

# Canceling a limit order

It is assumed that RFQ orders will be created with a short lifetime. But, if it becomes necessary to cancel the created RFQ order, then this can be done as follows:

## Parameters:

Parameter Type Description info string information about an RFQ order (see above in section [RFQ structure](#) )

## Creating with a typescript/javascript:

```
import Web3 from
'web3' ; import
{ LimitOrderProtocolFacade , Web3ProviderConnector , RFQOrder }
from
'@1inch/limit-order-protocol-utils' ;
const contractAddress =
'0x7643b8c2457c1f36dc6e3b8f8e112fdf6da7698a' ; const walletAddress =
'0xd337163ef588f2ee7cdd30a3387660019be415c9' ;
const web3 =
new
Web3 ( '...' ) ; // You can create and use a custom provider connector (for example: ethers) const connector =
new
Web3ProviderConnector ( web3 ) ;
const limitOrderProtocolFacade =
new
LimitOrderProtocolFacade ( contractAddress , chainId , connector , ) ;
const RFQOrder : RFQOrder =
{ ... } ;
const callData = limitOrderProtocolFacade . cancelRFQOrder ( RFQOrder . info ) ;
// Send transaction for the RFQ order canceling // Must be implemented sendTransaction ( { from : walletAddress , gas :
50_000 ,
// Set your gas limit gasPrice :
600000000 ,
// Set your gas price to : contractAddress , data : callData , } ) ;
```

## Canceling via CLI (with arguments):

```
gasPrice - in units of GWEI
npx limit-order-rfq-utils -- \ --operation = cancel \ --chainId = 56
\ --privateKey = { xxx }
\ --gasPrice = 6
\ --orderInfo = 29941961886664662336741887180811
```

## Canceling via CLI (through prompt):

`npx limit-order-rfq-utils` As result, you will receive a link to the transaction hash.[Edit this page](#) [Previous](#) [Filling a RFQ order](#)  
[Next](#) [Breaking changes in V3](#)