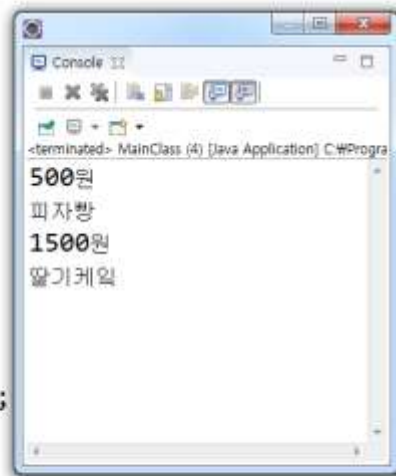

자바 Quiz (클래스~API)

문제

1. 아래의 중복으로 사용되는 빵의 정보를 Bread 클래스로 정의 하세요.

실행 화면

```
public class MainClass {  
    public static void main(String[] args) {  
        info(); //피자빵  
        info2(); //딸기케익  
    }  
  
    //피자빵  
    static int price = 500;  
    static String name = "피자빵";  
    static void info() {  
        System.out.println(price + "원");  
        System.out.println(name);  
    }  
  
    //딸기케익  
    static int price2 = 1500;  
    static String name2 = "딸기케익";  
    static void info2() {  
        System.out.println(price2 + "원");  
        System.out.println(name2);  
    }  
}
```

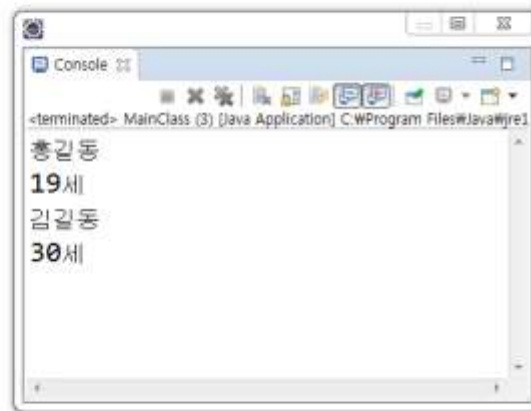


문제

1. 다음은 사람1, 사람2의 똑같은 기능을 중복으로 사용하고 있습니다.
2. 사람3 이 필요하다면 똑같은 기능을 또 만들어야 합니다.
3. 아래 코드를 재활용 할 수 있도록 Person 클래스로 정의하세요.
4. 정의 했다면, 2명의 사람을 만들고, 이름, 나이를 각각 지정한 후 info()를 출력하세요.

실행 화면

```
public static void main(String[] args) {  
  
    info();    //사람1 의 정보  
    info2();   //사람2 의 정보  
}  
  
//사람1  
static String name = "홍길동";  
static int age = 19;  
static void info() {  
    System.out.println(name);  
    System.out.println(age + "세");  
}  
  
//사람2  
static String name2 = "김길동";  
static int age2 = 30;  
static void info2() {  
    System.out.println(name2);  
    System.out.println(age2 + "세");  
}  
}
```



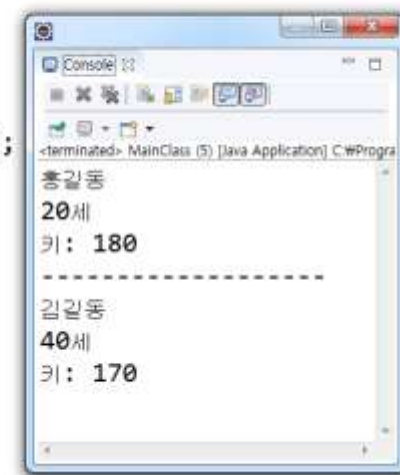
문제

1. 다음은 사람 클래스입니다.
2. 아래와 같이 실행 될 수 있도록 생성자를 만드세요.

