**SUBMITTED BY: NOOR UL AIN SAGHEER (BCS223020)**

**SUBMITTED TO: MAM MAHAM TARIQ**

**LAB 7 TASKS**

**PRACTICE TASK 1:**

**CODE:**

#include<iostream>

#include<string>

#include<fstream>

#include<sstream>

using namespace std;

class student\_info

{

private:

int reg[5];

string name[5];

int age[5];

public:

void writing();

void searching();

void display();

};

void student\_info:: writing()

{

int index;

ofstream write("student.txt", ios::app);

cout << " ENTER INDEX NUMBER (0-4) AT WHICH YOU WANT TO PLACE DATA : " << endl;

cin >> index;

cout << " ENTER NAME OF STUDENT :" << endl;

getline(cin, name[index]);

write << name << endl;

cout << " ENTER AGE OF STUDENT : " << endl;

cin >> age[index];

write << age << endl;

cout << " ENTER REGISTRATION NUMBER OF STUDENT : " << endl;

cin >> reg[index];

write << reg << endl;

write.close();

}

void student\_info::searching()

{

string search;

int i;

fstream read("student.txt");

cout << " ENTER NAME OR AGE OR REGISTRATION NUMBER OF STUDENT FOR SEARCHING DATA : " << endl;

getline(read, search);

for (int n = 0;n <= 4;n++)

{

if (search[n] == n)

{

i = n;

}

}

cout << " NAME " << name[i] << endl;

cout << " REGISTRATION MUMBER " << reg[i] << endl;

cout << " AGE " << age[i] << endl;

read.close();

}

void student\_info::display()

{

cout << " student information " << endl;

cout << " name " << endl;

for (int i = 0;i < 5;i++)

{

cout << name[i] << endl;

}

cout << " age " << endl;

for (int i = 0;i < 5;i++)

{

cout << age[i] << endl;

}

cout << " registration number " << endl;

for (int i = 0;i < 5;i++)

{

cout << reg[i] << endl;

}

}

void main()

{

student\_info obj;

char ch;

cout << "------ MENU ------" << endl;

cout << " ENTER WHAT DO YOU WANT WRITING IN FILE OR SEARCHING ? " << endl;

cout << " FOR WRITING SELECT 1 :" << endl;

cout << " FOR SEARCHING SELECT 2 :" << endl;

cin >> ch;

switch (ch)

{

case 1:

obj.writing();

obj.display();

break;

case 2:

obj.searching();

break;

default:

cout << " WRONG INPUT!!" << endl;

break;

}

}

**PRACTICE TASK 2:  
CODE:**

#include <iostream>

#include <fstream>

#include <string>

#include <vector>

using namespace std;

class Employee {

private:

string name;

int age;

double salary;

public:

// Setters

void setName(const string& newName) {

name = newName;

}

void setAge(int newAge) {

age = newAge;

}

void setSalary(double newSalary) {

salary = newSalary;

}

// Getters

string getName() const {

return name;

}

int getAge() const {

return age;

}

double getSalary() const {

return salary;

}

};

// Function to write employee information to a file

void writeEmployeeInfo(const Employee& employee) {

ofstream outFile("employee.txt", ios::app);

if (!outFile.is\_open()) {

cerr << "Error opening the file for writing." << endl;

return;

}

// Write the employee information to the file

outFile << employee.getName() << "," << employee.getAge() << "," << employee.getSalary() << endl;

outFile.close();

}

// Function to read employee information from a file

void readEmployeeInfo() {

ifstream inFile("employee.txt");

if (!inFile.is\_open()) {

cerr << "Error opening the file for reading." << endl;

return;

}

vector<Employee> employees;

Employee employee;

string line;

while (getline(inFile, line)) {

// Assuming the data in the file is stored as "name,age,salary"

size\_t pos1 = line.find(",");

size\_t pos2 = line.find(",", pos1 + 1);

employee.setName(line.substr(0, pos1));

employee.setAge(stoi(line.substr(pos1 + 1, pos2 - pos1 - 1)));

employee.setSalary(stod(line.substr(pos2 + 1)));

employees.push\_back(employee);

