M. Ali. Arslan

19F-0348

**Major Assignment 2**

**Problem # 1**

**Main.cpp**

#include<iostream>

#include<cstdlib>

#include<ctime>

#include"Header.h"

using namespace std;

int main() {

Book b1;

Book\* ptr = &b1;

int choice, d, edi, num;

string n, a;

cout << "Enter 1 Details of Books\n" "Choice: ";

cin >> choice;

switch (choice) {

case 1:

{

cout << "Enter number of books, you want to add: ";

cin >> num;

for (int i = 0; i < num; i++) {

cout << "Enter name of book " << i + 1 << ": ";

cin >> n;

cout << "Enter name of Author: ";

cin >> a;

cout << "Enter Edition of book: ";

cin >> edi;

cout << "Enter Date of Publish(1-31): ";

cin >> d;

}

}

cout << "Enter 2 to Display the Details of Latest 5 Books\n";

cin >> choice;

{

switch (choice)

{

case 2:

{

system("cls");

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

{

cout << "\t\t\t\t\tBook " << i + 1 << endl << endl;

cout << "Id of Book is: " << ptr->getID() << endl;

ptr->setName(n);

cout << "The Name of Book is: " << ptr->getName() << endl;

ptr->setAuthor(a);

cout << "The Name of Author is: " << ptr->getAuthor() << endl;

ptr->setEdition(edi);

cout << "The Edition of Book is: " << ptr->getEdition() << endl;

ptr->setDate(d);

cout << "The Date of Publish is: " << ptr->getDate() << endl;

cout << endl << endl;

}

break;

}

}

}

break;

default:

{

cout << "Invalid Entry!" << endl;

}

}

}

**Class.cpp**

#include<iostream>

#include<cstdlib>

#include<ctime>

#include "Header.h"

using namespace std;

Book::Book() {

id = 0;

name = " ";

author = " ";

edition = 0;

DateofPublish = 0;

}

int Book::getID() {

srand(time(0));

id = rand() % 5 + 1;

return id;

}

void Book::setName(string n)

{

name = n;

}

string Book::getName() {

return name;

}

void Book::setAuthor(string a) {

author = a;

}

string Book::getAuthor() {

return author;

}

void Book::setEdition(int edi) {

edition = edi;

}

int Book::getEdition() {

return edition;

}

void Book::setDate(int d) {

DateofPublish = d;

}

int Book::getDate() {

return DateofPublish;

}

Book::~Book()

{

id = 0;

name = " ";

author = " ";

edition = 0;

DateofPublish = 0;

}

**Header.h**

#include<iostream>

#include<cstdlib>

#include<ctime>

using namespace std;

class Book {

public:

int id;

string name;

string author;

int edition;

int DateofPublish;

Book();

int getID();

void setName(string n);

string getName();

void setAuthor(string a);

string getAuthor();

void setEdition(int edi);

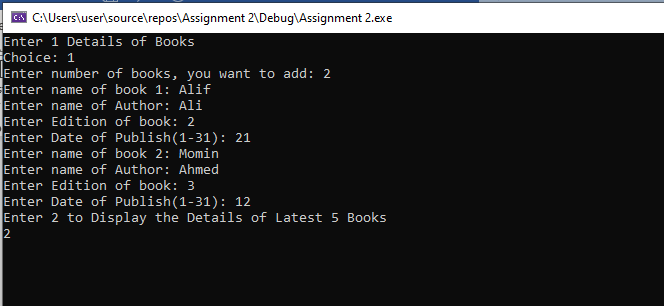
int getEdition();

void setDate(int d);

int getDate();

~Book();

};



**Problem # 2**

**Main.cpp**

#include<iostream>

#include<string>

#include"student.h"

using namespace std;

int main() {

Student s;

int count=0, choice;

do

{

system("cls");

cout << "\t\t\t\t\t\t\*Academics Data\*" << endl << endl;

cout << "\t1. Register a Student\n"

"\t2. Register Courses\n"

"\t3. Drop Courses\n"

"\t4. Update Record\n"

"\t5. Search a Student\n"

"\t6. Total Students Enrolled\n"

"\t7. Exit\n"

"\nChoice:";

cin >> choice;

system("cls");

switch (choice)

{

case 1:

{

s.reg();

count++;

}break;

case 2:

{

s.course();

}break;

case 3:

{

s.drop();

}break;

case 4:

{

s.update();

}break;

case 5:

{

s.byname();

}break;

case 6:

{

cout << "\t\t\t\t\t\t\*STUDENTS DATABASE\*" << endl << endl;

cout << "Total Students Enrolled are: " << count;

