SdkPool

SDK class encapsulating stableswap pool functions.

This class will either interact with internal StableSwapFacet pools or external StableSwap pools depending on which type of pool is being used for each asset. Note: SDK currently only supports internal StableSwapFacet pools.

Hierarchy

- SdkShared
- ↳SdkPool
-

Methods

addLiquidity

▸addLiquidity (domainId ,tokenAddress ,amounts ,minToMint? ,deadline?):Promise <TransactionRequest

Prepares the transaction request for adding liquidity to a pool.

Parameters

Name Type Default value Description domainId string undefined The domain ID of the pool. tokenAddress string undefined The address of local or adopted token. amounts string [] undefined The amounts of the tokens to swap. minToMint string "0" (optional) The minimum acceptable amount of LP tokens to mint. deadline number undefined (optional) The deadline for the operation. Returns

Promise <TransactionRequest

providers.TransactionRequest object.

calculateAddLiquidityPriceImpact

▸calculateAddLiquidityPriceImpact (domainId ,tokenAddress ,amountX ,amountY):Promise <undefined |BigNumber

Calculates the price impact of adding liquidity to a pool.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. amountX string The amount of token X (index 0 of the pool), in the token's native precision. amountY string The amount of token Y (index 1 of the pool), in the token's native precision. Returns

Promise <undefined |BigNumber

Price impact for adding liquidity, in 1e18 precision.

calculatePriceImpact

▸calculatePriceImpact (tokenInputAmount ,tokenOutputAmount ,virtualPrice? ,isDeposit?):BigNumber

Calculates the price impact depending on whether liquidity is being deposited or withdrawn.

Parameters

Name Type Default value Description tokenInputAmount BigNumber undefined The amount of inbound tokens (LP tokens for withdrawals, total tokens for deposits, dx for swaps), in 1e18 precision. tokenOutputAmount BigNumber undefined The amount of outbound tokens (total tokens for withdrawals, LP tokens for deposits, dy for swaps), in 1e18 precision. virtualPrice BigNumber undefined (optional) The current virtual price of the pool. isDeposit boolean true (optional) Whether this is a deposit or withdrawal. Returns

BigNumber

The price impact.

calculateRemoveLiquidityPriceImpact

▸calculateRemoveLiquidityPriceImpact (domainId ,tokenAddress ,amountX ,amountY):Promise <undefined |BigNumber

Returns the price impact of removing liquidity from a pool.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. amountX string The amount of token X (index 0 of the pool), in the token's native precision. amountY string The amount of token Y (index 1 of the pool), in the token's native precision. Returns

Promise <undefined |BigNumber

The price impact for removing liquidity, in 1e18 precision.

calculateRemoveSwapLiquidityOneToken

▸calculateRemoveSwapLiquidity (domainId ,tokenAddress ,amount ,index):Promise <BigNumber

Calculates the amounts of underlying tokens returned.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. amount string The amount of the LP token to burn on withdrawal. index number The index of the token to withdraw. Returns

Promise <BigNumber

Calculated amount of underlying token returned.

calculateRemoveSwapLiquidity

▸calculateRemoveSwapLiquidity (domainId ,tokenAddress ,amount):Promise

Calculates the amounts of underlying tokens returned.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. amount string The amount of the LP token to burn on withdrawal. Returns

Promise

Array containing amount of each underlying token returned, in correct index order.

calculateSwap

▸calculateSwap (domainId ,tokenAddress ,tokenIndexFrom ,tokenIndexTo ,amount):Promise <BigNumber

Calculates the amount of tokens received on a swap.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. tokenIndexFrom number The index of the token to sell. tokenIndexTo number The index of the token to buy. amount BigNumberish The number of tokens to sell, in the "From" token's native decimal precision. Returns

Promise <BigNumber

Minimum amount received, in the "To" token's native decimal precision.

calculateSwapPriceImpact

▸calculateSwapPriceImpact (domainId ,amountX ,tokenX ,tokenY):Promise <BigNumber

Calculates the price impact of a swap.

Parameters

Name Type Description domainId string The domain ID of the pool. amountX string The amount of tokens to swap, in the token's native precision. tokenX string The address of the token to swap from. tokenY string The address of the token to swap to. Returns

Promise <BigNumber

The price impact for swapping, in 1e18 precision.

calculateTokenAmount

▸calculateTokenAmount (domainId ,tokenAddress ,amounts ,isDeposit?):Promise <BigNumber

Calculates the minimum LP token amount from deposits or withdrawals.

Parameters

Name Type Default value Description domainId string undefined The domain ID of the pool. tokenAddress string undefined The address of local or adopted token. amounts string [] undefined The amounts of the tokens to deposit/withdraw, in the correct index order and in each token's native precision. isDeposit boolean true (optional) Whether this is a deposit or withdrawal. Returns

Promise <BigNumber

Minimum LP tokens received, in 1e18 precision.

calculateYield

▸calculateYield (feesEarned ,principal ,days):Object

Calculates apr and apy.

Parameters

Name Type Description feesEarned number The total fees earned in the period. principal number The principal amount at the start of the period. days number The number of days to look back. Returns

Object

Object containing apr and apy.

Name Type apr number apy number

getAdopted

▸getAdopted (domainId ,tokenAddress):Promise <string

Reads the adopted token.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. Returns

Promise <string

The adopted token.

getDefaultDeadline

▸getDefaultDeadline ():number

Set to 1 hour from current time.

Returns

number

The default deadline, in unix time.

getLPTokenAddress

▸getLPTokenAddress (domainId ,tokenAddress):Promise <string

Reads the LP token address of a pool.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. Returns

Promise <string

The LP token address.

getPoolTokenAddress

▸getPoolTokenAddress (domainId ,tokenAddress ,index):Promise <string

Reads the token address of a specified index in a pool.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. index number The index of the token in the pool. Returns

Promise <string

The address of the specified token in the pool.

getPoolTokenBalance

▸getPoolTokenBalance (domainId ,tokenAddress ,poolTokenAddress):Promise <BigNumber

Reads the balance of a pool token.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. poolTokenAddress string The address of the pool token. Returns

Promise <BigNumber

The balance of the pool token.

getPoolTokenIndex

▸getPoolTokenIndex (domainId ,tokenAddress ,poolTokenAddress):Promise <number

Reads the index of a token in a pool.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of the local or adopted token. poolTokenAddress string The address of the token in the pool to get the index for. Returns

Promise <number

The index of the specified token in the pool or -1 if not found.

getPoolTokenDecimals

▸getPoolTokenDecimals (domainId ,tokenAddress ,poolTokenAddress):Promise <number

Reads the decimal precision of a token in a pool.

Parameters

Name Type Description domainId string The domain id of the pool. tokenAddress string The address of local or adopted token. poolTokenAddress string The address of the token in the pool to get the precision for. Returns

Promise <number

The decimal precision of the specified token in the pool or -1 if not found.

getRepresentation

▸getRepresentation (domainId ,tokenAddress):Promise <string

Reads the representation asset of the pool. The representation asset is the adopted asset on the canonical domain and local (nextAsset) otherwise.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. Returns

Promise <string

The representation (local or adopted) token.

getTokenPrice

▸getTokenPrice (tokenSymbol):Promise <any

Fetches the current price of a token.

Parameters

Name Type Description tokenSymbol string The symbol for the token. Returns

Promise <any

The price of the token.

getTokenSupply

▸getTokenSupply (domainId ,tokenAddress):Promise <BigNumber

Reads the ERC20 token supply.

Parameters

Name Type Description domainId string The domain ID of the ERC20 token. tokenAddress string The address of the ERC20 token. Returns

Promise <BigNumber

The balance of the address.

getTokenUserBalance

▸getTokenUserBalance (domainId ,tokenAddress ,userAddress):Promise <BigNumber

Reads the ERC20 token balance of an address.

Parameters

Name Type Description domainId string The domain ID of the ERC20 token. tokenAddress string The address of the ERC20 token. userAddress string The address to get the balance of. Returns

Promise <BigNumber

The balance of the address.

getUserPools

▸getUserPools (domainId ,userAddress):Promise <{info :Pool ;lpTokenBalance :BigNumber ;poolTokenBalances :BigNumber [] }[]>

Retrieves the Pools that a user has LP tokens for.

Parameters

Name Type Description domainId string The domain ID of the pool. userAddress string The address of the user to get the pools for. Returns

Promise <{info :Pool ;lpTokenBalance :BigNumber ;poolTokenBalances :BigNumber [] }[]>

Array of Pool objects.

getVirtualPrice

▸getVirtualPrice (domainId ,tokenAddress):Promise <BigNumber

Reads the virtual price of a pool.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. Returns

Promise <BigNumber

The virtual price, scaled to the pool's decimal precision (10^18).

getYieldStatsForDays

▸getYieldStatsForDays (domainId ,tokenAddress ,unixTimestamp ,days):Promise

Calculates the fees, liquidity, and volume of a pool for the days prior to the specified unix time.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. unixTimestamp number The unix time to start the look back from. days number The number of days to look back. Returns

Promise

Object containing fees, liquidity, and volume, in 1e18 precision.

removeLiquidity

▸removeLiquidity (domainId ,tokenAddress ,amount ,minAmounts? ,deadline?):Promise <TransactionRequest

Returns the transaction request for removing liquidity from a pool.

