Task 5:

#include<iostream>

using namespace std;

class bank{

private:

double balance;

double interestRate;

double interest;

int transaction;

static int count;

public:

bank() {

interestRate = 3.5;

balance = 0;

transaction = 0;

interest = 0;

}

bank(int interest, double amount) {

interestRate = interest;

balance = amount;

transaction = 0;

interest = 0;

}

void setInterestRate(double x) {

interestRate = x;

}

void makeDeposite(double x) {

balance += x;

cout << "deposited\n";

}

void WithDraw(double x) {

if (x > balance) {

cout << "Error! Entered Amount for with Drawal is not valid " << endl;

}

else {

balance -= x;

}

}

void calcInterest() {

interest = interestRate \* balance;

balance += interest;

}

void incCount() {

count++;

}

int getCount()const {

return count;

}

double getInterestRate()const {

return interestRate;

}

double GetBalance()const {

return balance;

}

double getInterest()const {

return interest;

}

int getTransaction()const {

return transaction;

}

void incTransaction() {

transaction++;

}

};

int bank::count = 0;

int main() {

bank B(1.5, 1000);

int choice;

while (true) {

system("pause");

cout << endl;

cout << "1- Display the account Balance" << endl;

cout << "2- Display thr number of Transactions" << endl;

cout << "3- Display interest earned for this period" << endl;

cout << "4- Make a deposit" << endl;

cout << "5- Make a withdrawal" << endl;

cout << "6- Add interest for this period" << endl;

cout << "7- Exit the program" << endl;

cout << "8- Number of times the Program has taken Choice : " << B.getCount() << endl;

cin >> choice;

switch (choice) {

case 1: {

system("cls");

system("pause");

cout << "\nAccount Balance is : $" << B.GetBalance() << endl;

B.incCount();

break;

}

case 2: {

system("cls");

system("pause");

cout << "Number of Transaction : " << B.getTransaction() << endl;

B.incCount();

break;

}

case'C': {

system("cls");

system("pause");

cout << "Interest Earned for this period : " << B.getInterest() << endl;

B.incCount();

break;

}

case 3: {

system("cls");

system("pause");

double temp;

cout << "Enter the Ammount of Withdrawal : ";

cin >> temp;

B.WithDraw(temp);

cout << endl;

B.incCount();

B.incTransaction();

break;

}

case 4: {

system("cls");

system("pause");

double temp;

cout << "Enter the Ammount to deposit : ";

cin >> temp;

B.makeDeposite(temp);

cout << endl;

B.incCount();

B.incTransaction();

break;

}

case 5: {

system("cls");

system("pause");

B.calcInterest();

cout << "Interst Added to your Balance! \n";

B.incCount();

}

case 6: {

break;

break;

}

}

}

}