실행 화면

```
public class Person {  
  
    String name;  
    int age;  
    int tall;  
  
    void info() {  
        System.out.println(name);  
        System.out.println(age + "세");  
        System.out.println("키: " + tall);  
    }  
}
```

```
public class MainClass {  
  
    public static void main(String[] args) {  
  
        Person p1 = new Person("홍길동", 20, 180);  
        p1.info();  
  
        System.out.println("-----");  
  
        Person p2 = new Person();  
        p2.age = 40;  
        p2.name = "김길동";  
        p2.tall = 170;  
  
        p2.info();  
  
    }  
}
```

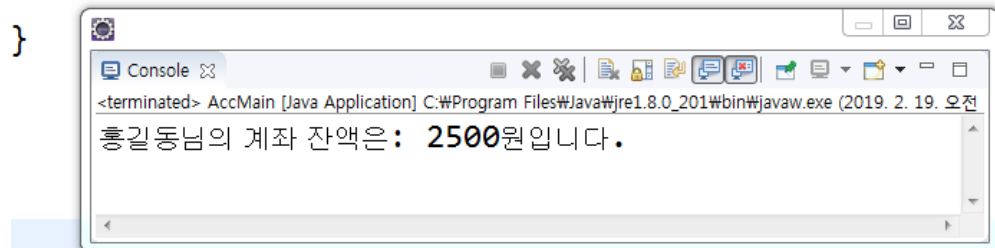


문제

1. 멤버 변수- 예금주이름(name : String), 비밀번호(password : int), 잔액(balace: int) 로 선언하세요.
2. 생성자 - 3가지 멤버 변수를 모두 매개값으로 받아 초기화하는 생성자를 선언하세요.
3. 메서드 - 입금기능 메서드(deposit : void), 출금기능 메서드(withDrwa: void), 잔액 조회 메서드(getbalance: int)
4. 아래와 같이 실행 될 수 있도록 클래스를 만드세요.

실행 화면

```
public static void main(String[] args) {  
  
    Account myAcc = new Account("홍길동", "1234", 3600);  
  
    myAcc.deposit(800);  
    myAcc.withDraw(1900);  
  
    int bal = myAcc.getBalance();  
    System.out.println(myAcc.name + "님의 계좌 잔액은: " + bal + "원입니다.");  
  
}
```



문제

1. Warrior 클래스 와 Wizard 클래스는 동일 속성과 기능을 갖고 있습니다.
2. 동일한 속성, 기능들을 Player 클래스로 정의 하고 상속 시키세요.
3. main에서 두 클래스를 만들어 info()를 사용하여 다음 결과가 나오도록 하세요.

실행 화면



```
<terminated> Main [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (2019. 2. 22 오후 9:55:02)
-----마법사-----
케릭터명 : 구르미그린달빛
체력 : 500
마나 : 1000
얼음화살 사용
-----전사-----
케릭터명 : 강한천구대한전사
체력 : 1000
마나 : 500
때리기 스킬 사용
```

```
public class Warrior {

    String name;
    int hp;
    int mp;

    void info() {
        System.out.println("케릭터명:" + name);
        System.out.println("체력:" + hp);
        System.out.println("마나:" + mp);
    }

    void bash() {
        System.out.println("때리기 스킬 사용");
    }
}
```

```
public class Wizard {

    String name;
    int hp;
    int mp;

    void info() {
        System.out.println("케릭터명:" + name);
        System.out.println("체력:" + hp);
        System.out.println("마나:" + mp);
    }

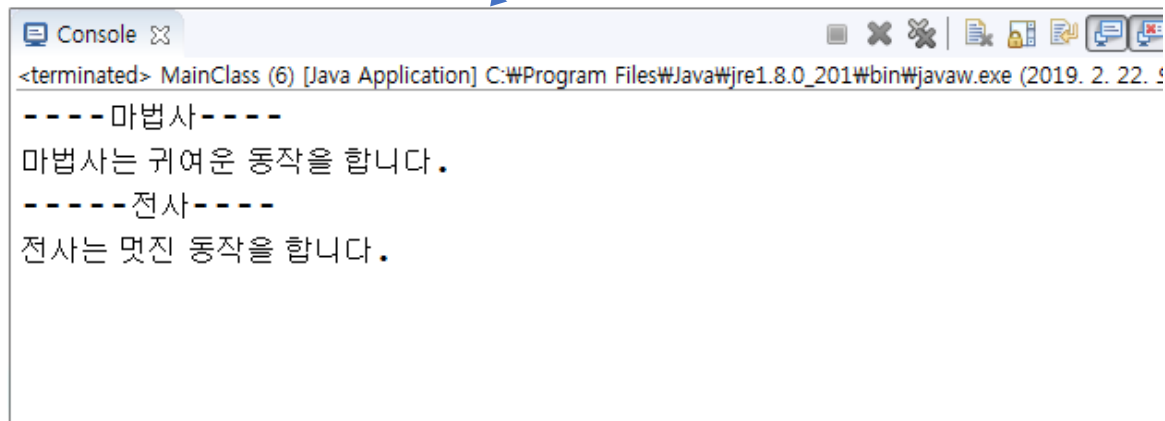
    void iceArrow() {
        System.out.println("얼음화살 사용");
    }
}
```

