

**\_\_INIT\_\_ METODINING AHAMIYATI VA MISOLLAR BILAN MUSTAQIL TAHLIL****INDEPENDENT ANALYSIS OF THE IMPORTANCE OF THE \_\_INIT\_\_ METHOD  
AND EXAMPLES****САМОСТОЯТЕЛЬНЫЙ АНАЛИЗ ЗНАЧЕНИЯ МЕТОДА \_\_INIT\_\_ И  
ПРИМЕРОВ****Mirsaid Yusupov Abdulaziz o'g'li**Farg'ona davlat universiteti amaliy matematika va informatika kafedrası o'qituvchisi  
[mirsaidbeks@gmail.com](mailto:mirsaidbeks@gmail.com)**Kamolova Rayhona Foziljon qizi**Farg'ona davlat universiteti Amaliy matematika yo'nalishi 3 - kurs talabasi  
[Kamolovarayhona14@gmail.com](mailto:Kamolovarayhona14@gmail.com)<https://doi.org/10.5281/zenodo.17959542>

**Annotatsiya:** Ushbu maqolada Python dasturlash tilida obyektga yo'naltirilgan dasturlash (OOP) doirasida ishlatiladigan `__init__` metodining ahamiyati nazariy va amaliy jihatdan tahlil qilinadi. `__init__` metodi Python klasslarida konstruktorga javob beruvchi maxsus metod bo'lib, obyekt yaratilganda avtomatik chaqiriladi va uning atributlarini boshlang'ich qiymat bilan initsializatsiya qiladi. Maqolada metodning vazifasi, ishlash prinsipi va turli amaliy misollar orqali uni qo'llash usullari batafsil yoritilgan. Tadqiqot yangi boshlovchi va amaliyotchi dasturchilar uchun foydali bo'lib, OOP konseptlarini chuqurroq tushunishga yordam beradi.

**Annotation:** This paper provides a theoretical and practical analysis of the significance of the `__init__` method used in object-oriented programming (OOP) in the Python programming language. The `__init__` method is a special constructor-like method in Python classes that is automatically invoked when an object is created, initializing the object's attributes with their initial values. The article thoroughly explores the method's purpose, operating principles, and various practical applications through examples. This study is valuable for both novice and experienced programmers, helping them gain a deeper understanding of OOP concepts.

**Аннотация:** В данной статье проводится теоретический и практический анализ значения метода `__init__`, используемого в объектно-ориентированном программировании (ООП) на языке Python. Метод `__init__` является специальным методом, выполняющим функцию конструктора в классах Python: он автоматически вызывается при создании объекта и инициализирует его атрибуты начальными значениями. В статье подробно рассматриваются назначение метода, принципы его работы и различные практические примеры применения. Исследование будет полезно как начинающим, так и опытным программистам, способствуя более глубокому пониманию концепций ООП.

**Kalit so'zlar:** `__init__` metodi (`__init__` method), Konstruktori (Constructor), Klass (Class), Obyekt (Object), Atributlar (Attributes), Self (self), Initsializatsiya (Initialization), Inkapsulyatsiya (Encapsulation), Python OOP, Amaliy misollar (Practical examples), Sintaksis (Syntax), Dastur tuzilishi (Program structure)

**Keywords:** `__init__` method, Constructor, Class, Object, Attributes, Self, Initialization, Encapsulation, Python OOP, Practical examples, Syntax, Program structure

**Ключевые слова:** Метод `__init__` (`__init__` method), Конструктор (Constructor), Класс (Class), Объект (Object), Атрибуты (Attributes), Self, Инициализация (Initialization), Инкапсуляция (Encapsulation), Python OOP, Практические примеры (Practical examples), Синтаксис (Syntax), Структура программы (Program structure)

**Kirish:** Obyektga yo'naltirilgan dasturlash (OOP) zamonaviy dasturlashning asosiy paradigmasidir. Python tilida OOP imkoniyatlarini amalga oshirishda `__init__` metodi markaziy rol o'ynaydi. Bu metod orqali har bir obyektning boshlang'ich atributlari belgilanadi, kodni tartibli, modul va qayta ishlatishga mos qilish imkonini beradi. Shu bois, `__init__` metodini tushunish va amalda qo'llash Python dasturchilari uchun muhim ko'nikmadir.

**Introduction:** Object-oriented programming (OOP) is a fundamental paradigm in modern programming. In Python, the `__init__` method plays a central role in implementing OOP capabilities. This method allows the initialization of each object's attributes, ensuring that code remains organized, modular, and reusable. Therefore, understanding and effectively applying the `__init__` method is an essential skill for Python programmers.

**Введение:** Объектно-ориентированное программирование (ООП) является основной парадигмой современного программирования. В Python метод `__init__` играет центральную роль в реализации возможностей ООП. Этот метод позволяет инициализировать атрибуты каждого объекта, обеспечивая организованность, модульность и повторное использование кода. Поэтому понимание и эффективное применение метода `__init__` является важным навыком для программистов на Python.

