

Voting

Vote/Unvote

- Token : vevc
- op type : Vote(0x03)
- Interacting pool = gauge address
-

Vote interacts with gauge. Gauge is same with pool address for volatile pairs. Check [Stake](#) to know how to get stable pool's gauge address.

Use positive amount to vote and negative amount to unvote.

...

```
Copy // you vote with vevc token which is ERC20 not ERC721 Token[]memorytokens=newToken
tokens[0]=toToken(IERC20(vevc));
```

```
VelocoreOperation[]memoryops=newVelocoreOperation; // Vote interacts with gauge. Gauge is same with pool for volatile
pairs. // Check Stake section of the docs to find out way to get stablepool's gauge address.
int128voteAmount=int128(int256(IERC20(vevc).balanceOf(address(this)))); ops[0].poolId=toPoolId(VOTE,usdc_eth_pool);
ops[0].tokenInformations=newbytes32; //Vote : positive flow in gauge's perspective. Unvote : negative flow. so just put - in
the amount to unvote. ops[0].tokenInformations[0]=toTokenInfo(0x00,EXACTLY,voteAmount); ops[0].data="";
returnexecute(tokens,newint128,ops);
```

...

Harvesting vote reward

If the gauge has already been voted on, there will be a voting reward to claim.

In this case, you'll also need to send additional tokenInfo in the Op to claim it. If you don't, it will be credited to your internal balance and you can claim it later by performing a separate claim operation.

- Token : Since voting rewards are accumulated in the LP itself, you need to include the pool LP address.
- opType : VOTE(0x03)
- Interacting pool : gauge address
- use AT_MOST 0 for amount since you are receiving.
-

...

```
Copy // Same as vote. just lp token is added on the operation. // If you are not changing the vote, just send lp token
operation without vevc. Token[]memorytokens=newToken; tokens[0]=toToken(IERC20(vevc));
tokens[1]=toToken(IERC20(usdc_eth_pool));
```

```
VelocoreOperation[]memoryops=newVelocoreOperation;
int128voteAmount=int128(int256(IERC20(vevc).balanceOf(address(this)))); ops[0].poolId=toPoolId(VOTE,usdc_eth_pool); //
this part is added. send 2 tokenInformation instead of 1. ops[0].tokenInformations=newbytes32;
ops[0].tokenInformations[0]=toTokenInfo(0x00,EXACTLY,voteAmount); // Since you don't know how many tokens you will
claim, but you are sure that you are receiving, not giving, // so use AT_MOST 0 for the LP token amount.
ops[0].tokenInformations[1]=toTokenInfo(0x01,AT_MOST,0); ops[0].data="";
```

```
returnexecute(tokens,newint128,ops);
```

...

Last updated 4 months ago On this page