የኮንትራት መዋቅር

በ ሶሊዲቲ ውስጥ ያሉ ኮንትራቶች ኦብጀክት ኦረንትድ ቋንቋዎች ካሉ ክፍሎች ጋር ተመሳሳይ ናቸው።እያንዳንዱ ውል የስቴት ተለዋዋጮች፣ ፈንክሽንስ፣ የተግባር ማስተካከያዎች፣ ክስተቶች፣ ስህተቶች፣ የአወቃቀር አይነቶች እና የኢነም አይነቶች መግለጫዎችን ሊይዝ ይችላል።በተጨማሪም ኮንትራቶች ከሌሎች ኮንትራቶች ሊወርሱ ይችላሉ።

ቤተመጻሕፍት እና መገናኛዎች የሚባሉ ልዩ የውል ዓይነቶችም አሉ።

ስለ ኮንትራቶች ያለው ክፍል ከዚህ ክፍል የበለጠ ዝርዝሮችን ይዟል, ይህም ፈጣን አጠቃላይ እይታን ለማቅረብ ያገለግላል.

የስቴት ተለዋዋጮች

የስቴት ተለዋዋጮች እሴቶቻቸው በቋሚነት በኮንትራት ማከማቻ ውስጥ የሚቀመጡ ተለዋዋጮች ናቸው።

// SPDX-License-Identifier: GPL-3.0

pragma solidity >=0.4.0 <0.9.0;

contract SimpleStorage {

uint storedData; // State variable

// ...

}

ፈንክሽንስ

ፈንክሽንስ የሚፈጸሙ የኮድ አሃዶች ናቸው።ፈንክሽንስ አብዛኛውን ጊዜ የሚገለጹት በውል ውስጥ ነው፣ ነገር ግን ከውል ውጪም ሊገለጹ ይችላሉ።

// SPDX-License-Identifier: GPL-3.0

pragma solidity >=0.7.1 <0.9.0;

contract SimpleAuction {

function bid() public payable { // Function

// ...

}

}

// Helper function defined outside of a contract

function helper(uint x) pure returns (uint) {

return x \* 2;

}

የፈንክሽን ጥሪዎች ከውስጥም ከውጪም ሊከሰቱ ይችላሉ እና ለሌሎች ኮንትራቶች የተለያየ የታይነት ደረጃ አላቸው።ፈንክሽንስ መለኪያዎችን ይቀበላሉ እና ተለዋዋጮችን ይመለሳሉ ግቤቶችን እና እሴቶችን በመካከላቸው ለማለፍ።

የፈንክሽንስ ማስተካከያዎች

የፈንክሽንስ መቀየሪያዎች የተግባርን የትርጓሜ ሂደት በገላጭ መንገድ ለማሻሻል ጥቅም ላይ ሊውሉ ይችላሉ (በኮንትራቶች ክፍል ውስጥ የተግባር ማስተካከያዎችን ይመልከቱ)።

ከመጠን በላይ መጫን ፣ ማለትም ፣ ተመሳሳይ የመቀየሪያ ስም ከተለያዩ ልኬቶች ጋር መኖር አይቻልም።ልክ እንደ ፈንክሽን፣ መቀየሪያዎች ሊሻሩ ይችላሉ።

// SPDX-License-Identifier: GPL-3.0

pragma solidity >=0.4.22 <0.9.0;

contract Purchase {

address public seller;

modifier onlySeller() { // Modifier

require(

msg.sender == seller,

"Only seller can call this."

);

\_;

}

function abort() public view onlySeller { // Modifier usage

// ...

}

}

ክስተቶች

ክስተቶች ከEVM ምዝግብ ማስታወሻ መገልገያዎች ጋር ምቹ በይነገጽ ናቸው።

// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.22;

event HighestBidIncreased(address bidder, uint amount); // Event

contract SimpleAuction {

function bid() public payable {

// ...

emit HighestBidIncreased(msg.sender, msg.value); // Triggering event

}

}

ክስተቶች እንዴት እንደሚታወጁ እና ከdapp ውስጥ ጥቅም ላይ ሊውሉ እንደሚችሉ መረጃ ለማግኘት በኮንትራቶች ውስጥ ያሉትን ክስተቶች ይመልከቱ።

ስህተቶች

ስህተቶች ለውድቀት ሁኔታዎች ገላጭ ስሞችን እና መረጃዎችን እንዲገልጹ ያስችሉዎታል። በተገላቢጦሽ መግለጫዎች ውስጥ ስህተቶችን መጠቀም ይቻላል. ከሕብረቁምፊ መግለጫዎች ጋር ሲነጻጸር፣ስህተቶች በጣም ርካሽ ናቸው እና ተጨማሪ ውሂብን ኮድ እንዲያደርጉ ያስችሉዎታል። ስህተቱን ለተጠቃሚው ለመግለጽ NatSpec ን መጠቀም ይችላሉ።

// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.4;

/// Not enough funds for transfer. Requested `requested`,

/// but only `available` available.

error NotEnoughFunds(uint requested, uint available);

contract Token {

mapping(address => uint) balances;

function transfer(address to, uint amount) public {

uint balance = balances[msg.sender];

if (balance < amount)

revert NotEnoughFunds(amount, balance);

balances[msg.sender] -= amount;

balances[to] += amount;

// ...

}

}

ለበለጠ መረጃ በውሉ ክፍል ውስጥ ስህተቶችን እና የተመለስን መግለጫ ይመልከቱ።

የመዋቅር ዓይነቶች

መዋቅሮች ብዙ ተለዋዋጮችን ማቧደን የሚችሉ ብጁ የተገለጹ ዓይነቶች ናቸው (በዓይነት ክፍል ውስጥ ያሉትን መዋቅሮች ይመልከቱ)።

// SPDX-License-Identifier: GPL-3.0

pragma solidity >=0.4.0 <0.9.0;

contract Ballot {

struct Voter { // Struct

uint weight;

bool voted;

address delegate;

uint vote;

}

}

ኢነም ዓይነቶች

ኢነም ውሱን የሆነ 'የቋሚ እሴቶች' ስብስብ ያላቸው ብጁ ዓይነቶችን ለመፍጠር ጥቅም ላይ ሊውል ይችላል (በዓይነት ክፍል ውስጥ ያሉትን ኢነም ይመልከቱ)።

// SPDX-License-Identifier: GPL-3.0

pragma solidity >=0.4.0 <0.9.0;

contract Purchase {

enum State { Created, Locked, Inactive } // Enum

}