


```
mapping(bytes32⇒bool) private
```

```
mapping(bytes32⇒uint) private
```

```
mapping(bytes32⇒int) private
```

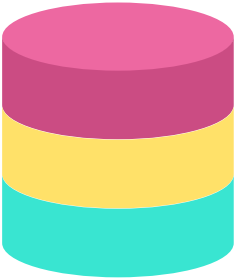
```
mapping(bytes32⇒address) private
```

```
mapping(bytes32⇒string) private
```

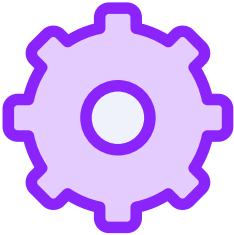
```
mapping(bytes32⇒bytes) private
```

```
function setBool string bool  
keccak256
```

```
function getBool string returns bool  
return keccak256
```







address


```
contract EternalStorage {

mapping(bytes32⇒bool) private _bool;
mapping(bytes32⇒uint) private _uint;
mapping(bytes32⇒int) private _int;
mapping(bytes32⇒address) private _address;
mapping(bytes32⇒string) private _string;
mapping(bytes32⇒bytes) private _bytes;

function setBool(string _key, bool _value) {
    _bool[keccak256(_key)] = _value;
}

function getBool(string _key) returns(bool) {
    return _bool[keccak256(_key)];
}

}
```

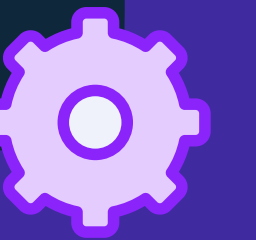
```
contract Token_v1 {

    address store;

}
```



ETERNAL STORAGE



```
contract EternalStorage {

    mapping(bytes32⇒bool) private _bool;
    mapping(bytes32⇒uint) private _uint;
    mapping(bytes32⇒int) private _int;
    mapping(bytes32⇒address) private _address;
    mapping(bytes32⇒string) private _string;
    mapping(bytes32⇒bytes) private _bytes;

    function setBool(string _key, bool _value) {
        _bool[keccak256(_key)] = _value;
    }

    function getBool(string _key) returns(bool) {
        return _bool[keccak256(_key)];
    }

}
```

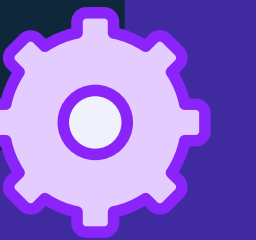


```
contract Token_v1 {

    address store;

    function setIsActive(bool _value) {
        store.setBool("token.is_active") = _value;
    }

}
```



ETERNAL STORAGE