A

PROJECT REPORT

ON

“**Staff management system”**

SUBMITTED BY: Sanika Amol Patare(2124UCEF1058)

SUBJECT

C**++ PROGRAMMING**

Under the guidance of

Miss. Ishwari Tirse



**Department of Computer Science and Engineering**

**Sanjivani Rural Education Society’s**

**SANJIVANI UNIVERSITY**

**KOPARGAON - 423603, DIST : AHEMDNAGAR 2024 - 2025**

|  |  |  |
| --- | --- | --- |
| **SR.**  **NO** | **CONTENT** | **PAGE NO.** |
| 1 | **INTRODUCTION** | 3 |
| 2 | **CODE** | 4 |
| 3 | **OUTPUT** | 6 |
| 4 | **CONCLUSION** | 8 |

INTRODUCTION

A C++ Staff Management System is a software application developed in C++ to manage employee-related information efficiently. It automates various administrative tasks such as maintaining staff records, monitoring attendance, handling payroll, and tracking performance. The system aims to simplify workforce management by organizing employee data in a structured manner, making it easy to retrieve, update, and analyze.

CODE

#include <iostream>

#include <vector>

#include <string>

using namespace std;

class Staff {

public:

string name;

int id;

string position;

Staff(int id, string name, string position)

: id(id), name(name), position(position) {}

void display() const {

cout << "ID: " << id << ", Name: " << name << ", Position: " << position << endl;

}

};

class StaffManagementSystem {

private:

vector<Staff> staffList;

public:

void addStaff(int id, string name, string position) {

Staff newStaff(id, name, position);

staffList.push\_back(newStaff);

cout << "Staff added successfully!" << endl;

}

void displayStaff() const {

if (staffList.empty()) {

cout << "No staff available." << endl;

return;

}

for (const auto& staff : staffList) {

staff.display();

}

}

void findStaff(int id) const {

for (const auto& staff : staffList) {

if (staff.id == id) {

staff.display();

return;

}

}

cout << "Staff with ID " << id << " not found." << endl;

}

};

int main() {

StaffManagementSystem sms;

int choice;

do {

cout << "\nStaff Management System\n";

cout << "1. Add Staff\n";

cout << "2. Display Staff\n";

cout << "3. Find Staff by ID\n";

cout << "4. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1: {

int id;

string name, position;

cout << "Enter ID: ";

cin >> id;

cout << "Enter Name: ";

cin.ignore(); // Clear newline character from input buffer

getline(cin, name);

cout << "Enter Position: ";

getline(cin, position);

sms.addStaff(id, name, position);

break;

}

case 2:

sms.displayStaff();

break;

case 3: {

int id;

cout << "Enter ID to find: ";

cin >> id;

sms.findStaff(id);

break;

}

case 4:

cout << "Exiting the program." << endl;

break;

default:

cout << "Invalid choice! Please try again." << endl;

