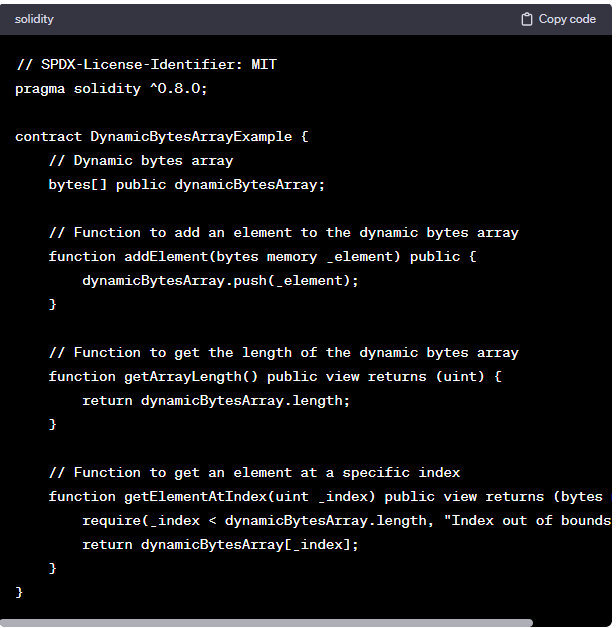
**(Bytes[] | Dynamic Size Array):**

In Solidity, you can use a dynamic array of bytes, which is essentially a **bytes** array. Unlike fixed-size arrays, dynamic arrays can change in size during execution. 

You can use this dynamic bytes array to store variable-length binary data or other types of data that can be represented using the **bytes** type. Just be aware that, as with other dynamic arrays, operations that increase the size of the array may have associated gas costs, so it's essential to consider gas efficiency in your smart contract design.

Top of Form

**Code:**

//SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

contract DynamicSizeArray{

    bytes public b1="abc";

    function pushElement() public

    {

        b1.push('d');

    }

    function getElement(uint i) public view returns(bytes1)

    {

        return b1[i];

    }

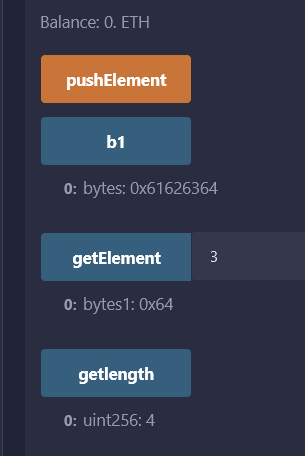
    function getlength() public view returns(uint)

    {

        return b1.length;

    }

}



//SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

contract DynamicSizeArray{

    bytes public temp;

    constructor()

    {

        temp = "123abcd45";

    }

    function pushElement() public

    {

        temp.push('c');

    }

    function popElement() public

    {

        temp.pop();

    }

    function getlength() public view returns(uint){

        return temp.length;

    }

    function getElement(uint \_idx) public view returns(bytes1){

        return temp[\_idx];

    }

}

