

# Module API

MikksChain HDWallet

## 문서변경이력

[illegible]

## 목차

API List.....	4
실행 .....	4
기능 .....	4
MIB-0039 : BIP-0039 대체 알고리즘 .....	4
BIP-0032 & BIP-0044 적용 .....	4
예제 .....	5

## API List

require('./miks.js')	
miks.passphraseToSeed()	
miks.hdnnode.fromSeedBuffer()	

## 실행

node를 실행하고 miks.js 모듈을 변수로 등록하여 사용한다.

```
> miks = require('./miks.js')
```

## 기능

기능 설명이므로 세부 동작에 대한 내용은 'Miks 지갑 구조 문서'를 참고하세요.

MIB-0039 : BIP-0039 대체 알고리즘

1. RandomByte를 생성하고 그 값과 문장, 비밀번호를 조합하여 Seed를 생성한다.
2. Seed와 RandomByte를 반환한다.

```
> seed = miks.passphraseToSeed("사용자 정의 문장","비밀번호")  
  
> seed.seed // => Buffer로 정의된 Seed  
  
> seed.rng // => Buffer로 정의된 RandomByte
```

BIP-0032 & BIP-0044 적용

1. Seed 값을 이용하여 MasterNode 생성

```
> masterNode = miks.hdnnode.fromSeedBuffer(seed.seed)
```

2. MasterNode의 Data 출력

```

> extendedPrivateKey = masterNode.toBase58()

> extendedPublicKey = masterNode.neutered().toBase58()

> publicKey = masterNode.getPublicKeyBuffer().toString('hex')

> privateKey = masterNode.keyPair.d.toString(16)

> chainCode = masterNode.chainCode.toString('hex')

```

### 3. MasterNode로 부터 Child 파생

```

> ChildNode = MasterNode.derivePath("'44'/0'/0'/0/0'")

```

## 예제

지갑에 있는 example.js 파일입니다.

```

> node example.js

```

### example.js 실행화면

```

{ '입력값':
  { personalSentence: '테스트 문장입니다.',
    personalPassword: '테스트 비밀번호입니다.' },
  '출력값':
  { Password: '테스트 문장입니다. 테스트 비밀번호입니다.',
    seed: '874de1fcc3842ea1319d8aead02371b04263d434d9ab364262d3dae24c9e509b3eafe9aaa6ea654b7e25ba9efdc0b506ffffcf48fd75431d5fe69284fb7c0adbf',
    rng: 1056 },
  masterNode:
  HDNode {
    keyPair:
    ECPair {
      d: [Object],
      compressed: true,
      network: [Object],
      _0: [Object] },
    chainCode: <Buffer e8 b3 cb 48 77 73 3f 81 50 70 20 3a 09 8d 49 39 a9 2a 6c 07 37 04 c7 49 d0 c2 f3 c8 6b c2 c0 d5>,
    depth: 0,
    index: 0,
    parentFingerprint: 0 },
  '세팅정보':
  { privateKey: 'b29f4842f5c187fdcd889b114431907f82a7b4382bebd9c368dfe2fce1dec8f9',
    publicKey: '020521849667f79c7715d6bf422469ed775982a0502e256fc553c99f59f2815c15',
    chainCode: 'e8b3cb4877733f815070203a098d4939a92a6c073704c749d0c2f3c86bc2c0d5',
    extendedPrivateKey: 'xprv9s212r0H149K4NjMf01UCnSap6dD8Ro1G4XjwiGZvzXKTnZfZeyejH8x2oVaWjWb7F76Xk4ci54o2E0vGnG7oTx3xaUWrgGFRz6DteFrGo',
    extendedPublicKey: 'xpub661MyMwAqRbcGropmFYUZvPKN8ThXtWrdHTLk6aBVL4JLatp7CHuH5TRt3CwMUTokk46WbFCf4EmVL85Mw0Fh4CLfZcWcRSEbpj5wFrkCE2',
    walletAddress: '1Kqazx0wF68KazYjvs99qdCAfipZX7RqPX' } }

```