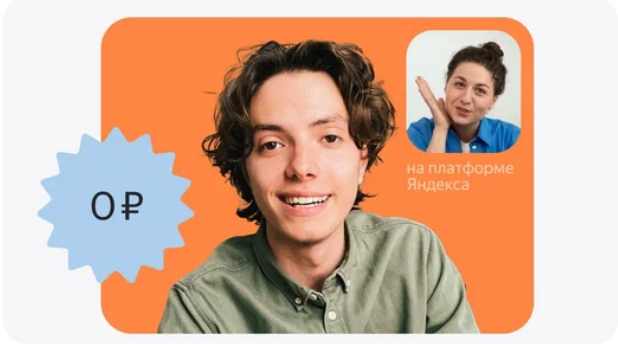


Операции над числами в Python



practicum.yandex.ru

РЕКЛАМА · 18+ Я

Бесплатное занятие английским в Яндекс Практикуме

Полноценное занятие с преподавателем, а не презентация курсов

Узнать больше

[Справочник по языку Python3.](#) / Операции над числами в Python

Числовые операции

Python поддерживает все обычные [операции, выполняемые над числами](#). Числа - неизменяемые объекты. Выполняя операции над числовыми объектами, вы всегда получаете новые числовые объекты, а не изменяете существующие. Разрешается обращаться к мнимой и действительной частям комплексного объекта `complex` как к атрибутам `complex.real` и `complex.imag`, доступным только для чтения. Попытка присвоить этим атрибутам новые значения приведет к возбуждению исключения.

Необязательные знаки `+` и `-` перед числами, а также знак `+`, присоединяющий литерал с плавающей точкой к мнимому для образования комплексного числа, не являются частью синтаксиса литералов. Они являются обычными операторами, которые подчиняются обычным правилам, регламентирующим приоритет операторов.

Например, вычисление выражения `-2**2` дает результат `-4`. Возведение в степень имеет более высокий приоритет, чем унарный "минус", поэтому анализатор воспринимает данное выражение как `-(2**2)`, а не как `(-2)**2`.

[Преобразование типов чисел в Python](#)

В арифметических операциях и операциях сравнения могут принимать участие любые два числа, относящиеся к встроенным типам Python. Если типы операндов различаются, выполняется автоматическое повышение типов.

[Арифметические операции в Python](#)

Синтаксис арифметических выражений прост. Операторы `+`, `-`, `*` и `/` работают так же, как и в большинстве других языков программирования. Так же имеются операции целочисленного деления, остатка от деления, возведения в степень

[Битовые операции над числами в Python](#)

Целые числа можно интерпретировать как битовые строки и использовать в битовых операциях. Битовые операторы имеют более низкий приоритет по сравнению с арифметическими операторами

[Проблемы точности чисел float в Python](#)

Большинство десятичных дробей не могут быть представлены точно как двоичные дроби. Следствием этого является то, что, как правило, вводимые десятичные числа с плавающей запятой аппроксимируются только двоичными числами с плавающей запятой, фактически сохраненными в машине.

ХОЧУ ПОМОЧЬ
ПРОЕКТУ

mango-office.ru



Развивайте
и масштабируйте
бизнес
легко

2 000 ₽

Виртуальная АТС Расширенная



2 900 Р
Виртуальная АТС Максимальная

[DOCS-Python.ru](#)™, 2023 г.

(Внимание! При копировании материала ссылка на источник обязательна)

[@docs_python_ru](#)