

## :mod:`uuid` --- UUID objects according to RFC 4122

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 1); [backlink](#)**

Unknown interpreted text role "mod".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 4)**

Unknown directive type "module".

```
.. module:: uuid
   :synopsis: UUID objects (universally unique identifiers) according to RFC 4122
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 6)**

Unknown directive type "moduleauthor".

```
.. moduleauthor:: Ka-Ping Yee <ping@zesty.ca>
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 7)**

Unknown directive type "sectionauthor".

```
.. sectionauthor:: George Yoshida <quiver@users.sourceforge.net>
```

Source code: `:source:`Lib/uuid.py``

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 9); [backlink](#)**

Unknown interpreted text role "source".

This module provides immutable `:class:`UUID`` objects (the `:class:`UUID`` class) and the functions `:func:`uuid1``, `:func:`uuid3``, `:func:`uuid4``, `:func:`uuid5`` for generating version 1, 3, 4, and 5 UUIDs as specified in [RFC 4122](#).

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 13); [backlink](#)**

Unknown interpreted text role "class".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 13); [backlink](#)**

Unknown interpreted text role "class".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 13); [backlink](#)**

Unknown interpreted text role "func".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 13); [backlink](#)**

Unknown interpreted text role "func".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 13); [backlink](#)**

Unknown interpreted text role "func".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 13); [backlink](#)**

Unknown interpreted text role "func".

If all you want is a unique ID, you should probably call `:func:`uuid1`` or `:func:`uuid4``. Note that `:func:`uuid1`` may compromise privacy since it creates a UUID containing the computer's network address. `:func:`uuid4`` creates a random UUID.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 17); [backlink](#)**

Unknown interpreted text role "func".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 17); [backlink](#)**

Unknown interpreted text role "func".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 17); [backlink](#)**

Unknown interpreted text role "func".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 17); [backlink](#)**

Unknown interpreted text role "func".

Depending on support from the underlying platform, `.func:uuid1` may or may not return a "safe" UUID. A safe UUID is one which is generated using synchronization methods that ensure no two processes can obtain the same UUID. All instances of `class:UUID` have an `attr:'is_safe'` attribute which relays any information about the UUID's safety, using this enumeration:

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 22); [backlink](#)**

Unknown interpreted text role "func".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 22); [backlink](#)**

Unknown interpreted text role "class".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 22); [backlink](#)**

Unknown interpreted text role "attr".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 30)**

Unknown directive type "versionadded".

```
.. versionadded:: 3.7
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 32)**

Unknown directive type "attribute".

```
.. attribute:: SafeUUID.safe
```

The UUID was generated by the platform in a multiprocessing-safe way.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 36)**

Unknown directive type "attribute".

```
.. attribute:: SafeUUID.unsafe
```

The UUID was not generated in a multiprocessing-safe way.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 40)**

Unknown directive type "attribute".

```
.. attribute:: SafeUUID.unknown
```

The platform does not provide information on whether the UUID was generated safely or not.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 45)**

Invalid class attribute value for "class" directive: "UUID(hex=None, bytes=None, bytes\_le=None, fields=None, int=None, version=None, \*, is\_safe=SafeUUID.unknown)".

```
.. class:: UUID(hex=None, bytes=None, bytes_le=None, fields=None, int=None, version=None, *, is_safe=SafeUUID.unknown)
```

Create a UUID from either a string of 32 hexadecimal digits, a string of 16 bytes in big-endian order as the `*bytes*` argument, a string of 16 bytes in little-endian order as the `*bytes_le*` argument, a tuple of six integers (32-bit `*time_low*`, 16-bit `*time_mid*`, 16-bit `*time_hi_version*`, 8-bit `*clock_seq_hi_variant*`, 8-bit `*clock_seq_low*`, 48-bit `*node*`) as the `*fields*` argument, or a single 128-bit integer as the `*int*` argument. When a string of hex digits is given, curly braces, hyphens, and a URN prefix are all optional. For example, these expressions all yield the same UUID::

