Query the Pool

The Pool shared object represents a market, such as a SUI/USDC market. That Pool is the only one representing that unique pairing (SUI/USDC) and the pairing is the only member of that particular Pool . See DeepBook Design to learn more about the structure of pools.

To perform trades, you pass a BalanceManager and TradeProof into the relevant Pool . Unlike Pool s, BalanceManager shared objects can contain any type of token, such that the same BalanceManager can access multiple Pool s to interact with many different trade pairings. See BalanceManager to learn more.

DeepBook exposes a set of endpoints that can be used to query any pool.

Accessor to check whether the pool is whitelisted.

Dry run to determine the quote quantity out for a given base quantity. Uses DEEP as fee.

Dry run to determine the base quantity out for a given quote quantity. Uses DEEP as fee.

Dry run to determine the quote quantity out for a given base quantity. Uses input token as fee.

Dry run to determine the base quantity out for a given quote quantity. Uses input token as fee.

Dry run to determine the quantity out for a given base or quote quantity. Only one out of base or quote quantity should be non-zero. Returns the (base_quantity_out , quote_quantity_out , deep_quantity_required).

Returns the DEEP required for an order if it's a taker or maker given quantity and price (deep_required_taker , deep_required_maker).

Returns the mid price of the pool.

Returns the order_id for all open orders for the balance_manager in the pool.

Returns vectors holding the prices (price_vec) and quantities (quantity_vec) for the level2 order book. The price_low and price_high are inclusive, all orders within the range are returned. is_bid is true for bids and false for asks.

Returns vectors holding the prices (price_vec) and quantities (quantity_vec) for the level2 order book. ticks are the maximum number of ticks to return starting from best bid and best ask. (bid_price , bid_quantity , ask_price , ask_quantity) are returned as four vectors. The price vectors are sorted in descending order for bids and ascending order for asks.

Get all balances held in this pool.

Get the ID of the pool given the asset types.

Returns the Order struct using the order ID.

Returns a vector of Order structs using a vector of order IDs.

Returns a vector of Order structs for all orders that belong to a BalanceManager in the pool.

Returns the locked balance for a BalanceManager in the pool (base_quantity , quote_quantity , deep_quantity).

Returns the trade parameters for the pool (taker_fee , maker_fee , stake_required).

Returns the trade parameters for the next epoch for the currently leading proposal of the pool (taker_fee , maker_fee , stake_required).

Returns the quorum needed to pass proposal in the current epoch.

Returns the book parameters for the pool (tick_size , lot_size , min_size).

Returns the OrderDeepPrice struct for the pool, which determines the conversion for DEEP fees.

API

DeepBook exposes a set of endpoints that can be used to query any pool.

Accessor to check whether the pool is whitelisted.

Dry run to determine the quote quantity out for a given base quantity. Uses DEEP as fee.

Dry run to determine the base quantity out for a given quote quantity. Uses DEEP as fee.

Dry run to determine the quote quantity out for a given base quantity. Uses input token as fee.

Dry run to determine the base quantity out for a given quote quantity. Uses input token as fee.

Dry run to determine the quantity out for a given base or quote quantity. Only one out of base or quote quantity should be non-zero. Returns the (base_quantity_out , quote_quantity_out , deep_quantity_required).

Returns the DEEP required for an order if it's a taker or maker given quantity and price (deep_required_taker , deep_required_maker).

Returns the mid price of the pool.

Returns the order_id for all open orders for the balance_manager in the pool.

Returns vectors holding the prices (price_vec) and quantities (quantity_vec) for the level2 order book. The price_low and price_high are inclusive, all orders within the range are returned. is_bid is true for bids and false for asks.

Returns vectors holding the prices (price_vec) and quantities (quantity_vec) for the level2 order book. ticks are the maximum number of ticks to return starting from best bid and best ask. (bid_price , bid_quantity , ask_price , ask_quantity) are returned as four vectors. The price vectors are sorted in descending order for bids and ascending order for asks.

Get all balances held in this pool.

Get the ID of the pool given the asset types.

Returns the Order struct using the order ID.

Returns a vector of Order structs using a vector of order IDs.

Returns a vector of Order structs for all orders that belong to a BalanceManager in the pool.

Returns the locked balance for a BalanceManager in the pool (base_quantity , quote_quantity , deep_quantity).

Returns the trade parameters for the pool (taker_fee , maker_fee , stake_required).

Returns the trade parameters for the next epoch for the currently leading proposal of the pool (taker_fee , maker_fee , stake_required).

Returns the quorum needed to pass proposal in the current epoch.

Returns the book parameters for the pool (tick_size , lot_size , min_size).

Returns the OrderDeepPrice struct for the pool, which determines the conversion for DEEP fees.