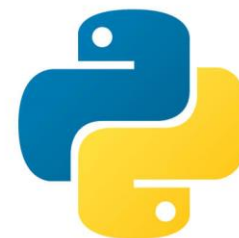


МАГІЧНІ
МЕТОДИ В
PYTHON ТА ЇХ
ПРИЗНАЧЕННЯ

*Мельник Владислав
Юрійович*

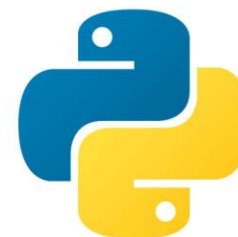
ВСТУПЛЕНИЕ

- Що таке магічні методи? Вони все в об'єктно-орієнтованому Пітоні. Це спеціальні методи, за допомогою яких ви можете додати в ваші класи «магію». Вони завжди обрамлені двома нижніми підкресленнями (наприклад, `__init__` або `__lt__`).



ВСІМ ВІДОМІЙ НАЙБІЛЬШИЙ БАЗОВИЙ МАГІЧНИЙ МЕТОД, `__INIT__`

- З його допомогою ми можемо ініціювати об'єкт. Однак, коли я пишу `x = SomeClass ()`, `__init__` не найперше, що викликається. Насправді, екземпляр об'єкта створює метод `__new__`, а потім аргументи передаються в ініціалізатор. На іншому кінці життєвого циклу об'єкта знаходиться метод `__del__`.



`__NEW__(CLS,
[...])`

- Це перший метод, який буде викликаний при ініціалізації об'єкта. Він приймає в якості параметрів клас і потім будь-які інші аргументи, які будуть передані в `__init__`. `__new__` використовується досить рідко, але іноді буває корисний, зокрема, коли клас успадковується від незмінного (immutable) типу, такого як кортеж (tuple) або рядок.

`__DEL__(SELF)`

- Якщо `__new__` і `__init__` утворюють конструктор об'єкта, `__del__` це його деструкцію. Він не визначає поведінку для вираження `del x` (тому цей код не еквівалентний `x.__del__()`). Швидше, він визначає поведінку об'єкта в той час, коли об'єкт потрапляє в збирач сміття. Це може бути досить зручно для об'єктів, які можуть вимагати додаткових чисток під час видалення, таких як сокети або файлові об'єкти. Однак, потрібно бути обережним, так як немає гарантії, що `__del__` буде викликаний, якщо об'єкт продовжує жити, коли інтерпретатор завершує роботу.

МАГІЧНІ МЕТОДИ ПОРІВНЯННЯ

- В Питоне уйма магических методов, созданных для определения интуитивного сравнения между объектами используя операторы, а не неуклюжие методы. Кроме того, они предоставляют способ переопределить поведение Питона по умолчанию для сравнения объектов (по ссылке). Вот список этих методов и что они делают:

ЧИСЛОВІ МАГІЧНІ МЕТОДИ

- Точно так же, як ви можете визначити, яким чином ваші об'єкти будуть порівнюватися операторами порівняння, ви можете визначити їх поведінку для числових операторів.

НАВІЩО
ПОТРІБНІ
МАГІЧНІ МЕТОДИ
PYTHON?

- Одним з їх поширених застосувань є перевантаження операторів.
- Це дозволяє перевизначити оператори для призначених для користувача класів, які дозволяють використовувати такі оператори, як `+` і `*`.
- Приклад магічного методу:
`__add__` для `+` або `__mul__` для `*`.



Дякую за увагу!

