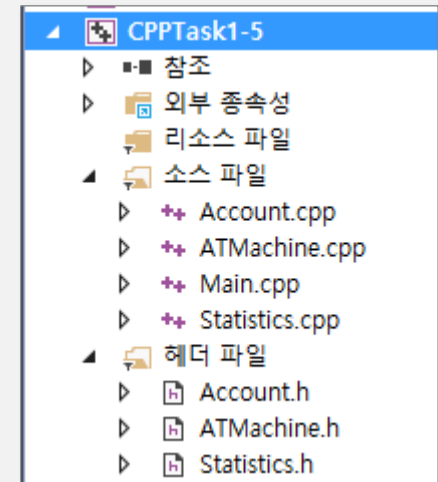


1. 과제 내용 : ATM 프로그램 작성
2. 수행 방법 : 프로젝트 명 : **영문이니셜_ATM**
3. 제출 항목 : **프로젝트 폴더를 압축하여 제출**
압축 파일 명: **이름_ATM.zip**
4. 제출 기한 : 4월 12일까지
5. 기능 설명



기능(배점)		내용	프로젝트 파일
기본 (2점)	1	계좌 개설 기능, 계좌 조회 기능, 업무 종료 기능	Main.cpp, Account.cpp, ATMachine.cpp Account.h, ATMachine.h
추가 (2점)	1	계좌 입금 기능	
	2	계좌 출금 기능	
	3	계좌 이체 기능	
	4	고객 관리 기능	Statistics.cpp, Statistics.h 추가

* 추가 기능은 순서대로 구현할 것

기본기능 : Main.cpp와 결과 화면

```
int main() {
    int select;
    // 고객수(100명), ATM 잔금 초기화, 관리자 암호
    ATMMachine atm(100, 50000, "admin");

    while(1) {
        atm.DisplayMenu();
        cout << "메뉴를 선택하세요: ";
        cin >> select;
        cout << endl;

        switch(select) {
            case 1:
                atm.createAccount();
                break;

            .....

            default:
                cout << "번호 확인 후 다시 입력하세요." << endl;
        }
    }
    return 0;
}
```

Main.cpp

```
C:\WINDOWS\system32\cmd.exe

----- KPU bank -----
1. 계좌 개설
2. 계좌 조회
9. 업무 종료
메뉴를 선택하세요: 1

----- 개설 -----
이름 입력: 류현진
암호 입력: 1234

류현진님 100번 계좌번호가 정상적으로 개설되었습니다. 감사합니다

----- KPU bank -----
1. 계좌 개설
2. 계좌 조회
9. 업무 종료
메뉴를 선택하세요: 2

----- 조회 -----
계좌번호 입력: 100
비밀번호 입력: 1234
현재 잔액 : 0

----- KPU bank -----
1. 계좌 개설
2. 계좌 조회
9. 업무 종료
메뉴를 선택하세요: 9

안녕히 가세요
계속하려면 아무 키나 누르십시오 . . .
```

기본기능 : ATMachine.h

```
#ifndef ATMACHINE_H
#define ATMACHINE_H

using std::string;

#define NEGATIVE_ATM_BALANCE    -10           // ATM 잔액 부족

class ATMachine {
private:
    Account * pAcctArray;           // 동적 생성된 고객계좌 배열 포인터
    int nMachineBalance;           // ATM 잔고
    int nMaxAccountNum;            // 동적 생성된 고객계좌 배열 크기
    int nCurrentAccountNum;        // 개설된 고객 계좌 수
    string strManagerPassword;     // 관리자 비밀번호

public:
    ATMachine(int size, int balance, string password); // 계좌 배열크기, ATM 잔고, 관리자 암호 초기화
    ~ATMachine();

    void displayMenu();             // ATM 기능 선택 화면
    void createAccount();           // 계좌 개설
    void checkMoney();              // 계좌 조회
};

#endif
```

ATMachine.h

기본기능 : Account.h

Account.h

```
#ifndef ACCOUNT_H
#define ACCOUNT_H

using std::string;

#define AUTHENTICATION_FAIL    -1    // 계정 인증 실패
#define AUTHENTICATION_SUCCESS 1     // 계정 인증 성공
#define BASE_ACCOUNT_ID      100    // 계좌번호는 100번 부터

class Account {
private:
    int nID;                // 계좌 번호
    int nBalance;           // 잔고
    string strAccountName;  // 고객 명
    string strPassword;     // 계좌 비밀번호

private:
    inline bool authenticate(int id, string passwd); // 인증 성공 : true(1), 인증 실패 : false(0)

public:
    Account();
    void create(int id, int money, string name, string password); // 계좌 개설
    int check(int id, string password); // return 값 : nBalance (잔고) or 인증 실패(AUTHENTICATION_FAIL)

    int getAcctID() { return nID; } // 계좌 번호 읽어오기
};

#endif
```

