

# 자바 팀 프로젝트

”

쇼핑 기능을 추가한 A-pay 뱅킹시스템 구현

피피티 시작 >

보기만해도  
흐 — 못한

# 3조

조원소개 >



백재현  
메뉴 및 입금 구현



서진혁  
쇼핑 구현



서정우  
쇼핑 구현



선우사랑  
메뉴 및 출금 및 송금 구현

# 피피티 순서

”

**01**

사용 프로그램 정보

**02**

프로젝트 진행 순서

**03**

프로그램 설명

**04**

스토리보드

**05**

실제 계획했던 부분과  
실패했던 것

**06**

프로젝트 소감



”

☆ ☆ ☆  
사용 프로그램 정보

01



SOFTWARE : jdk 1.8



IDE : eclipse



개발 도구를 앞으로 늘려나가야 한다...



”

프로젝트 진행 순서

02

## 프로젝트 진행 순서

날짜 별 흐름

20 %

7.21.-22.(목-금요일)

1. 콘솔프로그램 파악
2. 프로젝트 주제 선정 및 개별 구현할 파트 분담

WOW !

70 %

7.23.-24.(토-일요일)

1. 각자 구현할 클래스 작업
  - 백재현 : 가입메뉴 및 입금메서드
  - 서정우 : 쇼핑메뉴 구현, 디자인 수정
  - 서진혁 : 쇼핑/선물하기 기능구현
  - 선우사랑 : 가입메뉴 및 출/송금 메서드, 디자인

90 %

7.25.(월요일)

1. 개인별 변수와 작업물 통합
2. 오류 수정

100 %

7.26.-27.(화-수요일)

1. 디자인 및 자잘한 오류 수정
2. 피피티 제작



”

프로그램 설명

03



# 프로그램 : A-pay

”

해당 프로그램을 선택한 이유 : बैं킹과 atm 프로그램은 자바 콘솔에서 많이 구현하는 프로그램 중 하나로 회원을 등록하는 과정이나 메뉴를 구현하는 과정 등에서 그동안 배운 컬렉션이나 여러 문들을 복습하고 활용하기 용이한 프로그램이라 생각이 들어서 선택했습니다.

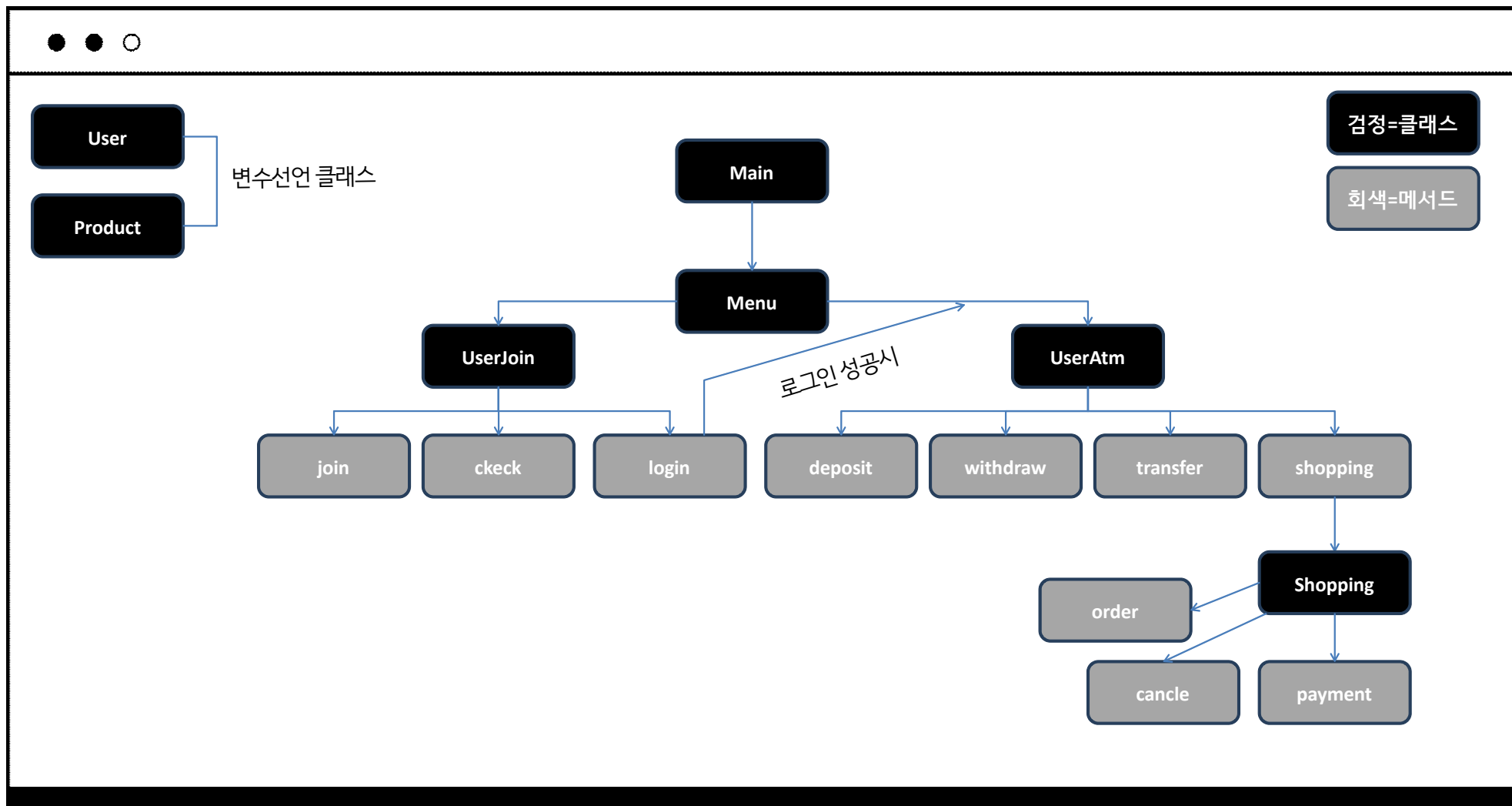
A-pay 기능 : 회원을 등록후에 로그인하여 입금 출금 그리고 등록된 유저간 송금과 같은 बैं킹시스템이 가능하며 단순한 बैं킹시스템을 넘어서 카카오페이처럼 쇼핑과 선물하기 기능을 구현했습니다.



