

Account

You can interact with, create or delete NEAR accounts.

Load Account

This will return an `Account` object for you to interact with.

```
const account =
```

```
await nearConnection . account ( "example-account.testnet" ) ;ClassAccount
```

Create Account

```
// create a new account using funds from the account used to create it. const account =
```

```
await nearConnection . account ( "example-account.testnet" ) ; await account . createAccount ( "example-account2.testnet" ,
```

```
// new account name "8hSHprDq2StXwMtNd43wDTXQYsjXcD4MJTXQYsjXcc",
```

```
// public key for new account "1000000000000000000000"
```

```
// initial balance for new account in yoctoNEAR ) :MethodAccount.createAccount
```

Delete Account

```
// deletes account found in the account object // transfers remaining account balance to the accountId passed as an argument
const account =
```

```
await nearConnection . account ( "example-account.testnet" ) ; await account . deleteAccount ( "beneficiary-account.testnet"
) ; MethodAccount.deleteAccount
```

Get Account Balance

```
// gets account balance const account =
```

```
await nearConnection . account ( "example-account.testnet" ) ; await account . getAccountBalance ( ) ;
MethodAccount.getAccountBalance
```

Get Account details

Returns information about an account, such as authorized apps.

```
// gets account details in terms of authorized apps and transactions const account =
```

```
await nearConnection . account ( "example-account.testnet" ) ; await account . getAccountDetails ( ) ;
MethodAccount.getAccountDetails
```

Deploy a Contract

You can deploy a contract from a compiled WASM file. This returns an object with transaction and receipts outcomes and status.

```
const account =
```

```
await nearConnection . account ( "example-account.testnet" ) ; const transactionOutcome =
```

await account . deployContract (fs . readFileSync ("example-file.wasm")) [MethodAccount.deployContract](#)
[InterfaceFinalExecutionOutcome](#)

Send Tokens

Transfer NEAR tokens between accounts. This returns an object with transaction and receipts outcomes and status.

```
const account =
```

```
await nearConnection . account ( "sender-account.testnet" ) ; await account . sendMoney ( "receiver-account.testnet" ,
```

```
// receiver account "10000000000000000000000000000000"
```

// amount in yctoNEAR) ;[MethodAccount.sendMoney InterfaceFinalExecutionOutcome](#)

State

Get basic account information, such as amount of tokens the account has or the amount of storage it uses.

const account =

await nearConnection . account ("example-account.testnet") ; const accountState =

await account . state () ;[MethodAccount.state InterfaceAccountView](#)

Access Keys

You can get and manage keys for an account.

Add Full Access Key

// takes public key as string for argument const account =

await nearConnection . account ("example-account.testnet") ; await account . addKey ("8hSHprDq2StXwMtNd43wDTXQYsjXcD4MJTXQYsjXcc") ; [MethodAccount.addKey](#)

Add Function Access Key

const account =

await nearConnection . account ("example-account.testnet") ; await account . addKey ("8hSHprDq2StXwMtNd43wDTXQYsjXcD4MJTXQYsjXcc" ,

// public key for new account "example-account.testnet" ,

// contract this key is allowed to call (optional) "example_method" ,

// methods this key is allowed to call (optional) "25000000000000"

// allowance key can use to call methods (optional)) ;[MethodAccount.addKey](#)

Get All Access Keys

const account =

await nearConnection . account ("example-account.testnet") ; await account . getAccessKeys () ; [MethodAccount.getAccessKeys InterfaceAccessKeyInfoView](#)

Delete Access Key

const account =

await nearConnection . account ("example-account.testnet") ; await account . deleteKey ("8hSHprDq2StXwMtNd43wDTXQYsjXcD4MJTXQYsjXcc") ; [MethodAccount.deleteKey](#) [Edit this page](#) Last updated on Jan 31, 2024 by gagdiez Was this page helpful? Yes No

[Previous Wallet](#) [Next Contract](#)