1. 패스트푸드점에서 메뉴를 선택하고 가격을 계산하는 클래스이다 주석에 해당하는 문장을 완성하시오

public class Menu {

private String menu; //메뉴명

private int danga; //단가

private int cnt; //갯수

//가격 필드를 추가하시오. 전용멤버로 하고 자료형은 int

//생성자를 추가하시오. 단, 필드초기화에 필요한 값을 매개변수로 받고 가격은 단가\*개수를 계산하여 초기화하시오

//가격 필드 값을 반환하는 접근자 메소드

//객체 내용(메뉴, 단가, 개수, 가격) 을 문자열로 반환하는 toString()

}

import java.util.\*;

public class MenuTest {

public static void main(String[] args) {

Scanner in=new Scanner(System.*in*);

Menu obj; //Menu 객체 선언

//메뉴 항목에 해당하는 문자열을 배열로 초기화

String[] menu={"햄버거", "샌드위치", "치즈스틱", "포테이토"};

//메뉴 별 단가를 정수형 배열로 초기화

int[] danga={2500, 1000, 1500, 2000};

int choice; //선택한 메뉴 번호

int total=0; //전체 구입 가격

while(true){

System.*out*.println("\n1. 햄버거\t2. 샌드위치\t3. 치즈스틱\t4. 포테이토\t5. 종료");

System.*out*.print("해당 메뉴를 선택하세요 : ");

choice = in.nextInt();

if(choice == 5)

break;

System.*out*.print("개수를 선택하세요 : ");

//개수를 입력받는다 – 본인작성

// Menu 객체 생성, 메뉴와 가격, 개수를 생성자 매개변수로 전달한다 – 본인작성

//가격을 계산하여 변수 total에 누적, 가격필드에 대한 접근자 메소드 사용 – 본인작성

//객체 내용 출력 – toString() 사용 –본인 작성

}

//전체 구입 가격 출력 – 본인 작성

}

}

|  |
| --- |
| 소스  **package** 멘토링0524연습;  **import** java.util.\*;  **class** Menu {  **private** String menu; //메뉴명  **private** **int** danga; //단가  **private** **int** cnt; //갯수  **private** **int** ga;//가격  **public** Menu(String menu , **int** danga , **int** cnt){  **this**.menu=menu;  **this**.danga=danga;  **this**.cnt=cnt;  **this**.ga=danga\*cnt;  }  **public** **int** getter(){  **return** ga;  }  **public** String toString(){  **return** "매뉴"+menu +" "+"단가"+danga+" "+"개수"+cnt+" "+"가격"+getter();  }  }  **public** **class** t01 {  **public** **static** **void** main(String[] args) {  Scanner in=**new** Scanner(System.***in***);  Menu obj; //Menu 객체 선언  //메뉴 항목에 해당하는 문자열을 배열로 초기  String[] menu={"햄버거", "샌드위치", "치즈스틱", "포테이토"};  //메뉴 별 단가를 정수형 배열로 초기화  **int**[] danga={2500, 1000, 1500, 2000};  **int** choice; //선택한 메뉴 번호  **int** total=0; //전체 구입 가격    **while**(**true**){  System.***out***.println("\n1. 햄버거\t2. 샌드위치\t3. 치즈스틱\t4. 포테이토\t5. 종료");  System.***out***.print("해당 메뉴를 선택하세요 : ");  choice = in.nextInt();  **if**(choice == 5)  **break**;  System.***out***.print("개수를 선택하세요 : ");  **int** z=in.nextInt();  obj = **new** Menu( menu[choice] ,danga[choice] , z );  total+=obj.getter();  System.***out***.println(obj);  }  System.***out***.print("전체 구입 가격:"+total);  }  } |
| 실행 결과 |

1. 고객 관리를 위한 클래스 Membership를 제시된 조건대로 작성하시오
2. 필드 구성 - 전용 필드로 할 것
3. 이름(String), 전화번호(String), 나이(int), 회원구분(VIP(‘v’), 일반(‘g’) char), 대출이율(int), 포인트(int)
4. 대출이율은 다음과 같이 계산할 것

