**LAB TASK # 6**

**Question # 1 :**

**INPUT :**

#include<iostream>

#include<string>

using namespace std;

class Rectangle

{

protected:

double length;

double width;

public:

Rectangle()

{}

Rectangle(double l, double w)

{

length = l;

width = w;

}

double getlength()

{

return length;

}

double getwidth()

{

return width;

}

void setlength(double l)

{

length = l;

}

void setwidth(double w)

{

width = w;

}

double area()

{

return length \* width;

}

virtual void showData()

{

cout << "Length = " << getlength() << endl;

cout << "Width = " << getwidth() << endl;

}

};

class DormRoom :public Rectangle

{

private:

double height;

public:

DormRoom()

{}

DormRoom(int l, int b, int h) :Rectangle(l, b), height(h)

{}

double getheight()

{

return height;

}

void setheight(double h)

{

height = h;

}

double volume()

{

return length \* width \* height;

}

virtual void showData()

{

Rectangle::showData();

cout << "Height = " << getheight() << endl;

}

};

**Question # 2 :**

**INPUT :**

#include <iostream>

using namespace std;

class MealBill {

private:

int cost;

string nameOfResturant="AAK'S CAFE";

int localSalesRate=100;

int tip;

int amountPayed;

public:

MealBill(int price, int amPayed)

{

cost = price;

tip = 0.15 \* cost;

amountPayed = amPayed;

}

void displayBill() {

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

cout << "Meal Cost: " << cost << endl;

cout << "Tax: " << localSalesRate << endl;

cout << "Tip: " << tip << endl;

cout << "Total Cost: " << cost + localSalesRate + tip << endl;

}

void displayReceipt() {

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

cout << "Amount Payed: " << amountPayed;

cout << endl;

cout << "Total Cost" << cost + localSalesRate + tip << endl;

cout << "Change: " << amountPayed - (cost + localSalesRate + tip) << endl;

cout << endl;

cout << "Thank you for dining at " << nameOfResturant << endl;

}

};

int main() {

string name;

int salesTax, cost1, amountPayed;

cout << "Enter Meal Cost: ";

cin >> cost1;

cout << "Enter Amount Payed: " ;

cin >> amountPayed;

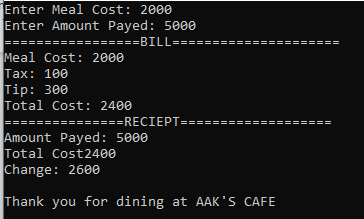
MealBill bill( cost1, amountPayed);

bill.displayBill();

bill.displayReceipt();

}

**OUTPUT :**



**Question # 3 :**

**INPUT :**

#include <iostream>

using namespace std;

class MealBill {

private:

int cost;

string nameOfResturant="AAK'S CAFE";

int localSalesRate=100;

int tip;

int amountPayed;

public:

MealBill(int price, int amPayed)

{

cost = price;

tip = 0.15 \* cost;

amountPayed = amPayed;

}

void displayBill() {

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

cout << "Meal Cost: " << cost << endl;

cout << "Tax: " << localSalesRate << endl;

cout << "Tip: " << tip << endl;

cout << "Total Cost: " << cost + localSalesRate + tip << endl;

}

void displayReceipt() {

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

cout << "Amount Payed: " << amountPayed;

cout << endl;

cout << "Total Cost" << cost + localSalesRate + tip << endl;

cout << "Change: " << amountPayed - (cost + localSalesRate + tip) << endl;

cout << endl;

cout << "Thank you for dining at " << nameOfResturant << endl;

}

void displayBalance() {

cout << "================AHMED ACCOUNT ====================" << endl;

cout << "ID: 0210-BCS-06"<<endl;

cout << "Name: Ahmed Ali" << endl;

cout << endl;

cout << "ACCOUNT BALANCE: " << 2000 - amountPayed + amountPayed - (cost + localSalesRate + tip) << endl;

}

};

int main() {

string name;

int cost1, amountPayed;

cout << "Enter Meal Cost: ";

cin >> cost1;

cout << "Enter Amount Payed: " ;

cin >> amountPayed;

MealBill bill( cost1, amountPayed);

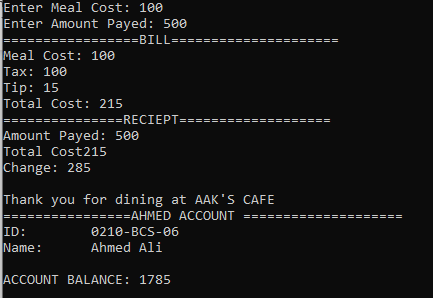
bill.displayBill();

bill.displayReceipt();

bill.displayBalance();

}

**OUTPUT :**



**Question # 4 :**

**INPUT :**

#include <iostream>

#include <conio.h>

#include <string>

using namespace std;

class GradeRecord

{

private:

string student\_id;

int number\_of\_credits;

double grade\_points, GPA;

public:

GradeRecord()

{};

GradeRecord(string id, int c, double gp)

{

student\_id = id;

number\_of\_credits = c;

grade\_points = gp;

};

void calculatedata()

{

char grade;

string subject\_name;

double credit\_hours, gradepoints;

