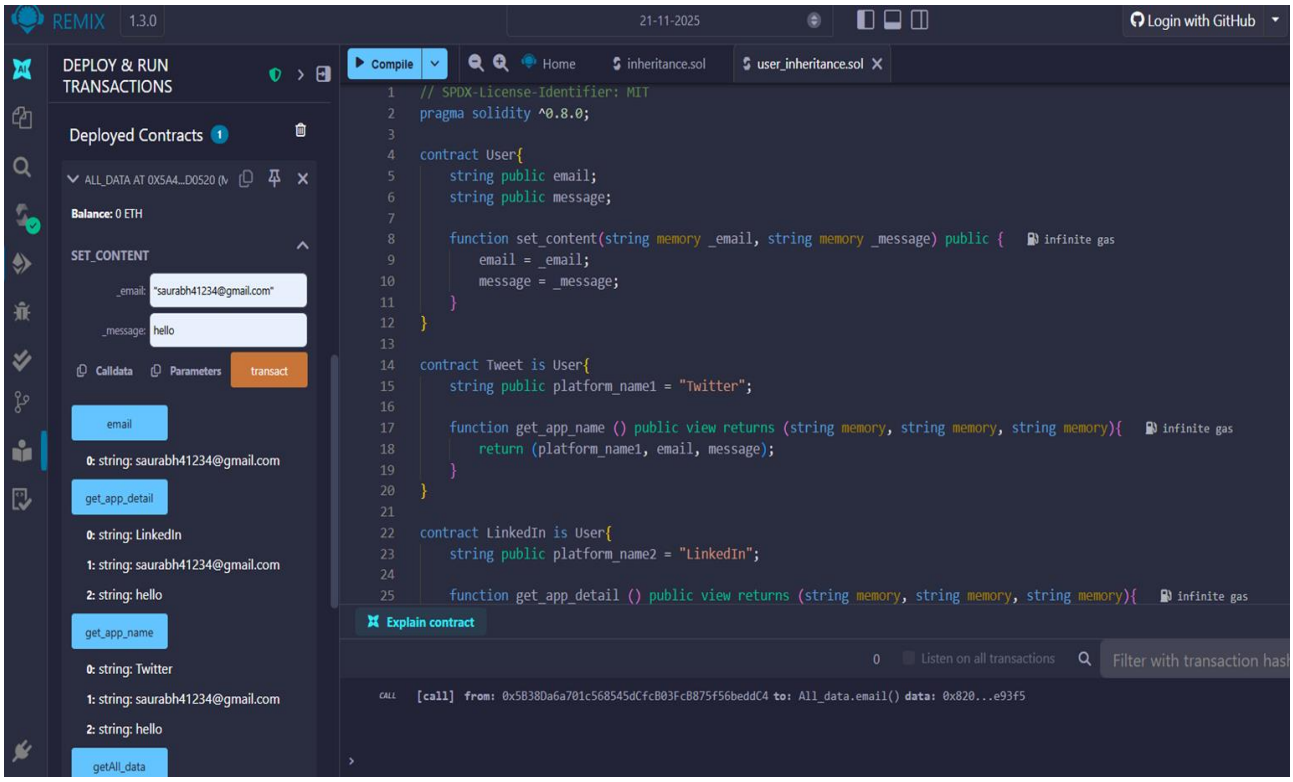
```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

// Base contract that stores user information
contract User {
    string public email;     // Stores user email
    string public message;   // Stores user message/post content

    // Function to update user email and message
    function set_content(string memory _email, string memory _message) public {
        email = _email;
        message = _message;
    }
}
// Tweet contract inherits User contract
contract Tweet is User {
    string public platform_name1 = "Twitter";   // Platform name

    // Returns platform name + inherited email + message
    function get_app_name()
        public
        view
        returns (string memory, string memory, string memory)
    {
        return (platform_name1, email, message);
    }
}
// LinkedIn contract also inherits User contract
contract LinkedIn is User {
    string public platform_name2 = "LinkedIn";  // Platform name

    // Returns platform name + inherited email + message
    function get_app_detail()
        public
        view
        returns (string memory, string memory, string memory)
    {
        return (platform_name2, email, message);
    }
}

// Multiple inheritance: All_data inherits Tweet and LinkedIn
contract All_data is Tweet, LinkedIn {

    function getAll_data()
        public
        view
        returns (string memory, string memory, string memory, string memory)
    {
        return (platform_name1, platform_name2, email, message);
    }
}
```


REMIX 1.3.0
21-11-2025
Login with GitHub
DEPLOY & RUN TRANSACTIONS
Deployed Contracts 1
ALL_DATA AT 0X5A4...D0520 (M
Balance: 0 ETH
SET_CONTENT
_email: "saurabh41234@gmail.com"
_message: hello
Calldata
Parameters
transact
email
0: string: saurabh41234@gmail.com
get_app_detail
0: string: LinkedIn
1: string: saurabh41234@gmail.com
2: string: hello
get_app_name
0: string: Twitter
1: string: saurabh41234@gmail.com
2: string: hello
getAll_data
Compile
Home
inheritance.sol
user_inheritance.sol
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract User{
string public email;
string public message;
function set_content(string memory _email, string memory _message) public { infinite gas
email = _email;
message = _message;
}
}
contract Tweet is User{
string public platform_name1 = "Twitter";
function get_app_name () public view returns (string memory, string memory, string memory){ infinite gas
return (platform_name1, email, message);
}
}
contract LinkedIn is User{
string public platform_name2 = "LinkedIn";
function get_app_detail () public view returns (string memory, string memory, string memory){ infinite gas
Explain contract
0
Listen on all transactions
Filter with transaction hash
CALL [call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: All_data.email() data: 0x820...e93f5