



89 lines (66 loc) · 1.55 KB

Preview

Code

Blame

Raw



marp

true

Mappings

- 🔑 key/value hash lookup
- 📦 storage only
- 🙅 cannot be passed as an argument

Example:

```
mapping(address => bool) isMember;
```



```
function join() external {  
    isMember[msg.sender] = true;  
}
```

```
function belongs() external view returns(bool) {  
    return isMember[msg.sender];  
}
```

Combining Types

```
// mapping to a struct  
mapping(address => User) users;
```



```
// mapping to an array of structs
mapping(address => Order[]) ordersByAddress;

// mapping an id to bool
mapping(uint256 => bool) allowedIds;

// mapping in a struct
struct Person {
    mapping(uint256 => Vote) proposalVotes;
}
```



Be Careful With Nested Mappings in Structs!

```
contract X {
    struct Proposal {
        bytes data;
        address target;
        mapping(address => bool) votes;
    }

    Proposal[] proposals;

    function newProposal(bytes memory data, address target) external {
        // 🚫 Struct containing a (nested) mapping cannot be constructed.
        // Proposal memory proposal = Proposal(data, target);

        // ✅ build it in storage first, then assign fields
        Proposal storage proposal = proposals.push();
        proposal.data = data;
        proposal.target = target;
    }
}
```



Implementation

```
contract X {
    mapping(address => bool) isMember; // base slot 0x0

    function join() external {
        // SSTORE(keccak256(msg.sender + 0x0), true)
        isMember[msg.sender] = true;
    }

    function belongs() external view returns(bool) {
```



```
// SLOAD(keccak256(msg.sender + 0x0))  
return isMember[msg.sender];  
}  
}
```