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

cin >> subject\_name;

cout << "Enter grade achieved in this subject: ";

cin >> grade;

cout << "Enter the credit hours of this subject: ";

cin >> credit\_hours;

if (grade == 'A')

{

gradepoints = 4 \* credit\_hours;

grade\_points += gradepoints;

number\_of\_credits += credit\_hours;

}

else if (grade == 'B')

{

gradepoints = 3 \* credit\_hours;

grade\_points += gradepoints;

number\_of\_credits += credit\_hours;

}

else if (grade == 'C')

{

gradepoints = 2 \* credit\_hours;

grade\_points += gradepoints;

number\_of\_credits += credit\_hours;

}

else if (grade == 'D')

{

gradepoints = credit\_hours;

grade\_points += gradepoints;

number\_of\_credits += credit\_hours;

}

else if (grade == 'E')

{

gradepoints = 0;

grade\_points += gradepoints;

number\_of\_credits += credit\_hours;

}

else

{

cout << "Invalid grade entered!!" << endl;

}

}

double getgradepoints()

{

return grade\_points;

}

double getnumberofcredits()

{

return number\_of\_credits;

}

double calculategpa()

{

GPA = grade\_points / number\_of\_credits;

return GPA;

}

void showstudentrecord()

{

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

cout << "Student: " << student\_id << "\tUnits: " << number\_of\_credits << "\tGradePts: " << grade\_points << "\tGPA: " << GPA << endl;

}

};

int main()

{

GradeRecord gr("0120-BESE-56", 100, 345);

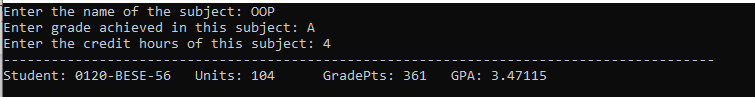
gr.calculatedata();

gr.calculategpa();

gr.showstudentrecord();

}

**OUTPUT :**



**Question # 5 :**

**INPUT :**

#include<conio.h>

#include<string.h>

#include<iostream>

using namespace std;

class GradeRecord {

public:

string studentID;

int credit;

int gradepoint;

public:

GradeRecord(string i)

{

studentID = i;

credit = 0;

gradepoint = 0;

};

GradeRecord(string i, int c, int gr\_p)

{

studentID = i;

credit = c;

gradepoint = gr\_p;

};

double calculate\_GPA()

{

if (credit > 0)

{

return (double)gradepoint / (double)credit;

}

else

{

return 0;

}

}

void gp\_info()

{

cout << "STUDENT:" << studentID;

cout << "\tUnits:" << credit;

cout << "\tGradePts:" << gradepoint;

cout << "\tGPA";

printf("%.2f", calculate\_GPA());

}

void updaterec(int c, int gr\_p) {

credit += c;

gradepoint += gr\_p;

}

};

int main() {

GradeRecord g1("0210-BCS-01");

cout << "Ali's GPA is " << g1.calculate\_GPA() << endl;

int c, g;

cout << "1st semester grade points and units are:";

cin >> g >> c;

g1.updaterec( c, g);

g1.gp\_info();

}

**OUTPUT :**



**Question # 6 :**

**INPUT :**

#include <iostream>

#include<iomanip>

#include <string>

using namespace std;

class Person

{

protected:

string name;

string gender;

int age;

public:

Person()

{};

Person(string n, string g, int a) :name(n), gender(g), age(a)

{};

string getname()

{

return name;

}

string getgender()

{

return gender;

}

int getage()

{

return age;

}

void setname(string n)

{

name = n;

}

void setgender(string g)

{

gender = g;

}

void setage(int a)

{

age = a;

}

virtual void showinfo()

{

cout << "Name : " << name << endl;

cout << "Gender : " << gender << endl;

cout << "Age : " << age << endl;

}

};

class Student :public Person

{

protected:

int rollNo;

public:

Student()

{};

Student(string n, string g, int a, int r) :Person(n, g, a), rollNo(r)

{};

int getrollNo()

{

return rollNo;

}

void setrollNo(int r)

{

rollNo = r;

}

virtual void showinfo()

{

Person::showinfo();

cout << "Roll No : " << rollNo << endl;

}

};

class Graduate :public Student

{

private:

double gradePoints;

int creditHours;

public:

Graduate()

{};

Graduate(string n, string g, int a, int r, double gp, int ch) :Student(n, g, a, r), gradePoints(gp), creditHours(ch)

{};

int getcredit()

{

return creditHours;

}

double getgrade()

{

return gradePoints;

}

void setcredit(int ch)

{

creditHours = ch;

}

void setgrade(double gp)

{

gradePoints = gp;

}

double cgpa()

{

return gradePoints / creditHours;

}

virtual void showinfo()

{

Student::showinfo();

cout << "Credit Hours : " << creditHours << endl;

cout << "Grade Points : " << gradePoints << endl;

cout << "GPA : " << setprecision(3) << cgpa() << endl;

}

};

int main()

{

Graduate grad("Asad Ali Khan", "Male", 19, 56, 60, 16);

grad.showinfo();

}

**OUTPUT :**