문제

1. Player 클래스에 action() 기능이 추가 되었습니다.
2. Warrior, Wizard 클래스에서 다음과 같은 실행 결과가 나오도록 overriding 하세요.

실행 화면

```
public class Player {  
    String name;  
    int hp;  
    int mp;  
  
    void info() {  
        System.out.println("캐릭터명:" + name);  
        System.out.println("체력:" + hp);  
        System.out.println("마나:" + mp);  
    }  
  
    void action() {  
  
    }  
}
```



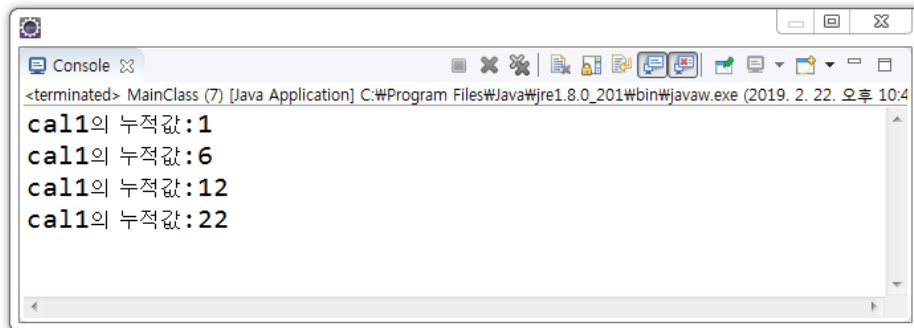
```
<terminated> MainClass (6) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (2019. 2. 22. 9)  
-----마법사-----  
마법사는 귀여운 동작을 합니다.  
-----전사-----  
전사는 멋진 동작을 합니다.
```

문제

1. 다음과 같은 실행 결과가 나올 수 있도록 Calculator 클래스에 add() 를 오버로딩 하세요.
힌트: Calculator클래스에는 값 누적하여 저장하는 result(멤버변수)가 있음.

실행 화면

```
public class MainClass {  
  
    public static void main(String[] args) {  
  
        Calculator cal = new Calculator();  
        cal.add(1); //매개변수 1개를 누적에 더함  
        System.out.println("cal1의 누적값:" + cal.result);  
        cal.add(2, 3); //매개변수 2개를 누적에 더함  
        System.out.println("cal1의 누적값:" + cal.result);  
        cal.add(1, 2, 3); //매개 변수 3개를 누적에 더함  
        System.out.println("cal1의 누적값:" + cal.result);  
        cal.add(1, 2, 3, 4); //매개 변수 4개를 누적에 더함  
        System.out.println("cal1의 누적값:" + cal.result);  
  
    }  
}
```