### 1 `__init__` metodining vazifasi

`__init__` — Python klasslari ichidagi maxsus metod bo'lib, quyidagi funksiyalarni bajaradi:

- Obyekt yaratilishi bilan avtomatik chaqiriladi
- Obyekt atributlariga boshlang'ich qiymat beradi
- Konstruktor sifatida obyektни dastlabki holatga tayyorlaydi
- `self` orqali obyektning o'ziga murojaat qiladi
- Kodni tartibli va modulli qilishga yordam beradi

Mazkur metod mavjud bo'lmasa ham klassdan obyekt yaratish mumkin, biroq obyektning atributlarini belgilash keyinchalik qo'lda amalga oshiriladi va bu xatolarga olib kelishi mumkin.

### 2. `__init__` metodining sintaksisi

Python tilida klassning umumiy sintaksisi quyidagicha:

```
class ClassName:
    def __init__(self,param1,param2):
        self.param1=param1
        self.param2=param2
```

- `self.param1` — obyekt ichidagi atribut
- `param1` — tashqi qiymat
- Har bir obyektning atributlari alohida saqlanadi

### Oddiy konstruktordan foydalanish

```
class Student:
    def __init__(self,name,age):
```

```

self.name=name
self.age=age

def show(self):
    print(f"Talaba:{self.name},Yosh: {self.age}")
s = Student("Rayhona",20)
s.show()

```

Bu misolda Student obyektining name va age atributlari darhol initsializatsiya qilinadi.

### **Inkapsulyatsiyada \_\_init\_\_ metodining o'ri**

```

class Student:
    def __init__(self,name,age):
        self.name=name
        self.age=age

    def show(self):
        print(f"Talaba:{self.name},Yosh: {self.age}")
s = Student("Rayhona",20)
s.show()

```

D

Bu yerda \_\_balance tashqi aralashuvdan himoyalangan.

### **Meros olish va super() funksiyasi**

```

class Animal
    def __init__(self,name):
        self.name=name

def dog(Animal):
    def __init__(elf,name,breed):
        super().__init__(name)
        self.breed=breed

```

super() orqali ota-klass konstruktori chaqirilgan.

**\_\_init\_\_** metodi:

- obyekt yaratishni tartibga soladi;
- atributlarga to'g'ri boshlang'ich qiymat berilishini ta'minlaydi;
- xatoliklar ehtimolini kamaytiradi;

Masalan: \_\_init\_\_ metodi mavzusi uchun sodda va umumiy misol:

# Car klassi misoli

```

class Car:
    def __init__(self, brand, model, year):
        # __init__ metodi obyekt yaratilganda atributlarni avtomatik initsializatsiya qiladi
        self.brand = brand

```

```
self.model = model
self.year = year
```

```
def car_info(self):
    # Obyekt atributlarini chiqarish
    return f"{self.brand} {self.model}, {self.year}"
```

```
# Obyekt yaratish
my_car = Car("Toyota", "Corolla", 2022)
print(my_car.car_info()) # Natija: Toyota Corolla, 2022
```

```
# Yana bir obyekt
another_car = Car("Honda", "Civic", 2020)
print(another_car.car_info()) # Natija: Honda Civic, 2020
```

1. Car klassida `__init__` metodi ishlatiladi, bu metod obyekt yaratilganda avtomatik chaqiriladi.
2. Har bir obyekt uchun atributlar (brand, model, year) alohida initsializatsiya qilinadi.
3. `car_info()` metodi yordamida obyekt ma'lumotlarini chiqaramiz.

Shu misol orqali siz `init` metodining asosiy vazifasi va ishlash prinsipini tushunasiz: obyekt yaratilganda avtomatik ravishda atributlarni belgilash va obyektни tayyor holatga keltirish.

**Xulosa:** Yuqoridagi tahlillar shuni ko'rsatadiki, `__init__` metodi Python tilidagi OOP tizimining markaziy bo'g'ini hisoblanadi. Obyektlarning boshlang'ich holatini aniqlash, atributlar bilan ishlash, inkapsulyatsiya va meros mexanizmlarini yo'lga qo'yish aynan ushbu metod yordamida amalga oshiriladi. Shunday ekan, Python dasturchisi uchun `__init__` metodining ahamiyatini to'g'ri anglash va uni amaliyotda samarali qo'llash dasturlash malakasini sezilarli darajada oshiradi.

- OOP konsepsiyasining barcha asosiy elementlarini qo'llab-quvvatlaydi (inkapsulyatsiya, meros, polimorfizm).

Katta loyihalarda `__init__` metodidan oqilona foydalanish kodni massiv, tartibli va oson boshqariladigan holga keltiradi. Shuningdek, kodni qayta ishlatish imkonini kengaytiradi, chunki bir marta yozilgan klass turli obyektlar yaratishda takroran xizmat qiladi.

### Adabiyotlar, References, Литературы:

1. Lutz M. *Learning Python*. O'Reilly Media, 2013.
2. Sweigart A. *Automate the Boring Stuff with Python*. No Starch Press, 2019.
3. Van Rossum G., Drake F. L. *Python Tutorial*. Python Software Foundation.
4. Python rasmiy hujjatlari – *Classes and OOP*:  
<https://docs.python.org/3/tutorial/classes.html>
5. Zelle J. *Python Programming: An Introduction to Computer Science*. Franklin, Beedle & Associates, 2010.
6. Summerfield M. *Programming in Python 3*. Addison-Wesley, 2009.
7. Ramalho L. *Fluent Python*. O'Reilly Media, 2015.