”

☆ ☆ ☆  
스토리보드

04



## 메뉴 스토리보드

Main을 실행시 menuImpl을 통해 회원가입과 로그인을 위한 loginMenu메서드를 구현함

loginMenu메서드는 while문을 통해 이미지와 같은 메뉴창을 반복한다.

try~catch문을 통해 정수를 입력하지 않으면 다음과 같이 되돌아온다.

```
┌────────── A-pay ──────────┐
| 1. 회원가입                  |
| 2. 로그인                    |
| 3. 회원확인                  |
| 0. 프로그램종료              |
└────────────────────────────────┘
```

선택 :

숫자로 메뉴를 선택해주세요.

## 메뉴 스토리보드

1번 메뉴를 선택할 경우 menuImpl에 연결된 UserJoinImpl클래스의 join메서드로 가서 이름과 아이디와 비밀번호와 거래시 필요한 핀넘버(6자리)를 입력하게 한다.

핀번호를 정수로 입력하지 않으면 다시 처음 메뉴로 돌아온다.,

```
===== A-pay =====  
| 1. 회원가입  
| 2. 로그인  
| 3. 회원확인  
| 0. 프로그램종료  
=====
```

선택 : 1

```
===== 회원가입 =====  
| 이름 등록 : 백재현  
| 아이디 등록 : a  
| 비밀번호 등록 : 1  
| 핀번호 [ 6자리 ] 등록 : 123456  
숫자로 메뉴를 선택해주세요.
```

# 메뉴 스토리보드

다음과 같이 핀번호를 6자리로 입력하지 않으면 다시 입력하게 하고 핀번호까지 제대로 입력하면 계좌번호를 Math.Random을 이용하여 6자리의 랜덤한 숫자로 만들어준다.

```
===== A-pay =====
| 1. 회원가입
| 2. 로그인
| 3. 회원확인
| 0. 프로그램종료
=====
```

선택 : 1

```
===== 회원가입 =====
| 이름 등록 : 백재현
| 아이디 등록 : a
| 비밀번호 등록 : 1
| 핀번호 [6자리] 등록 : 12345
| 핀번호는 6자리로 입력해주세요.
| 핀번호 [6자리] 등록 : 123456
| 계좌번호 생성 중..
| 계좌번호 : 228160
=====
```

# 메뉴 스토리보드

변수를 선언해 둔 User클래스를 List컬렉션을 이용해 배열화 했기 때문에 계속해서 회원을 등록할 수 있다.

```
===== A-pay =====  
| 1. 회원가입  
| 2. 로그인  
| 3. 회원확인  
| 0. 프로그램종료  
=====
```

선택 : 1

```
===== 회원가입 =====  
| 이름 등록 : 서정우  
| 아이디 등록 : b  
| 비밀번호 등록 : 2  
| 핀번호 [ 6자리 ] 등록 : 234567  
| 계좌번호 생성 중..  
| 계좌번호 : 957853  
=====
```

## 메뉴 스토리보드

이름이나 패스워드 등은 겹칠 수 있지만 아이디는 중복이 불가하게 설정하여 같은 아이디로 회원가입시 가입이 불가하다.

즉 아이디를 통해 회원의 구분이 가능하다.

```
===== A-pay =====  
| 1. 회원가입  
| 2. 로그인  
| 3. 회원확인  
| 0. 프로그램종료  
=====
```

선택 : 1

```
===== 회원가입 =====  
| 이름 등록 : 김영석  
| 아이디 등록 : a  
| 같은 아이디가 존재합니다.  
=====
```



# 메뉴 스토리보드

회원확인 은 UserJoinImpl클래스의 check메서드로 이동한다. 회원확인을 통해 그동안 입력했던 회원들을 모두 확인할 수 있다.

