladi. Pythonning soddaligi, funksional qulayligi va keng kutubxonalar ekotizimi uni zamonaviy dasturlash jarayonining eng ommabop vositalaridan biriga aylantirgan. Tadqiqot davomida tilning asosiy imkoniyatlari tahlil qilinib, ularning dastur yaratishdagi samaradorligi yoritilgan. Maqola Python tilini o'rganishni boshlovchilar uchun nazariy asos bo'lib xizmat qiladi hamda uni amaliy jarayonda qo'llashning metodik yo'nalishlarini ko'rsatib beradi.

Maqolada Python dasturlash tilining asosiy tushunchalari, jumladan o'zgaruvchilar (variables), ma'lumot turlari (data types), operatorlar (operators), shart operatorlari (conditional statements), sikllar (loops), funksiyalar (functions), modullar (modules), paketlar (packages), sintaksis (syntax) va dastur tuzilishi (program structure) mazmunan yoritilgan. Har bir tushuncha uchun kod misollari, amaliy tavsiflar va ularning dasturiy ekvivalentlari keltirilgan. Shuningdek, maqolada Python tilining sintaksisi, dastur tuzilishi va amaliy dasturlash jarayonida ularning qo'llanilishi tahlil qilingan. Tadqiqot Python dasturlash tilini o'rganayotgan yangi boshlovchilar va amaliyotchi dasturchilar uchun nazariy va metodik asos vazifasini bajaradi. Ushbu maqolada har bir komponentning amaliy ahamiyati, dastur samaradorligi va dasturiy kodni tizimli shaklda yaratish imkoniyatlari ham yoritilgan.

Annotation: This article examines the fundamental concepts of the Python programming language, including its syntax structure, data types, operators, functions, and the use of modules, from both theoretical and practical perspectives. Python's simplicity, functional convenience, and extensive library ecosystem have made it one of the most popular tools in modern programming. During the study, the main features of the language were analyzed, highlighting their effectiveness in program development. The article serves as a theoretical foundation for beginners learning Python and demonstrates methodological approaches to applying it in practical programming.

The article covers the main concepts of the Python programming language, including variables, data types, operators, conditional statements, loops, functions, modules, packages, syntax, and program structure. For each concept, code examples, practical descriptions, and their programming



equivalents are provided. Furthermore, the article analyzes Python's syntax, program structure, and their application in practical programming processes. The study serves as a theoretical and methodological foundation for beginners and practicing programmers learning Python. It also highlights the practical significance of each component, the efficiency of programs, and the possibility of creating structured and systematic code.

Аннотация: В данной статье рассматриваются основные концепции языка программирования Python, включая его синтаксическую структуру, типы данных, операторы, функции и использование модулей, с теоретической и практической точки зрения. Простота Python, функциональная удобство и широкая экосистема библиотек сделали его одним из самых популярных инструментов в современном программировании. В ходе исследования были проанализированы основные возможности языка, подчеркнута их эффективность при создании программ. Статья служит теоретической основой для начинающих изучающих Python и демонстрирует методологические подходы к его применению в практическом программировании.

Статья охватывает основные концепции языка программирования Python, включая переменные, типы данных, операторы, условные конструкции, циклы, функции, модули, пакеты, синтаксис и структуру программы. Для каждой концепции приведены примеры кода, практические описания и их программные эквиваленты. Кроме того, в статье анализируются синтаксис Python, структура программы и их применение в практическом программировании. Исследование служит теоретической и методической основой для начинающих и практикующих программистов, изучающих Python. Также подчеркивается практическое значение каждого компонента, эффективность программ и возможность создания структурированного и систематизированного кода.

Kalit so 'zlar: o'zgaruvchilar (variables), ma'lumot turlari (data types), operatorlar (operators), shart operatorlari (conditional statements), sikllar (loops), funksiyalar (functions), modullar (modules), paketlar (packages), sintaksis (syntax), dastur tuzilishi (program structure), amaliy dasturlash (practical programming)

Ключевые слова: переменные (variables), типы данных (data types), операторы (operators), условные операторы (conditional statements), циклы (loops), функции (functions), модули (modules), пакеты (packages), синтаксис (syntax), структура программы (program structure), практическое программирование (practical programming)

Keywords: variables, data types, operators, conditional statements, loops, functions, modules, packages, syntax, program structure, practical programming

Kirish:Bugungi kunda dasturlash tillari hayotimizning turli sohalarida keng qo'llanilmoqda. Python — sodda sintaksisga ega, oson o'qiladigan va ko'p platformalarda ishlay oladigan universal dasturlash tili hisoblanadi. U sun'iy intellekt, ma'lumotlarni tahlil qilish, veb-dasturlash, avtomatlashtirish va ilmiy hisoblashlar kabi sohalarida samarali qo'llaniladi.

