
aioethereum Documentation

Release 0.2.0

DeV1doR

Oct 05, 2017

Contents

1	Features	3
2	Installation	5
3	Requirements	7
4	License	9
5	Getting started	11
5.1	Getting started	11
6	API Documentation	13
6.1	aioethereum API	13
6.2	Glossary	26
7	Indices and tables	27
	Python Module Index	29

Ethereum RPC client library for the [\(PEP 3156\)](#) Python event loop.

The library is intended to provide simple and clear interface to Ethereum node based on [asyncio](#).

CHAPTER 1

Features

<i>ujson</i> support	Yes
High-level APIs	Yes
HTTP support	Yes
Unix domain socket (IPC) support	Yes
SSL/TLS support	Yes
Tested CPython versions	3.4, 3.5, 3.6
Tested Geth versions	1.7.0
Implemented RPC apis	<i>db, eth, miner, net, personal, shh, txpool, web3</i>

CHAPTER 2

Installation

The easiest way to install aioethereum is by using the package on PyPi:

```
pip install aioethereum
```


CHAPTER 3

Requirements

- Python 3.3 and *asyncio* or Python 3.4+
- *ujson*

CHAPTER 4

License

The aioethereum is offered under MIT license.

5.1 Getting started

This part of the documentation is a quick start for writing applications that interact with the client for ethereum nodes (geth, parity, etc.)

5.1.1 Basic API description

This client provides APIs from:

- <https://github.com/ethereum/wiki/wiki/JSON-RPC>
- <https://github.com/ethereum/go-ethereum/wiki/Management-APIs>

It is simple http(s)/unix connector for ethereum node which implements standart RPC methods.

6.1 aioethereum API

Documentation related to various APIs available in this package.

6.1.1 aioethereum

class `aioethereum.AsyncIOHTTPClient` (*host='127.0.0.1', port=8545, tls=False, timeout=60, *, loop=None*)

Bases: `aioethereum.client.BaseAsyncIOClient`, `aioethereum.management.RpcMixin`

Creates AsyncIOHTTPClient client to communicate via HTTP(s).

Parameters

- **host** (*str*) – Host on ethereum node
- **port** (*int*) – Port on ethereum node
- **tls** (*bool*) – Use SSL connection
- **timeout** (*int*) – Total time of timeout call
- **loop** (`EventLoop`) – An optional *event loop* instance (uses `asyncio.get_event_loop()` if not specified).

Returns `AsyncIOHTTPClient` instance.

class `aioethereum.AsyncIOIPCClient` (*reader, writer, unix_path, timeout=60, *, loop=None*)

Bases: `aioethereum.client.BaseAsyncIOClient`, `aioethereum.management.RpcMixin`

Creates AsyncIOIPCClient client to communicate via IPC.

Parameters

- **reader** (`asyncio.streams.StreamReader`) – Instance of the stream reader
- **writer** (`asyncio.streams.StreamWriter`) – Instance of the stream writer

- **unix_path** (*str*) – Unix domain path
- **timeout** (*int*) – An optional total time of timeout call
- **loop** (*EventLoop*) – An optional *event loop* instance (uses `asyncio.get_event_loop()` if not specified).

Returns *AsyncIOIPCClient* instance.

class aioethereum.**BaseAsyncIOClient**

Bases: `abc.ABC`

Abstract class for creating client.

rpc_call (*method, params=None, id_=None*)

Implements RPC 2.0 call to node server.

Parameters

- **method** (*str*) – RPC node method
- **params** (*list*) – Input params for method
- **id** (*str or int*) – ID of current request

Returns *Response from RPC*

coroutine aioethereum.**create_ethereum_client** (*uri, timeout=60, *, loop=None*)

Create client to ethereum node based on schema.

Parameters

- **uri** (*str*) – Host on ethereum node
- **timeout** (*int*) – An optional total time of timeout call
- **loop** (*EventLoop*) – An optional *event loop* instance (uses `asyncio.get_event_loop()` if not specified).

Returns *BaseAsyncIOClient* instance.

6.1.2 management

This package contains all high level features of the *aioethereum.AsyncIOIPCClient* and *aioethereum.AsyncIOHTTPClient*.

admin

Admin related features

class aioethereum.management.admin.**AdminMixin**

Bases: `object`

db

Db related features

class aioethereum.management.db.**DbMixin**

Bases: `object`

db_getHex (*db_name*, *key*)
https://github.com/ethereum/wiki/wiki/JSON-RPC#db_gethex
 DEPRECATED

coroutine db_getString (*db_name*, *key*)
https://github.com/ethereum/wiki/wiki/JSON-RPC#db_getstring
 DEPRECATED

coroutine db_putHex (*db_name*, *key*, *value*)
https://github.com/ethereum/wiki/wiki/JSON-RPC#db_puthex
 DEPRECATED

coroutine db_putString (*db_name*, *key*, *value*)
https://github.com/ethereum/wiki/wiki/JSON-RPC#db_putstring
 DEPRECATED

debug

Debug related features

class aioethereum.management.debug.**DebugMixin**
 Bases: object

eth

Eth related features

class aioethereum.management.eth.**EthMixin**
 Bases: object

coroutine eth_accounts ()
https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_accounts
Returns accounts
Return type list

coroutine eth_blockNumber ()
https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_blocknumber
Returns bnumber
Return type int

coroutine eth_call (*from_*, *to=None*, *gas=None*, *gas_price=None*, *value=None*, *data=None*,
block='latest')
https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_call

Parameters

- **from** (*str*) – From account address
- **to** (*str*) – To account address (optional)
- **gas** (*int*) – Gas amount for current transaction (optional)
- **gas_price** (*int*) – Gas price for current transaction (optional)
- **value** (*int*) – Amount of ether to send (optional)

- **data** (*hex*) – Additional data for transaction (optional)
- **block** (*int or BLOCK_TAGS*) – Block tag or number (optional)

Return type str

coroutine eth_coinbase ()

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_coinbase

Returns address

Return type str

coroutine eth_compileLLL (*code*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_compilelll

DEPRECATED

coroutine eth_compileSerpent (*code*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_compileserpent

DEPRECATED

coroutine eth_compileSolidity (*code*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_compilesolidity

DEPRECATED

coroutine eth_estimateGas (*from_, to=None, gas=None, gas_price=None, value=None, data=None*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_estimategas

Parameters

- **from** (*str*) – From account address
- **to** (*str*) – To account address (optional)
- **gas** (*int*) – Gas amount for current transaction (optional)
- **gas_price** (*int*) – Gas price for current transaction (optional)
- **value** (*int*) – Amount of ether to send (optional)
- **data** (*hex*) – Additional data for transaction (optional)

Returns gas amount

Return type int

coroutine eth_gasPrice ()

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_gasprice

Returns wei

Return type int

coroutine eth_getBalance (*address, block='latest'*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getbalance

Parameters

- **address** (*str*) – Account address
- **block** (*int or BLOCK_TAGS*) – Block tag or number (optional)

Returns wei

Return type int

coroutine eth_getBlockByHash (*bhash*, *tx_objects=True*)
https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getblockbyhash

Parameters

- **bhash** (*str*) – Block hash
- **tx_objects** (*bool*) – Return txs full object (optional)

Returns block

Return type dict or None

coroutine eth_getBlockByNumber (*block='latest'*, *tx_objects=True*)
https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getblockbynumber

Parameters

- **block** (*int or BLOCK_TAGS*) – Block tag or number (optional)
- **tx_objects** (*bool*) – Return txs full object (optional)

Returns block

Return type dict or None

coroutine eth_getBlockTransactionCountByHash (*bhash*)
https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getblocktransactioncountbyhash

Parameters **bhash** (*str*) – Block hash

Returns count

Return type int or None

coroutine eth_getBlockTransactionCountByNumber (*block='latest'*)
https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getblocktransactioncountbynumber

Parameters **block** (*int or BLOCK_TAGS*) – Block tag or number (optional)

Returns count

Return type int

coroutine eth_getCode (*address*, *block='latest'*)
https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getcode

Parameters

- **address** (*str*) – Address of contract
- **block** (*int or BLOCK_TAGS*) – Block tag or number (optional)

Returns code

Return type str

coroutine eth_getCompilers ()
https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getcompilers

DEPRECATED

coroutine eth_getFilterChanges (*filter_id*)
https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getfilterchanges

Parameters **filter_id** (*str*) – Id of created filter

Returns logs

Return type list

coroutine eth_getFilterLogs (*filter_id*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getfilterlogs

Parameters **filter_id** (*str*) – Id of created filter

Returns logs

Return type list

coroutine eth_getLogs (*from_block='latest', to_block='latest', address=None, topics=None*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getlogs

Parameters

- **from_block** (*int or BLOCK_TAGS*) – Block tag or number (optional)
- **to_block** (*int or BLOCK_TAGS*) – Block tag or number (optional)
- **address** (*str*) – Contract address (optional)
- **topics** (*list*) – Topics (optional)

Returns logs

Return type list

coroutine eth_getStorageAt (*address, position=0, block='latest'*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getstorageat

Parameters

- **address** (*str*) – Storage address
- **position** (*int*) – Position in storage (optional)
- **block** (*int or BLOCK_TAGS*) – Block tag or number (optional)

Return type int

coroutine eth_getTransactionByBlockHashAndIndex (*bhash, index=0*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_gettransactionbyblockhashandindex

Parameters

- **bhash** (*str*) – Block hash
- **index** (*int*) – Index position (optional)

coroutine eth_getTransactionByBlockNumberAndIndex (*block='latest', index=0*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_gettransactionbyblocknumberandindex

Parameters

- **block** (*int or BLOCK_TAGS*) – Block tag or number (optional)
- **index** (*int*) – Index position (optional)

Returns transaction

Return type dict or None

coroutine eth_getTransactionByHash (*txhash*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_gettransactionbyhash

Parameters **txhash** (*str*) – Transaction hash

Returns transaction

Return type dict or None

coroutine eth_getTransactionCount (*address*, *block*='latest')

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_gettransactioncount

Parameters

- **address** (*str*) – Account address
- **block** (*int* or *BLOCK_TAGS*) – Block tag or number (optional)

Returns count

Return type int

coroutine eth_getTransactionReceipt (*txhash*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_gettransactionreceipt

Parameters **txhash** (*str*) – Transaction hash

Returns transaction

Return type dict or None

coroutine eth_getUncleByBlockHashAndIndex (*bhash*, *index*=0)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getunclebyblockhashandindex

Parameters

- **bhash** (*str*) – Block hash
- **index** (*int*) – Index position (optional)

Returns block

Return type dict or None

coroutine eth_getUncleByBlockNumberAndIndex (*block*='latest', *index*=0)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getunclebyblocknumberandindex

Parameters

- **block** (*int* or *BLOCK_TAGS*) – Block tag or number (optional)
- **index** (*int*) – Index position (optional)

Returns block

Return type dict or None

coroutine eth_getUncleCountByBlockHash (*bhash*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getunclecountbyblockhash

Parameters **bhash** (*str*) – Block hash

Returns count

Return type int or None

coroutine eth_getUncleCountByBlockNumber (*block*='latest')

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getunclecountbyblocknumber

Parameters **block** (*int* or *BLOCK_TAGS*) – Block tag or number (optional)

Returns count

Return type int

coroutine eth_getWork()

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getwork

Returns work

Return type list

coroutine eth_hashrate()

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_hashrate

Returns hashrate

Return type int

coroutine eth_mining()

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_mining

Return type bool

coroutine eth_newBlockFilter()

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_newblockfilter

Returns filter_id

Return type str

coroutine eth_newFilter (*from_block='latest', to_block='latest', address=None, topics=None*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_newfilter

Parameters

- **from_block** (*int or BLOCK_TAGS*) – Block tag or number (optional)
- **to_block** (*int or BLOCK_TAGS*) – Block tag or number (optional)
- **address** (*str*) – Contract address (optional)
- **topics** (*list*) – Topics (optional)

Returns filter_id

Return type str

coroutine eth_newPendingTransactionFilter()

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_newpendingtransactionfilter

Returns filter_id

Return type str

coroutine eth_protocolVersion()

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_protocolversion

Return type int

coroutine eth_sendRawTransaction (*data*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_sendrawtransaction

Parameters **data** (*str*) – Signed transaction data

Returns txhash

Return type str

coroutine eth_sendTransaction (*from_, to=None, gas=None, gas_price=None, value=None, data=None, nonce=None*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_sendtransaction

