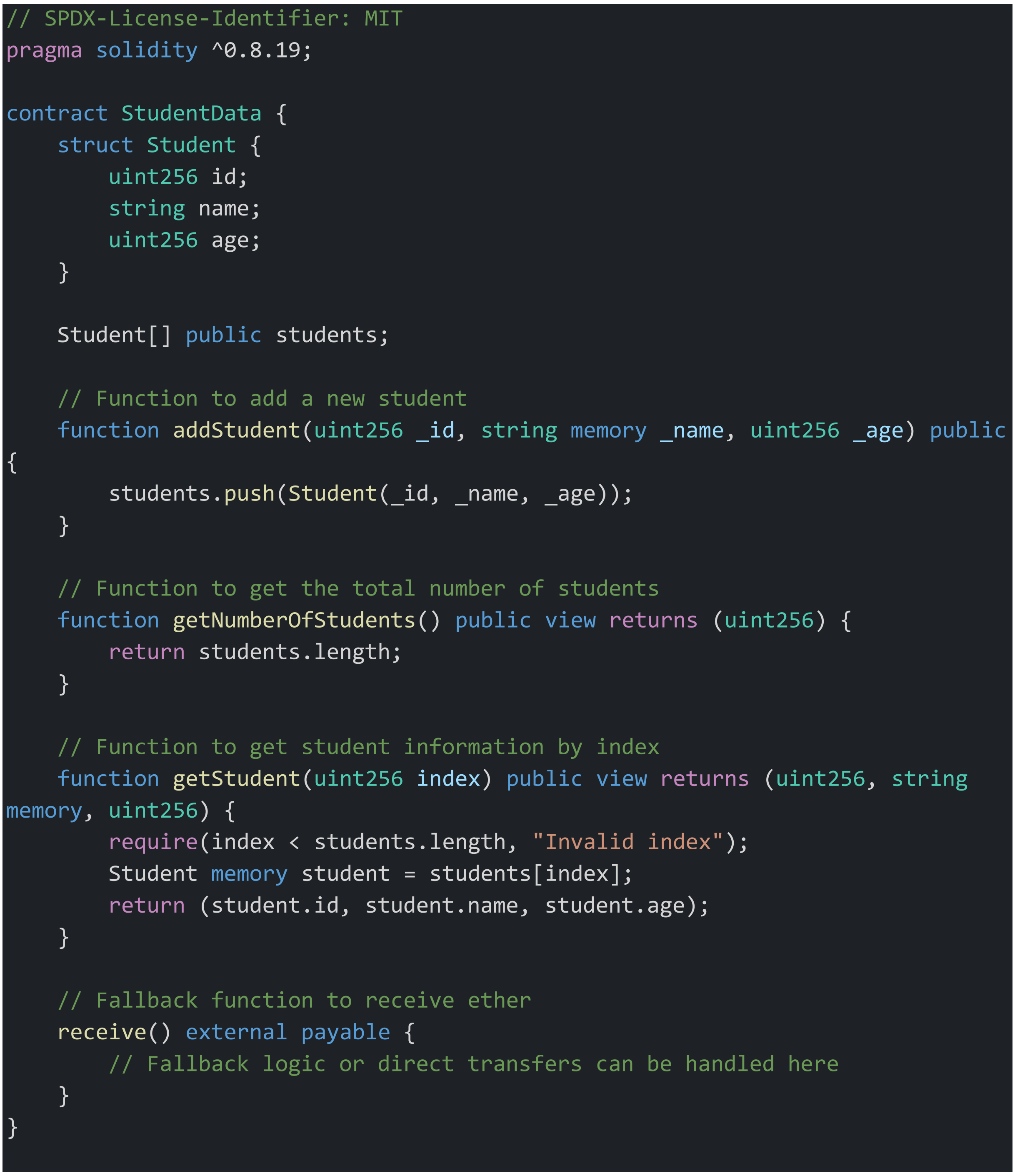

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.19;

contract StudentData {
    struct Student {
        uint256 id;
        string name;
        uint256 age;
    }

    Student[] public students;

    // Function to add a new student
    function addStudent(uint256 _id, string memory _name, uint256 _age) public
{
        students.push(Student(_id, _name, _age));
    }

    // Function to get the total number of students
    function getNumberOfStudents() public view returns (uint256) {
        return students.length;
    }

    // Function to get student information by index
    function getStudent(uint256 index) public view returns (uint256, string
memory, uint256) {
        require(index < students.length, "Invalid index");
        Student memory student = students[index];
        return (student.id, student.name, student.age);
    }

    // Fallback function to receive ether
    receive() external payable {
        // Fallback logic or direct transfers can be handled here
    }
}
```