}break;

case 7:

{

exit(0);

}break;

}

cout << endl;

cout << "Enter 1 to go to main menu & 0 to exit: ";

cin >> choice;

} while (choice!=0);

system("pause");

return 0;

}

**Student.cpp**

#include<iostream>

#include<string>

#include"student.h"

using namespace std;

void Student::reg() {

cout << "\t\t\t\t\t\t\*STUDENT REGISTRATION\*" << endl << endl;

cout << "Enter the name of Student: ";

cin >> name;

cout << "Enter ID of Student(3-Digit): ";

cin >> id;

cout << "Enter CNIC number of Student: ";

cin >> cnic;

cout << "Enter Address of Student: ";

cin >> address;

cout << "Enter Gender of Student: ";

cin >> gender;

cout << "Congrats! The Student is registered" << endl;

}

void Student::course() {

int c;

cout << "\t\t\t\t\t\t\*COURSES REGISTRATION\*" << endl << endl;

cout << "Enter Name of Student: ";

cin >> name;

cout << "The Offered courses are:\n 1)DLD\n 2)OOP\n 3)COAL\n 4)CALCULAS\n 5)P.S\n";

cout << "You can register maximum 5 & minimum 3 courses\n";

cout << "Choice: ";

cin >> c;

if (c == 3)

{

cout << "Enter name of Course 1: ";

cin >> c1;

cout << "Enter name of Course 2: ";

cin >> c2;

cout << "Enter name of Course 3: ";

cin >> c3;

cout << endl << endl << name << " has registered the following courses: 1)" << c1 << " 2)"

<< c2 << " 3)" << c3 << endl;

}

else if (c == 4)

{

cout << "Enter name of Course 1: ";

cin >> c1;

cout << "Enter name of Course 2: ";

cin >> c2;

cout << "Enter name of Course 3: ";

cin >> c3;

cout << "Enter name of Course 4: ";

cin >> c4;

cout << endl << endl << name << " has registered the following courses: 1)" << c1 << " 2)"

<< c2 << " 3)" << c3 << " 4)" << c4 << endl;

}

else if (c == 5)

{

cout << "Enter name of Course 1: ";

cin >> c1;

cout << "Enter name of Course 2: ";

cin >> c2;

cout << "Enter name of Course 3: ";

cin >> c3;

cout << "Enter name of Course 4: ";

cin >> c4;

cout << "Enter name of Course 5: ";

cin >> c5;

cout << endl << endl << name << " has registered the following courses: 1)" << c1 << " 2)"

<< c2 << " 3)" << c3 << " 4)" << c4 << " 5)" << c5 << endl;

}

}

void Student::drop() {

int d;

cout << "\t\t\t\t\t\t\*COURSE DROP\*" << endl << endl;

cout << "Enter Name of student: ";

cin >> name;

cout << "Enter Total Number of Registered Courses: ";

cin >> registered\_courses;

cout << "Enter number of Courses you want to drop: ";

cin >> d;

cout << endl << endl << name << " has dropped " << d << " courses out of " << registered\_courses << " courses" << endl;

}

void Student::update()

{

cout << "\t\t\t\t\t\t\*UPDATE RECORD\*" << endl << endl;

cout << "ENTER THE Name OF THE Student of which you want to update the Information: ";

cin >> name;

cout << "Enter CNIC of Student: ";

cin >> cnic;

cout << "Enter Address of Student: ";

cin >> address;

cout << "The Record of " << name << " has been updated!";

cout << endl;

}

void Student::byname()

{

cout << "\t\t\t\t\t\t\*SEARCH A STUDENT\*" << endl << endl;

string str;

cout << "PLEASE ENTER THE NAME OF THE STUDENT OF WHICH YOU WANT TO SEARCH : ";

cin >> str;

if (str == name)

{

cout << " ->The ID of Student is: " << id << " " << endl;

cout << " ->The CNIC number of Student is: " << cnic << " " << endl;

cout << " ->The Address of Student is: " << address << " " << endl;

cout << " ->The Gender of Student is: " << gender << " " << endl;

}

else

{

cout << "Sorry! There is no such Student" << endl;

}

cout << endl;

}

**Student.h**

#include<iostream>

#include<string>

using namespace std;

class Student {

int id;

int cnic;

int registered\_courses;

string name;

string address;

string gender;

string c1, c2, c3, c4, c5;

public:

Student() {

id = 0;

name = " ";

cnic = 0;

registered\_courses = 0;

address = " ";

gender = " ";

