// SPDX-License-Identifier: MIT

pragma solidity ^0.8.0;

contract StudentVerificat{

    address public admin;

    struct Certificat{

        string studentName;

        string course;

        string certId;

        bool isCertificateValid;

    }

    address public owner;

    mapping (string => Certificat) public certifications;

     constructor(){

        owner = msg.sender;

     }

     modifier onlyOwner(){

        require(msg.sender==owner, "only owner can add certificat");

        \_;

     }

    function addCertificat(string memory \_studentName, string memory \_course, string memory \_certId) public onlyOwner{

        certifications[\_certId] = Certificat(\_studentName , \_course ,\_certId,true);

    }

    function verifyStudentsCertificate(string memory \_certId )public view returns (string memory,string memory ,bool){

        Certificat memory cert = certifications[\_certId];

        return (cert.studentName,cert.course, cert.isCertificateValid);

    }

}