Maqolaning maqsadi — Python dasturlash tilining asosiy tushunchalarini tizimli tarzda yoritish va ularning dastur yaratish jarayonidagi amaliy ahamiyatini tushuntirish. Tadqiqot nafaqat boshlovchilar, balki amaliy dasturchilar uchun ham foydali bo'lib, ularning bilimlarini mustahkamlashga xizmat qiladi.

1.O'zgaruvchilar (Variables)

O'zgaruvchilar — ma'lumotlarni saqlash uchun ajratilgan nomlangan joydir. Python'da o'zgaruvchi yaratish uchun faqat nom berish va qiymat tayinlash kifoya qiladi:

```
File Edit Format Run Options Window Help
```

```
x = 10
name = "Rayhona"
```

2.Ma'lumot turlari (Data Types)

Python'da asosiy ma'lumot turlari:

- int — butun sonlar
- float — o'nlik sonlar
- str — matn
- bool — mantiqiy qiymatlar (True/False)

Shuningdek, Python list, tuple, dict kabi kompleks tuzilmalarni ham qo'llab-quvvatlaydi.

3.Operatorlar (Operators)

Python operatorlari arifmetik, taqqoslash, mantiqiy va tayinlash operatorlariga bo'linadi:

```
File Edit Format Run Options Window Help
```

```
a = 5
b = 2
c = a + b # arifmetik operator|
```

4.Shart operatorlari (Conditional Statements)

Shart operatorlari dasturga qaror qabul qilish imkonini beradi:

```
File Edit Format Run Options Window Help
```

```
if a > b:
    print("a katta")
else:
    print("b katta")|
```

5. Sikllar (Loops)

Sikllar biror amallarni takrorlash uchun ishlatiladi:

```
File Edit Format Run Options Window Help
```

```
for i in range(5):
    print(i)|
```

6. Funksiyalar (Functions)



Funksiyalar kodni modulga ajratib, takrorlanuvchi amallarni soddalashtiradi:

```
File Edit Format Run Options Window Help
```

```
def greet(name):
    print("Hello", name)

greet("Rayhona")
```

7. Modullar va Paketlar (Modules & Packages)

Python'da kodni modul va paketlarga ajratish dasturiy loyihani boshqarishni osonlashtiradi:

```
File Edit Format Run Options Window Help
```

```
import math
print(math.sqrt(16))
```

8. Sintaksis va Dastur Tuzilishi (Syntax & Program Structure)

Python sintaksisi soddaligi va o'qiluvchanligi bilan mashhur. Dastur tuzilishi kodni modulga ajratilgan, tizimli shaklda yaratish imkonini beradi, bu esa katta loyihalarda samaradorlikni oshiradi.

Bir misolni korib chiqsak misol Python tilida dastur yaratishda **nazariy va amaliy bilimlarni birlashtirishni** ko'rsatadi. Dastur yangi boshlovchilar va amaliyotchi dasturchilar uchun **tizimli fikrlash, kodni qayta ishlatish, modul va funksiyalardan foydalanish** bo'yicha mustahkam asos yaratadi. Shu orqali dasturchi **ma'lumotlarni boshqarish, hisob-kitob qilish va natijalarni tartibli chiqarishni** osonlik bilan amalga oshira oladi.

