1. Write a Basic NFT Contract made by using Solidity, Remix IDE, and OpenZeppelin.

Ans: // SPDX-License-Identifier: MIT

pragma solidity ^0.8.4;

import "@openzeppelin/contracts/token/ERC721/IERC721.sol";

import "@openzeppelin/contracts/access/Ownable.sol";

contract NFT is ERC721, Ownable {

    uint256 public minPrice= 1 ether;

    uint256 public totalNFT;

    uint256 public maxSupply;

    bool public isMintEnabled;

    mapping (address=>uint256) public mintedWallet;

    constructor()payable ERC721('Basic NFT','BSC'){

        maxSupply=10;

    }

    function toggleisMintEnabled(uint256 \_maxsupply)external onlyOwner{

        isMintEnabled= !isMintEnabled;

    }

    function setMaxSupply(uint256 \_maxsupply)external onlyOwner{

        maxSupply=\_maxsupply;

    }

    function mint() external payable{

        require(isMintEnabled,'error, minting is not enabled');

        require(msg.value== minPrice,'wrong price, please place right price');

        require(mintedWallet[msg.sender]<2,'Limit reached of this wallet');

        require(maxSupply > totalNFT,'all NFTs are sold out');

        mintedWallet[msg.sender]++;

        totalNFT++;

        uint tokenID=totalNFT;

        \_safeMint(msg.sender,tokenID);

    }

}