}

// Display the read employee information

cout << "Employee Information:" << endl;

for (const auto& emp : employees) {

cout << "Name: " << emp.getName() << ", Age: " << emp.getAge() << ", Salary: " << emp.getSalary() << endl;

}

inFile.close();

}

// Function to search for an employee in the file

void searchEmployee(const string& searchKey) {

ifstream inFile("employee.txt");

if (!inFile.is\_open()) {

cerr << "Error opening the file for reading." << endl;

return;

}

Employee employee;

string line;

bool found = false;

while (getline(inFile, line)) {

// Assuming the data in the file is stored as "name,age,salary"

size\_t pos1 = line.find(",");

size\_t pos2 = line.find(",", pos1 + 1);

employee.setName(line.substr(0, pos1));

employee.setAge(stoi(line.substr(pos1 + 1, pos2 - pos1 - 1)));

employee.setSalary(stod(line.substr(pos2 + 1)));

if (employee.getName() == searchKey) {

found = true;

cout << "Employee found: Name: " << employee.getName() << ", Age: " << employee.getAge() << ", Salary: " << employee.getSalary() << endl;

break; // Assuming names are unique, no need to continue searching

}

}

if (!found) {

cout << "Employee not found." << endl;

}

inFile.close();

}

// Function to count the number of employees with salary more than 50,000

int countHighSalaryEmployees() {

ifstream inFile("employee.txt");

if (!inFile.is\_open()) {

cerr << "Error opening the file for reading." << endl;

return -1;

}

int count = 0;

Employee employee;

string line;

while (getline(inFile, line)) {

// Assuming the data in the file is stored as "name,age,salary"

size\_t pos1 = line.find(",");

size\_t pos2 = line.find(",", pos1 + 1);

employee.setName(line.substr(0, pos1));

employee.setAge(stoi(line.substr(pos1 + 1, pos2 - pos1 - 1)));

employee.setSalary(stod(line.substr(pos2 + 1)));

if (employee.getSalary() > 50000) {

count++;

}

}

inFile.close();

return count;

}

int main() {

int choice;

do {

cout << "Choose an option:" << endl;

cout << "1. Add employee information" << endl;

cout << "2. Read employee information" << endl;

cout << "3. Search for an employee" << endl;

cout << "4. Count employees with salary > 50,000" << endl;

cout << "5. Exit" << endl;

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1: {

Employee newEmployee;

cin.ignore(); // Clear the input buffer

cout << "Enter employee name: ";

getline(cin, newEmployee.setName);

cout << "Enter employee age: ";

cin >> newEmployee.setAge;

cout << "Enter employee salary: ";

cin >> newEmployee.setSalary;

writeEmployeeInfo(newEmployee);

break;

}

case 2:

readEmployeeInfo();

break;

case 3: {

cin.ignore(); // Clear the input buffer

string searchName;

cout << "Enter the name to search: ";

getline(cin, searchName);

searchEmployee(searchName);

break;

}

case 4: {

int count = countHighSalaryEmployees();

if (count != -1) {

cout << "Number of employees with salary > 50,000: " << count << endl;

}

break;

}

case 5:

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

break;

default:

cout << "Invalid choice. Please enter a valid option." << endl;

}

} while (choice != 5);

return 0;

}

**//SEARCHING IN FILES**

**CODE:**

#include<iostream>

#include<fstream>

#include<string>

using namespace std;

int main()

{

ofstream write("lab7.txt",ios::app);

write << " Noor ul ain ";

write << " maleeha ";

write << " maryam ";

write << " laiba ";

write.close();

ifstream read("lab7.txt");

if (!read.is\_open())

{

cout << " file not opened! " << endl;

return 1;

}

string line;

bool found = false;

string search\_name;

cout << " enter name you want to search " << endl;

cin >> search\_name;

while (getline(read, line))

{

if (line == search\_name)

{

found = true;

break;

}

}

read.close();

if (found)

{

cout << " found " << search\_name << endl;

}

else

{

cout << " not found " << search\_name << endl;

}

return 0;

}