2. O'zgaruvchilar va ma'lumot turlari (Variables & Data Types)

```
name = "Rayhona" # str
age = 25 # int
height = 1.68 # float
is_student = True # bool
```

3. Operatorlar va shart operatorlari (Operators & Conditional Statements)

```
bmi = age / (height ** 2) # BMI formula
```

```
if bmi < 18.5:
```

```
    bmi_status = "Underweight"
```

```
elif 18.5 <= bmi < 25:
```

```
    bmi_status = "Normal weight"
```

```
else:
```

```
    bmi_status = "Overweight"
```

4. Funksiya (Function) yaratish

```
def student_report(name, age, height, is_student, bmi_status):
```

```
    print("==== Student Report ====")
```

```
    print(f"Name: {name}")
```

```
    print(f"Age: {age}")
```

```
    print(f"Height: {height} m")
```



```
print(f"Student: {is_student}")
print(f"BMI Status: {bmi_status}")
```

5. Sikl (Loop) orqali amaliy dasturlash (Practical Programming)

```
print("Random grades for 5 subjects:")
grades = []
for i in range(5):
    grade = random.randint(50, 100) # tasodifiy ball
    grades.append(grade)
    print(f"Subject {i+1}: {grade}")
```

6. Funksiyani chaqirish va natijani ko'rsatish

```
student_report(name, age, height, is_student, bmi_status)
```

7. Moduldan foydalanish: Circle area hisoblash

```
radius = 3
circle_area = math.pi * radius**2
print(f"Circle area with radius {radius}: {circle_area}")
```

File Edit Shell Debug Options Window Help

```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
===== RESTART: C:/Users/asr_tech/Documents/suniy_intektekt/1.py =====
Random grades for 5 subjects:
Subject 1: 71
Subject 2: 56
Subject 3: 74
Subject 4: 95
Subject 5: 82
===== Student Report =====
Name: Rayhona
Age: 25
Height: 1.68 m
Student: True
BMI Status: Underweight
Circle area with radius 3: 28.274333882308138
>>> |
```

Xulosa (Conclusion)

Ushbu maqola Python dasturlash tilining **asosiy tushunchalarini** nazariy va amaliy jihatdan yoritadi. Tadqiqot davomida quyidagi jihatlar ko'rib chiqildi:

1. **O'zgaruvchilar va ma'lumot turlari (Variables & Data Types):** dastur ma'lumotlarni turli formatlarda saqlash va ulardan samarali foydalanish imkonini beradi.
2. **Operatorlar va shart operatorlari (Operators & Conditional Statements):** dasturga qaror qabul qilish va shartlarga mos natijalarni chiqarish imkonini yaratadi.

3. **Sikllar (Loops):** kodni takrorlanuvchi vazifalar uchun samarali ishlatish, masalan, tasodifiy sonlar yoki ballar yaratish imkonini beradi.
4. **Funksiyalar (Functions):** dastur tuzilishini tartibga soladi, kodni qayta ishlatish va modul sifatida ishlatish imkonini beradi.
5. **Modullar va paketlar (Modules & Packages):** Python tilining keng kutubxonalar ekotizimi orqali qo'shimcha funksiyalar va amaliy imkoniyatlar yaratadi.
6. **Sintaksis va dastur tuzilishi (Syntax & Program Structure):** kodni toza, tartibli va xatolardan holi yozish imkonini beradi, bu esa amaliy dasturlashda muhim ahamiyatga ega.

Maqola Python dasturlash tilini o'rganayotgan yangi boshlovchilar va amaliyotchi dasturchilar uchun **nazariy va metodik asos** vazifasini bajaradi. U dastur yaratish jarayonida **ma'lumotlarni boshqarish, hisob-kitoblar qilish, funksiyalar va modul imkoniyatlaridan foydalanish, natijalarni tizimli chiqarish** kabi ko'nikmalarni shakllantirishga yordam beradi. Shu orqali Python tilining zamonaviy dasturlashdagi amaliy ahamiyati va samaradorligi yoritiladi.

Foydalanilgan adabiyotlar

1. Lutz, M. Learning Python. O'Reilly Media, 2013.
2. Sweigart, A. Automate the Boring Stuff with Python. No Starch Press, 2015.
3. Beazley, D. Python Cookbook. O'Reilly Media, 2013.
4. Downey, A. Think Python: How to Think Like a Computer Scientist. Green Tea Press, 2015.
5. Python Software Foundation. Python Documentation.
<https://docs.python.org/3/>
6. Pilgrim, M. Dive Into Python 3. Apress, 2009.
7. Hetland, M. Beginning Python: From Novice to Professional. Apress, 2017.
8. Zelle, J. Python Programming: An Introduction to Computer Science. Franklin, Beedle & Associates, 2010.

