**Bytes Array { Fixed Size Array}:**

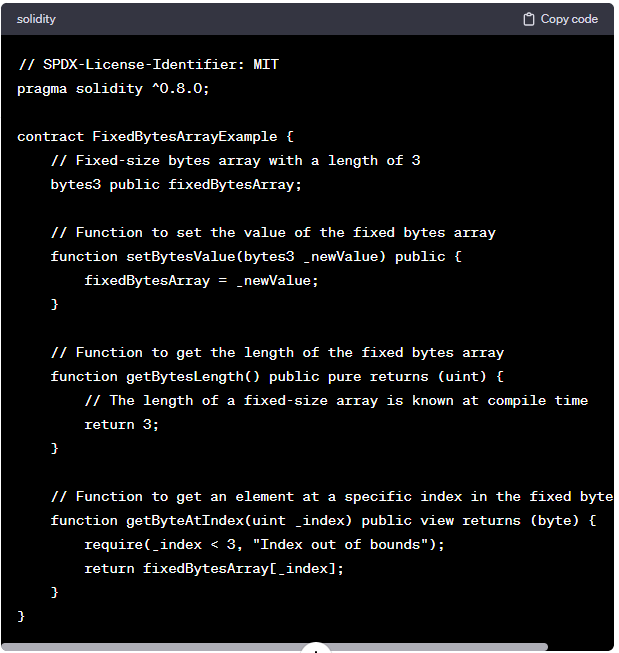
1 byte = 8 bit

1 hexadecimal = 4 bit

1 byte = 2 hex

Everything that will be stored in the byte array will be in the form of the hexadecimal digits.

In Solidity, you can use a fixed-size array of bytes, denoted by **bytesN** where **N** is the size of the array.



**Details:**

1) Byte arrays cannot be modified.

2) Padding of 0 is added at the end if the value (by which the array is initialized) does not occupy the entire space.

**Code:**

Here b2 contains 2 byte and b3 contains 3 bytes and we have the value in hexadecimal.

//SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

contract ByteFixedSizeArray{

    bytes3 public b3; //3 bytes array

    bytes2 public b2; //2 bytes array

    function setter() public

    {

        b3='abc';

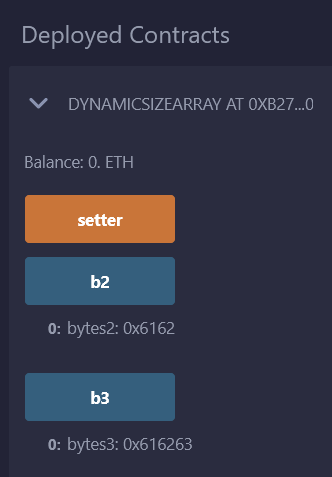
        b2='ab';

       // b3[0]='d';

    }

}

If we are are using in place of 2 byte only 1 byte is used then it auto fill by 00 at the end.



//SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

contract Bytes{

    bytes5 public temp1;

    bytes7 public temp2;

    function setvalue() public

    {

        temp1 = "abcde";

        temp2 = "12abcde";

    }

    function getdigit() public view returns(bytes1){

        return temp1[2];

    }

    function getlen() public view returns(uint){

        return temp1.length;

    }

}



Padding concept if we are putting less value than the actual size of the byte variable then in the end it automatically add "00".