잔액은 당연히 0원으로 시작

```
===== A-pay =====  
1. 회원가입  
2. 로그인  
3. 회원확인  
0. 프로그램종료
```

선택 : 3

```
===== 회원정보 =====  
이름 : 박재현  
아이디 : a  
비밀번호 : 1  
전화번호 : 123456  
계좌번호 : 228160  
계좌잔액 : 0
```

```
===== 회원정보 =====  
이름 : 서정우  
아이디 : b  
비밀번호 : 2  
전화번호 : 234567  
계좌번호 : 957853  
계좌잔액 : 0
```

```
===== 회원정보 =====  
이름 : 서진혁  
아이디 : c  
비밀번호 : 3  
전화번호 : 345678  
계좌번호 : 338406  
계좌잔액 : 0
```

```
===== 회원정보 =====  
이름 : 선우사랑  
아이디 : d  
비밀번호 : 4  
전화번호 : 456789  
계좌번호 : 800519  
계좌잔액 : 0
```

# 메뉴 스토리보드

로그인시 UserJoinImpl클래스의 login메서드로 이동하고 비밀번호가 틀리거나 없는 아이디 입력시 로그인이 불가능하다.

```
===== A-pay =====  
| 1. 회원가입  
| 2. 로그인  
| 3. 회원확인  
| 0. 프로그램종료  
=====
```

선택 : 2

```
===== 로그인 =====  
| 아이디 입력 : a  
| 비밀번호 입력 : 3  
| 입력한 정보가 맞지 않습니다.  
=====
```

```
===== A-pay =====  
| 1. 회원가입  
| 2. 로그인  
| 3. 회원확인  
| 0. 프로그램종료  
=====
```

선택 : 2

```
===== 로그인 =====  
| 아이디 입력 : f  
| 비밀번호 입력 : 1  
| 입력한 정보가 맞지 않습니다.  
=====
```

# 메뉴 스토리보드

로그인이 성공하면 menuImpl의 userMenu메서드로 이동해 회원 메뉴창을 반복한다.

이 때 login메서드에 입력한 아이디와 패스워드를 owner변수에 저장해서 로그인한 계정의 정보를 회원 메뉴에 보존한다.

```
===== A-pay =====  
| 1. 회원가입  
| 2. 로그인  
| 3. 회원확인  
| 0. 프로그램종료  
=====
```

선택 : 2

```
===== 로그인 =====  
| 아이디 입력 : a  
| 비밀번호 입력 : 1  
| 백재현님 환영합니다.  
=====
```

```
===== 회원 메뉴 =====  
| 1. 입금  
| 2. 출금  
| 3. 송금  
| 4. 쇼핑하기  
| 0. 로그아웃  
=====
```

# 입금 스토리보드

입금시 핀번호 입력은 for문을 통해 3번까지 시도가 가능하며 3번 오류시 회원 메뉴로 되돌아간다.

회원 메뉴

1. 입금

2. 출금

3. 송금

4. 쇼핑하기

0. 로그아웃

선택 : 1  
입금할 금액 입력 : 50000  
핀번호[6자리] 입력 : 3  
1회 핀번호 오류! 3회 실패시 종료됩니다.  
핀번호[6자리] 입력 : 4  
2회 핀번호 오류! 3회 실패시 종료됩니다.  
핀번호[6자리] 입력 : 5  
3회 핀번호 오류! 3회 실패시 종료됩니다.

# 입금 스토리보드

입금은 다른 뱅킹시스템처럼 최소금액의 제한(1만원)을 두어 금액이 적으면 핀번호가 일치해도 입금이 불가능하다.

회원 메뉴

- | 1. 입금
- | 2. 출금
- | 3. 송금
- | 4. 쇼핑하기
- | 0. 로그아웃

선택 : 1

입금할 금액 입력 : 5

핀번호 [ 6자리 ] 입력 : 123456

10,000원이상 입금가능합니다.

# 입금 스토리보드

입금액이 1만원 이상에 핀번호가 로그인 한 계정과 일치하면 정상적으로 입금완료

## 회원 메뉴

- 1. 입금
- 2. 출금
- 3. 송금
- 4. 쇼핑하기
- 0. 로그아웃

선택 : 1

입금할 금액 입력 : 50000

핀번호 [6자리] 입력 : 123456

50000원이 입금 되었습니다.

잔액 : 50000

# 입금 스토리보드

로그아웃 후 check메서드로 확인해보면 다음과 같이 로그인했던 계정에 입력했던 5만원이 정상적으로 입금됐음을 확인 할 수 있다.

```
===== 회원 메뉴 =====
| 1. 입금
| 2. 출금
| 3. 송금
| 4. 쇼핑하기
| 0. 로그아웃
```

선택 : 0

로그아웃 완료!!!

