コントラクト設計

* コントラクト名 : UserArea
* 概要 : ユーザーの地域登録、参照。
* 関数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 関数名 | 引数 | 返り値 | 修飾詞 | 概要 |
| areaRegister | String areaName | 無し | Public |  |
| getAreaName | Address userAddr | String | View, internal |  |

* コントラクト名 : Times
* 概要 : 現在時刻の取得など。
* 関数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 関数名 | 引数 | 返り値 | 修飾詞 | 概要 |
| areaRegister | 無し | uint | internal | 現在時刻の取得 |
|  |  |  |  |  |

* コントラクト名 : IoTDataManager(UserAreaを継承)
* 概要 : データの登録、参照などを行うためのコントラクト。

データの種類（温度、湿度など）はIDで管理。

* 関数 :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 関数名 | 引数 | 返り値 | 修飾詞 | 概要 |
| SetDataInfo | Int64 value,  Int dataID  uint time | 無し | Internal | データの登録用 |
| SetDataName | String dataName | 無し | Ownable | データの種類登録 |
| getDataInfoByID | Int id | DataInfo構造体 | Internal,  View | データ検索 |
| getDataInfoByAddr | Address  Addr | DataInfo構造体の配列 | Internai,  View | データ検索 |
| getDataInfoByAreaName | String  areaName | DataInfo構造体の配列 | Internal,  view | データ検索 |
| getAllDataInfo | 無し | DataInfo構造体の配列 | Ownable,  view | デバッグ用 |
|  |  |  |  |  |

* 構造体 :
* 名前 : DataInfo

|  |  |  |
| --- | --- | --- |
| 変数名 | データ型 | 概要 |
| dataName | string | 値の種類 |
| value | int64 | 値 |
| areaName | string | 地域名 |
| timestamp | uint | 時間 |
| owner | address | 所持者 |

* コントラクト名 : IoTDataService(IoTDataManagerを継承)
* 概要 : ユーザー間でデータのやり取りをする為のコントラクト。

ユーザー間の取引は構造体の配列で管理。ユーザ情報登録やデータ登録の管理も行う。

* 関数 :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 関数名 | 引数 | 戻り値 | オプション | 概要 |
| checkLimit | Address to,  Address from | Bool | Internal | データ閲覧の有効期限が有効なものかチェック。 |
| sendMoney | Address to | 無し | Payable  Public | データ閲覧のために、対象のアドレスに対して支払いを行う。また、手数料として少額コントラクトのオーナーに支払い。 |
| setUser | String areaName | 無し | Public | ユーザーの基本情報を登録する。 |
| setData |  |  | payable | 基本情報が登録されているか、確認した上で、データの登録を行う。また、少額オーナーに支払う。 |

* 構造体 :
* 名前 : tradeInfo

概要 : 取引の履歴を管理するための構造体。期限の確認などに使用する。

|  |  |  |
| --- | --- | --- |
| 名前 | データ型 | 概要 |
| To | address | 支払い先 |
| From | Address | 支払い元 |
| limit | Uint | 閲覧期限 |