

DIA

DIA token price feeds provide smart contract real-time price information of 3,000+ cryptocurrencies, transparently sourced from 80+ trusted, high-volume DEXs and CEXs. The feeds facilitate the development of DeFi use cases such as money markets, lending/borrowing, synthetic asset issuance, options, derivatives and futures markets, and many more.

How to access DIA's oracle?

Here is an example of how to retrieve price value from a standard DIA oracle. For the purpose of this example, we will be using the following demo oracle on Ethereum: [0xa935...5856](#).

1. Access any DIA oracle smart contract.
2. Call `getValue(pair_name)` with `pair_name` being the full pair name such as BTC/USD. You can use the "Read" section on Etherscan to execute this call.
3. The response of the call contains four values:
 - The current asset price in USD with a fix-comma notation of 8 decimals.
 - the UNIX timestamp of the last update.

Oracle Integration Example

Here is an example on how you can integrate DIA's oracle into your smart contract with Solidity: `pragma solidity ^0.8.13;`

```
interface IDIAOracleV2{
    function getValue(string memory) external returns (uint128, uint128);
}

contract IntegrationSample{

    address immutable ORACLE = 0xa93546947f3015c986695750b8bbEa8e26D65856;
    uint128 public latestPrice;
    uint128 public timestampOflatestPrice;

    function getPriceInfo(string memory key) external {
        (latestPrice, timestampOflatestPrice) = IDIAOracleV2(ORACLE).getValue(key);
    }

    function checkPriceAge(uint128 maxTimePassed) external view returns (bool inTime){
        if((block.timestamp - timestampOflatestPrice) < maxTimePassed){
            inTime = true;
        } else {
```

```
        inTime = false;
    }
}
```

Find more detailed description of the functions and how to run test in this [GitHub repository](#)

DIA has a dedicated Solidity library to facilitate integration of DIA oracles in your own contracts. The library consists of two functions, getPrice and getPriceIfNotOlderThan. You can learn more about the library and how to use it in the [DIA documentation](#).