메인 페이지로 돌아갑니다. 이용해주셔서 감사합니다.

```
===== A-pay =====
| 1. 회원가입
| 2. 로그인
| 3. 회원확인
| 0. 프로그램종료
```

선택 : 3

```
===== 회원정보 =====
| 이름 : 백재현
| 아이디 : a
| 비밀번호 : 1
| 핀번호 : 123456
| 계좌번호 : 228160
| 계좌잔액 : 50000
```

```
===== 회원정보 =====
| 이름 : 서정우
| 아이디 : b
| 비밀번호 : 2
| 핀번호 : 234567
| 계좌번호 : 957853
| 계좌잔액 : 0
```

```
===== 회원정보 =====
| 이름 : 서진형
| 아이디 : c
| 비밀번호 : 3
| 핀번호 : 345678
| 계좌번호 : 338406
| 계좌잔액 : 0
```

```
===== 회원정보 =====
| 이름 : 선우사랑
| 아이디 : d
| 비밀번호 : 4
| 핀번호 : 456789
| 계좌번호 : 800519
| 계좌잔액 : 0
```

# 입금 스토리보드

다른 계정으로 로그인후 입금을 실행해도 정상적으로 실행된다.

```
A-pay
1. 회원가입
2. 로그인
3. 회원확인
0. 프로그램종료
```

선택 : 2

```
로그인
아이디 입력 : b
비밀번호 입력 : 2
서정우님 환영합니다.
```

```
회원 메뉴
1. 입금
2. 출금
3. 송금
4. 쇼핑하기
0. 로그아웃
```

선택 : 1

입금할 금액 입력 : 30000  
편번호[6자리] 입력 : 234567  
30000원이 입금 되었습니다.  
잔액 : 30000

```
A-pay
1. 회원가입
2. 로그인
3. 회원확인
0. 프로그램종료
```

선택 : 3

```
회원정보
이름 : 백재현
아이디 : a
비밀번호 : 1
편번호 : 123456
계좌번호 : 228160
계좌잔액 : 50000
```

```
회원정보
이름 : 서정우
아이디 : b
비밀번호 : 2
편번호 : 234567
계좌번호 : 957853
계좌잔액 : 30000
```

```
회원정보
이름 : 서진혁
아이디 : c
비밀번호 : 3
편번호 : 345678
계좌번호 : 338406
계좌잔액 : 0
```

```
회원정보
이름 : 선우사랑
아이디 : d
비밀번호 : 4
편번호 : 456789
계좌번호 : 800519
계좌잔액 : 0
```



# 출금 스토리보드

회원정보 확인 후 로그인

```
===== 회원정보 =====  
| 이름 : 홍길동  
| 아이디 : hong  
| 비밀번호 : hong  
| 핀번호 : 111111  
| 계좌번호 : 929038  
| 계좌잔액 : 0  
=====
```

```
===== 회원정보 =====  
| 이름 : 이순신  
| 아이디 : lee  
| 비밀번호 : lee  
| 핀번호 : 222222  
| 계좌번호 : 409283  
| 계좌잔액 : 0  
=====
```



```
===== A-pay =====  
| 1. 회원가입  
| 2. 로그인  
| 3. 회원확인  
| 0. 프로그램종료  
=====
```

선택 : 2

```
===== 로그인 =====  
| 아이디 입력 : hong  
| 비밀번호 입력 : hong  
| 홍길동님 환영합니다.  
=====
```

# 출금 스토리보드

로그인 한 유저의 계좌에 잔고가 없을경우 출금 불가  
입금 후 출금 진행

회원 메뉴

- 1. 입금
- 2. 출금
- 3. 송금
- 4. 쇼핑하기
- 0. 로그아웃

선택 : 2

출금할 금액 입력 : 10000

핀번호[6자리] 입력 : 111111

잔액이 부족합니다. 잔고를 확인해주세요



회원 메뉴

- 1. 입금
- 2. 출금
- 3. 송금
- 4. 쇼핑하기
- 0. 로그아웃

선택 : 1

입금할 금액 입력 : 1000000

핀번호[6자리] 입력 : 111111

1000000원이 입금 되었습니다.

잔액 : 1000000



회원 메뉴

- 1. 입금
- 2. 출금
- 3. 송금
- 4. 쇼핑하기
- 0. 로그아웃

선택 : 2

출금할 금액 입력 : 10000

핀번호[6자리] 입력 : 111111

출금액을 확인해 주세요. 출금 하시겠습니까?

<1. 확인> <2. 취소>

선택 : 1

출금이 완료 되었습니다.

잔액 : 990000

# 출금 스토리보드

출금액이 잔고보다 작고 핀번호 오류가 없을 경우 출금 완료  
출금액이 잔고보다 크면 출금서비스 종료

회원 메뉴

1. 입금  
2. 출금  
3. 송금  
4. 쇼핑하기  
0. 로그아웃

선택 : 2

출금할 금액 입력 : 10000

핀번호[6자리] 입력 : 111111

출금액을 확인해 주세요. 출금 하시겠습니까?

