Activity: Sending and Receiving ETH with Solidity

Introduction

Previously you have been exposed on the mechanics of compiling and deploying a contract to the Ethereum blockchain. We will not go through the step-by-step mechanics here again.

You will need some testnet ETH in your Account (say Account 1). Please go to the Roptein faucet to obtain some test ETH.

You will to connect your Metamask to Roptein test net. Start up the Remix compiling and write or key in the code required.

This week we will study the code for sending and receiving Ether.

Code 1.

We use the *msg.sender* property to retrieve the account address that calls the contract and check the Ether balance. If you are interacting via Account 1 (in Metamask) then you should see its balance in Remix, and it should be the same amount in Metamask, except that in remix it is in Wei (10^{-18} Ether)

Key in this code, compile and deploy in Remix.

```
// SPDX-License-Identifier: GPL-3.
pragma solidity >=0.7.0 <0.9.0;

contract sendBal{

    function Bal() public view returns (uint256){
        return msg.sender.balance;
    }

}
```

Code 2.

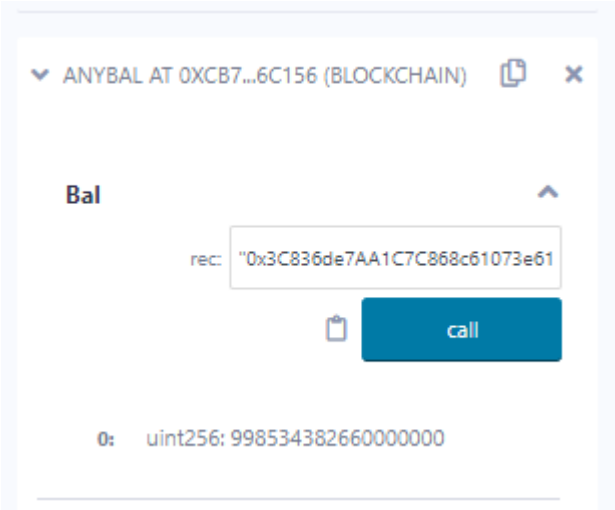
We use (this) to refer to the current executing contract, then we cast it with address() to get its address. Once we have the address we can use the balance property.

Key in this code, compile and deploy in Remix.

```
// SPDX-License-Identifier: GPL-3.
pragma solidity >=0.7.0 <0.9.0;

contract contractBal{

    function Bal() public view returns (uint256){
        return address(this).balance;
    }

}
```

Code 3.

We can also create a function that allows user to key in an account address and obtain the balance. Remember the function must be within a Contract.



Key in this code, compile and deploy in Remix.

```
// SPDX-License-Identifier: GPL-3.
pragma solidity >=0.7.0 <0.9.0;

contract anyBal{

    function Bal(address payable rec) public view returns (uint256){
        return rec.balance;
    }
}
```

Code 4.

How do we send ETH to a contract and how can we check the contract balance?

You need to include the receive() function in the contract for the contract to receive ETH.

This is function to include into the contract:

```
receive() external payable {
}
```

Write the complete code that can receive payment and can check its balance. See below deployment window:

a. Deploy the contract and check its initial balance
b. You can send ETH to the contract using Metamask. You need to copy the Contract address to the Metamask Send ETH page.
c. Send ETH (from the Account 1 that has ETH) to the Contract. Send a small amount.
d. Check the Contract balance. Is the amount correct?
e. Check on the Metamask the Account balance. Is the amount correct?

Can the amount in the Contract be retrieved (or transferred)?

Code 5.

How to transfer out ETH from a contract? You will need to include function like this:

```
function give(uint256 amt, address payable receiver) public {
    receiver.transfer(amt);
}
```

Write a complete contract that can receive ETH and transfer ETH to another EOA. The contract should include these functions:

a. A function that makes the contract able to receive ETH
b. A function that makes the contract able to transfer ETH to another EOA
c. A function to check the contract's current balance.

It should look like this in the Remix Deployment Window: