Question 1:

#include<iostream>

#include<string>

using namespace std;

class person

{

private : // defaults have been set

string name;

string age;

char gender;

string occupation;

string cook;

public:

person(); // constructor

~person(); // destructor

void setname();

void getname();

void getage();

void setage();

void getoccupation();

void isfemale();

void getgender();

void cancook();

// all the requirements added as asked in mannual

};

person::person()

// implementing my default constructor

{

cout << " I AM IMPLEMENTING QUESTION 1 OF THIS ASSIGNMENT " << endl;

cout << "DEFAULT CONSTRUCTOR HAS INIALIZED AS ASKED " << endl;

name = "null";

age = "0";

gender = 'm';

occupation = "student";

cook = "yes";

cout << endl;

}

// implementing my requirements

void person ::getname() {

cout << "PLEASE ENTER THE NAME OF PERSON = " << endl;

getline(cin, name);

}

void person::setname()

{

cout << "the name is = " << endl;

cout << name << endl;

}

void person::getage() {

cout << "PLEASE ENTER THE AGE OF PERSON = " << endl;

getline(cin, age);

}

void person::setage()

{

cout << "the age is = " << endl;

cout << age << endl;

}

void person::getgender() {

cout << "PLEASE ENTER THE GENDER OF PERSON = " << endl;

cin >> gender;

if (gender == 'f')

{

gender = 'f';

}

else if (gender == 'm')

{

gender = 'm';

}

else

{

cout << "invalid option " << endl;

}

}

void person::isfemale()

{

if (gender == 'f')

{

cout << "female" << endl;

}

else if (gender == 'm')

cout << "male" << endl;

cout << "GENDER IS = " << gender << endl;

}

void person::getoccupation() {

cout << "PLEASE ENTER THE OCCUPATION OF PERSON = " << endl;

getline(cin,occupation);

cout << "THE OCCUPATION IS = " << endl;

cout << occupation << endl;

}

void person::cancook() {

cout << "CAN THIS PERSON COOK IF YES ENTER 1 IF NO ENTER 0 " << endl;

int choice;

cin >> choice;

if (choice == 1)

{

cook = "true";

}

else if (choice == 0)

{

cook = "false";

}

cout << "cooking status= " << cook << endl;

}

person::~person() {

cout << "i ve done w my question" << endl;

}

int main()

{ // ive taken object as arham instead of p1 for unique code xD

person arham;

arham.getname();

arham.setname();

arham.getage();

arham.setage();

arham.getoccupation();

arham.getgender();

arham.isfemale();

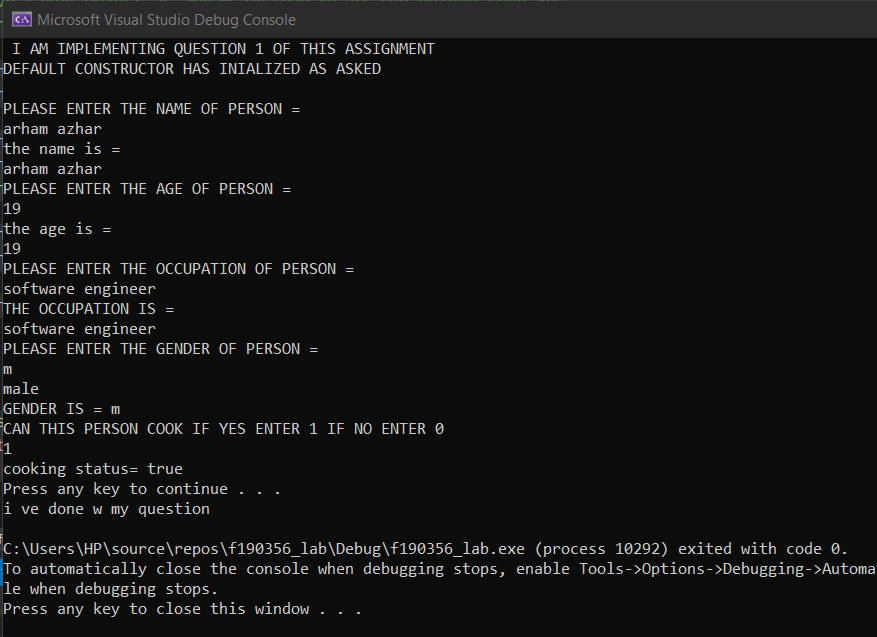
arham.cancook();

system("pause");

return 0;

// after return zero my destructor will be automatically called which will delete object arham

}



Question 2:

#include<iostream>

#include<string>

#include<iomanip>

using namespace std;

class employee {

private:

string name;

string Dept;

string position;

int idNumber;

public:

string name1;

int idinput;

string depart1;

string pos1;

// constructor for initailizing blank values

employee()

{

name = " ";

Dept = " ";

position = " ";

idNumber = 0;

}

// simple destructor

~employee()

{

cout << "destructor called and ive done w my question" << endl;

}

// defining functions

void setName(string);

string getName();

void setid(int);

int getID();

void setDepart(string);

string getDepart();

void setPosition(string);

string getPosition();

void output();

};

// objects of class employee

employee o, o1, o2, o3;

// funcations made one by one

void employee::setName(string a)

{

name = a;

}

string employee::getName()

{

// get name would return the name string

return name;

}

void employee::setid(int indentity\_key)

{

idinput = indentity\_key;

}

int employee::getID()

{

return idinput;

}

void employee::setDepart(string departt)

{

Dept = departt;

}

string employee::getDepart()

{

return Dept;

}

void employee::setPosition(string pos)

{

position = pos;

}

string employee::getPosition()

{

return position;

}

// output function to display the answer

void employee::output()

{

cout << getName() <<" " << getID() <<" " << getDepart() <<" " << getPosition() << endl;

}

int main()

{

// as sir said we have to do it for users so this is for first user

cout << "please enter name of person= ";

getline(cin, o1.name1);

cout << "please enter id of person= ";

cin >> o1.idinput;

cin.ignore(); // i have used it here because i ve used getline before xD

cout << "please enter department of person= ";

getline(cin, o1.depart1);

cout << "please enter position of person in department= ";

getline(cin, o1.pos1);

o1.setName(o1.name1);

o1.setDepart(o1.depart1);

o1.setPosition(o1.pos1);

o1.setid(o1.idinput);

cout << endl;

// as sir said we have to do it for users so this is for second user

cout << "please enter name of person= ";

getline(cin, o2.name1);

cout << "please enter id of person= ";

cin >> o2.idinput;

cin.ignore();

cout << "please enter department of person= ";

getline(cin, o2.depart1);

cout << "please enter position of person in department= ";

getline(cin, o2.pos1);

o2.setName(o2.name1);

o2.setDepart(o2.depart1);

o2.setPosition(o2.pos1);

o2.setid(o2.idinput);

cout << endl;

// as sir said we have to do it for users so this is for third user

cout << "please enter name of person= ";

getline(cin, o3.name1);

cout << "please enter id of person= ";

cin >> o3.idinput;

cin.ignore();

cout << "please enter department of person= ";

getline(cin, o3.depart1);

cout << "please enter position of person in department= ";

getline(cin, o3.pos1);

o3.setName(o3.name1);

o3.setDepart(o3.depart1);

o3.setPosition(o3.pos1);

o3.setid(o3.idinput);

cout << endl;

cout << "Name" << " " << "Id Number" << " " << "Department" << " " << "Position" << endl << endl;

// output for o1

o1.output();

cout << endl;

cout << "Name" << " " << "Id Number" << " " << "Department" << " " << "Position" << endl << endl;

// output for o2

o2.output();

cout << endl;

cout << "Name" << " " << "Id Number" << " " << "Department" << " " << "Position" << endl << endl;

// output for o3

o3.output();