<1. 확인> <2. 취소>

선택: 1

출금이 완료 되었습니다.

잔액 : 990000

회원 메뉴

1. 입금  
2. 출금  
3. 송금  
4. 쇼핑하기  
0. 로그아웃

선택 : 2

출금할 금액 입력 : 1000000

핀번호[6자리] 입력 : 111111

잔액이 부족합니다. 잔고를 확인해주세요

# 출금 스토리보드

출금시 핀번호가 3회 오류일 경우 출금페이지 자동 종료  
출금 취소시에도 출금페이지 종료

회원 메뉴

1. 입금  
2. 출금  
3. 송금  
4. 쇼핑하기  
0. 로그아웃

선택 : 2

출금할 금액 입력 : 10000

핀번호[6자리] 입력 : 123123

1회 핀번호 오류! 3회 실패시 종료됩니다.

핀번호[6자리] 입력 : 121212

2회 핀번호 오류! 3회 실패시 종료됩니다.

핀번호[6자리] 입력 : 222222

3회 핀번호 오류! 3회 실패시 종료됩니다.

회원 메뉴

1. 입금  
2. 출금  
3. 송금  
4. 쇼핑하기  
0. 로그아웃

선택 : 2

출금할 금액 입력 : 10000

핀번호[6자리] 입력 : 111111

출금액을 확인해 주세요. 출금 하시겠습니까?

<1. 확인> <2. 취소>

선택 : 2

이전단계로 돌아갑니다.



회원 메뉴

1. 입금  
2. 출금  
3. 송금  
4. 쇼핑하기  
0. 로그아웃

# 송금 스토리보드

존재하지 않는 계좌, 즉 회원가입하지 않은 회원에게 송금할 경우 송금시스템 종료  
또는 존재하는 계좌이나 유저의 핀번호가 3회 오류시 송금시스템 종료

회원 메뉴

- 1. 입금
- 2. 출금
- 3. 송금
- 4. 쇼핑하기
- 0. 로그아웃

선택 : 3

받으시는 분 계좌번호 입력 : 100001

없는 계좌번호입니다. 계좌번호를 다시 확인해주세요.

회원 메뉴

- 1. 입금
- 2. 출금
- 3. 송금
- 4. 쇼핑하기
- 0. 로그아웃

선택 : 3

받으시는 분 계좌번호 입력 : 409283

받으시는 분 성함 : 이순신

송금할 금액 입력 : 10000

핀번호[6자리] 입력 : 123123

1회 핀번호 오류! 3회 실패시 종료됩니다.

핀번호[6자리] 입력 : 222222

2회 핀번호 오류! 3회 실패시 종료됩니다.

핀번호[6자리] 입력 : 321321

3회 핀번호 오류! 3회 실패시 종료됩니다.



회원 메뉴

- 1. 입금
- 2. 출금
- 3. 송금
- 4. 쇼핑하기
- 0. 로그아웃

# 송금 스토리보드

송금받는 사람의 계좌가 존재하고 송금하는 사람의 핀번호가 일치하면 송금 완료

회원 메뉴

- 1. 입금
- 2. 출금
- 3. 송금
- 4. 쇼핑하기
- 0. 로그아웃

선택 : 3

받으시는 분 계좌번호 입력 : 409283

받으시는 분 성함: 이순신

송금할 금액 입력 : 10000

핀번호[6자리] 입력 : 111111

이순신님 계좌로 10000원이 송금되었습니다.  
잔액 : 980000



회원정보

이름 : 홍길동  
아이디 : hong  
비밀번호 : hong  
핀번호 : 111111  
계좌번호 : 929038  
계좌잔액 : 980000

회원정보

이름 : 이순신  
아이디 : lee  
비밀번호 : lee  
핀번호 : 222222  
계좌번호 : 409283  
계좌잔액 : 10000

## 출금/송금 스토리보드 - 예외처리

출금과 송금시 입력란에 문자로 잘못 입력하였을 시 예외처리하여 메뉴로 돌아가게 함.

회원 메뉴

1. 입금

2. 출금

3. 송금

4. 쇼핑하기

0. 로그아웃

선택 : 2

출금할 금액 입력 : rk

숫자로 메뉴를 선택해주세요.

회원 메뉴

1. 입금

2. 출금

3. 송금

4. 쇼핑하기

0. 로그아웃

선택 : 2

출금할 금액 입력 :

회원 메뉴

1. 입금

2. 출금

3. 송금

4. 쇼핑하기

0. 로그아웃

선택 : 3

받으시는 분 계좌번호 입력 : rk

숫자로 메뉴를 선택해주세요.

회원 메뉴

1. 입금

2. 출금

3. 송금

4. 쇼핑하기

0. 로그아웃

선택 :

# 쇼핑 스토리보드

로그인 성공시 menuImp의 userMenu메서드로 이동해 회원 메뉴창으로 들어옵니다.  
회원메뉴창에서 4입력시 쇼핑페이지로 들어옵니다.  
while문을 통해 이미지와 같은 메뉴창을 반복합니다.

