Сообщить об ошибке.

про**Модуль uuid, уникальные идентификаторы в Python**



<u>Стандартная библиотека Python3.</u> / Модуль uuid, уникальные идентификаторы в Python

Создание различных универсальных уникальных идентификаторов

Модуль uuid реализует универсальные уникальные идентификаторы, как описано в RFC 4122.

RFC 4122 определяет систему для создания универсально уникальных идентификаторов для ресурсов таким образом, чтобы не требовался центральный регистратор. Значения UUID имеют длину 128 бит и, как говорится в справочном руководстве, "могут гарантировать уникальность в пространстве и времени".

Значения UUID полезны для генерации идентификаторов для документов, хостов, клиентов приложений и других ситуаций, когда необходимо уникальное значение. RFC специально сфокусирован на создании пространства имен Uniform Resource Name и охватывает три основных алгоритма:

- Использование МАС-адресов IEEE 802 в качестве источника уникальности,
- Использование псевдослучайных чисел,
- Использование известных строк в сочетании с криптографическим хешированием.

Во всех случаях начальное значение объединяется с системными часами и значением тактовой последовательности, используемым для поддержания уникальности в случае, если часы установлены неправильно.

Если **нужен только уникальный идентификатор**, то вероятно, следует вызвать функции <u>uuid.uuid1()</u> или <u>uuid.uuid4()</u>. Обратите внимание, что uuid.uuid1() может нарушить конфиденциальность, поскольку создает идентификатор, содержащий сетевой адрес компьютера. Функция uuid.uuid4() создает случайный UUID.

В зависимости от поддержки базовой платформы функция uuid.uuid1() может возвращать или не возвращать "безопасный" UUID. Безопасный UUID - это тот, который генерируется с использованием методов синхронизации, которые гарантируют, что два процесса не смогут получить один и тот же UUID. Все экземпляры UUID с версии Python-3.7 имеют атрибут UUID.is_safe, который передает любую информацию о безопасности UUID, используя это перечисление:

class uuid.SafeUUID:

- safe UUID был сгенерирован платформой безопасным для многих процессов способом.
- unsafe UUID не был сгенерирован многопроцессорным безопасным способом.
- unknown Платформа не предоставляет информацию о том, был ли UUID сгенерирован безопасно или нет.

Примеры использования:

Вот несколько примеров типичного использования модуля uuid:

```
>>> import uuid

# UUID на основе идентификатора хоста и текущего времени
>>> uuid.uuid1()

# UUID('a8098c1a-f86e-11da-bd1a-00112444be1e')

Вверх

# основе хеша MD5 пространства имен UUID и имени
```

```
>>> uuid.uuid3(uuid.NAMESPACE_DNS, 'python.org')
# UUID('6fa459ea-ee8a-3ca4-894e-db77e160355e')
# Случайный UUID
>> \mid.uuid4()
# UUID('16fd2706-8baf-433b: 82eb-8c7fada847da')
# UUID на основе хеша SHA-1 пространства имен UUID и имени
>>> uuid.uuid5(uuid.NAMESPACE_DNS, 'python.org')
# UUID('886313e1-3b8a-5372-9b90-0c9aee199e5d')
# Создать UUID из строки шестнадцатеричных цифр
# (фигурные скобки и дефисы игнорируются)
>>> x = uuid.UUID('{00010203-0405-0607-0809-0a0b0c0d0e0f}')
# Преобразовать UUID в строку шестнадцатеричных цифр
>>> str(x)
# '00010203-0405-0607-0809-0a0b0c0d0e0f'
# Получить необработанные 16 байтов UUID
>>> x.bytes
# b'\x00\x01\x02\x03\x04\x05\x06\x07\x08\t\n\x0b\x0c\r\x0e\x0f'
>>> # Создать UUID из 16-байтовой строки
>>> uuid.UUID(bytes=x.bytes)
UUID('00010203-0405-0607-0809-0a0b0c0d0e0f')
```

Содержание раздела:

- КРАТКИЙ ОБЗОР МАТЕРИАЛА.
- <u>Функциb uuid1() модуля uuid</u>
- <u>Функциb uuid3() и uuid5() модуля uuid</u>
- <u>Функция uuid4() модуля uuid</u>
- <u>Knacc UUID() модуля uuid</u>
- <u>Функция getnode() модуля uuid</u>

DOCS-Python.ru™, 2023 г.

(Внимание! При копировании материала ссылка на источник обязательна)

@docs_python_ru

Вверх