```
UUID('{12345678-1234-5678-1234-567812345678}')
UUID('12345678123456781234567812345678')
UUID('urn:uuid:12345678-1234-5678-1234-567812345678')
UUID(bytes=b'\x12\x34\x56\x78'*4)
UUID(bytes_le=b'\x78\x56\x34\x12\x34\x12\x78\x56' +
           b'\x12\x34\x56\x78\x12\x34\x56\x78')
UUID(fields=(0x12345678, 0x1234, 0x5678, 0x12, 0x34, 0x567812345678))
UUID(int=0x12345678123456781234567812345678)
```

Exactly one of `*hex*`, `*bytes*`, `*bytes_le*`, `*fields*`, or `*int*` must be given. The `*version*` argument is optional; if given, the resulting UUID will have its variant and version number set according to `:rfc:4122`, overriding bits in the given `*hex*`, `*bytes*`, `*bytes_le*`, `*fields*`, or `*int*`.

Comparison of UUID objects are made by way of comparing their `:attr:'UUID.int'` attributes. Comparison with a non-UUID object raises a `:exc:'TypeError'`.

`str(uuid)` returns a string in the form `'12345678-1234-5678-1234-567812345678'` where the 32 hexadecimal digits represent the UUID.

`:class:'UUID'` instances have these read-only attributes:

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 79); [backlink](#)**

Unknown interpreted text role "class".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 81)**

Unknown directive type "attribute".

```
.. attribute:: UUID.bytes
```

The UUID as a 16-byte string (containing the six integer fields in big-endian byte order).

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 87)**

Unknown directive type "attribute".

```
.. attribute:: UUID.bytes_le
```

The UUID as a 16-byte string (with `*time_low*`, `*time_mid*`, and `*time_hi_version*` in little-endian byte order).

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 93)**

Unknown directive type "attribute".

```
.. attribute:: UUID.fields
```

A tuple of the six integer fields of the UUID, which are also available as six individual attributes and two derived attributes:

Field	Meaning
<code>:attr:'time_low'</code>	the first 32 bits of the UUID
<code>:attr:'time_mid'</code>	the next 16 bits of the UUID
<code>:attr:'time_hi_version'</code>	the next 16 bits of the UUID
<code>:attr:'clock_seq_hi_variant'</code>	the next 8 bits of the UUID
<code>:attr:'clock_seq_low'</code>	the next 8 bits of the UUID
<code>:attr:'node'</code>	the last 48 bits of the UUID
<code>:attr:'time'</code>	the 60-bit timestamp

```
| :attr:`clock_seq` | the 14-bit sequence number |  
+-----+-----+
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 119)**

Unknown directive type "attribute".

```
.. attribute:: UUID.hex
```

The UUID as a 32-character lowercase hexadecimal string.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 124)**

Unknown directive type "attribute".

```
.. attribute:: UUID.int
```

The UUID as a 128-bit integer.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 129)**

Unknown directive type "attribute".

```
.. attribute:: UUID.urn
```

The UUID as a URN as specified in :rfc:`4122`.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 134)**

Unknown directive type "attribute".

```
.. attribute:: UUID.variant
```

The UUID variant, which determines the internal layout of the UUID. This will be one of the constants :const:`RESERVED\_NCS`, :const:`RFC\_4122`, :const:`RESERVED\_MICROSOFT`, or :const:`RESERVED\_FUTURE`.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 141)**

Unknown directive type "attribute".

```
.. attribute:: UUID.version
```

The UUID version number (1 through 5, meaningful only when the variant is :const:`RFC\_4122`).

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 146)**

Unknown directive type "attribute".

```
.. attribute:: UUID.is_safe
```

An enumeration of :class:`SafeUUID` which indicates whether the platform generated the UUID in a multiprocessing-safe way.

```
.. versionadded:: 3.7
```

The `mod:uuid` module defines the following functions:

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 153); [backlink](#)**

Unknown interpreted text role "mod".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 156)**

Unknown directive type "function".

```
.. function:: getnode()
```

Get the hardware address as a 48-bit positive integer. The first time this runs, it may launch a separate program, which could be quite slow. If all attempts to obtain the hardware address fail, we choose a random 48-bit number with the multicast bit (least significant bit of the first octet) set to 1 as recommended in :rfc:`4122`. "Hardware address" means the MAC address of a network interface. On a machine with multiple network interfaces, universally administered MAC addresses (i.e. where the second least significant bit of the first octet is \*unset\*) will be preferred over locally administered MAC addresses, but with no other ordering guarantees.

```
.. versionchanged:: 3.7
   Universally administered MAC addresses are preferred over locally
   administered MAC addresses, since the former are guaranteed to be
   globally unique, while the latter are not.
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 173)**

Unknown directive type "index".

```
.. index:: single: getnode
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 176)**

Unknown directive type "function".

```
.. function:: uuid1(node=None, clock_seq=None)
```

Generate a UUID from a host ID, sequence number, and the current time. If \*node\* is not given, :func:`getnode` is used to obtain the hardware address. If \*clock\_seq\* is given, it is used as the sequence number; otherwise a random 14-bit sequence number is chosen.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 183)**

Unknown directive type "index".

```
.. index:: single: uuid1
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 186)**

Unknown directive type "function".

