**Omm SCORE APIs Documentation**

1. **Readonly methods**
   1. Get all addresses ( USDb, sICX, oICX , oUSDb, LendingPool, LendingPoolDataProvider)

*{ “to” : AddressProvider ,*

*“method”: “getAddress”,*

*“params”: { “\_name”: “USDb”}*

*}*

Example response:

*cx3444…...*

* 1. Get reserve data for a specific reserve

*{ “to” : LendingPoolDataProvider*

*“method”: “getReserveData”,*

*“params”: {“\_reserve”: Address using 1 a for USDb and sICX }*

*}*

Example response:

*{"totalLiquidity": 100000 ,*

*"availableLiquidity":1000,*

*“liquidityRate”: 3.4,*

*“borrowRate”: 5.6,*

*“utilizationRate”: 65.5,*

*“liquidityRate”: 0.5,*

*“oTokenAddress”: “cx8u88…”,*

*“lastUpdateTimestamp” : 23454543 }*

* 1. Get user reserve data for a specific reserve

*{ “to” : LendingPoolDataProvider,*

*“method”: “getUserReserveData”,*

*“params”: {“\_reserve”: Address using 1 a for USDb and sICX ,*

*“\_user”: “hx09dd……”}*

*}*

Example response:

*{"currentOTokenBalance": 100000 ,*

*"currentBorrowBalance":1000,*

*“principalBorrowBalance”: 900,*

*“liquidityRate”: 3.4,*

*“borrowRate”: 5.6,*

*“originationFee”: 0.0001,*

*“variableBorrowIndex”: 0.1*

*“lastUpdateTimestamp” : 23454543,*

*“usageAsCollateralEnabled”: True }*

* 1. Get user account data

*{ “to” : LendingPoolDataProvider,*

*“method”: “getUserAccountData”,*

*“params”: {“\_user”: “hx09dd……”}*

*}*

Example response:

*{"totalLiquidityICX": 100000 ,*

*"totalCollateralICX":1000,*

*“totalBorrowsICX”: 100,*

*“totalFeesICX”: 5.6,*

*“availableBorrowsICX”:100,*

*“currentLiquidationThreshold”: 0.65,*

*“ltv” :0.75,*

*“healthFactor”: 1.2 }*

* 1. Get reserve configuration data

*{ “to” : LendingPoolDataProvider,*

*“method”: “getReserveConfigurationData”,*

*“params”: {“reserve”: “Address using 1 a for USDb and sICX ”}*

*}*

Example response:

*{"ltv": 0.6*

*"liquidationThreshold":0.65,*

*“liquidationBonus”: 0.1,*

*“usageAsCollateralEnabled”: True,*

*“borrowingEnabled”:True,*

*“isActive”: True,*

*}*

1. **External methods**
   1. [Deposit ICX](https://app.lucidchart.com/invitations/accept/e2de6c8f-91c4-4ddd-a07a-8186a89cd664)

*{ “to” : from response in 1 a for LendingPool,*

*“method”: “deposit”,*

*“params”: { “\_amount”: 10 \*\* 18 },*

*“value”: 10 \*\* 18*

*}*

* 1. [Deposit USDb](https://app.lucidchart.com/invitations/accept/65cdc890-fb7e-4ddc-8d33-75259c2a17d8)

*{ “to” : from response in 1 a for USDb,*

*“method”: “transfer”,*

*“params”: {“\_to”: from response in 1 a for LendingPool,*

*“\_value”: 10 \*\* 18,*

*“\_data”: bytes('{ "method": "deposit", "params": {“\_amount": 10 \*\* 18}}') })*

*}*

*}*

* 1. [Redeem USDb](https://app.lucidchart.com/invitations/accept/0bd317b0-4214-433a-9817-97955def300b)

{ *“to” : from response in 1 a for oUSDb,*

*“method”: “redeem”,*

*“params” : {“\_amount” : 10 \*\* 18}*

}

* 1. [Redeem ICX](https://app.lucidchart.com/invitations/accept/1f2b8e6a-b3b8-4be7-ac91-037f0d6b1e9e) (

{ *“to” : from response in 1 a for oICX,*

*“method”: “redeem”,*

*“params” : {“\_amount” : 10 \*\* 18,*

*“\_waitForUnstaking”: True}*

}

*\*\*Note: If \_waitForUnstaking is set to true, it waits for an unstaking period to transfer ICX amount back to the user wallet else if set to false then it transfers the equivalent amount of sICX to the user wallet instantly. \*\**

* 1. [Borrow](https://app.lucidchart.com/invitations/accept/5916d025-5030-4d44-8860-c7548759acaa) (

{ *“to” : from response in 1 a for LendingPool,*

*“method”: “borrow”,*

*“params” : {“\_amount” : 10 \*\* 18,*

*“\_reserveAddress”: from response in 1 a for USDb or sICX}*

}

* 1. [Repay](https://app.lucidchart.com/invitations/accept/82ec9b41-e0db-463e-a933-1040c4cf8177) (

{ *“to” : from response in 1 a for USDb or sICX,*

*“method”: “transfer”,*

*“params”: {“\_to”: from response in 1 a for LendingPool,*

*“\_value”: 10 \*\* 18,*

*“\_data”: bytes('{ "method": "repay", "params": {“\_reserveAddress“:from response in 1 a for USDb or sICX, \_amount": 10 \*\* 18}}') }*

*}*

* 1. [Liquidation](https://app.lucidchart.com/invitations/accept/746bb070-ab05-4966-aad4-0ca369ecde19) (

{ *“to” : from response in 1 a for USDb or sICX,*

*“method”: “transfer”,*

*“params”: {“\_to”: from response in 1 a for LendingPool,*

*“\_value”: 10 \*\* 18,*

*“\_data”: bytes('{ "method": "liquidationCall",*

*"params": {“\_collateral“:from response in 1 a for USDb or sICX,*

*“\_reserve”: from response in 1 a for sICX or USDb,*

*”\_user”:”hx4555...”,*

*“\_purchaseAmount": 10 \*\* 18}}') }*

*}*