

SDK

Description: provides high level functionality to working with fusion mode

Real world example

```
import
{ FusionSDK , NetworkEnum }
from
'@1inch/fusion-sdk'
async
function
main ( )
{ const sdk =
new
FusionSDK ( { url :
'https://fusion.1inch.io' , network : NetworkEnum . ETHEREUM } )
const orders =
await sdk . getActiveOrders ( { page :
1 , limit :
2 } ) }
main ( )
```

Creation

Constructor arguments: params: FusionSDKConfigParams

interface

HttpProviderConnector

{ get < T

(url :

string) :

Promise < T

post < T

(url :

string , data :

unknown) :

Promise < T

}

interface

BlockchainProviderConnector

{ signTypedData (walletAddress :

```

string , typedData : EIP712TypedData ) :
    Promise < string
ethCall ( contractAddress :
    string , callData :
        string ) :
        Promise < string
    }
type
FusionSDKConfigParams
=
{ url :
    string network : NetworkEnum blockchainProvider ? : BlockchainProviderConnector httpProvider ? : HttpProviderConnector
    // by default we are using axios } Example with custom httpProvider:
import
{ api }
from
'my-api-lib'
class
CustomHttpProvider
implements
HttpProviderConnector
{ get < T
    ( url :
        string ) :
        Promise < T
    { return api . get ( url ) }
    post < T
    ( url :
        string , data :
        unknown ) :
        Promise < T
    { return api . post ( url , data ) } }

```

Methods

getActiveOrders

Description: used to get the list of active ordersArguments :

- [0]PaginationParams

Example:

```
import
{ FusionSDK , NetworkEnum }

from
'@1inch/fusion-sdk' const sdk =
new
FusionSDK ( { url :
'https://fusion.1inch.io' , network : NetworkEnum . ETHEREUM } ) const orders =
await sdk . getActiveOrders ( { page :
1 , limit :
2 } )
```

getOrdersByMaker

Description: used to get orders by maker

Arguments :

- [0]params: PaginationParams & {address: string}

Example:

```
import
{ FusionSDK , NetworkEnum }

from
'@1inch/fusion-sdk' const sdk =
new
FusionSDK ( { url :
'https://fusion.1inch.io' , network : NetworkEnum . ETHEREUM } )

const orders =
await sdk . getOrdersByMaker ( { page :
1 , limit :
2 , address :
'0xfa80cd9b3becc0b4403b0f421384724f2810775f' } )
```

getQuote

Description: Get quote details based on input data

Arguments:

- [0]params: QuoteParams

Example:

```
import
{ FusionSDK , NetworkEnum , QuoteParams }

from
'@1inch/fusion-sdk' const sdk =
new
```

```

FusionSDK ( { url :
'https://fusion.1inch.io' , network : NetworkEnum . ETHEREUM } )

const params =
{ fromTokenAddress :
'0x6b175474e89094c44da98b954eedeac495271d0f' , toTokenAddress :
'0xc02aaa39b223fe8d0a0e5c4f27ead9083c756cc2' , amount :
'1000000000000000000000' }

const quote =
await sdk . getQuote ( params )

```

placeOrder

Description: used to create a fusion order

Arguments:

- [0]params: OrderParams

Example:

```

const makerPrivateKey =
'0x123....' const makerAddress =
'0x123....'

const nodeUrl =
'....'

const blockchainProvider =
new
PrivateKeyProviderConnector ( makerPrivateKey , new
Web3 ( nodeUrl ) )

const sdk =
new
FusionSDK ( { url :
'https://fusion.1inch.io' , network :
1 , blockchainProvider } )

sdk . placeOrder ( { fromTokenAddress :
'0xc02aaa39b223fe8d0a0e5c4f27ead9083c756cc2' ,
// WETH toTokenAddress :
'0xa0b86991c6218b36c1d19d4a2e9eb0ce3606eb48' ,
// USDC amount :
'5000000000000000000' ,
// 0.05 ETH walletAddress : makerAddress , // fee is an optional field fee :
{ takingFeeBps :
100 ,

```

```
// 1% as we use bps format, 1% is equal to 100bps takingFeeReceiver :  
'0x0000000000000000000000000000000000000000'  
  
// fee receiver address } } ) . then ( console . log )
```

Types

PaginationParams

```
type  
PaginationParams  
=  
{ page ? :  
  number  
  
  // default is 1 limit ? :  
  number  
  
  // default is 2, min is 1, max is 500 }
```

QuoteParams

```
type  
QuoteParams  
=  
{ fromTokenAddress :  
  string toTokenAddress :  
  string amount :  
  string permit ? :  
  string  
  
  // a permit (EIP-2612) call data, user approval sign takingFeeBps ? :  
  number  
  
  // 100 == 1% }
```

OrderParams

```
enum PresetEnum { fast =  
  'fast' , medium =  
  'medium' , slow =  
  'slow' }  
  
type  
OrderParams  
=  
{ fromTokenAddress :  
  string toTokenAddress :  
  string amount :
```

```
string walletAddress :
string permit ? :
string
// a permit (EIP-2612) call data, user approval sign receiver ? :
string
// address preset ? : PresetEnum nonce ? : OrderNonce |
string
|
number
// allows to batch cancel orders. by default: not used fee ? : TakingFeeInfo }
export
type
TakingFeeInfo
=
{ takingFeeBps :
number
// 100 == 1% takingFeeReceiver :
string } Edit this page Previous Contract interaction Next Creating Fusion Orders
```