QNO:2

CSStudents.h

```
#include<iostream>
#include<string>
using namespace std;
class CSStudents
{
        string Name;
        float Fee;
public:
        CSStudents()
        {
                Name ="";
                Fee = 0;
        }
        void setValues(string N,int F)
        {
                Name = N;
                Fee = F;
        }
        void getName();
        void getFee();
        float calculateFee(float);
};
```

CSStudents.cpp

```
#include "CSStudents.h"
#include<iostream>
#include<string>
using namespace std;
void CSStudents::getName()
{
	cout << "Name of student:"<<Name;
}
void CSStudents::getFee()
{
	cout << "\nFee of student:" << Fee<<endl;
}
float CSStudents::calculateFee(float percentage)
{
	if (percentage >= 0)
	{
		 Fee += ((Fee*percentage) / 100);
		return Fee;
	}
	else
	{
		cout << "Error.";
		return Fee;
	}
}
int main()
{
	CSStudents students[3];
	students[0].setValues("SAAD", 23455);
	students[1].setValues("SHAHZAIN", 34767);
	students[2].setValues("SHARIQ", 87564);
	for (int i = 0; i < 3; i++)
	{
		cout << "Details of student:" << i + 1 << endl;
		students[i].getName();
		students[i].getFee();
	}
	float per;
	cout << "Enter raise percentage :";
	cin >> per;
	cout << "Detail of all student after raise percentage of" << per << "%" << "in fee" <<
endl;
	for (int i = 0; i < 3; i++)
	{
		float result = students[i].calculateFee(per);
```

```
            cout << "Details of student:" << i + 1 << endl;
            students[i].getName();
            cout << "\nUpdated Fee of student:" << result << endl;
    }
    system("pause");
    return 0;
}
```

QNO:1

Bank.h

```
#include<iostream>
#include<string>
using namespace std;
class Bank
{
        string Name, AccountType;
        int AccountNumber, Balance;
public:
        void getdetail()
        {
                cout << "Enter name of customer:";
                cin >> Name;
                cout << "Enter account number:";
                cin >> AccountNumber;
                cout << "Enter account type:";
                cin >> AccountType;
                cout << "Enter total balance:";
                cin >> Balance;
        }
        void deposit();
        void withDraw();
        void displayDetail()
        {
                        cout << "Name of customer:" << Name << endl
                                << "Account number:" << AccountNumber << endl
                                << "Account type:" << AccountType << endl
                                << "Total balance:" << Balance << endl;
        }
};
```

Bank.cpp

```
#include "Bank.h"
#include<iostream>
#include<string>
using namespace std;
void Bank::deposit()
{
        int depositAmount;
        cout << "Enter amount to deposit:";
        cin >> depositAmount;
        Balance += depositAmount;
        cout << "Balance is=" << Balance << endl;
}
```

```
void Bank::withDraw()
{
        int withDrawAmount;
        cout << "Enter amount to withdraw:";
        cin >> withDrawAmount;
        Balance -= withDrawAmount;
        cout << "Balance is=" << Balance<<endl;
}
void menu()
{
        cout <<"***********MENU***********"<<endl
                <<"^^^^^Enter 0 to exit^^^^^" << endl
                << "^^^^^Enter 1 to intialize new account^^^^^"<< endl
                << "^^^^^Enter 2 to deposit^^^^^" << endl
                << "^^^^^Enter 3 to withdraw^^^^^" << endl
                << "^^^^^Enter 4 to see account status^^^^^" << endl;
}
int main()
{
        Bank obj;
        int option;
        while (true)
        {
                menu();
                cout << "Enter option:";
                cin >> option;
                switch (option)
                {
                case 0: exit(0);
                case 1:
                {
                        obj.getdetail();
                        break;
                }
                case 2:
                {
                        obj.deposit();
                        break;
                }
                case 3:
                {
                        obj.withDraw();
                        break;
                }
                case 4:
                {
```

```
                obj.displayDetail();
                break;
            }
            default: cout << "Invalid option.";
            }
    }
    system("pause");
    return 0;
}
```