Parameters

Name Type Description domainId string The domain ID of the pool. tokenAddress string The address of local or adopted token. amount string The amount of LP tokens to burn. minAmounts string [] (optional) The minimum amounts of each token to withdraw. deadline number (optional) The deadline for the operation. Returns

Promise <TransactionRequest

providers.TransactionRequest object.

removeLiquidityOneToken

▸removeLiquidityOneToken (domainId ,tokenAddress ,amount ,minAmount? ,deadline?):Promise <TransactionRequest

Returns the transaction request for removing liquidity from a pool.

Parameters

Name Type Default value Description domainId string undefined The domain ID of the pool. tokenAddress string undefined The address of local or adopted token. withdrawTokenAddress string undefined The address of the token to withdraw. amount string undefined The amount of LP tokens to burn. minAmount string "0" (optional) The minimum acceptable amount of the token to withdraw. deadline number One hour from current time. (optional) The deadline for the operation. Returns

Promise <TransactionRequest

providers.TransactionRequest object.

removeLiquidityImbalance

▸removeLiquidityImbalance (domainId ,tokenAddress ,amounts ,maxBurnAmount? ,deadline?):Promise <TransactionRequest

Returns the transaction request for removing liquidity from a pool.

Parameters

Name Type Default value Description domainId string undefined The domain ID of the pool. tokenAddress string undefined The address of local or adopted token. amounts string [] undefined The amount of LP tokens to burn. maxBurnAmount string "0" (Use total LP balance) (optional) The max LP tokens the user is willing to burn. deadline number One hour from current time. (optional) The deadline for the operation. Returns

Promise <TransactionRequest

providers.TransactionRequest object.

swap

▸swap (domainId ,tokenAddress ,from ,to ,amount ,minDy? ,deadline?):Promise <TransactionRequest

Returns the transaction request for performing a swap in a pool.

Parameters

Name Type Default value Description domainId string undefined The domain ID of the pool. tokenAddress string undefined The address of local or adopted token. from string undefined The address of the token to sell. to string undefined The address of the token to buy. amount string undefined The amount of the selling token to swap. minDy number 0 (optional) The minimum amount of the buying token to receive. deadline number undefined (optional) The deadline for the operation. Returns

Promise <TransactionRequest

providers.TransactionRequest object.

create

▸Static create (_config):Promise <SdkPool

Create a singleton instance of the SdkPool class.

Parameters

Name Type Default value Description _config Object undefined SdkConfig object. _config.chains Recordundefined Chain config, at minimum with providers for each chain. _config.signerAddress string undefined Signer address for transactions. _config.logLevel "fatal" |"error" |"warn" |"info" |"debug" |"trace" |"silent" "info" (optional) Logging severity level. _config.network "testnet" |"mainnet" "mainnet" (optional) Blockchain environment to interact with. Returns

Promise <SdkPool

providers.TransactionRequest object.

Example

```

Copy import{ SdkPool }from"@connext/sdk";

constconfig={ signerAddress:"", network:"mainnet", chains:{ 6648936:{// the domain ID for Ethereum Mainnet providers:["https://rpc.ankr.com/eth"], }, 1869640809:{// the domain ID for Optimism providers:["https://mainnet.optimism.io"] }, 1886350457:{// the domain ID for Polygon providers:["https://polygon-rpc.com"] }, }, }

constsdkPool=awaitSdkPool.create(config);

```

See theDeployments page for all domain IDs and asset addresses.

Previous SdkBase Next SdkRouter Last updated3 months ago On this page *Hierarchy * Methods * addLiquidity * calculateAddLiquidityPriceImpact * calculatePriceImpact * calculateRemoveLiquidityPriceImpact * calculateRemoveSwapLiquidityOneToken * calculateRemoveSwapLiquidity * calculateSwap * calculateSwapPriceImpact * calculateTokenAmount * calculateYield * getAdopted * getDefaultDeadline * getLPTokenAddress * getPoolTokenAddress * getPoolTokenBalance * getPoolTokenIndex * getPoolTokenDecimals * getRepresentation * getTokenPrice * getTokenSupply * getTokenUserBalance * getUserPools * getVirtualPrice * getYieldStatsForDays * removeLiquidity * removeLiquidityOneToken * removeLiquidityImbalance * swap * create

Edit on GitHub