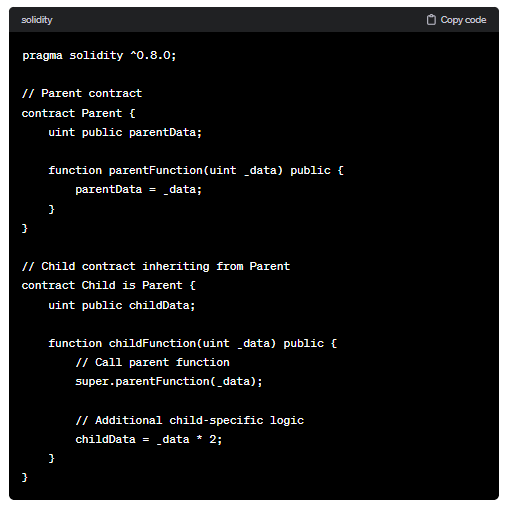
**Calling Parent Function:**

In Solidity, if you have a contract that inherits from another contract, you can call functions from the parent contract using the **super** keyword. This allows you to execute the logic defined in the parent contract's functions within the context of the child contract.



By using **super**, you ensure that both the parent and child contract logic is executed when you call the child function.

**Code:**

//SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

contract ConA{

    event log(string name, uint age);

    function function1() public virtual {

        emit log("ConA.fun1",21);

    }

    function function2() public virtual{

        emit log("ConA.fun2",21);

    }

}

contract ConB is ConA{

    function function1() public virtual override {

        emit log("ConB.fun1", 31);

        ConA.function1();

    }

    function function2() public virtual override{

        emit log("B.fun2",31);

        super.function2();

    }

}

contract ConC is ConA{

    function function1() public virtual override {

        emit log("C.fun1",41);

    }

    function function2() public virtual override{

        emit log("C.fun2",41);

    }

}

contract ConD is ConB,ConC{

    function function1() public override(ConB,ConC) {

        emit log("D.fun1",51);

    }

    function function2() public override(ConB,ConC){

        emit log("D.fun2",51);

        super.function2();

    }

}

**Output:**