추가 기능 1 : 계좌 입금

```
// return 값 :
// nBalance (최종잔고) or 인증 실패(AUTHENTICATION_FAIL)

Account.h

public:
    int deposit(int id, string password, int money);
```

```
ATMachine.h

public:
    void depositMoney(); // 계좌 입금
```

```
C:\WINDOWS\system32\cmd.exe

----- KPU bank -----
1. 계좌 개설
2. 계좌 조회
3. 계좌 입금
9. 업무 종료
메뉴를 선택하세요: 1

----- 개설 -----
이름 입력: 류현진
암호 입력: 1234

류현진님 100번 계좌번호가 정상적으로 개설되었습니다. 감사합니다

----- KPU bank -----
1. 계좌 개설
2. 계좌 조회
3. 계좌 입금
9. 업무 종료
메뉴를 선택하세요: 3

----- 입금 -----
계좌번호 입력: 100
비밀번호 입력: 1234
입금액 입력: 4000
현재 잔액 : 4000
입금 완료

----- KPU bank -----
1. 계좌 개설
2. 계좌 조회
3. 계좌 입금
9. 업무 종료
메뉴를 선택하세요: 9

안녕히 가세요
계속하려면 아무 키나 누르십시오 . . .
```

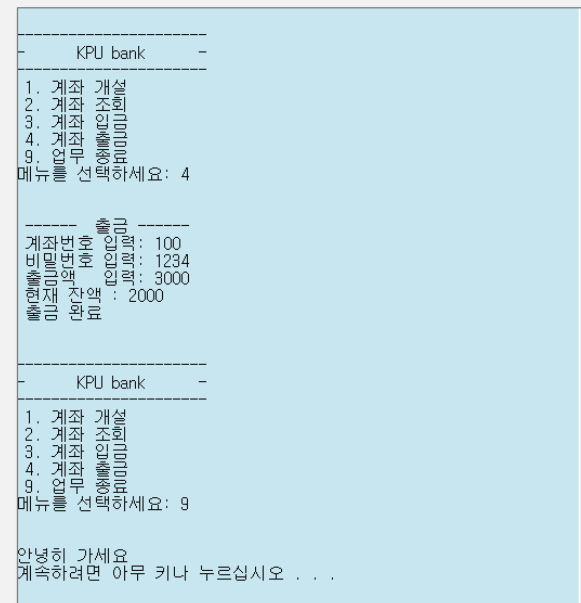
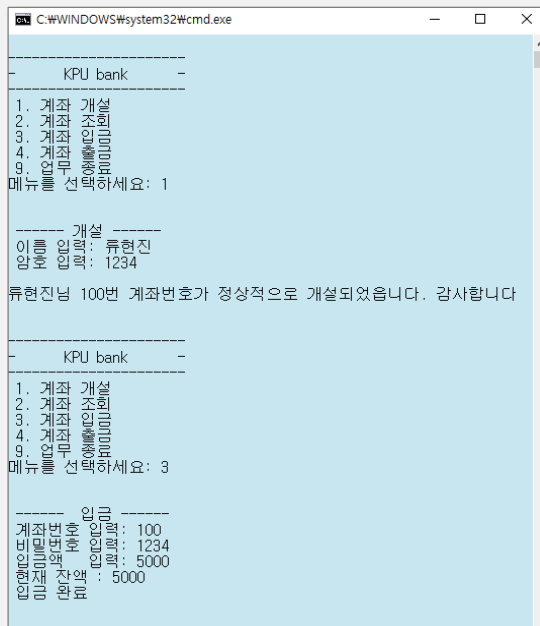
추가 기능 2 : 계좌 출금

Account.h

```
// return 값 :  
// nBalance (최종잔고) or 인증 실패(AUTHENTICATION_FAIL)  
  
public:  
    int withdraw(int id, string password, int money);
```

ATMachine.h

```
public:  
    void withdrawMoney(); // 계좌 출금
```



추가 기능 3 : 계좌 이체

```
// return 값 :  
// nBalance (최종잔고)
```

```
public:  
    int deposit(int id, int money);
```

