

Flashloan

Velocore supports flashloans, and they are completely free. The benefit of using our flashloan is that since it is treated as a part of normal swaps, you can repay with any tokens you want, and chain it with normal swaps or pay the difference with tokens in your wallet.

You can request flashloan with [vault.execute\(\)](#) , specifying the recipient as a converter pool.

continuing from [this code](#) :

...

```
Copy await compileAndExecute(0, [ poolId(2, your_flashloan_recipient), // flashloan [eth, "flashloan", INT128_MAX], // flashloan
entire balance of the vault [dai, "flashloan", INT128_MAX], [usdt, "exactly", 12340560], // you can also specify the amount to
borrow [vc, "exactly", 0], // if you are planning to use another token to repay, // you must specify that token too. ]], // at this
point, the internal balance will be (tokens returned) - (tokens borrowed). // if you return less than what you've borrowed, they
may be negative. ])
```

...

to receive flashloan, your contract must implement the method below:

...

```
Copy function velocore__convert(address user, Token[] calldata tokens, int128[] memory, bytes calldata data) external {
require(msg.sender == VAULT_ADDRESS);
```

```
// do something with the tokens
```

```
// return tokens for (uint256 i = 0; i < tokens.length; i++) {
tokens[i].transferFrom(address(this), msg.sender, tokens[i].balanceOf(address(this))); } }
```

...

Last updated 5 months ago On this page