```
.. function:: uuid3(namespace, name)
```

Generate a UUID based on the MD5 hash of a namespace identifier (which is a UUID) and a name (which is a string).

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 191)**

Unknown directive type "index".

```
.. index:: single: uuid3
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 194)**

Unknown directive type "function".

```
.. function:: uuid4()
```

Generate a random UUID.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 198)**

Unknown directive type "index".

```
.. index:: single: uuid4
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 201)**

Unknown directive type "function".

```
.. function:: uuid5(namespace, name)
```

Generate a UUID based on the SHA-1 hash of a namespace identifier (which is a UUID) and a name (which is a string).

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 206)**

Unknown directive type "index".

```
.. index:: single: uuid5
```

The `mod:uuid` module defines the following namespace identifiers for use with `func:uuid3` or `func:uuid5`.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 208); [backlink](#)**

Unknown interpreted text role "mod".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 208); [backlink](#)**

Unknown interpreted text role "func".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 208); [backlink](#)**

Unknown interpreted text role "func".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 212)**

Unknown directive type "data".

```
.. data:: NAMESPACE_DNS
```

When this namespace is specified, the `*name*` string is a fully-qualified domain name.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 218)**

Unknown directive type "data".

```
.. data:: NAMESPACE_URL
```

When this namespace is specified, the `*name*` string is a URL.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 223)**

Unknown directive type "data".

```
.. data:: NAMESPACE_OID
```

When this namespace is specified, the `*name*` string is an ISO OID.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 228)**

Unknown directive type "data".

```
.. data:: NAMESPACE_X500
```

When this namespace is specified, the `*name*` string is an X.500 DN in DER or a text output format.

The `mod:uuid` module defines the following constants for the possible values of the `attr:variant` attribute:

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main) (Doc) (library) uuid.rst, line 233); [backlink](#)**

Unknown interpreted text role "mod".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 233); [backlink](#)**

Unknown interpreted text role "attr".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 237)**

Unknown directive type "data".

```
.. data:: RESERVED_NCS

    Reserved for NCS compatibility.
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 242)**

Unknown directive type "data".

```
.. data:: RFC_4122

    Specifies the UUID layout given in :rfc:`4122`.
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 247)**

Unknown directive type "data".

```
.. data:: RESERVED_MICROSOFT

    Reserved for Microsoft compatibility.
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 252)**

Unknown directive type "data".

```
.. data:: RESERVED_FUTURE

    Reserved for future definition.
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 257)**

Unknown directive type "seealso".

```
.. seealso::

    :rfc:`4122` - A Universally Unique IDentifier (UUID) URN Namespace
    This specification defines a Uniform Resource Name namespace for UUIDs, the
    internal format of UUIDs, and methods of generating UUIDs.
```

## Example

Here are some examples of typical usage of the `mod:uuid` module:

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) uuid.rst, line 269); [backlink](#)**

Unknown interpreted text role "mod".

```
>>> import uuid

>>> # make a UUID based on the host ID and current time
>>> uuid.uuid1()
UUID('a8098c1a-f86e-11da-bd1a-00112444be1e')

>>> # make a UUID using an MD5 hash of a namespace UUID and a name
>>> uuid.uuid3(uuid.NAMESPACE_DNS, 'python.org')
UUID('6fa459ea-ee8a-3ca4-894e-db77e160355e')

>>> # make a random UUID
>>> uuid.uuid4()
UUID('16fd2706-8baf-433b-82eb-8c7fada847da')

>>> # make a UUID using a SHA-1 hash of a namespace UUID and a name
>>> uuid.uuid5(uuid.NAMESPACE_DNS, 'python.org')
UUID('886313e1-3b8a-5372-9b90-0c9aee199e5d')
```

```
>>> # make a UUID from a string of hex digits (braces and hyphens ignored)
>>> x = uuid.UUID('{00010203-0405-0607-0809-0a0b0c0d0e0f}')

>>> # convert a UUID to a string of hex digits in standard form
>>> str(x)
'00010203-0405-0607-0809-0a0b0c0d0e0f'

>>> # get the raw 16 bytes of the UUID
>>> x.bytes
b'\x00\x01\x02\x03\x04\x05\x06\x07\x08\t\n\x0b\x0c\r\x0e\x0f'

>>> # make a UUID from a 16-byte string
>>> uuid.UUID(bytes=x.bytes)
UUID('00010203-0405-0607-0809-0a0b0c0d0e0f')
```