}

void reg();

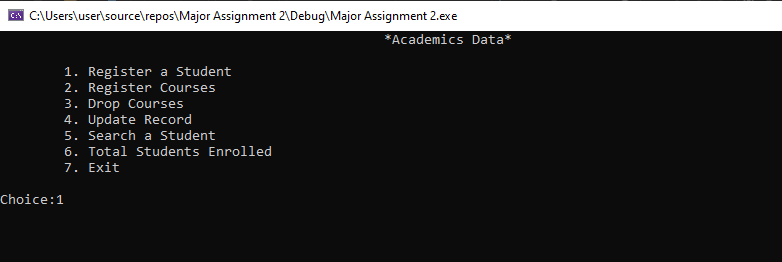
void course();

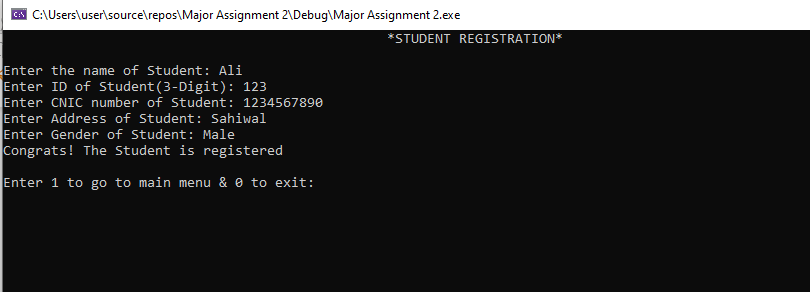
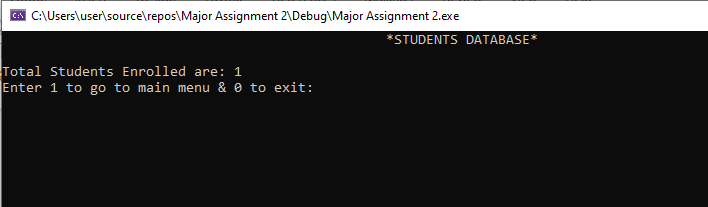
void drop();

void update();

void byname();

};



**Problem # 3**

**Source.cpp**

#include<iostream>

#include"Employee.h"

using namespace std;

int main()

{

int ID;

int press = 1;

Employee Emp[2];

for (int i = 0; i < 2; i++)

{

cout << "\t\t\t\t\t\tEmployee #" << i + 1 << endl << endl;

cout << "Enter Employee's ID:";

cin >> ID;

cin.ignore();

Emp[i].GetData();

system("cls");

}

while (press != 4)

{

cout << "1) Update Monthly hours" << endl;

cout << "2) Update salary per hour" << endl;

cout << "3) Calculate Monthly salary" << endl;

cout << "4) Full receipt" << endl;

//cout << "Press 0 to exit." << endl;

cout << "Choice:";

cin >> press;

if (press == 1)

{

int ID1 = 0;

cout << "Enter Employee's ID:";

cin >> ID1;

if (ID1 > 0)

{

Emp[ID1 - 1].UpdateHours();

//Emp[ID1 - 1].PutData();

}

else

{

Emp[ID1].UpdateHours();

//Emp[ID1].PutData();

}

}

else if (press == 2)

{

int ID2 = 0;

cout << "Enter Employee's ID:";

cin >> ID2;

if (ID2 > 0)

{

Emp[ID2 - 1].chngeSalary();

// Emp[ID2 - 1].PutData1();

}

else

{

Emp[ID2].chngeSalary();

//Emp[ID2].PutData1();

}

}

else if (press == 3)

{

int ID3 = 0;

cout << "Enter Employee's ID:";

cin >> ID3;

if (ID3 > 0)

{

cout << "The Datails of Employee no." << ID3 << " are." << endl;

Emp[ID3 - 1].monthlySalary();

Emp[ID3 - 1].PutData1();

}

else

{

cout << "The Datails of Employee no." << ID3 << " are." << endl;

Emp[ID3].monthlySalary();

Emp[ID3].PutData1();

}

}

else if (press == 4)

{

int ID3 = 0;

cout << "Enter Employee's ID:";

cin >> ID3;

cout << "The Receipt of Employee no." << ID3 << " is." << endl;

if (ID3 > 0)

{

Emp[ID3 - 1].Receipt();

}

else

{

Emp[ID3].Receipt();

}

}

}

system("pause");

return 0;

}

**Employee.cpp**

#include "Employee.h"

#include<iostream>

#include<string>

using namespace std;

Employee::Employee()

