

ChainlinkCalculator

Quick links

[Functions](#)[Errors](#)

Functions

doublePrice

```
function doublePrice(contract AggregatorV3Interface oracle1, contract AggregatorV3Interface oracle2, int256 decimalsScale, uint256 amount) external view returns (uint256 result)
```

Calculates price of token A relative to token B. Note that order is important

Return Values

Name	Type	Description
result	uint256	Token A relative price times amount

getMakingAmount

```
function getMakingAmount(struct IOrderMixin.Order, bytes, bytes32, address, uint256 takingAmount, uint256, bytes extraData) external view returns (uint256)
```

getTakingAmount

```
function getTakingAmount(struct IOrderMixin.Order, bytes, bytes32, address, uint256 makingAmount, uint256, bytes extraData) external view returns (uint256)
```

_getSpreadAmount

`function _getSpreadedAmount(uint256 amount, bytes blob) internal view returns (uint256)`

Calculates price of token relative to oracle unit (ETH or USD) The first byte of the blob contain inverse and useDoublePrice flags, The inverse flag is set when oracle price should be inverted, e.g. for DAI-ETH oracle, inverse=false means that we request DAI price in ETH and inverse=true means that we request ETH price in DAI The useDoublePrice flag is set when needs price for two custom tokens (other than ETH or USD)

Return Values

Name	Type	Description
[0]	uint256	Amount _ spread _ oracle price

_doublePrice

`function _doublePrice(contract AggregatorV3Interface oracle1, contract AggregatorV3Interface oracle2, int256 decimalsScale, uint256 amount) internal view returns (uint256 result)`

Errors

DifferentOracleDecimals

`error DifferentOracleDecimals()`

StaleOraclePrice

`error StaleOraclePrice()`

Previous

[< ApprovalPreInteraction](#)

Next

[DutchAuctionCalculator >](#)