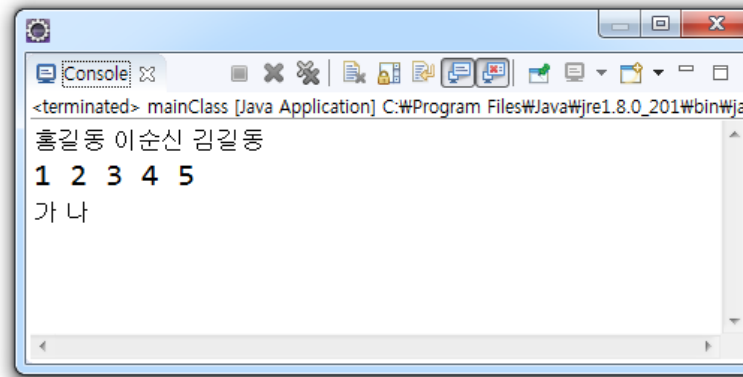


문제

- 다음과 같은 실행 결과가 나올 수 있도록, ArrayPrint클래스의 printArray() 메서드를 오버로딩 하세요.

실행 화면

```
public class mainClass {  
  
    public static void main(String[] args) {  
  
        ArrayPrint ap = new ArrayPrint();  
  
        String[] sArr = {"홍길동", "이순신", "김길동"};  
        int[] iArr = {1,2,3,4,5};  
        char[] cArr = {'가', '나'};  
  
        ap.printArray(sArr);  
        ap.printArray(iArr);  
        ap.printArray(cArr);  
  
    }  
}
```

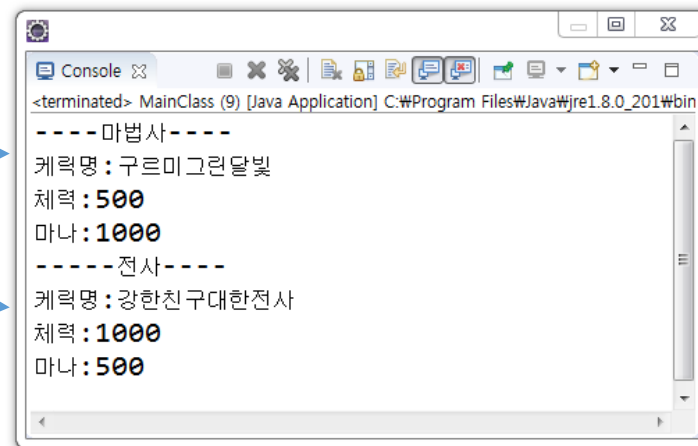


문제

- 다음과 같은 결과가 나올 수 있도록, Wizard, Warrior 클래스의 생성자를 this를 이용하여 만드세요.

실행 화면

```
public class MainClass {  
  
    public static void main(String[] args) {  
        System.out.println("----마법사----");  
        Wizard play1 = new Wizard("구르미그린달빛");  
        play1.info();  
  
        System.out.println("-----전사-----");  
        Warrior play2 = new Warrior("강한친구대한전사");  
        play2.info();  
    }  
}
```



문제

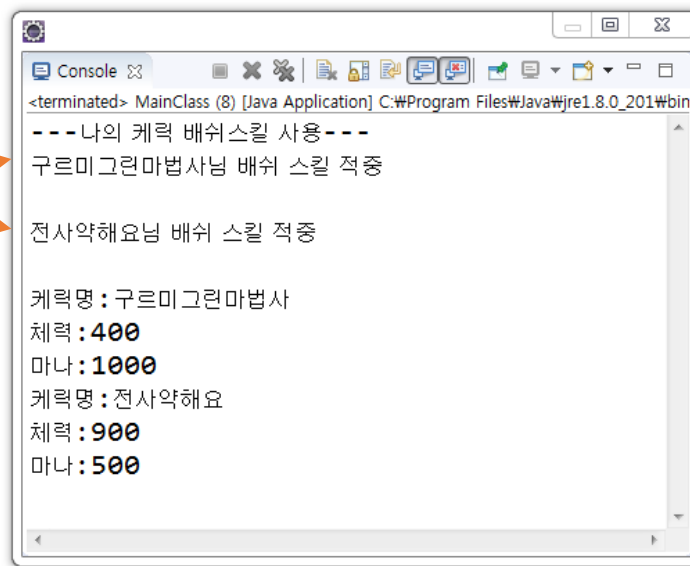
실행 화면

문제

1. Warrior 클래스의 bash()에 매개변수로 Warrior, Wizard가 모두 전달 될 수 있도록 변경해보세요.
2. bash스킬에 play들이 적중되면 "~~~님 배쉬 스킬 적중" 을 출력하고
2. 전달 된 객체의 hp를 -100감소하도록 기능을 만들어 보세요.

실행 화면

```
public class MainClass {  
    public static void main(String[] args) {  
  
        Warrior me = new Warrior("강한친구대한전사");  
  
        Wizard play2 = new Wizard("구르미그린마법사"); //체력: 500  
        Warrior play3 = new Warrior("전사악해요"); //체력: 1000  
  
        System.out.println("---나의 케릭 배쉬스킬 사용---");  
        me.bash(play2); //강한친구대한전사 bash스킬 사용(play2 적중)  
        me.bash(play3); //강한친구대한전사 bash스킬 사용(play3 적중)  
  
        play2.info();  
        play3.info();  
  
    }  
}
```



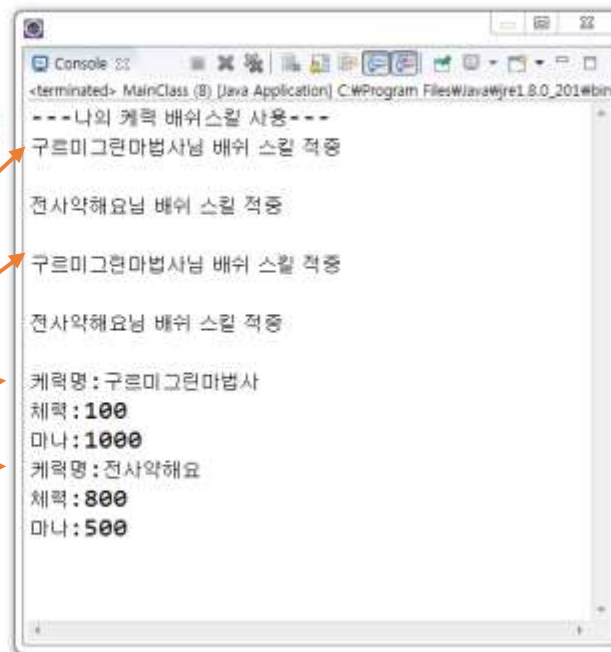
문제

1. bash() 에 적중 당한 클래스에 따라서 hp의 감소가 다르게 처리되도록 bash()를 수정해보세요.
2. 스킬을 사용할 때마다 me캐릭터의 mp를 100씩 감소 시킵니다.
캐릭터의 mp가 100 이하라면 스킬을 사용하지 못하고 메서드를 종료 시키세요

실행 화면

```
public class MainClass {  
    public static void main(String[] args) {
```

```
        Warrior me = new Warrior("강한친구대한전사");  
  
        Wizard play2 = new Wizard("구르미그린마법사"); //체력: 500  
        Warrior play3 = new Warrior("전사악해요"); //체력: 1000  
  
        System.out.println("---나의 캐릭 배워스킬 사용---");  
        me.bash(play2); //강한친구대한전사 bash스킬 사용(play2 적중)  
        me.bash(play3); //강한친구대한전사 bash스킬 사용(play3 적중)  
  
        me.bash(play2); //강한친구대한전사 bash스킬 사용(play2 적중)  
        me.bash(play3); //강한친구대한전사 bash스킬 사용(play3 적중)  
  
        play2.info();  
        play3.info();
```



```
    }  
}
```

문제

1. Wizard클래스에 Blizzard() 메서드를 만들겠습니다.
2. Blizzard()는 Player타입 배열을 받습니다.
3. 기능을 채워 넣고 main에서 메서드를 실행시키세요

실행 화면

```
public class Wizard extends Player {

    Wizard(String name) {
        this.name = name;
        this.hp = 500;
        this.mp = 1000;
    }

    void iceArrow() {
        System.out.println("얼음화살 사용");
    }

    public void blizzard(Player[] players) {
        System.out.println("-----");
        System.out.println(this.name + "님의 눈보라 시전!");
        //기능
        //1.매개변수 players배열을 향상된 for문으로 회전.
        //2.for문 안에서는 10~15사이의 랜덤난수를 발생.
        //3.players배열의 hp를 각각 랜덤난수 만큼 감소시킵니다.
        //4.plyaers배열의 이름과 함께 랜덤 피해를 각각 출력하세요

    }

}
```

