

Dobre i loše prakse prilikom pisanja pametnih ugovora

Osnovne dobre prakse

- Zaključane verzije compilera

```
pragma solidity 0.8.22;
```

- Bezbednost pre optimizacije
- Validacija unosa

```
pragma solidity 0.8.22;

contract InputValidationExample {
    mapping(address => uint256) public balances;

    function transfer(address to, uint256 value) external {
        // validacija unosa kako bi se sprecile ranjivosti u kodu
        require(balances[msg.sender] >= value, "Insufficient balance");

        balances[msg.sender] -= value;
        balances[to] += value;
    }
}
```

- Očekivanje grešaka
- Korišćenje evenata

```
pragma solidity 0.8.22;

contract EventsLoggingExample {
    mapping(address => uint256) public balances;

    event Transfer(address indexed from, address indexed to, uint256 value);

    function transfer(address to, uint256 value) external {
        balances[msg.sender] -= value;
        balances[to] += value;

        // Korišćenje eventa za logovanje promena
        emit Transfer(msg.sender, to, value);
    }
}
```

- Korišćenje msg.sender za autorizaciju

Loše prakse

- Ignorisanje reentrancy napada
- Ignorisanje ograničenja gasa
- Hardcoding važnih/osetljivih podataka