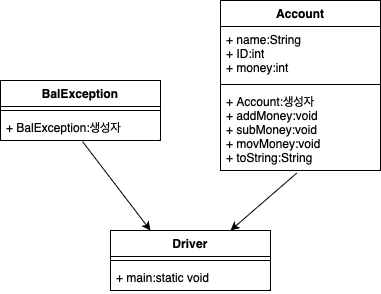
1. 클래스들
   1. BalException
   2. Account
   3. Driver



1. 각 클래스 구현
   1. Bank
      1. 변수
      2. 메소드
         1. BalException
   2. Account
      1. 변수
         1. name
         2. ID
         3. money
      2. 메소드
         1. Account
            1. 매개변수

n\_ID

n\_name

n\_money

* + - * 1. 알고리즘

변수들을 전부 새로 선언하면서 생성

* + - 1. addMoney
         1. 매개변수

p\_money

* + - * 1. 알고리즘

money+=p\_money

* + - 1. subMoney
         1. 매개변수

m\_money

* + - * 1. 알고리즘

subMoney

try {

if(m\_money>money) {

throw new BalException();

}

else {

money-=m\_money;

}

}

catch(BalException e) {

System.***out***.println("예외 처리함:출금시 잔고 부족");

}

end

* + - 1. movMoney
         1. 매개변수

to

mon

* + - * 1. 알고리즘

movMoney

try {

if(mon>money) {

throw new BalException();

}

else {

to.addMoney(mon);

this.subMoney(mon);

}

}

catch(BalException e){

System.***out***.println("예외 처리함:이체시 잔고 부족");

}

end

* + - 1. toString
         1. 알고리즘

return ID+” “+name+” “+money;

* 1. Driver
     1. 변수
     2. 메소드
        1. main
           1. 변수

a

b

* + - * 1. 알고리즘

main

Account a = new Account(101,"선남",100);

Account b = new Account(201,"선녀",200);

System.***out***.println("번호 이름 잔고")

System.out.println(a.toString())

System.out.println(b.toString())

System.***out***.println("101계좌로부터 50원을 출금한다")

a.subMoney(50)

System.out.println(a.toString())

System.***out***.println("201계좌에서 101 계좌로 300원을 계좌이체한다")

b.movMoney(300)

System.***out***.println("201계좌에서 101 계좌로 100원을 계좌이체한다")

b.movMoney(100)

System.out.println(a.toString())

System.out.println(b.toString())

end