1. 기초적인 제네릭 코드를 작성해보자 (초 기초)
   1. Sort 클래스를 생성해라 (객체 생성시 타입에 제한이 없게 해줘라(제네릭 사용하라는 말))
      1. 클래스 내부에서 만 사용할 수 있는 배열 변수를 하나 선언해라(물론 제네릭 타입 위에서 넘겨 받은 제네릭 타입 그대로 사용)
      2. 반환형이 없는 sort 메소드를 생성해라
         1. Sort 메소드의 매개변수는 정해지지 않은 타입의 배열 변수를 받는다.(즉슨 객체 생성시 받은 자료형 배열을 받으라는 말)
         2. 필드에 선언된 배열변수에 매개변수로 받은 배열을 대입 해라
      3. 반환형이 없는 print 메소드를 생성해라
         1. For each로 필드에 있는 배열을 출력해라(무조건 for each로 할 것)
   2. 위에서 만든 클래스를 테스트 하기위한 테스트 메인 클래스를 작성해라
      1. 문자열 배열을 선언하고 아무 값이나 초기화 해줘라(값 3개이상)
      2. 정수형 배열을 선언하고 아무 값이나 초기화 해줘라(값 3개 이상)
      3. 실수형 배열을 선언하고 아무 값이나 초기화 해줘라(값 3개 이상)
      4. 문자열 타입으로 클래스를 생성해라
      5. Sort 메소드를 호출하여 문자열 배열 값을 넘겨줘라
      6. Print 메소드를 호출하여 출력을 확인 해봐라

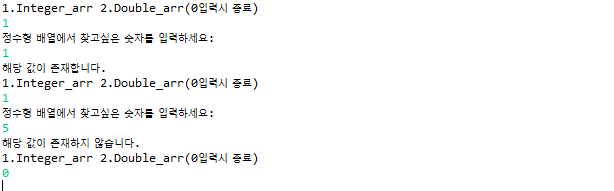
(iv , v ,vi 번의 과정을 정수형, 실수형 클래스도 생성해서 확인 할 것)



1. 기초적인 제네릭 메소드를 작성 해보자 (기초)
   1. Method 클래스를 작성해라
      1. 넘겨받는 배열의 타입에 따라 반환형이 결정되는 getLast메소드를 생성해라(즉슨 제네릭 메소드 사용하라는 말)
         1. 매개변수는 타입이 정해지지 않은 배열이다.(즉슨 제네릭 사용하라는 말)
         2. 넘겨받은 배열의 마지막 원소에 있는 값을 반환해라
      2. 넘겨받는 배열의 타입에 따라 반환형이 결정되는 getFIrst메소드를 생성해라(즉슨 제네릭 메소드 사용하라는 말)
         1. 매개변수는 타입이 정해지지 않은 배열이다.(즉슨 제네릭 사용하라는 말)
         2. 넘겨받은 배열의 마지막 원소에 잇는 값을 반환해라
   2. Test 클래스를 작성해라
      1. 정수형 랩퍼클래스 배열을 생성 및 초기화 해라
      2. 실수형 랩퍼클래스 배열을 생성 및 초기화 해라
      3. Method 객체를 생성해라.
      4. 정수형 배열, 실수형 배열의 첫번째 마지막 원소를 메소드를 이용하여 출력해라



1. 리스트 생성시 타입에 제한을 둬 클래스를 작성해봐라 (와일드 카드 이용) (기초)
   1. 숫자 자료형만 리스트로 만들 수 있게 제한해라
      1. 필드에는 타입이 제한 없는 리스트를 선언해라
      2. 반복문에 사용할 정수형 변수를 0으로 초기화 해라
   2. 생성자
      1. 숫자 자료형만 받을 수 있게 매개변수에 Number클래스를 상속받은 리스트를 받아라
      2. 필드에 선언한 list에 대입해라
   3. 반환형이 없는 find 메소드를 작성해라
      1. 매개변수로는 타입이 정해지지 않은 변수를 넘겨줘라(제너릭 타입으로 형식 매개변수를 만들어라)
         1. Public <T> void find(T d)
      2. 반복문을 이용하여 매개변수로 넘겨진 값이 존재하면 값이 있다고 출력해주고 없으면 없다고 출력해줘



