

A payable function in Solidity is a function that can receive Ether and respond to an Ether deposit for record-keeping purposes.

All about Payable

- In a smart contract, Payable ensures that money goes into and out of the contract. Solidity functions with a modifier Payable can send and receive Ether transactions, but cannot handle transactions with zero Ether values. If a function does not include the payable keyword, the transaction will be automatically rejected. For example, a receive() function with the payable modifier can receive money in the contract, but a send() function without the payable modifier will reject the transaction.
- Fallback payable functions in Solidity are helpful for ensuring transactions go through if someone sends money to the contract without the payable modifier. It is recommended to use a version of a function with the no_name and payable modifiers, with the function name being "no_name" instead of "payable." "Payable" is the word that describes the function.

Function declaration with no name:

```
function () public payable {}
```

You can define a payable function using the following syntax:

```
function receive() payable {}  
function send() payable {}
```



Solidity supports several methods of transferring ether between the contracts.

1. `address.send(amount)`
2. `address.transfer(amount)`
3. `address.call.value(amount)()`

address.send(amountValue)

Send is the first method for sending ether between contracts. It has a 2300 gas limit for a contract's fallback function, which is not enough for multiple events. If `send()` fails due to gas shortage, it returns false but does not throw an exception. Therefore, it should be inside the `require` function. If gas is not paid, transactions cannot be submitted to the blockchain, and changes will be rolled back.

address.transfer(amountValue)

The transfer method has a limit of 2300 gas, but developers proposed adding a `gas()` modifier to change the limit. Unlike the `send()` method, the `transfer()` method throws an exception when it fails, indicating that the transaction was executed incorrectly.

address.call.value(amountValue)()

The call function is the most personalized method for sending ether, but it will return false if an error occurs. The main difference from previous functions is that the `gas(gasLimit)` modifier allows setting the gas limit, especially for complex ether payment functions that require a significant amount of gas.



YouTube Link:



https://www.youtube.com/embed/4Clg68-HiQU?si=w-hDSi_5BhIWXXpl

Solidity Source File



.....

Bibliography/References

<https://www.showwcase.com/show/33658/payable-functions-in-solidity>

