

SNIP721

[SecretNET.NFT](#) is a layer on top of the Secret.NET which supports all methods of the [reference implementation](#) of the [SNIP721 contract](#) .

nuget install SecretNET.NFT

Add Minter

...

Copy `varsnip721Client=newSecretNET.NFT.Snip721Client(secretClient);`// SecretNET.NFT

```
varaddMinterMsg=newSecretNET.NFT.AddMinterRequest(minters:new[] {""});
varaddMinter=awaitsnip721Client.Tx.AddMinter(
msg:newSecretNET.NFT.MsgAddMinter(addMinterMsg,contractAddress,codeHash), txOptions:newTxOptions() {
GasLimit=100_000});
```

...

Mint SNIP721 Token

...

Copy `varmintNftMsg=SecretNET.NFT.MintNftRequest.Create(tokenId:"1");` `varmintNft=awaitsnip721Client.Tx.MintNft(`
`msg:newSecretNET.NFT.MsgMintNft(mintNftMsg,contractAddress,codeHash), txOptions:newTxOptions() {`
`GasLimit=200_000});`

...

Transfer SNIP721 Token

...

Copy `vartransferNftMsg=newSecretNET.NFT.TransferNftRequest(recipient,tokenId);`
`vartransferNft=awaitsnip721Client.Tx.TransferNft(`
`msg:newSecretNET.NFT.MsgTransferNft(transferNftMsg,contractAddress,codeHash), txOptions:newTxOptions() {`
`GasLimit=50_000});`

...

Query Tokens with Permit

...

Copy `varpermit=awaitsecretClient.Permit.Sign(owner:wallet.Address, chainId:secretClient.ChainId, permitName:"test",`
`allowedContracts:newstring[] { contractAddress }, permissions:newPermissionType[] { PermissionType.Owner });`

`vartokens=awaitsnip721Client.Query.GetTokens(contractAddress,wallet.Address,permit:permit,codeHash:codeHash);`

...

Query Tokens with Viewing Key

...

Copy `vartxExec=awaitsnip20Client.Tx.SetViewingKey(contractAddress, "hello", txOptions:newTxOptions() {`
`GasLimit=100_000});`

`vartokens=awaitsnip721Client.Query.GetTokens(contractAddress,wallet.Address,viewingKey:"hello",codeHash:codeHash);`

...

Last updated1 year ago On this page * [Add Minter](#) * [Mint SNIP721 Token](#) * [Transfer SNIP721 Token](#) * [Query Tokens with Permit](#) * [Query Tokens with Viewing Key](#)

Was this helpful? [Edit on GitHub](#) [Export as PDF](#)