Account.h

```
public:  
    void void transfer(); // 계좌 이체
```

ATMachine.h

```
C:\WINDOWS\system32\cmd.exe

KPU bank
1. 계좌 개설
2. 계좌 잔액조회
3. 계좌입금
4. 계좌출금
5. 계좌이체
6. 인출
7. 종료
메뉴를 선택하세요: 1

----- 개설 -----
이름 입력: 류현진
암호 입력: 1234
류현진님 100번 계좌번호가 정상적으로 개설되었습니다. 감사합니다

KPU bank
1. 계좌 개설
2. 계좌 잔액조회
3. 계좌입금
4. 계좌출금
5. 계좌이체
6. 인출
7. 종료
메뉴를 선택하세요: 3

----- 이체 -----
계좌번호: 100
비밀번호: 1234
이체금액: 101
이체금의 암호: 3000
현재잔액: 5000
이체 완료
```

```
KPU bank
1. 계좌 개설
2. 계좌 잔액조회
3. 계좌입금
4. 계좌출금
5. 계좌이체
6. 인출
7. 종료
메뉴를 선택하세요: 1

----- 개설 -----
이름 입력: 배지현
암호 입력: 1234
배지현님 101번 계좌번호가 정상적으로 개설되었습니다. 감사합니다
```

```
KPU bank
1. 계좌 개설
2. 계좌 잔액조회
3. 계좌입금
4. 계좌출금
5. 계좌이체
6. 인출
7. 종료
메뉴를 선택하세요: 5

----- 이체 -----
계좌번호: 100
비밀번호: 1234
이체금액: 101
이체금의 암호: 3000
현재잔액: 5000
이체 완료

KPU bank
1. 계좌 개설
2. 계좌 잔액조회
3. 계좌입금
4. 계좌출금
5. 계좌이체
6. 인출
7. 종료
메뉴를 선택하세요: 9

안녕히 가세요
계속하려면 아무 키나 누르십시오 . . .
```

추가 기능 4 : 고객 관리

Account.h

```
// return 값 :  
// nBalance (최종잔고) or 인증 실패(AUTHENTICATION_FAIL)  
public:  
    int getBalance() { return nBalance; }  
    string getAccountName() { return strAccountName; }
```

ATMachine.h

```
private:  
    bool isManager(string password); // 메니저 암호 확인  
    void displayReport();           // 통계 화면 처리  
public:  
    void managerMode();             // 고객 관리
```

Statistics.h

```
#ifndef STATISCIS_H  
#define STATISCIS_H  
  
class Statistics {  
  
public:  
    int sum(Account * pArray, int size); // 계좌 잔고 총합  
    int average(Account * pArray, int size); // 계좌 잔고 평균  
    int max(Account * pArray, int size); // 계좌 최고 값  
};  
  
#endif
```


추가 기능 4 : 고객 관리

```
C:\WINDOWS\system32\cmd.exe

KPU bank
-----
1. 계좌 개설
2. 계좌 조회
3. 계좌입금
4. 계좌출금
5. 계좌이체
6. 고객관리
9. 종료
메뉴를 선택하세요: 1

----- 개설 -----
이름 입력: 류현진
암호 입력: 1234

류현진님 100번 계좌번호가 정상적으로 개설되었습니다. 감사합니다

KPU bank
-----
1. 계좌 개설
2. 계좌 조회
3. 계좌입금
4. 계좌출금
5. 계좌이체
6. 고객관리
9. 종료
메뉴를 선택하세요: 3

----- 입금 -----
계좌번호: 100
입금금액: 1234
잔액: 9000
메뉴를 선택하세요: 3
```

```
KPU bank
-----
1. 계좌 개설
2. 계좌 조회
3. 계좌입금
4. 계좌출금
5. 계좌이체
6. 고객관리
9. 종료
메뉴를 선택하세요: 1

----- 개설 -----
이름 입력: 배지현
암호 입력: 1234

배지현님 101번 계좌번호가 정상적으로 개설되었습니다. 감사합니다

KPU bank
-----
1. 계좌 개설
2. 계좌 조회
3. 계좌입금
4. 계좌출금
5. 계좌이체
6. 고객관리
9. 종료
메뉴를 선택하세요: 3

----- 입금 -----
계좌번호: 101
입금금액: 1234
잔액: 5000
메뉴를 선택하세요: 3
```

```
C:\WINDOWS\system32\cmd.exe

KPU bank
-----
1. 계좌 개설
2. 계좌 조회
3. 계좌입금
4. 계좌출금
5. 계좌이체
6. 고객관리
9. 종료
메뉴를 선택하세요: 6

----- 관리 -----
관리자 비밀번호 입력: admin

관리자입니다.

ATM 현금 잔고: 64000원
고객 잔고 총액: 14000원 (총2명)
고객 잔고 평균: 7000원
고객 잔고 최고: 9000원

----- 계좌현황 -----
류현진 100 9000
배지현 101 5000

KPU bank
-----
1. 계좌 개설
2. 계좌 조회
3. 계좌입금
4. 계좌출금
5. 계좌이체
6. 고객관리
9. 종료
메뉴를 선택하세요: 9

안녕히 가세요
계속하려면 아무 키나 누르십시오 . . .
```