Ardor Developers Cheat Sheet

Java

Conversion Functions

From	To	Code
Secret Phrase	Public Key	`Crypto.getPublicKey(String s)`
Public Key	Numeric ID	`Account.getId(byte[] pk)`
Numeric ID	Public Key	`Account.getPublicKey(long id)`
Numeric ID	String ID	`Long.toUnsignedString(long id)`
String ID	Numeric ID	`Convert.parseUnsignedLong(String s)`
Numeric ID	RS Format	`Convert.rsAccount(id)`
RS Format or String ID	Numeric ID	`Convert.parseAccountId(String s)`
byte[]	Hex String	`Convert.toHexString(b)`
Hex String	byte[]	`Convert.parseHexString(hex)`
Full Hash	ID	`Convert.fullHashToId(byte[] b)`
Epoch Time	Time Millis	`Convert.fromEpochTime(int t)`
Time Millis	Epoch Time	`Convert.toEpochTime(long t)`
String	byte[]	`Convert.toString(byte[] b, Boolean isText)`
byte[]	String	`Convert.toBytes(String s, boolean isText)`

Common Operations

Operation	Code	Return
Sign	`Crypto.sign(byte[] message, String secretPhrase)`	byte[]
Verify	`Crypto.verify(byte[] signature, byte[] message, byte[] publicKey)`	Boolean
Encrypt	`Account.encryptTo(byte[] publicKey, byte[] data, String senderSecretPhrase, boolean compress)`	`EncryptedData`
Decrypt	`Account.decryptFrom(EncryptedData encryptedData, String recipientSecretPhrase, boolean uncompress)`	`byte[]`
Sha256	`Crypto.sha256().digest(byte[] b)`	`byte[]`
Compress	`Convert.compress(byte[] bytes)`	`byte[]`
Uncompress	`Convert.uncompress(byte[] bytes)`	`byte[]`

Useful Constants - see nxt.Constants class

Reading from properties file

`Nxt.getBooleanProperty(), Nxt.getIntProperty(), Nxt.getStringProperty(), Nxt.getStringListProperty()`

Ardor Developers Cheat Sheet

Javascript

Conversion Functions

From	To	Code
Secret Phrase	Public Key	`NRS.getPublicKey(converters.stringToHexString(secretPhrase))`
Public Key	String ID	`NRS.getAccountIdFromPublicKey(publicKeyHexString)`
String ID	Public Key	`NRS.getPublicKey(id, true)`
String ID	RS Format	`NRS.convertNumericToRSAccountFormat(id)`
byte[]	Hex String	`converters.byteArrayToHexString(bytes)`
Hex String	byte[]	`converters.hexStringToByteArray(hex)`
Full Hash	ID	`NRS.fullHashToId(fullHash)`
Epoch Time	Time Millis	`NRS.fromEpochTime(epochTime)`
Time Millis	Epoch Time	`NRS.toEpochTime(time)`
String	byte[]	`NRS.getUtf8Bytes(String), converters.stringToByteArray(String)`
byte[]	String	`converters.byteArrayToString(bytes)`
NQT	NXT	`NRS.intToFloat(amountNQT, decimals)`
NQT	formatted	`NRS.formatQuantity(amountNQT, decimals)` – many different variants
NXT	NQT	`NRS.floatToInt(amountNXT, decimals)`

Common Operations

Operation	Code	Return
Sign	`NRS.signBytes(hexStringMessage, secretPhrase)`	`Hex String`
Verify	`NRS.verifySignature(hexSignature, hexStringMessage, hexPublicKey, callback)`	`boolean`
Encrypt	`NRS.encryptNote(), NRS.encryptData()` – see `test.nrs.encryption.js`	`encrypted message, nonce`
Decrypt	`NRS.decryptNote(), NRS.decryptData()` – see `test.nrs.encryption.js`	`Decrypted message, shared key`
Sha256	`CryptoJS.algo.SHA256.create(), update(), finalize()`	`word array`
Compress	`pako.gzip(new Uint8Array(bytes))`	`byte array`
Uncompress	`pako.inflate(new Uint8Array(bytes))`	`byte array`

Useful Constants

See nrs.constants.js

Reading account settings

`NRS.settings[<setting name>]` – account specific
`NRS.mobileSettings[<setting name>]` – device specific