**Passing Parameter to Parent Constructor In Multiple Inheritance:**

In Solidity, when dealing with multiple inheritance, you can pass parameters to parent constructors by specifying them in the derived contract's constructor and explicitly calling the constructors of the parent contracts.



**Code:**

//SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

contract ConA{

    string public name;

    uint public age;

    constructor(string memory \_name,uint \_age){

        name = \_name;

        age = \_age;

    }

}

contract ConB{

    string public subject;

    uint public grade;

    constructor(string memory \_subject,uint \_grade){

        subject = \_subject;

        grade = \_grade;

    }

}

contract ConC is ConA("sam",21),ConB("Mumbai",100000){}

contract ConD is ConA,ConB{

    constructor() ConA("sam",21) ConB("mumbai",100000){

    }

}

contract ConE is ConA,ConB{

    string public a;

    constructor(string memory \_name,uint \_age,string memory \_subject,uint \_grade, string memory \_a) ConA(\_name,\_age+2) ConB(\_subject,\_grade){

        a = \_a;

    }

}

contract ConF is ConA("sam",21),ConB{

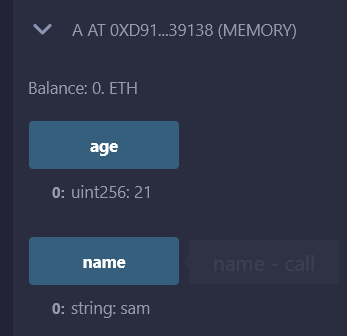
    constructor(string memory \_subject,uint \_grade) ConB(\_subject,\_grade){

    }

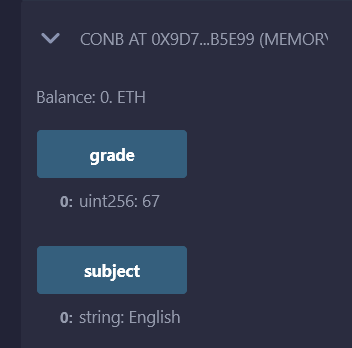
}

**Output:**

**Contract A:**



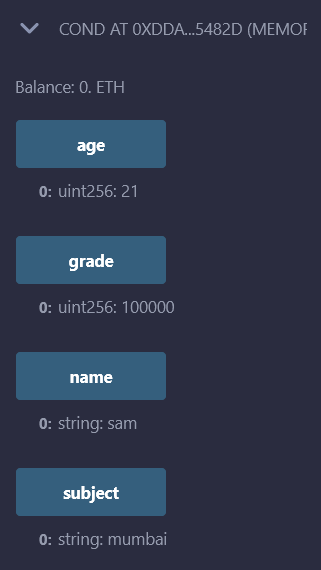
**Contract B:**



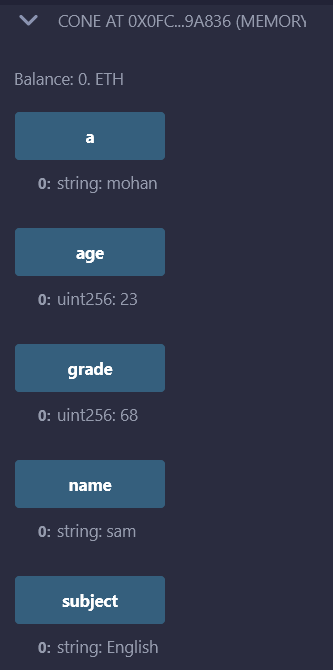
**Contract C:**



**Contract D:**



**Contract E:**



**Contract F:**