1. ArrayList를 이용해 최대값을 구하는 프로그램을 작성해봐라 (기초)
   1. 메인클래스를 생성해라
      1. 정수형 타입의 ArrayList를 생성해라
      2. 무한 반복문을 시작해라
         1. 정수를 입력 받아라
         2. 0을 입력하면 프로그램은 종료된다.
         3. 0이 아닌 다른 숫자는 계속 List에 추가된다.
      3. 무한반복문이 종료 됬다면
         1. 여태까지 리스트에 넣은 값들 중에서 최대값을 구해라
      4. 최대값을 출력해라



문제5. 제너릭 을 이용한 ArrayList(중급)

Tformal class (제너릭)

필드 :

One (타입 :T)

메소드 :

Set,Get 메소드

Tformaltest class

Random 랩퍼클래스 선언 합니다.

타입이 Tformal 인 ArrayList 를 선언합니다.

Ex ) 타입 : int

ArrayList <Integer> =new ArrayList<Integer>();

ArrayList에 얼만큼 넣을껀지 입력 받습니다.

입력 받은 값까지 반복문을 돌립니다.

반복문 안의 내용

List에 새로운 객체를 선언하여 넣어줍니다.

다음과 같이 작성 합니다.

System.***out***.println("1.실수 랜덤 2.정수 랜덤 3. 영어 대문자 랜덤 4.참,거짓 랜덤 (단,정수와 실수는 1~100사이 값을 가지도록 한다.)");

Swirch , if ~else if 을 사용하여 입력받은 값이 1인경우,

2인경우 1~100사이 값의 실수,정수 랜덤값을 방금 List에 넣은 객체의 필드값 one에 넣어줍니다.

3인경우 아스키코드를 사용해 대문자 영어만 랜덤으로 대입해줍니다.

4인경우 참,거짓을 랜덤으로 넣어줍니다.

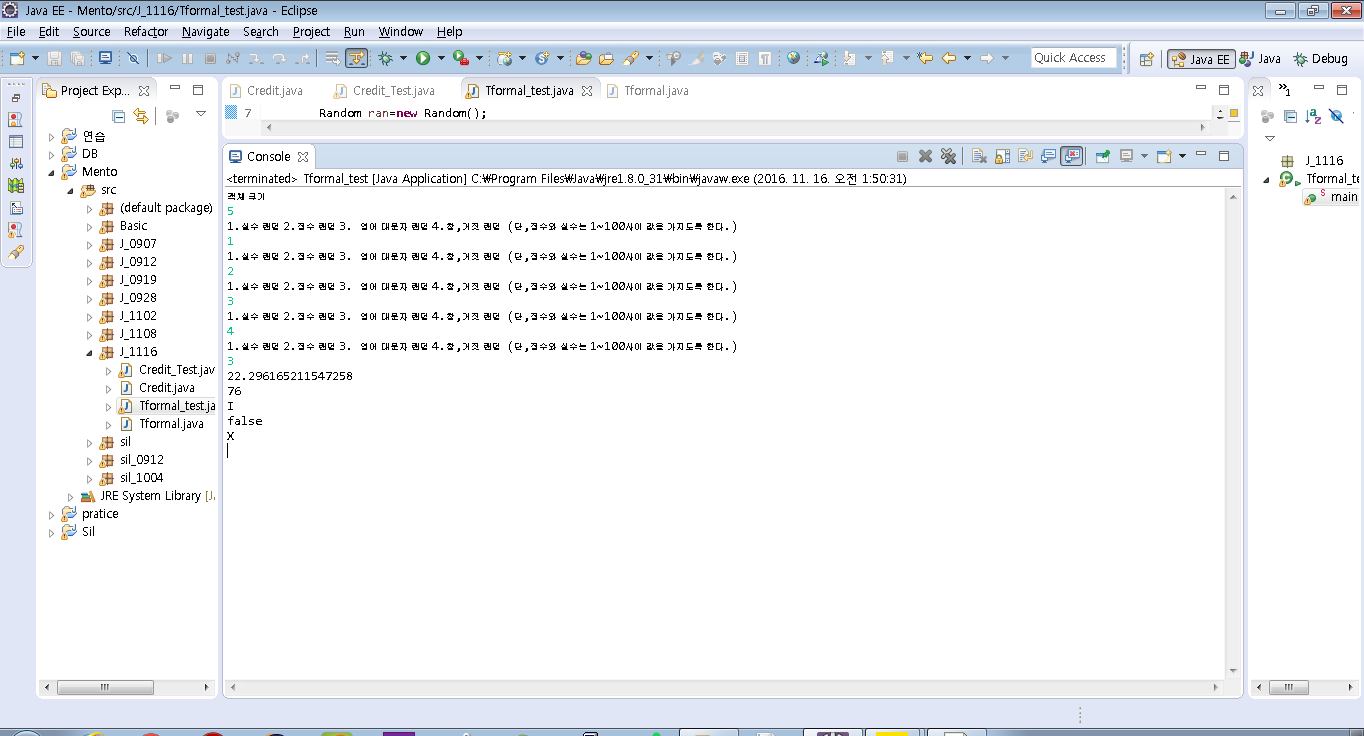
4인경우 소는 생소하니 보여줄께요!. 보고 위 1,2,3번에 응용하시면 됩니다.

list.get(i).setOne(ran.nextBoolean());

입력받은 크기까지 반복문이 돌고 나면 list에있는 모든 원소들을 출력 하는 반복을을 만들어주세요!

For,for~each 둘 중 하나 사용 하시면 됩니당

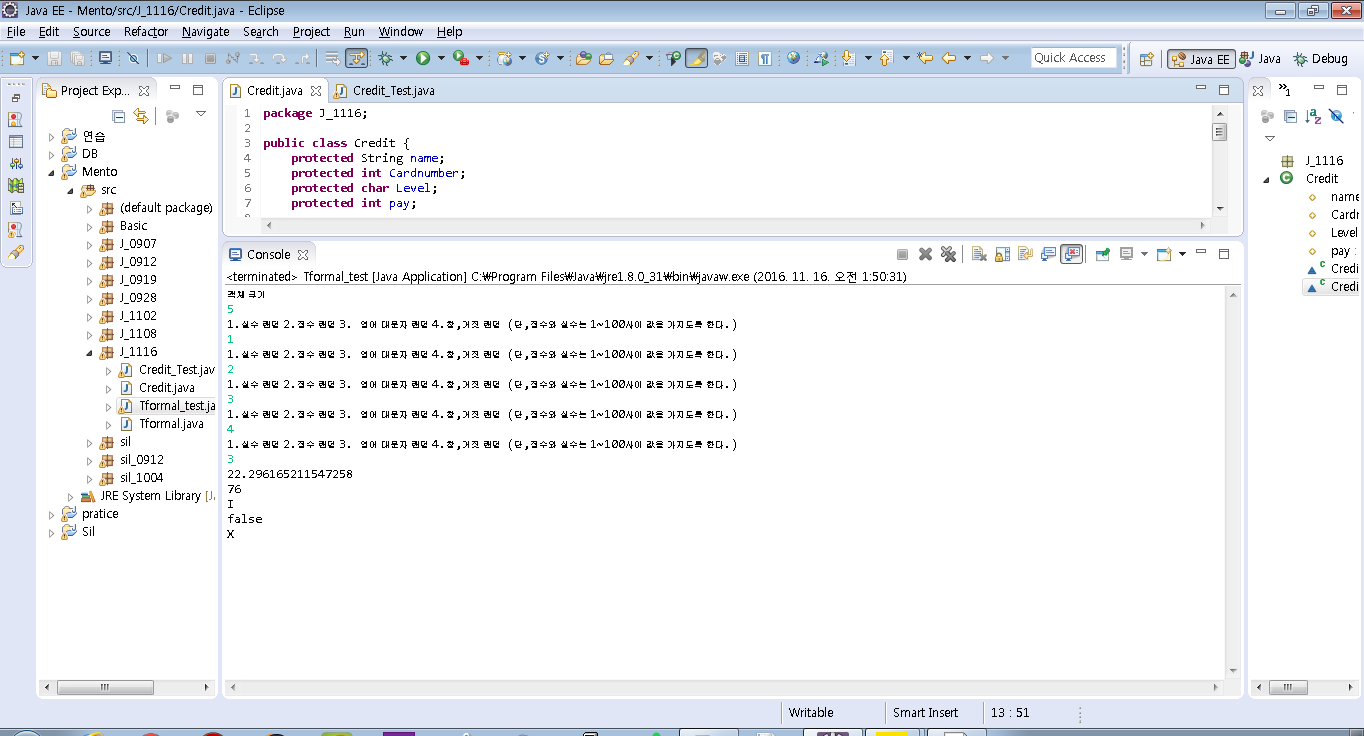
실행결과



문제6. 제너릭 응용 (고급)

생각을 잘 하셔야 됩니다. 신미영교수님 과제와 각 교수님들 과제, 멘토링 과제 를 잘 찾아보시면 해결 할 수 있을 것 입니다.

Credit class



생성자 :

Credit() : 아무것도 생성하지 않았기에 매개변수를 받고 초기화하는 생성자에게 null,0,0,을 보내줍니다.(this)

Credit(String name,int Cardnumber,int pay) :

각각 필드 값에 넣어줍니다. (this. 꼭 사용)

Level은 pay 가 10000원 이상인경우 A

5000원 이상인경우 B

1000원 이상인경우 C

그이하 D로 넣어줍니다.

CreditTest

CreditTest 위에 Void info(Credit t) 의 메소드 정의를 한

Credit\_t 인터페이스를 만드세요.

Main 에서

Credit 타입을 가진 ArrayList list 를 선언하세요

list에는 다음과 같이 넣어주세요.

list.add(**new** Credit());

list.add(**new** Credit("Joosang", 123, 5002));

list.add(**new** Credit("Inkyu", 324, 1003));

list.add(**new** Credit("Goeun", 342, 200));

list.add(**new** Credit("Seyoung", 552, 10009));

list.add(**new** Credit("SeokJin", 223, 4038));

list.add(**new** Credit("Jisu", 142, 3321));

list.add(**new** Credit("A-young", 814, 12000));

list.add(**new** Credit("Dajung", 772, 6021));

list.add(**new** Credit("Jihye", 552, 2221));

list.add(**new** Credit("Subin", 901, 9560));

list.add(**new** Credit("Eunji", 111, 820));

Credit\_t 인터페이스를 람다식으로 구현해주세요(매개변수=Credit 객체변수)

람다식 안의 내용 :

System.***out***.printf("회원 이름 : %-8s 카드 번호 : %3d 회원 등급 : %c 현재 잔액 : %05d 원\n", t.name, t.Cardnumber, t.Level,t.pay);

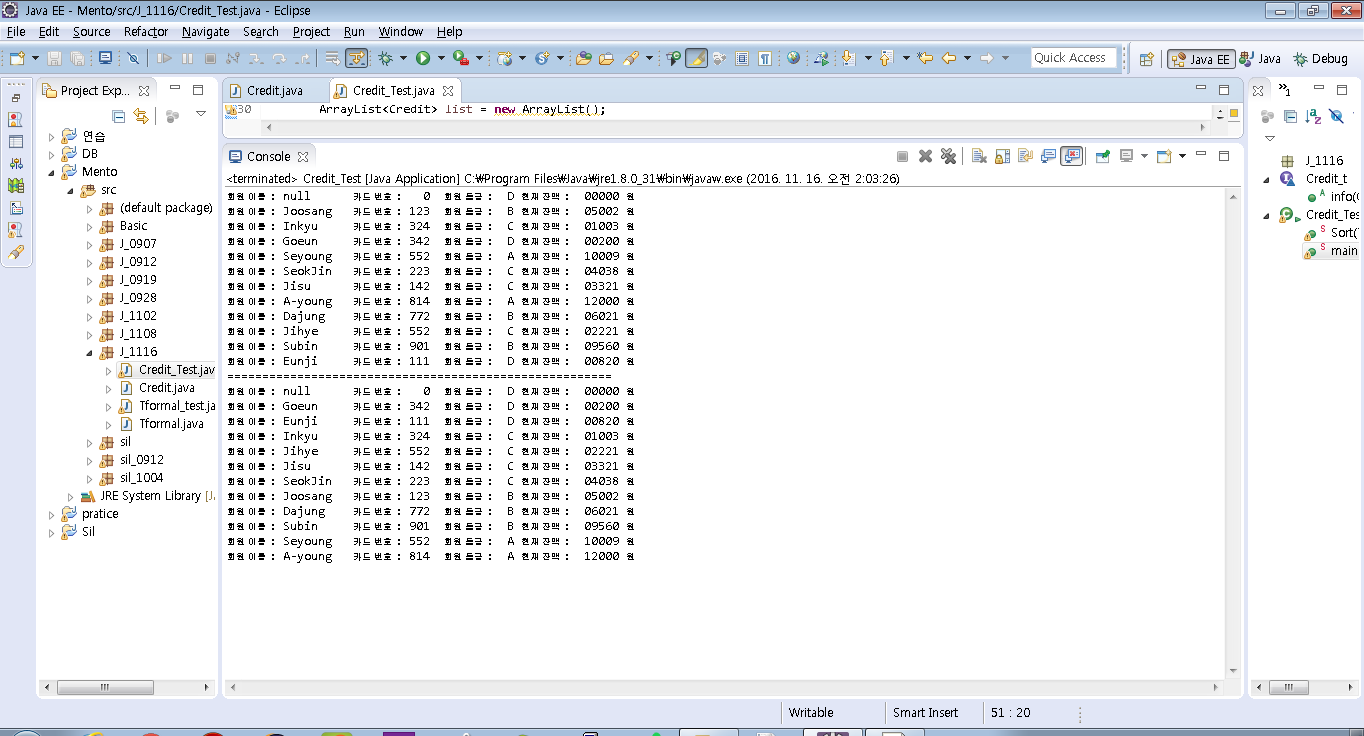
이제 list에 있는 원소들을 출력해보아요

출력문은 람다식 에 있습니다.

list 크기만큼 반복문을 돌려 정보를 출력해주세요

(람다식 사용법은 (Credit\_T 객체변수).info(내용) 으로 사용하여 출력 하면 됩니다.)

자 출력이 되었나요?



이렇게 나와야 성공 입니다!

이제 대망의 하이라이튼데요.

이 리스트를 각 객체안의 현재 잔액 이 작은 순서대로 정렬을 할겁니다.

정렬할 메소드는 이 CreditTest안에 만들 겁니다. 그리고 제너릭을 이용한 제너릭 메소드를 만들 겁니다.

제너릭 클래스와 제너릭 메소드 다들 배우셨죠?ㅎㅎ

제너릭 메소드 Sort 를 만드세요 . Sort메소드의 형식 매개변수는

T list 입니다. 다음과 같이 만들어 주세요.

**public** **static** <T> **void** Sort(T list)

T 는 최상위 계층이기 때문에

List 가 어떤 타입인지를 모릅니다.

우리는 Main에서 ArrayList 타입인 list를 Sort메소드로 호출 시킬겁니다.

list 는 배열이 아니기 때문에 list.length를 사용 할 수 없는 건아시죠? 그래서 다른 ArrayList안의 메소드를 사용해줘야되는데 그 메소드를 사용하려면 T list 에서 T가 ArrayList 인지 알아야 합니다.

예전에 배웠을 겁니다. 이 객체의 클래스가 이거냐 라는 식? ^\_^ instanceof 를 사용합니다.