{

}

void Employee::GetData() //Statement 1 : Defining GetData()

{

cout << "Enter Employee's Name:";

cin.getline(Name, 15);

cout << "Enter Employee's Salary per hour:";

cin >> salaryperhour;

cout << "Enter Employee's Monthly worked Hours:";

cin >> monthlyWorkedHours;

cout << "Enter Employee's Year of joining:";

cin >> yearofJoining;

}

void Employee::PutData1()

{

cout << "Employee's Name:" << Name << endl;

cout << "Salary is:" << salary << endl;

}

void Employee::UpdateHours()

{

cout << "Enter Updated hours:";

cin >> monthlyWorkedHours;

salary = 100 \* monthlyWorkedHours;

}

void Employee::chngeSalary()

{

cout << "Enter increment in Salary per hour:";

cin >> salaryperhour;

salary = 100 \* monthlyWorkedHours;

}

void Employee::isSeniorEmployee()

{

salary = 100 \* monthlyWorkedHours;

int tax = (salary \* 10) / 100;

int Bonus = (salary \* 10) / 100;

salary = salary - tax;

salary = salary + Bonus;

taxPaid = 10;

salary = salary - 1000;

//cout << "Your Salary is:" << salary << endl;

}

void Employee::monthlySalary()

{

if (yearofJoining <= 2012)

{

isSeniorEmployee();

}

else

{

salary = 100 \* monthlyWorkedHours;

int tax = (salary \* 10) / 100;

salary = salary - tax;

taxPaid = 10;

salary = salary - 1000;

}

}

void Employee::Receipt()

{

cout << " SALARY RECEIPT! " << endl;

cout << "Name" << " " << " Salary per Hour Monthly Hour Tax paid Health fee Salary" << endl;

cout << Name << " " << salaryperhour << " " << monthlyWorkedHours << " " << taxPaid << " " << 1000 << " " << salary << endl;

}

Employee::~Employee()

{

}

**Employee.h**

#include<iostream>

class Employee

{

private:

char Name[15];

float salaryperhour;

float monthlyWorkedHours;

float taxPaid;

int yearofJoining;

float salary;

public:

Employee();

~Employee();

void GetData();

void PutData1();

void UpdateHours();

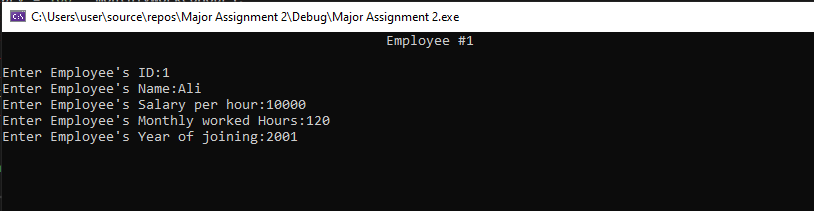
void chngeSalary();

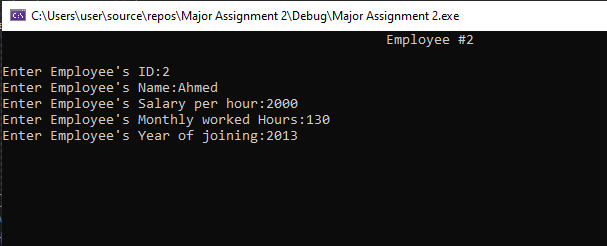
void monthlySalary();

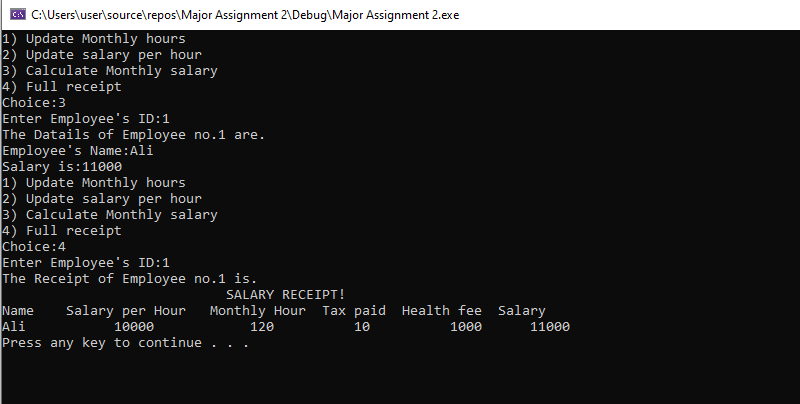
void isSeniorEmployee();

void Receipt();

};







**Problem # 4**