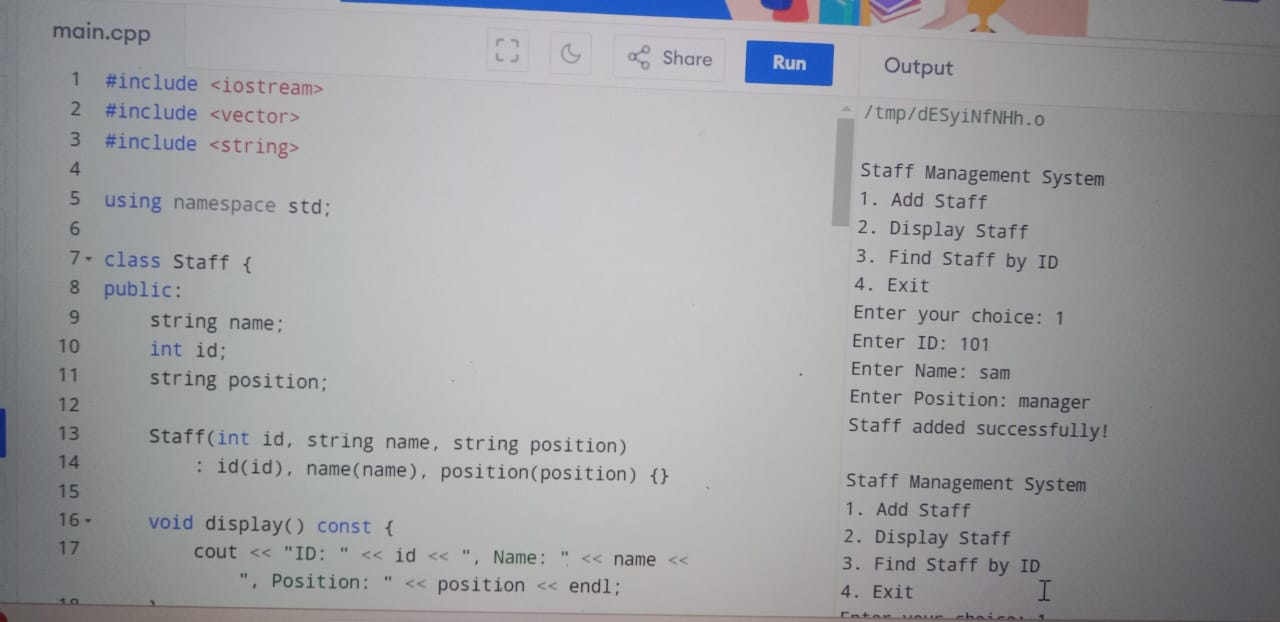
}

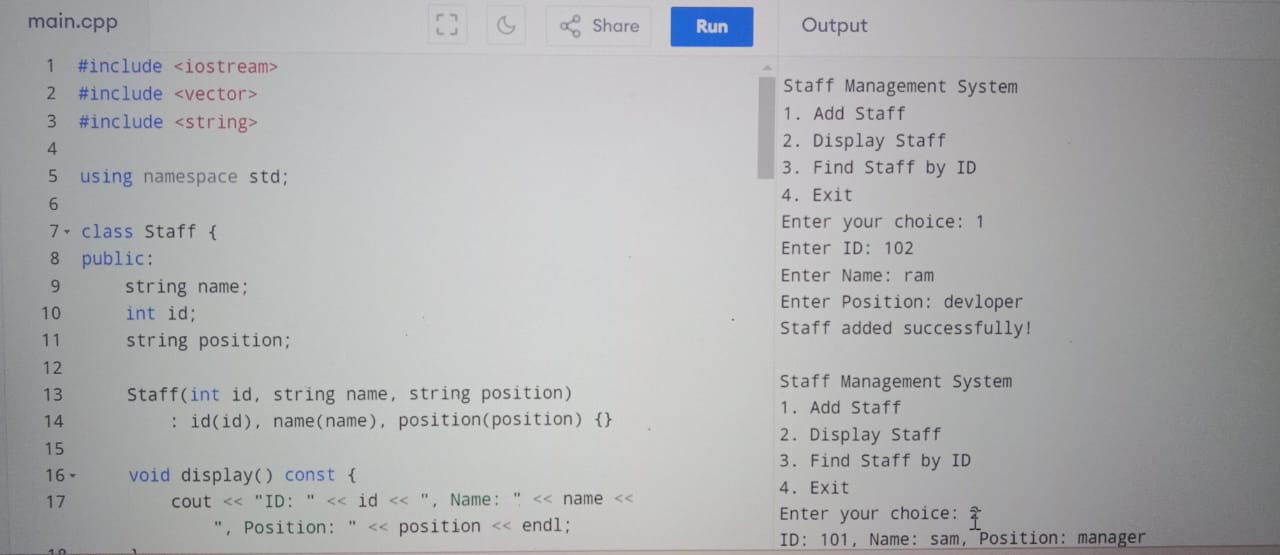
} while (choice != 4);

return 0;

}

OUTPUT





conclusion

The staff management system implemented in C++ effectively addresses the needs of modern organizational structures by streamlining employee management processes. The system's modular design allows for easy maintenance and scalability, accommodating various functionalities such as adding, updating, and deleting employee records, managing attendance, and generating reports.

Utilizing object-oriented programming principles, the system promotes code reusability and organization. By implementing data structures like arrays or linked lists, it efficiently handles multiple employee records. Additionally, integrating file handling ensures that data persists beyond program execution, allowing for easy retrieval and management.