//SPDX-License-Identifier: MIT

pragma solidity ^0.8.0;

// Build the Contract

contract StudentData

{

// Create a structure for

// student details

struct Student

{

int ID;

string fName;

string lName;

int[2] marks;

}

// fallback () external payable {

// }

address owner;

int public stdCount = 0;

mapping(int => Student) public stdRecords;

modifier onlyOwner

{

require(owner == msg.sender);

\_;

}

constructor()

{

owner=msg.sender;

}

// Create a function to add

// the new records

function addNewRecords(int \_ID,string memory \_fName,string memory \_lName,int[2] memory \_marks) public onlyOwner

{

// Increase the count by 1

stdCount = stdCount + 1;

// Fetch the student details

// with the help of stdCount

stdRecords[stdCount] = Student(\_ID, \_fName,\_lName, \_marks);

}

}