

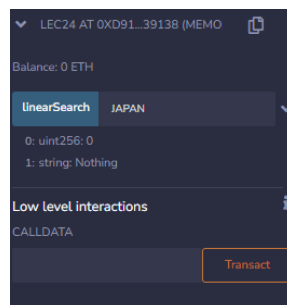
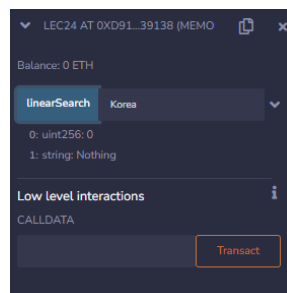
solidity-기본-5

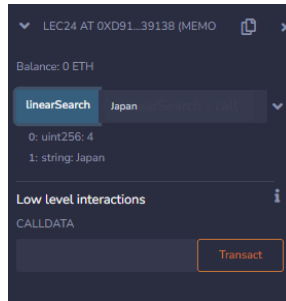
☰ 태그	
📅 날짜	@2023년 6월 20일

linear search

```
contract lec24{
    event CountryIndexName(uint256 indexed _index, string _name);
    string[] private countryList = ["South Korea", "North Korea", "USA", "China","Japan"];

    function linearSearch(string memory _search) public view returns(uint256, string memory){
        for(uint256 i=0; i<countryList.length; i++){
            // "aaa" === "aaa" / "aa".isEqual 같은 문자열비교 기능은 solidity 엔 없다.
            // 아래처럼 byte화 후 hash화를 시켜서 비교해야 한다.
            if(keccak256(bytes(countryList[i])) == keccak256(bytes(_search))){
                return (i,countryList[i]);
            }
        }
        return(0, "Nothing");
    }
}
```





대소문자 유의

에러 핸들러 : require, revert, assert, try/catch

0.4.22 ~ 0.7.x version

assert : gas를 다 소비한 후, 특정한 조건을 부합하지 않으면 (false일 때) 에러를 발생시킨다.

revert : 조건없이 에러를 발생시키고, gas를 환불 시켜준다.

require : 특정한 조건에 부합하지 않으면(false일 때) 에러를 발생시키고, gas를 환불 시켜준다.

```
contract lec25{

    function assertNow() public pure{
        assert(조건문); // 조건문이 false일 때 에러를 발생, gas 소비
        // 가스를 다 소비 한 후 에러를 체크하기 때문에 환불 x
        // test용
    }

    function revertNow() public pure{
        revert("에러메시지 입력 가능"); // gas 환불
        // 가스를 지불 -> revert 에러 발생 -> revert의 대한 gas만 소비 후 gas환불
        //
    }

    function requireNow() public pure{ // require = if + revert
        require(조건문, "false일 때 출력 메시지");
    }

}
```

```

contract lec26{
    function onlyAdults(uint256 _age) public pure returns(string memory){
        if(_age < 19){
            revert("You are not allowed to pay for the cigarette");
        }
        return "Your payment is succeeded";
    }
    // _age가 19 미만 일 경우 revert 메시지 출력

    function onlyAdults2(uint256 _age) public pure returns(string memory){
        require(_age>19, "You are not allowed to pay for the cigarette");
        return "Your payment is succeeded";
    }
}

// revert 는 조건이 true 일 때 revert메시지 출력
// require은 조건이 false일 때 에러 발생
}

```

에러핸들러

0.8.0 포함 x , 0.8.1 ~ version
 assert : 오직 내부적 에러 테스트 용도, 불변성 체크 용도.
 assert가 에러를 발생시키면, panic(uint256) 이라는 에러타입의 에러 발생

0.8.1 이상 부터는 gas를 환불 받을 수 있다.