쇼핑페이지

- 1. 주문하기
- 2. 취소하기
- 3. 결제하기
- 4. 선물하기
- 0. 쇼핑종료

입력:





## 쇼핑 스토리보드

1번 주문하기를 선택할 경우 menuImpl에 연결된 ShoppingImpl클래스로가서 상품을 입력하게 됩니다. while문을 통해 이미지와 같은 상품창을 반복합니다.

- (1) 주문 가능한 메뉴를 출력합니다.
- (2) 주문 받을 상품을 입력받습니다.
- (3) 주문한 상품의 가격을 총 금액에 누적합니다.
- (4) 주문한 상품을 전체 상품리스트에 추가합니다.
- (5) 주문한 개수를 한 개 증가시킵니다.

입력: 1

1. 디퓨저 : 8000원
2. 향수 : 30000원
3. 핸드크림 : 15000원
4. 립밤 : 12000원
5. 종료

상품 번호 선택 : 2

1. 디퓨저 : 8000원
2. 향수 : 30000원
3. 핸드크림 : 15000원
4. 립밤 : 12000원
5. 종료

상품 번호 선택 : 3

1. 디퓨저 : 8000원
2. 향수 : 30000원
3. 핸드크림 : 15000원
4. 립밤 : 12000원
5. 종료

상품 번호 선택 : 4

1. 디퓨저 : 8000원
2. 향수 : 30000원
3. 핸드크림 : 15000원
4. 립밤 : 12000원
5. 종료

상품 번호 선택 : 5

쇼핑페이지

1. 주문하기

2. 취소하기

3. 결제하기

4. 선물하기

0. 쇼핑종료

입력: 2

## 쇼핑 스토리보드

### 2. 취소하기

쇼핑페이지에서 2번을 선택할 경우  
while문을 통해 이미지와 같은 창을 반복합니다.

- (1) 주문한 상품리스트를 출력합니다.
- (2) 그 중에서 취소한 상품을 입력받습니다.
- (3) 입력받은 상품을 상품리스트에서 제거합니다.
- (4) 입력받은 상품의 금액을 계좌충전금액에서 차감합니다.
- (5) 주문한 개수를 한 개 감소시킵니다.

### =====주문한 상품 리스트=====

1. 디퓨저
  2. 향수
  3. 핸드크림
  4. 립밤
  0. 이전화면으로 이동
- 취소할 상품번호:  
1

### =====주문한 상품 리스트=====

1. 향수
  2. 핸드크림
  3. 립밤
  0. 이전화면으로 이동
- 취소할 상품번호:

# 쇼핑 스토리보드

## 3.결제하기

- (1) 결제해야 할 총 금액을 출력합니다.
- (2) 결제한 금액과 총 금액을 비교 결제 취소 or 계산완료를 출력합니다.
- (3)보유금액이 없을경우 잔액 부족을 출력, 금액이 결제 금액보다 많을 경우 남은 페이머니 출력합니다.
- (4) 총 금액은 0원으로 주문리스트는 비워줍니다.
- (5) 전체 개수를 0개로 초기화합니다.

쇼핑페이지

1. 주문하기

2. 취소하기

3. 결제하기

4. 선물하기

0. 쇼핑종료

입력: 3  
1. 핸드크림  
결제할 금액:15000원  
잔액이 부족합니다.  
남은 페이머니는 0원 입니다.

쇼핑페이지

1. 주문하기

2. 취소하기

3. 결제하기

4. 선물하기

0. 쇼핑종료

입력: 3  
1. 디퓨저  
결제할 금액:8000원  
남은 페이머니는 992000원 입니다.

쇼핑페이지

1. 주문하기

2. 취소하기

3. 결제하기

4. 선물하기

0. 쇼핑종료

입력:

# 쇼핑 스토리보드

## 3. 결제하기

3번 메뉴를 선택하면 이와같은 화면이 뜬다.(10만원 입금했다는 가정) cart, 즉 productlist에 들어온 값이 아무것도 없기때문이다.

상품을 주문하기 메뉴에서 주문을 받아오면 즉, cart로 옮겨짐으로 for문을 통해 cart의 사이즈보다 작으면 상품의 이름과 값을 받아오게 한다.

결제할 총금액은 total로 설정 후 받아온 가격들의 총합이다.

여기까지 ShoppingImpl에서 반환되는 값들이다.

```
「===== 쇼핑페이지 =====」  
| 1. 주문하기  
| 2. 취소하기  
| 3. 결제하기  
| 4. 선물하기  
| 0. 쇼핑종료  
「=====」
```

입력: 3

결제할 금액:0원

남은 페이머니는 100000원 입니다.

# 쇼핑 스토리보드

## 3.결제하기

ShoppingImpl에서 cart값을 UserAtmImpl에 totalpayment 로 받아온다.  
결제 계좌인 l.get(owner).getAccount()를 받아오기 위해서다.  
totalPayment보다 적을시에는 잔액 부족 메시지를 가져오고  
아닐시에는 정상적으로 결제가 완료한다. 결제 후에는 cart를 초기화한다.

```
===== 쇼핑페이지 =====  
| 1. 주문하기  
| 2. 취소하기  
| 3. 결제하기  
| 4. 선물하기  
| 0. 쇼핑종료  
=====
```

입력: 3

1. 디퓨저  
2. 핸드크림  
결제할 금액: 23000원  
남은 페이머니는 77000원 입니다.

입력: 3

1. 핸드크림  
결제할 금액: 15000원  
잔액이 부족합니다.  
남은 페이머니는 0원 입니다.

# 쇼핑 스토리보드

## 4.선물하기

회원가입한 회원들끼리의 선물하기 기능이다.  
진혁과 정우 두 회원의 정보를 미리 입력했다.  
로그인은 진혁으로 진행하였다.

```
===== 회원가입 =====  
|이름 등록 : 정우  
|아이디 등록 : qwe  
|비밀번호 등록 : 321  
|핀번호[6자리] 등록 : 654321  
|계좌번호 생성 중..  
|계좌번호 : 127670  
=====
```

```
===== 회원가입 =====  
|이름 등록 : 진혁  
|아이디 등록 : abc  
|비밀번호 등록 : 123  
|핀번호[6자리] 등록 : 123456  
|계좌번호 생성 중..  
|계좌번호 : 993821  
=====
```

# 쇼핑 스토리보드

## 4.선물하기

지금까지 사용한 cart와는 다른 장바구니가 필요함으로써 선물하기를 위한 giftbox라는 productlist타입의 arraylist를 새로 만들었다.  
주문할 상품은 주문하기의 것과 동일함으로 shop.order 를 가져왔다.

```
|
|===== 선물하기 =====|
1. 디퓨저      : 8000원
2. 향수        : 30000원
3. 핸드크림    : 15000원
4. 립밤        : 12000원
0. 종료
상품 번호 선택 :
```

# 쇼핑 스토리보드

## 4.쇼핑하기

선물하기 위해 이름을 입력하기 위한 receiver 와 회원의 이름정보를 가져온 targetUser 비교  
과정을 통해 이름이 맞다면 결제가 진행된다.  
결제과정은 쇼핑(결제하기)의 과정과 동일하다.  
마찬가지로 결제과 완료되면 giftbox또한 초기화한다

상품 번호 선택 : 2

1. 디퓨저 : 8000원

2. 향수 : 30000원

3. 핸드크림 : 15000원

4. 립밤 : 12000원

0. 종료

상품 번호 선택 : 4

1. 디퓨저 : 8000원

2. 향수 : 30000원

3. 핸드크림 : 15000원

4. 립밤 : 12000원

0. 종료

상품 번호 선택 : 0

받는사람 이름 입력 : 정우

1. 향수

2. 립밤

결제할 금액:42000원

남은 페이머니는 35000원 입니다.

정우님에게 선물을 보냈습니다.

받는사람 이름 입력 : 재현

잘못된 입력입니다.





”

실제 계획했던  
부분과 실패했던 것

05

백재현 : 본래 거래내역 메서드를 구현하고자 했으나...

```
import java.util.List;
public interface UserAtm {

    void deposit(List<User> l);
    void withdraw(List<User> l);
    void transfer(List<User> l);
    void transInfo(List<User> l);
    // 로그인한 유저가 실행했던 입금 출금 송금 쇼핑 선물 등 모든 거래내역을 일괄 출력하는 메서드
    void shopping(List<User> l);
}
```

이런메서드를 구현하고 싶었지만...