Parameters

- **from** (*str*) – From account address
- **to** (*str*) – To account address (optional)
- **gas** (*int*) – Gas amount for current transaction (optional)
- **gas_price** (*int*) – Gas price for current transaction (optional)
- **value** (*int*) – Amount of ether to send (optional)
- **data** (*hex*) – Additional data for transaction (optional)
- **nonce** (*int*) – Unique nonce for transaction (optional)

Returns txhash**Return type** str**coroutine** **eth_sign** (*address, data*)https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_sign**Parameters**

- **address** (*str*) – Account address
- **data** (*str*) – Message to sign

Returns signature**Return type** str**coroutine** **eth_submitHashrate** (*hashrate, id_*)https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_submithashrate**Parameters**

- **hashrate** (*str*) – Hash rate
- **id** (*str*) – Random hex

Returns success**Return type** bool**coroutine** **eth_submitWork** (*nonce, header, mix_digest*)https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_submitwork**Parameters**

- **nonce** (*int*) – Nonce of work
- **header** (*str*) – Pow-hash header
- **mix_digest** (*str*) – Mix digest

Returns success**Return type** bool**coroutine** **eth_syncing** ()https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_syncing**Returns** sync status or false**Return type** dict or bool**coroutine** **eth_uninstallFilter** (*filter_id*)https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_uninstallfilter

Parameters `filter_id` (*str*) – Id of created filter

Returns success

Return type bool

miner

Miner related features

class aioethereum.management.miner.**MinerMixin**

Bases: object

coroutine `miner_setEtherBase` (*address*)

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#miner_setetherbase

NOT AVAILABLE

coroutine `miner_setExtra` (*data*)

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#miner_setextra

Parameters `data` (*str*) – Extra data a miner can include when miner blocks

Return type bool

coroutine `miner_setGasPrice` (*number*)

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#miner_setgasprice

Parameters `number` (*int*) – Minimal accepted gas price when mining transactions

Return type bool

coroutine `miner_start` (*number*)

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#miner_start

Parameters `number` (*int*) – CPU mining process with the given number of threads

Return type None

coroutine `miner_stop` ()

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#miner_stop

Return type bool

net

Net related features

class aioethereum.management.net.**NetMixin**

Bases: object

coroutine `net_listening` ()

https://github.com/ethereum/wiki/wiki/JSON-RPC#net_listening

Return type bool

coroutine `net_peerCount` ()

https://github.com/ethereum/wiki/wiki/JSON-RPC#net_peercount

Return type int

coroutine `net_version` ()

https://github.com/ethereum/wiki/wiki/JSON-RPC#net_version

Return type str

personal

Personal related features

class aioethereum.management.personal.**PersonalMixin**

Bases: object

coroutine **personal_ecRecover** (*message, signature*)

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#personal_ecrecover

Parameters

- **message** (*str*) – Message for sign
- **password** (*str*) – Signature of account (optional)

Returns address

Return type str

coroutine **personal_importRawKey** (*keydata, passphrase*)

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#personal_importrawkey

Parameters

- **keydata** (*str*) – Unencrypted private key
- **passphrase** (*str*) – Passphrase of private key

Returns address

Return type hex

coroutine **personal_listAccounts** ()

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#personal_listaccounts

Returns addresses

Return type list

coroutine **personal_lockAccount** (*address*)

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#personal_lockaccount

Parameters **address** (*str*) – Account address

Return type bool

coroutine **personal_newAccount** (*passphrase=None*)

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#personal_newaccount

Parameters **passphrase** (*str*) – Passphrase of account (optional)

Returns address

Return type str

coroutine **personal_sendTransaction** (*from_, to=None, gas=None, gas_price=None, value=None, data=None, nonce=None, passphrase=None*)

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#personal_sendtransaction

Parameters

- **from** (*str*) – From account address

- **to** (*str*) – To account address (optional)
- **gas** (*int*) – Gas amount for current transaction (optional)
- **gas_price** (*int*) – Gas price for current transaction (optional)
- **value** (*int*) – Amount of ether to send (optional)
- **data** (*hex*) – Additional data for transaction (optional)
- **nonce** (*int*) – Unique nonce for transaction (optional)
- **passphrase** (*str*) – Passphrase of account (optional)