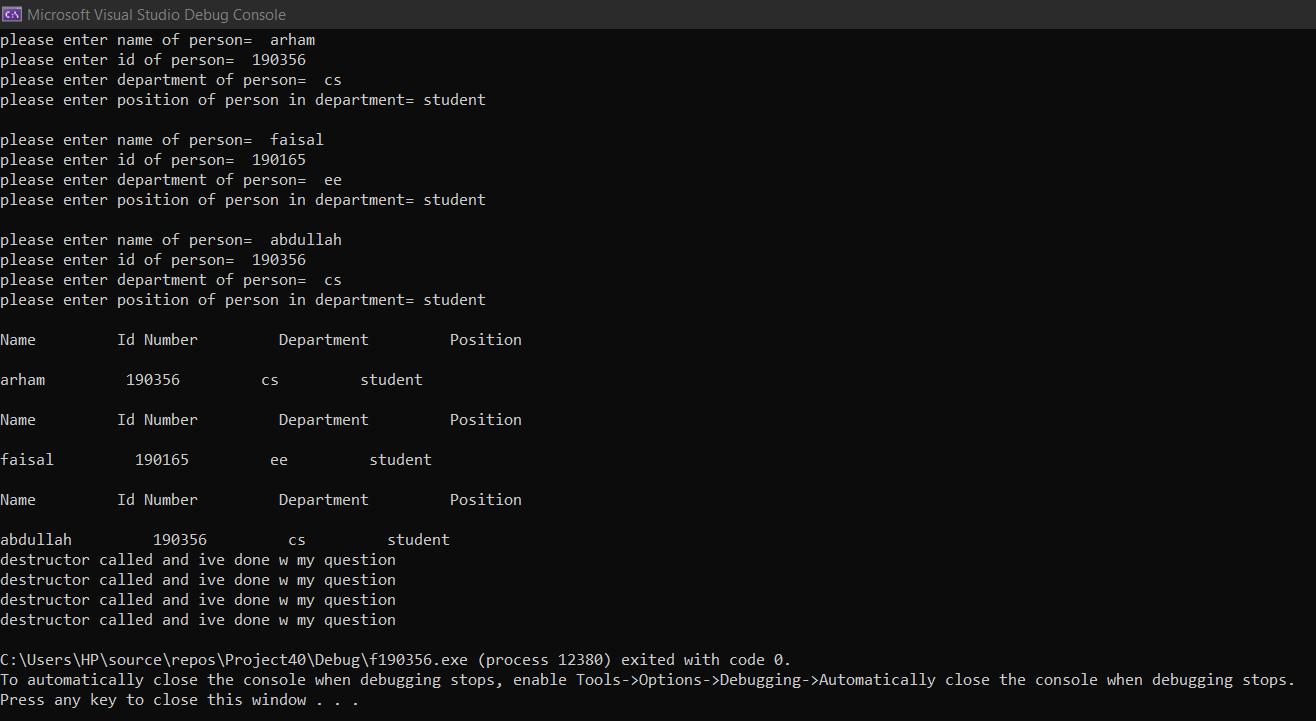
// destructor count

cout << "the destructor will be called 4 times , due to ojects o,o1,o2,o3 " << endl;

system("pause");

return 0;

}



Question 3:

#include<iostream>

#include<string>

using namespace std;

class studentID

{

private:

string stdId;

string stdName;

string dept;

public:

studentID()

{

stdId = " ";

stdName = "";

dept = "";

}

studentID(string id, string name)

{

stdId = id;

stdName = name;

}

studentID(string name, string dep, string id)

{

stdName = name;

dept = dep;

stdId = id;

}

studentID(string name, string dep,int o)

{

o = 0;

dept = dep;

stdName = name;

stdId = " ";

}

void inputData()

{

cout << "Enter your Name: ";

getline(cin, stdName);

cout << "Enter your Id: ";

cin >> stdId;

cout << "Enter your department: ";

getline(cin, dept);

cin.ignore();

}

void DisplayData()

{

cout << "Name: " << stdName << endl

<< "ID: " << stdId << endl

<< "Department: " << dept << endl;

}

};

int main()

{

studentID s1;

cout << " STUDENT RECORDS" << endl;

cout << "===================================" << endl;

s1.inputData();

studentID sd;

studentID s3("12345", "ali hassan");

studentID s4("fahad abbas", "Computer Science",0);

studentID s2("sheraz depp", "Computer Science", "2356");

cout << endl;

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

cout << "INPUT FUNCTION CALLED!" << endl;

s1.DisplayData();

cout << endl;

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

cout << "DEFAULT CONSTRUCTOR CALLED!" << endl;

sd.DisplayData();

cout << endl;

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

cout << "\nOVERLOADED CONSTRUCTOR 1 CALLED!" << endl;

s3.DisplayData();

cout << endl;

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

cout << "\OVERLOADED CONSTRUCTOR 2 CALLED!" << endl;

s2.DisplayData();

cout << endl;

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

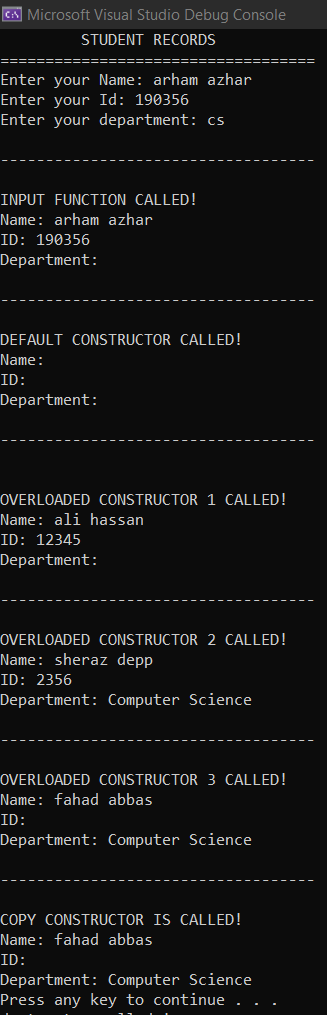
cout << "\OVERLOADED CONSTRUCTOR 3 CALLED!" << endl;

s4.DisplayData();

system("pause");

return 0;

}





Question 4:

#include<iostream>

#include<string>

using namespace std;

class areaaa

{

private:

float length, width; // as per mannual

int aarea;

public:

static int counter;

void DisplayData(int arr)

{

cout << "the area is = " << arr << endl;

cout << "total times the constructor was called is = " << areaaa::counter << endl;

}

areaaa(float a, float b, int aarea)

{

aarea = a \* b; // cal of area

counter++;

DisplayData(aarea); // display func

}

};

int areaaa::counter = 0; // counter (how many times contructor ran)

int main()

{

// taking new entries each time for better understanding xD

int answer=0;

float len, wid;

cout << "enter length = " << endl;

cin >> len;

cout << "enter width = " << endl;

cin >> wid;

areaaa a1(len,wid,answer);

cout << "enter length = " << endl;

cin >> len;

cout << "enter width = " << endl;

cin >> wid;

areaaa a2(len, wid, answer);

cout << "enter length = " << endl;

cin >> len;

cout << "enter width = " << endl;

cin >> wid;

areaaa a3(len, wid, answer);

system("pause");

return 0;

}