리스트 내의 다른 유저와도 관련된 모든  
거래내역을 저장하는 부분에서 어려움을  
느끼고 구현하지 못했음

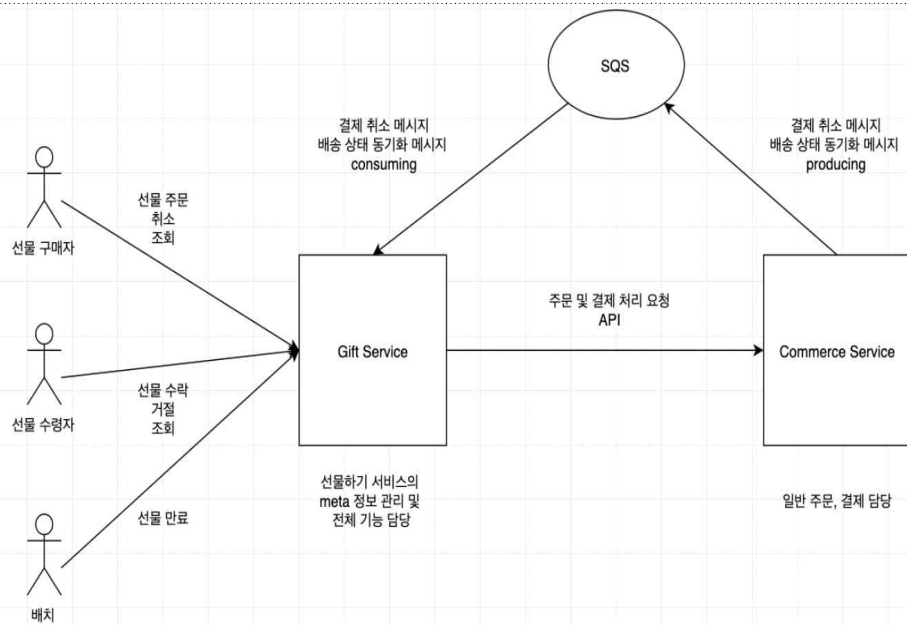


## 서정우 : 장바구니 추가 기능구현

장바구니에서의 여러기능을 구현하고  
싶었지만 실패했습니다.

장바구니 기능은 구현을 못했지만  
주문하기에서 주문수량이 쌓이고 그의  
총합과 상품명이 뜨게 하기위해 상품  
주문을 주문하기 메뉴에서 받아오면 cart  
로 옮겨지고 for문을 통해 cart의  
사이즈보다 작으면 상품의 이름과 값이  
받아지게 했습니다.

## 서진혁 : 선물하기 기능을 자바 뿐만아니라 다른 데이터베이스나 웹을 활용한다면?



더 다양한 기능을 추가할 수 있을 것 같다는 생각이 들었다.



## 선우사랑 : 계좌번호 설정과 로그인후 연동의 문제

자동 개설되는 계좌번호를 12자리로 지정하고 싶었으나 데이터 타입이 너무 커짐에 따라 자동 설정이 어려워서 계좌 자리를 6자리로 임의 지정하여 변경하였음.

회원가입 후 로그인 한 유저의 데이터를 바탕으로 은행 업무 및 쇼핑시스템을 연동시키고자 하였지만 로그인 한 유저의 정보들을 가져오는 데 실패하였고 로그인한 사람의 인덱스 번호를 'owner'라는 변수를 별도 설정하여 사용하였음.



”

프로젝트 소감

06

# 프로젝트 소감

일단 후련하다

그리고.....

“

백재현

팀프로젝트와 개인프로젝트의 많은 차이를 느꼈습니다. 그 중에서 좋은점은 개인으로 할 때는 깨닫지 못해서 전전긍긍하는 부분을 팀원에게 직접적으로 도움받거나 간접적인 힌트를 통해 깨달을 수 있다는 것이었습니다. 불편한 점으로는 변수나 각자 만든것을 통합하는 과정에서 생기는 오류가 아닌가 싶습니다. 팀프로젝트는 앞으로의 업무에 필수적이기에 다음부터는 더욱 소통의 적극성을 발휘해야한다고 느꼈습니다. 한편으로 리스트를 비롯한 컬렉션 쪽에 대한 이해도가 이번 팀프로젝트를 하면서 부족하다고 느꼈고 보강의 필요성을 인지했습니다.

“

서정우

팀프로젝트를 하면서 많이 배운것 같습니다. 혼자서 모르고 어려워 했던것을 개인 프로젝트가 아닌 팀 프로젝트이기에 물어보기도 수월하였고 서로의 부족한 점을 채워주었기 때문에 개인 프로젝트보다 훨씬 수월하였습니다. 팀 프로젝트를 진행하면서 협업의 중요성을 알게 되었고 커뮤니케이션의 중요성 또한 알게 되었습니다.다음에는 제가 더 노력하여 팀원들에게 도움을 줄 수 있는 프로젝트를 하고싶습니다.

“

서진혁

말로 표현한 것을 구현해내가는 것은 항상 어렵다는 것을 다시 느끼며 각자 맡은 부분에서 최선을 다해준 팀원에게 우선 고맙다 좋은 팀원이 아니었다면 프로젝트를 끝내지 못했을 것이란 생각을 했으며 내가 놓친부분이나 생각해내지 못한 것들을 피드백을 통해 고쳐나갈 수 있다는 장점이 있다고 생각했으며 다음 프로젝트는 좀 더 발전한 나로 팀원의 사기를 올려주고싶다.

“

선우사랑

온라인과 오프라인을 병행하여 수업이 진행되다 보니 팀원 간에 소통의 어려움이 있어 같은 기능을 각자 중복 구현한다든지 구현한 기능들을 서로 이해하고 피드백하는 부분이 다소 아쉬웠습니다. 또한, 다양한 이름으로 정의된 변수들을 논의하고 합치는 과정에서 난항도 있었습니다. 그러나 팀원 모두 적극적인 자세로 참여하여 머리를 모으고 서로의 아이디어를 공유함으로써 프로젝트를 잘 마무리할 수 있었으며 서로 짠 코드를 리뷰하면서 서로의 부족함을 채우고 배우면서 자바에 대한 이해도와 개발 활용 능력을 향상시킬 수 있었던 것 같습니다.

# 지금까지 상큼한 피피티

”

프로젝트에 대한 질의응답은 언제든지 환영합니다:)

피피티 끝!