

Uniswap compatible interface

Only for volatile pairs

Uniswap compatible interface for volatile pairs

Velocore supports a Uniswap-compatible interface. The difference is that Velocore doesn't use WETH internally. Use `address(0)` in place of ETH.

- This is a wrapper function for the direct execute function for easy swap integration. You could check the exact code how it is wrapped [in the repo here](#)
- .
- This only supports volatile pools. Can't swap on, for example, USDC-USDT pools with this interface.
- Like all the other functions on Velocore, you could call Vault contract to use this function.
-

...

Copy

```
functionswapExactTokensForTokens( uint256amountIn, uint256amountOutMin, address[]calldatapath, addresssto,
uint256deadline )externalreturns(uint256[]memoryamounts); functionswapTokensForExactTokens( uint256amountOut,
uint256amountInMax, address[]calldatapath, addresssto, uint256deadline )externalreturns(uint256[]memoryamounts);
functionswapExactETHForTokens(uint256amountOutMin,address[]calldatapath,addresssto,uint256deadline) external payable
returns(uint256[]memoryamounts); functionswapTokensForExactETH( uint256amountOut, uint256amountInMax,
address[]calldatapath, addresssto, uint256deadline )externalreturns(uint256[]memoryamounts);
functionswapExactTokensForETH( uint256amountIn, uint256amountOutMin, address[]calldatapath, addresssto,
uint256deadline )externalreturns(uint256[]memoryamounts);
functionswapETHForExactTokens(uint256amountOut,address[]calldatapath,addresssto,uint256deadline) external payable
returns(uint256[]memoryamounts);
```

```
functiongetAmountsOut(uint256amountIn,address[]calldatapath)externalreturns(uint256[]memoryamounts);
functiongetAmountsIn(uint256amountOut,address[]calldatapath)externalreturns(uint256[]memoryamounts);
```

```
functiongetPair(addresss0,addresss1)externalviewreturns(address);
```

```
functionallPairs(uint256i)externalviewreturns(address);
```

```
functionallPairsLength()externalviewreturns(uint256); ````
```

...

Last updated4 months ago On this page