# Lab: Data-Types

## State contract

Write a contract, that:

* Has 3 kinds of states: **locked**, **unlocked** and **restricted**
* The owner **can change the state**
* **Locked** means that **nobody can call public** contract functions (even **fallback**)
* **Unlocked** means that **everyone can call** contract functions
* **Restricted** means that **only the owner can call** contract functions
* Have a **structure** that contains a **counter**, a **timestamp** and an **address**
* Have a **function** that **increments** the counter by one, **sets the current** timestamp and **sets the address** to that of the **caller**

## Agreement contract

Write a contract, that:

* Can **accept** **ETH**
* Has **N** number of **owners** (a list of owners given in **constructor**)
* A **proposal** can be made to transfer funds to an account (make it a **struct**)
* For the proposal to be accepted, **each owner must agree** in the order defined in the list
  + **For the second owner to agree, the first one has to agree first.**
  + The same logic for all owners
* If **all agree within 5 minutes** of the proposal, the proposal is **accepted** and the funds are **transferred**
* **Include the necessary constraints**

## Multiple coins contract

Write a contract that:

* Should store the balance of two coins (e.g. – RedCoin, GreenCoin)
  + You should compose the coins in a single **struct**
  + You should store the balances of both coins in a single **\_\_\_\_\_\_\_\_\_**
* The contract creator should have 10000 RedCoins and 5000 GreenCoins
* Have two functions that are used to send coins (one for RedCoin and another for GreenCoin)

## Multiple coins contract vol.2

Write a contract that:

* Should store the balance of multiple coins (e.g. – RedCoin, GreenCoin, Blue, Purple, Yellow, Violet, Indigo etc.)
  + You should **store the balances** of all coins in a **\_\_\_\_\_\_\_\_\_**
* The contract creator should have 10000 of each coin
* Have one function that are used to send coins (think of a way to determine which coin you are sending)

## Pokemon Game

Create a Pokemon game, that:

* Knows ~10 different types of Pokemons
* Any player can say that he caught a Pokemon, but maximum once per 15 seconds (personalized timer)
* The contract can list the Pokemons that a player has caught
* The contract can list which player possess a Pokemon of certain type
* Watch out for duplicates!