즉, 이 list의 클래스가 ArrayList이냐 라고 조건문에 넣어주시면 됩니다.

충족이 된다면 if 문 안에서는 ArrayList 메소드를 사용 할 수있습니다.

하지만 번거롭기 때문에 T를 하향식 변환을 해줘야 합니다.

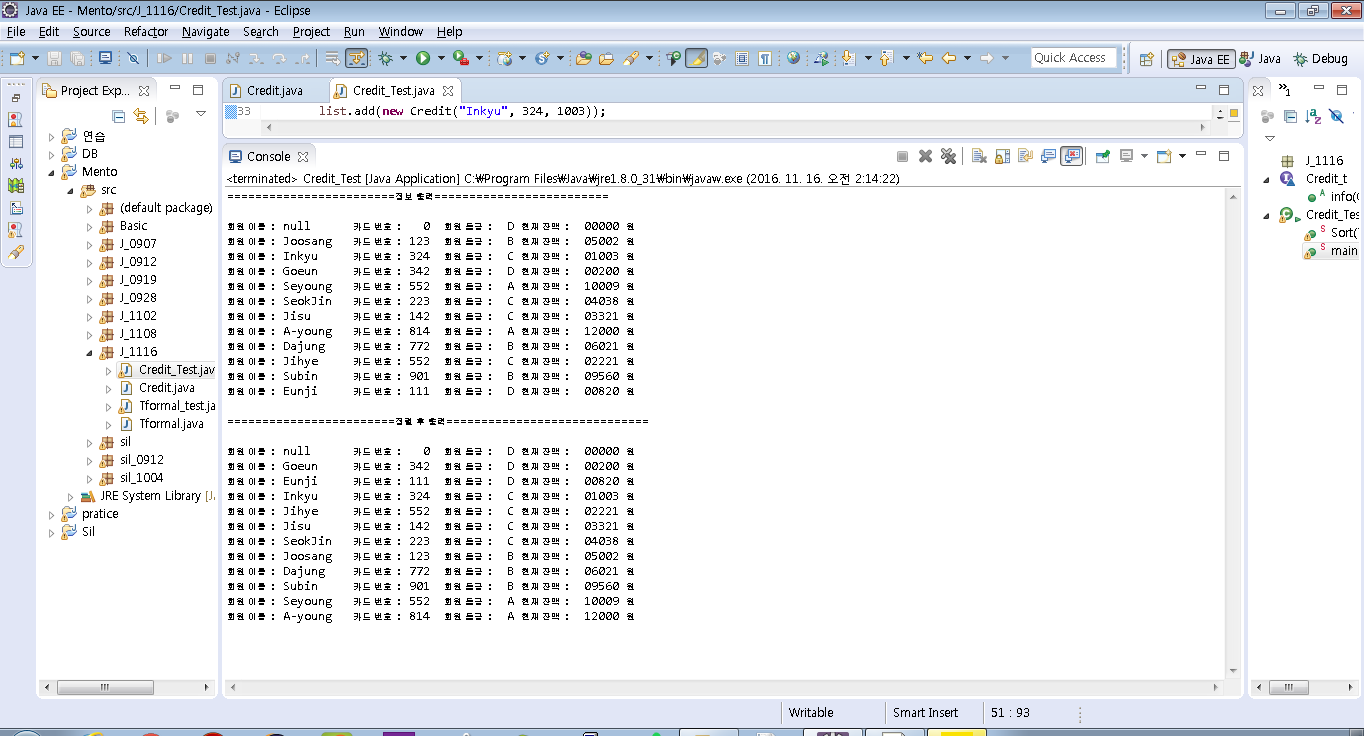
ArrayList<Credit> List\_1 = (ArrayList<Credit>) list;

위와 같이 사용 합니다.

그 다음 은 List\_1 에 있는 모든 원소(객체)들의 잔액 가지고 정렬 하시면 됩니다.

정렬은 매번 해왔으니 잘 하시겠죠?^\_^

실행 결과



그럼 파이팅 하시고 모르는건 항상 질문해서 이 문제를 자기것으로 만드세요~!