VIP회원 중에서 현재 포인트가 1000점 이상이면 대출이율은 6%, 아니면 7%이고

일반회원 중에서 현재 포인트가 2000점 이상이면 대출이율은 8%, 아니면 9%이다

1. 메소드 구성
2. 매개변수로 필드를 초기화하는 생성자. 단, 대출이율은 rateProc()를 호출하여 결정
3. rateProc() 메소드 : 회원구분과 포인트를 이용하여 대출이율을 계산하여 반환
4. memberType() 메소드 : 회원구분이 ‘v’이면 “VIP 회원”, ‘g’이면 “일반회원”을 반환
5. disPlay() 메소드

고객정보를 출력 - 이름, 전화번호, 나이, 포인트, 대출이율, 회원구분(문자열로) 출력, 반환값과 매개변수 없음

public class MembershipTest{

public static void main(String args[]){

// Membership객체 두 개 생성, 생성자 매개변수로 필드초기화에 필요한 값 전달

//display() 메소드를 호출하여 객체 내용 출력

}

}

1. main() 메소드 : Membership 클래스 객체를 생성하고 테스트, 단 객체는 두개를 생성하여 테스트 할 것
2. 도서 대여점 관리를 위한 Book 클래스를 작성하라.
3. 고객번호(id :String, private), 도서종류(book:String, private) 권수(cnt : int, private) 대여비(rental : int, private) 대여금액(money:int, private) 필드로 구성된다.
4. 고객번호만 초기화 하고 나머지 필드는 모두 0으로 하는 생성자
5. 고객번호, 북코드, 권수만 매개변수로 받아 도서종류와 대여비, 대여금액(권수 \* 대여비)을 계산하는 생성자
6. 객체 상태(고객번호, 도서종류, 권수, 대여비, 대여금액)를 문자열로 반환하는 toString()
7. 테스트 클래스 BookTest를 제시된 조건대로 작성하시오
8. 객체 생성시 고객번호, 북코드, 권수를 입력 받아 생성자 매개변수로 전달

|  |  |  |
| --- | --- | --- |
| 북코드 | 도서종류 | 대여비 |
| 1 | 만화 | 300 |
| 2 | 소설 | 400 |
| 3 | 잡지 | 500 |

1. 다음과 같은 요구사항을 만족하는 TelBook 클래스를 선언하시오.

[필드]

- 회원이름 (name) : private

- 전화번호 (tel) : private

- 회원 수 (NUM) : private

[생성자]

- 회원 객체가 생성될 때 마다 NUM 증가

- default 생성자

: default 생성자로 객체 생성 시 name 필드는 friend로 초기화

: default 생성자로 객체 생성 시 tel 필드는 000-0000-0000으로 초기화

- 일반 생성자

: 이름과 전화번호를 매개변수로 전달 받아 객체 생성

[기능]

- 전화번호에 해당하는 이름을 알려 주는 정적(static) 메소드

: 이름 getName(TelBook 객체배열, 전화번호); //메소드 형식

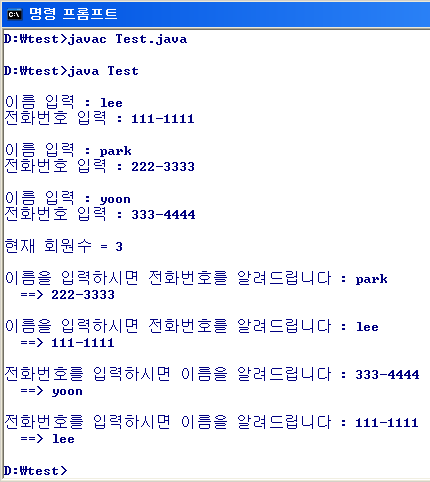
- 이름에 해당하는 전화번호를 알려 주는 정적(static) 메소드

: 전화번호 getTel(TelBook 객체배열, 이름); //메소드 형식

- 현재 회원 수를 알려 주는 정적(static) 메소드

: 회원수 getNUM() //메소드 형식

1. 위 4번에서 선언된 클래스를 이용하여 다음의 그림과 같이 수행되도록 메인 메소드의 빈칸을 채우시오.



|  |
| --- |
| 소스  **import** java.util.\*;  **class** TelBook{  **private** **static** String *name*;  **private** **static** String *tel*;  **private** **static** **int** *num*;  **public** **static** **int** *length*;  **public** TelBook(){  //num++;  **this**.*name*="friend";  **this**.*tel*="000-0000-0000";  }  **public** TelBook(String name ,String tel){  *num*++;  **this**.*name*=name;  **this**.*tel*=tel;  }  **public** **static** String teltoname(String name ,TelBook obj[]){  **int** x=0;  **for**(**int** i=0;i<obj.length;i++){  **if**(obj[i].equals(name)){  x=i;  //System.out.println("==>"+obj[x].tel);  **break**;  }  }  **return** obj[x].*tel*;  }  **public** **static** String nametotel(String tel ,TelBook obj[]){  **int** x=0;  **for**(**int** i=0;i<obj.length;i++){  **if**(obj[i].equals(tel)){  x=i;  //System.out.println("==>"+obj[x].name);  **break**;  }  }  **return** obj[x].*name*;  }  **public** **static** **int** getnum(){  **return** *num*;  }  }  **public** **class** t04 {  **public** **static** **void** main(String[] args) {  Scanner key = **new** Scanner(System.***in***);  TelBook[] obj = **new** TelBook[3];  **for**(**int** i=0; i<obj.length;i++){  System.***out***.println("이름과 전화번호를 입력하세요.");  obj[i]=**new** TelBook( key.next() , key.next() );  }  System.***out***.println("이름을 입력하시면 전화번호를 알려드립니다.");  System.***out***.println("==>"+TelBook.*teltoname*(key.next(),obj));  System.***out***.println("전화번호를 입력하시면 이름을 알려드립니다.");  System.***out***.println("==>"+TelBook.*nametotel*(key.next(),obj));  System.***out***.println(TelBook.*getnum*());  }  } |
| 실행화면 |

6. PC방 관리

-PC방 클래스,회원가입 클래스,Main 3가지를 만듭니다.

-회원가입 Class

Id,비번,이름,생년(ex : 94),사는 곳,전체 회원수

private String Account;

private String Name;

private String Live;

private String Password;

private int Born\_year;

private static int *count*=0;

-메소드는 각 필드의 get메소드

-생성자 사용합니다.(사용시마다 count증가)

-PC방 Class

필드-회원가입 객체배열,시간,관리자Id,관리자비번

private MembershipJoin MJ[];

private String Pcname = "Root";

private String Pcpassword = "1234";

private int time;

-생성자는(회원가입 객체배열,시간) 매개변수를 가집니다.

public boolean Login(String Account, String password) 메소드

로그인할 Id와 비밀번호 매개변수로 받아 확인합니다.

틀리면 오류 문구 출력 후 false 리턴 합니다.

밤10시~오전8시(22시~08시)까지는 청소년 출임 금지입니다.

98년생부터는 출입 금지를 시켜야됩니다.

98년생이 로그인 할 경우 오류문구 출력 후false 리턴합니다.

다 갖추어 졌다면 환영문구 출력후 true 리턴합니다.

public boolean Pcplay(MembershipJoin per) 메소드

미성년자인지 확인하는 메소드입니다.

로그인 할 id의 생년이 98년이상이면 false,아니면 true

를 리턴합니다.

public void Pass(String Account, String Name) 메소드

비밀번호 찾는 메소드입니다.

id 와 이름을 매개변수로 받아 확인합니다.

둘 다 일치하면 비밀번호를 알려줍니다.

하나라도 일치하지 않으면 오류문구 출력합니다.

public void PcM() 메소드

관리자 메소드입니다.

System.*out*.println("1. 모든 회원정보 2.종료");

1번 입력하면 회원정보 메소드를 호출 합니다.

2번 입력하면 메인으로 돌아갑니다.

public void Info() 메소드

관리자 메소드에서 1번을 입력하면 회원정보 메소드를 호출 합니다.

모든 회원 수 와 각 회원의 정보를 출력하면 됩니다.

public String getPcname() 메소드

public String getPcpassword() 메소드

●Main 소스●

package HS\_3;

import java.util.\*;

public class Pcmain {

public static void main(String[] args) {

MembershipJoin mj[] = new MembershipJoin[5];

mj[0] = new MembershipJoin("Happy", "희수", "춘천", "hs12", 96);

mj[1] = new MembershipJoin("Love", "고은", "월꼐", "ge12", 98);

mj[2] = new MembershipJoin("Great", "세영", "구리", "sy12", 97);

mj[3] = new MembershipJoin("Pretty", "지수", "서울", "js12", 97);

mj[4] = new MembershipJoin("Handsome", "주상", "별내", "ljs12", 93);

int time=(int)(Math.*random*()\*23+1);

PcManager pc = new PcManager(mj,time);

Scanner scan = new Scanner(System.*in*);

while (true) {

System.*out*.println("Noname 피시방에 오신걸 환영합니다. 현재 시각은 "+time+ "시 입니다.");

System.*out*.println("1.로그인");

System.*out*.println("2.비밀번호 찾기");

System.*out*.println("3.관리자 로그인");

System.*out*.println("4.종료");

int num = scan.nextInt();

if (num == 4) {

System.*out*.println("다음에 또 이용해주세요");

break;

}

switch (num) {

case 1:

System.*out*.println("ID입력");

String id = scan.next();

System.*out*.println("Password 입력");

String pw = scan.next();

if (pc.Login(id, pw) == true) {

System.*exit*(0);

}

break;

case 2:

System.*out*.println("ID입력");

id = scan.next();

System.*out*.println("이름 입력");

String name = scan.next();

pc.Pass(id, name);

break;

case 3:

System.*out*.println("관리자ID입력");

id = scan.next();

System.*out*.println("Password 입력");

pw = scan.next();

if (pc.getPcname().equals(id)) {

if (pc.getPcpassword().equals(pw))

pc.PcM();

}

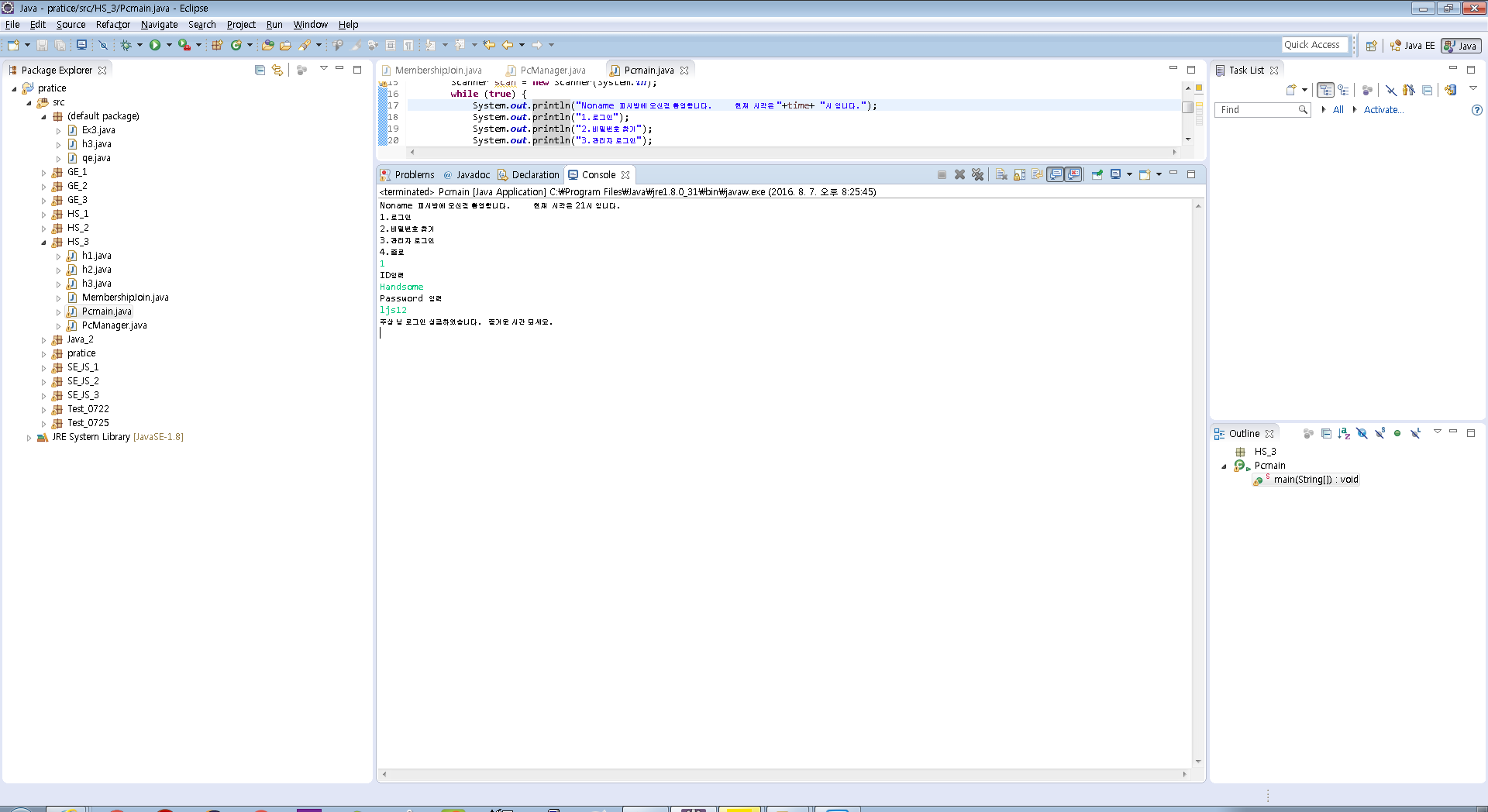
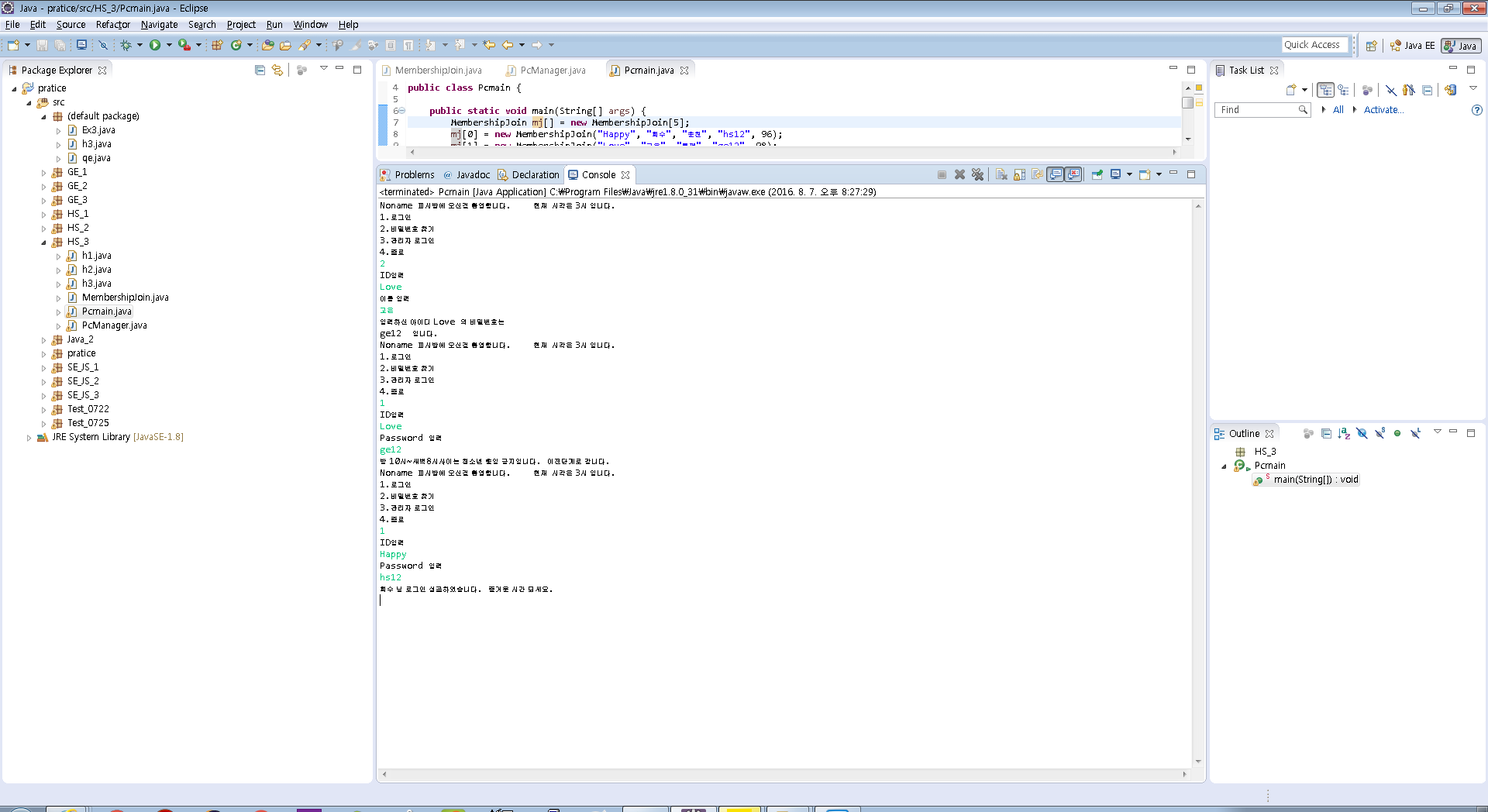
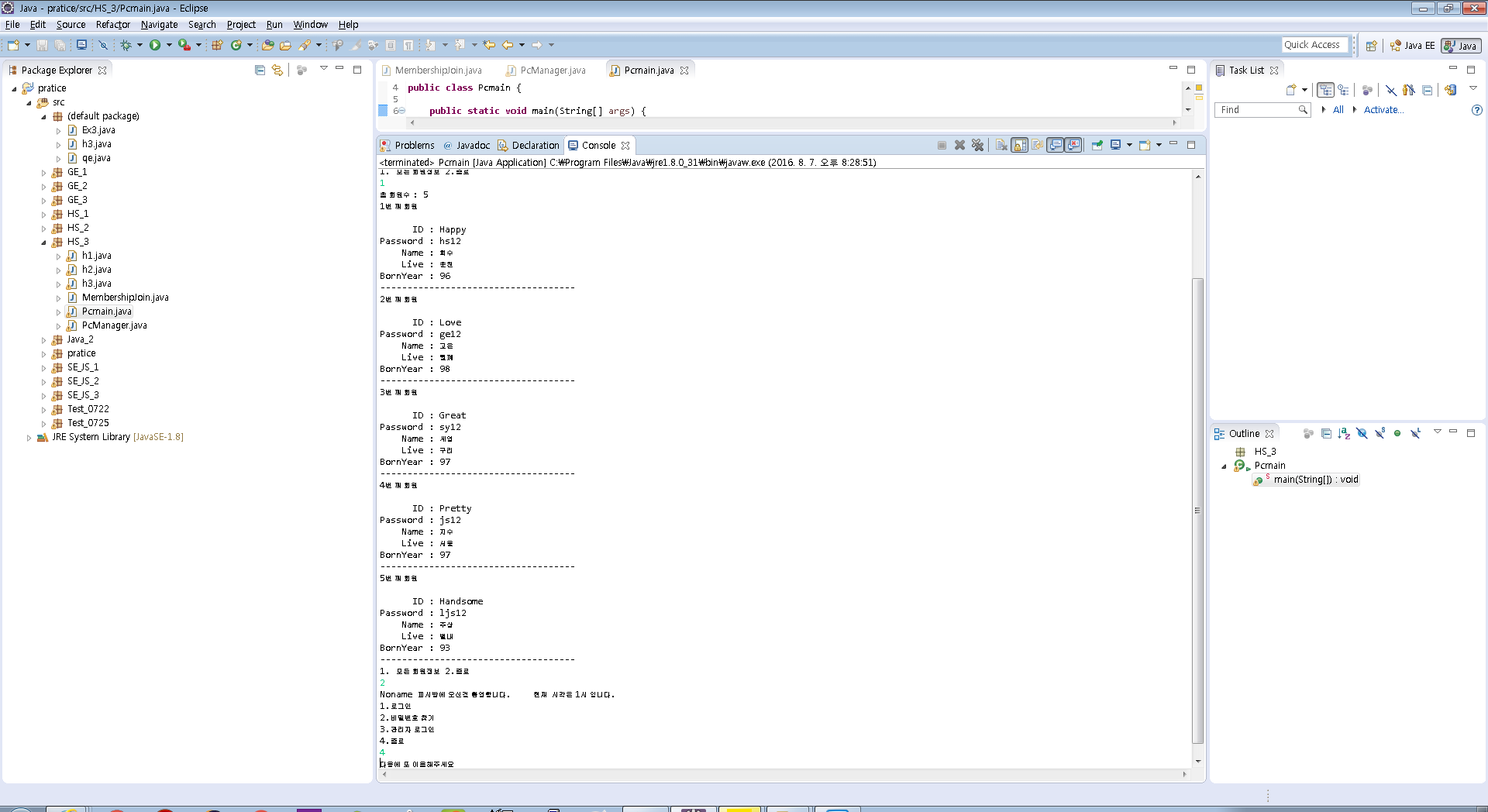
break;

}

}

}

}

●실행 결과●

◆실행결과 3그림은 다 다른 컴파일!◆