Solidity Code

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract MoneyTransaction {
    address public owner;

    constructor() {
        owner = msg.sender;  // The owner is the person who deploys the contract
    }
    // A function to receive Ether. 'payable' allows the contract to accept ETH.
    receive() external payable {}
   // Returns the contract's balance in Wei (smallest unit of Ether)
    function getBalance() public view returns (uint) {
        return address(this).balance;
    }
    // Function to withdraw all Ether from the contract by the owner
    function withdraw() public {
        require(msg.sender == owner, "Only the owner can withdraw");
        payable(msg.sender).transfer(address(this).balance);
    }
    // Function to transfer Ether to a specific address
    function transferTo(address payable _to, uint _amount) public {
        require(msg.sender == owner, "Only the owner can transfer");
        require(address(this).balance >= _amount, "Insufficient funds in contract");
        _to.transfer(_amount);
    }
}
```