Returns txhash

Return type str

coroutine personal_sign (*message, account, password=None*)

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#personal_sign

Parameters

- **message** (*str*) – Message for sign
- **account** (*str*) – Account address
- **password** (*str*) – Password of account (optional)

Returns signature

Return type str

coroutine personal_unlockAccount (*address, passphrase=None, duration=None*)

https://github.com/ethereum/go-ethereum/wiki/Management-APIs#personal_unlockaccount

Parameters

- **address** (*str*) – Account address
- **passphrase** (*str*) – Passphrase of account (optional)
- **duration** (*int*) – Duration to be unlocked (optional)

Return type bool

shh

Shh related features

class aioethereum.management.shh.ShhMixin

Bases: object

coroutine shh_addToGroup ()

https://github.com/ethereum/wiki/wiki/JSON-RPC#shh_addtogroup

DEPRECATED

coroutine shh_getFilterChanges (*filter_id*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#shh_getfilterchanges

DEPRECATED

coroutine shh_getMessages (*filter_id*)

https://github.com/ethereum/wiki/wiki/JSON-RPC#shh_getmessages

DEPRECATED

```

coroutine shh_hasIdentity (address)
    https://github.com/ethereum/wiki/wiki/JSON-RPC#shh\_hasidentity

    DEPRECATED

coroutine shh_newFilter (to=None, *, topics)
    https://github.com/ethereum/wiki/wiki/JSON-RPC#shh\_newfilter

    DEPRECATED

coroutine shh_newGroup ()
    https://github.com/ethereum/wiki/wiki/JSON-RPC#shh\_newgroup

    DEPRECATED

coroutine shh_newIdentity ()
    https://github.com/ethereum/wiki/wiki/JSON-RPC#shh\_newidentity

    DEPRECATED

coroutine shh_post (from_=None, to=None, *, topics, payload, priority, ttl)
    https://github.com/ethereum/wiki/wiki/JSON-RPC#shh\_post

coroutine shh_uninstallFilter (filter_id)
    https://github.com/ethereum/wiki/wiki/JSON-RPC#shh\_uninstallfilter

    DEPRECATED

coroutine shh_version ()
    https://github.com/ethereum/wiki/wiki/JSON-RPC#shh\_version

    Return type float

```

txpool

Txpool related features

```

class aioethereum.management.txpool.TxpoolMixin
    Bases: object

    coroutine txpool_content ()
        https://github.com/ethereum/go-ethereum/wiki/Management-APIs#txpool\_content

        Return type dict

    coroutine txpool_inspect ()
        https://github.com/ethereum/go-ethereum/wiki/Management-APIs#txpool\_inspect

        Return type dict

    coroutine txpool_status ()
        https://github.com/ethereum/go-ethereum/wiki/Management-APIs#txpool\_status

        Return type dict

```

web3

Web3 related features

```

class aioethereum.management.web3.Web3Mixin
    Bases: object

```

coroutine `web3_clientVersion()`
https://github.com/ethereum/wiki/wiki/JSON-RPC#web3_clientversion
Returns version
Return type str

coroutine `web3_sha3(data)`
https://github.com/ethereum/wiki/wiki/JSON-RPC#web3_sha3
Parameters `data` (*str*) – Data for hashing
Returns Keccak-256
Return type str

6.2 Glossary

asyncio Reference implementation of **PEP 3156**

See <https://pypi.python.org/pypi/asyncio>

pytest A mature full-featured Python testing tool. See <http://pytest.org/latest/>

ujson UltraJSON is an ultra fast JSON encoder and decoder written in pure C with bindings for Python 2.5+ and 3.
[ujson](#).

See <https://pypi.python.org/pypi/ujson>

CHAPTER 7

Indices and tables

- `genindex`
- `modindex`
- `search`

a

- `aioethereum`, [13](#)
- `aioethereum.management.admin`, [14](#)
- `aioethereum.management.db`, [14](#)
- `aioethereum.management.debug`, [15](#)
- `aioethereum.management.eth`, [15](#)
- `aioethereum.management.miner`, [22](#)
- `aioethereum.management.net`, [22](#)
- `aioethereum.management.personal`, [23](#)
- `aioethereum.management.shh`, [24](#)
- `aioethereum.management.txpool`, [25](#)
- `aioethereum.management.web3`, [25](#)

A

AdminMixin (class in aioethereum.management.admin), 14
aioethereum (module), 13
aioethereum.management.admin (module), 14
aioethereum.management.db (module), 14
aioethereum.management.debug (module), 15
aioethereum.management.eth (module), 15
aioethereum.management.miner (module), 22
aioethereum.management.net (module), 22
aioethereum.management.personal (module), 23
aioethereum.management.shh (module), 24
aioethereum.management.txpool (module), 25
aioethereum.management.web3 (module), 25
asyncio, 26
AsyncIOHTTPClient (class in aioethereum), 13
AsyncIOIPCCClient (class in aioethereum), 13

B

BaseAsyncIOClient (class in aioethereum), 14

C

create_ethereum_client() (in module aioethereum), 14

D

db_getHex() (aioethereum.management.db.DbMixin method), 14
db_getString() (aioethereum.management.db.DbMixin method), 15
db_putHex() (aioethereum.management.db.DbMixin method), 15
db_putString() (aioethereum.management.db.DbMixin method), 15
DbMixin (class in aioethereum.management.db), 14
DebugMixin (class in aioethereum.management.debug), 15

E

eth_accounts() (aioethereum.management.eth.EthMixin method), 15

eth_blockNumber() (aioethereum.management.eth.EthMixin method), 15
eth_call() (aioethereum.management.eth.EthMixin method), 15
eth_coinbase() (aioethereum.management.eth.EthMixin method), 16
eth_compileLLL() (aioethereum.management.eth.EthMixin method), 16
eth_compileSerpent() (aioethereum.management.eth.EthMixin method), 16
eth_compileSolidity() (aioethereum.management.eth.EthMixin method), 16
eth_estimateGas() (aioethereum.management.eth.EthMixin method), 16
eth_gasPrice() (aioethereum.management.eth.EthMixin method), 16
eth_getBalance() (aioethereum.management.eth.EthMixin method), 16
eth_getBlockByHash() (aioethereum.management.eth.EthMixin method), 16
eth_getBlockByNumber() (aioethereum.management.eth.EthMixin method), 17
eth_getBlockTransactionCountByHash() (aioethereum.management.eth.EthMixin method), 17
eth_getBlockTransactionCountByNumber() (aioethereum.management.eth.EthMixin method), 17
eth_getCode() (aioethereum.management.eth.EthMixin method), 17
eth_getCompilers() (aioethereum.management.eth.EthMixin method), 17
eth_getFilterChanges() (aioethereum.management.eth.EthMixin method), 17
eth_getFilterLogs() (aioethereum.management.eth.EthMixin method), 18
eth_getLogs() (aioethereum.management.eth.EthMixin method), 18
eth_getStorageAt() (aioethereum.management.eth.EthMixin method), 18

method), 18

`eth_getTransactionByBlockHashAndIndex()`
(`aioethereum.management.eth.EthMixin`
method), 18

`eth_getTransactionByBlockNumberAndIndex()`
(`aioethereum.management.eth.EthMixin`
method), 18

`eth_getTransactionByHash()`
(`aioethereum.management.eth.EthMixin`
method), 18

`eth_getTransactionCount()`
(`aioethereum.management.eth.EthMixin`
method), 19

`eth_getTransactionReceipt()`
(`aioethereum.management.eth.EthMixin`
method), 19

`eth_getUncleByBlockHashAndIndex()`
(`aioethereum.management.eth.EthMixin`
method), 19

`eth_getUncleByBlockNumberAndIndex()`
(`aioethereum.management.eth.EthMixin`
method), 19

`eth_getUncleCountByBlockHash()`
(`aioethereum.management.eth.EthMixin`
method), 19

`eth_getUncleCountByBlockNumber()`
(`aioethereum.management.eth.EthMixin`
method), 19

`eth_getWork()` (`aioethereum.management.eth.EthMixin`
method), 19

`eth_hashrate()` (`aioethereum.management.eth.EthMixin`
method), 20

`eth_mining()` (`aioethereum.management.eth.EthMixin`
method), 20

`eth_newBlockFilter()` (`aioethereum.management.eth.EthMixin`
method), 20

`eth_newFilter()` (`aioethereum.management.eth.EthMixin`
method), 20

`eth_newPendingTransactionFilter()`
(`aioethereum.management.eth.EthMixin`
method), 20

`eth_protocolVersion()` (`aioethereum.management.eth.EthMixin`
method), 20

`eth_sendRawTransaction()`
(`aioethereum.management.eth.EthMixin`
method), 20

`eth_sendTransaction()` (`aioethereum.management.eth.EthMixin`
method), 20

`eth_sign()` (`aioethereum.management.eth.EthMixin`
method), 21

`eth_submitHashrate()` (`aioethereum.management.eth.EthMixin`
method), 21

`eth_submitWork()` (`aioethereum.management.eth.EthMixin`
method), 21

`eth_syncing()` (`aioethereum.management.eth.EthMixin`
method), 21

`eth_uninstallFilter()` (`aioethereum.management.eth.EthMixin`
method), 21

`EthMixin` (class in `aioethereum.management.eth`), 15

M

`miner_setEtherBase()` (`aioethereum.management.miner.MinerMixin`
method), 22

`miner_setExtra()` (`aioethereum.management.miner.MinerMixin`
method), 22

`miner_setGasPrice()` (`aioethereum.management.miner.MinerMixin`
method), 22

`miner_start()` (`aioethereum.management.miner.MinerMixin`
method), 22

`miner_stop()` (`aioethereum.management.miner.MinerMixin`
method), 22

`MinerMixin` (class in `aioethereum.management.miner`),
22

N

`net_listening()` (`aioethereum.management.net.NetMixin`
method), 22

`net_peerCount()` (`aioethereum.management.net.NetMixin`
method), 22

`net_version()` (`aioethereum.management.net.NetMixin`
method), 22

`NetMixin` (class in `aioethereum.management.net`), 22

P

`personal_ecRecover()` (`aioethereum.management.personal.PersonalMixin`
method), 23

`personal_importRawKey()`
(`aioethereum.management.personal.PersonalMixin`
method), 23

`personal_listAccounts()` (`aioethereum.management.personal.PersonalMixin`
method), 23

`personal_lockAccount()` (`aioethereum.management.personal.PersonalMixin`
method), 23

`personal_newAccount()` (`aioethereum.management.personal.PersonalMixin`
method), 23

`personal_sendTransaction()`
(`aioethereum.management.personal.PersonalMixin`
method), 23

`personal_sign()` (`aioethereum.management.personal.PersonalMixin`
method), 24

`personal_unlockAccount()`
(`aioethereum.management.personal.PersonalMixin`
method), 24

`PersonalMixin` (class in
`aioethereum.management.personal`), 23

`pytest`, 26

Python Enhancement Proposals
PEP 3156, 1, 26

R

`rpc_call()` (aioethereum.BaseAsyncIOClient method), 14

S

`shh_addToGroup()` (aioethereum.management.shh.ShhMixin method), 24
`shh_getFilterChanges()` (aioethereum.management.shh.ShhMixin method), 24
`shh_getMessages()` (aioethereum.management.shh.ShhMixin method), 24
`shh_hasIdentity()` (aioethereum.management.shh.ShhMixin method), 24
`shh_newFilter()` (aioethereum.management.shh.ShhMixin method), 25
`shh_newGroup()` (aioethereum.management.shh.ShhMixin method), 25
`shh_newIdentity()` (aioethereum.management.shh.ShhMixin method), 25
`shh_post()` (aioethereum.management.shh.ShhMixin method), 25
`shh_uninstallFilter()` (aioethereum.management.shh.ShhMixin method), 25
`shh_version()` (aioethereum.management.shh.ShhMixin method), 25
`ShhMixin` (class in aioethereum.management.shh), 24

T

`txpool_content()` (aioethereum.management.txpool.TxpoolMixin method), 25
`txpool_inspect()` (aioethereum.management.txpool.TxpoolMixin method), 25
`txpool_status()` (aioethereum.management.txpool.TxpoolMixin method), 25
`TxpoolMixin` (class in aioethereum.management.txpool), 25

U

`ujson`, 26

W

`web3_clientVersion()` (aioethereum.management.web3.Web3Mixin method), 25
`web3_sha3()` (aioethereum.management.web3.Web3Mixin method), 26
`Web3Mixin` (class in aioethereum.management.web3), 25