

Name: Adil Khan

Department: BS AI

Roll no: 2330-0030

Subject: Object Oriented Programming

Teacher: Mam Saba Awan

**Lab # 05**

Task no 1:

#include <iostream>

#include <string>

using namespace std;

class BankAccount {

private:

string accountHolderName;

string accountNumber;

double balance;

static int accountCount;

bool validateAmount(double amount) {

return amount > 0;

}

public:

BankAccount(string name, string accNo, double initialBalance) {

accountHolderName = name;

accountNumber = accNo;

if (validateAmount(initialBalance)) {

this->balance = initialBalance;

} else {

this->balance = 0;

}

accountCount++;

}

void deposit(double amount) {

if (validateAmount(amount)) {

this->balance += amount;

cout << "Deposited: " << amount << " to your account.\n";

} else {

cout << "Invalid deposit amount. Please enter a positive value.\n";

}

}

void withdraw(double amount) {

if (validateAmount(amount)) {

if (this->balance >= amount) {

this->balance -= amount;

cout << "Withdrawn: " << amount << " from your account.\n";

} else {

cout << "Insufficient funds. Withdrawal failed.\n";

}

} else {

cout << "Invalid withdrawal amount. Please enter a positive value.\n";

}

}

double getBalance() const {

return balance;

}

void displayAccountInfo() {

cout << "Account Holder Name: " << accountHolderName << endl;

cout << "Account Number: " << accountNumber << endl;

cout << "Account Balance: Rs" << balance << endl;

}

static int getAccountCount() {

return accountCount;

}

};

int BankAccount::accountCount = 0;

int main() {

BankAccount account1("Adil", "1234", 1000);

BankAccount account2("Sara", "5678", 500);

account1.displayAccountInfo();

account1.deposit(500);

account1.withdraw(200);

account1.withdraw(1500);

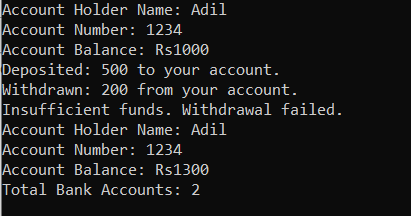
account1.displayAccountInfo();

cout << "Total Bank Accounts: " << BankAccount::getAccountCount() << endl;

return 0;

}

Output:



Task no 2:

#include <iostream>

#include <string>

using namespace std;

class HotelOrder {

private:

string name;

mutable string placedOrder;

int tableNo;

mutable int bill;

public:

HotelOrder(string nam, string plac, int tabno, int bil) {

name = nam;

placedOrder = plac;

tableNo = tabno;

bill = bil;

}

void changePlacedOrder(string newOrder) const {

placedOrder = newOrder;

cout << "Order changed successfully to: " << placedOrder << endl;

}

void updateBill(int newBill) const {

bill = newBill;

cout << "Bill updated successfully: Rs" << bill << endl;

}

void displayOrder() const {

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

cout << "Table Number: " << tableNo << endl;

cout << "Placed Order: " << placedOrder << endl;

cout << "Bill: Rs" << bill << endl;

}

};

int main() {

HotelOrder order("Adil", "Pizza", 5, 500);

order.displayOrder();

order.changePlacedOrder("Burger");

order.updateBill(400);

order.displayOrder();

return 0;

}

Output:

