Smart Account V1 Methods

export

interface

IBiconomySmartAccount

extends

ISmartAccount

{ init (initilizationData ? : InitilizationData) :

Promise < this

; initializeAccountAtIndex (accountIndex :

number) :

void ; getExecuteCallData (to :

string , value : BigNumberish , data : BytesLike) :

string ; getExecuteBatchCallData (to :

Array < string

, value :

Array < BigNumberish

, data :

Array < BytesLike

,) :

string ; buildUserOp (transactions : Transaction [ ] , overrides ? : Overrides ,) :

Promise < Partial < UserOperation

; getAllTokenBalances (balancesDto : BalancesDto) :

Promise < BalancesResponse

; getTotalBalanceInUsd (balancesDto : BalancesDto) :

Promise < UsdBalanceResponse

; getSmartAccountsByOwner (smartAccountByOwnerDto : SmartAccountByOwnerDto ,) :

Promise < SmartAccountsResponse

; getTransactionsByAddress (chainId :

number , address :

string ,) :

Promise < SCWTransactionResponse [ ]

; getTransactionByHash (txHash :

string) :

Promise < SCWTransactionResponse

; getAllSupportedChains ( ) :

Promise < SupportedChainsResponse

; }

IBiconomySmartAccount Interface

TheIBiconomySmartAccount interface extends theISmartAccount interface and provides additional methods for interacting with a Biconomy Smart Account.

Method Parameters Description init initilizationData?: InitilizationData Initializes the smart account with the providedinitilizationData , if provided. Returns a Promise that resolves to the initializedIBiconomySmartAccount . initializeAccountAtIndex accountIndex: number Initializes the smart account at the specifiedaccountIndex . getExecuteCallData to: string, value: BigNumberish, data: BytesLike Returns the call data for executing a single transaction to the specified address (to) with the givenvalue anddata . getExecuteBatchCallData to: Array string , value: Array BigNumberish, data: Array BytesLike Returns the call data for executing a batch of transactions to multiple addresses (to) with the correspondingvalue anddata arrays. buildUserOp transactions: Transaction[], overrides?: Overrides Builds aUserOperation object from the provided array ofTransaction s. It also accepts optionaloverrides for specifying gas limits and gas prices. Returns a Promise that resolves to the partialUserOperation . getAllTokenBalances balancesDto: BalancesDto Retrieves the balances of multiple tokens for the smart account based on the providedbalancesDto . Returns a Promise that resolves to theBalancesResponse containing the token balances. getTotalBalanceInUsd balancesDto: BalancesDto Retrieves the total balance of all tokens in USD for the smart account based on the providedbalancesDto . Returns a Promise that resolves to theUsdBalanceResponse . getSmartAccountsByOwner smartAccountByOwnerDto: SmartAccountByOwnerDto Retrieves all smart accounts owned by the specified address based on the providedsmartAccountByOwnerDto . Returns a Promise that resolves to theSmartAccountsResponse containing the array of smart account addresses. getTransactionsByAddress chainId: number, address: string Retrieves the transactions associated with the specifiedaddress on the givenchainId . Returns a Promise that resolves to an array ofSCWTransactionResponse objects containing information about the transactions. getTransactionByHash txHash: string Retrieves the transaction details for the specified transaction hash (txHash). Returns a Promise that resolves to aSCWTransactionResponse object containing information about the transaction. getAllSupportedChains N/A Retrieves information about all supported chains. Returns a Promise that resolves to theSupportedChainsResponse containing the list of supported chains.

ISmartAccount Interface

TheISmartAccount interface provides essential methods for interacting with a Biconomy Smart Account.

import

{ UserOperation }

from

"@biconomy/core-types" ; import

{ UserOpResponse }

from

"@biconomy/bundler" ; export

interface

ISmartAccount

{ getSmartAccountAddress (accountIndex :

number) :

Promise < string

; signUserOp (userOperation : UserOperation) :

Promise < UserOperation

; sendUserOp (userOperation : UserOperation) :

Promise < UserOpResponse

; sendSignedUserOp (userOperation : UserOperation) :

Promise < UserOpResponse

; } Method Parameters Description getSmartAccountAddress accountIndex: number Retrieves the address of the

smart account at the specifiedaccountIndex . Returns a Promise that resolves to the smart account address. signUserOp userOperation: UserOperation Signs the givenUserOperation with the necessary cryptographic signature. Returns a Promise that resolves to the signedUserOperation . sendUserOp userOperation: UserOperation Sends the providedUserOperation to the Biconomy network for execution. Returns a Promise that resolves to aUserOpResponse containing the response from the network. sendSignedUserOp userOperation: UserOperation Sends the pre-signedUserOperation to the Biconomy network for execution. Returns a Promise that resolves to aUserOpResponse containing the response from the network. Previous Quickstart Next Tutorials