<terminated> MainClass (16) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (2)

구르미님의 눈보라 시전!
 강한친구님이14피해를 입었습니다
 나의캐릭터님이12피해를 입었습니다

구르미님의 눈보라 시전!
 강한친구님이12피해를 입었습니다
 나의캐릭터님이14피해를 입었습니다

구르미님의 눈보라 시전!
 강한친구님이15피해를 입었습니다
 나의캐릭터님이11피해를 입었습니다

구르미님의 눈보라 시전!
 강한친구님이14피해를 입었습니다
 나의캐릭터님이14피해를 입었습니다

구르미님의 눈보라 시전!
 강한친구님이12피해를 입었습니다
 나의캐릭터님이12피해를 입었습니다

```
public class MainClass {

    public static void main(String[] args) {

        Warrior w1 = new Warrior("강한친구");
        Wizard w2 = new Wizard("구르미");
        Warrior me = new Warrior("나의캐릭터");

        //w2의 블리자드를 5번 실행시키세요

    }

}
```

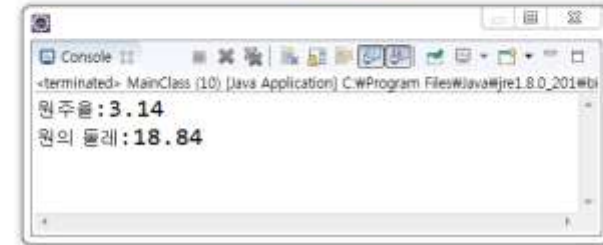
문제

1. 다음 Calculator 클래스에, static 변수 pi를 선언하세요. (pi = 3.14 입니다)
2. 원의 둘레를 구하여 리턴하는 static 메서드 circle(int)을 만드세요. (둘레공식 : $2 * r * \pi$)
3. 접근 제한자는 public 입니다.
3. main에서 ??를 채워 사용하세요.

실행 화면

```
public class Calculator {  
  
    int result; //누적 시킬 멤버변수  
  
    int add(int a) {  
        result +=a;  
        return result;  
    }  
  
    int add(int a, int b) {  
        result += a + b; // result = result + a + b;  
        return result;  
    }  
  
    int add(int a, int b, int c) {  
        result += a + b + c;  
        return result;  
    }  
  
    int add(int a, int b, int c, int d) {  
        result += a + b + c + d;  
        return result;  
    }  
}
```

```
public class MainClass {  
  
    public static void main(String[] args) {  
  
        System.out.println("원주율:" + ??);  
  
        //원의 둘레공식 : 2 * pi * 반지름  
        System.out.println("원의 둘레:" + ??); //매개변수 3전달  
  
    }  
}
```



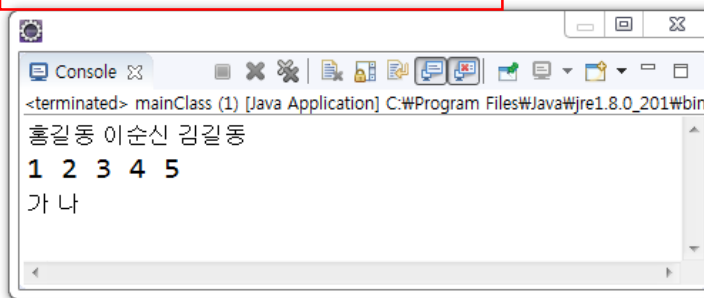
문제

1. ArrayPrint 클래스는 불필요하게 객체 생성을 하여 기능을 사용할 필요가 없습니다.
2. 객체 생성을 할 수 없도록 변경하세요.
3. printArray() 의 사용을 다음과 같이 할 수 있도록 변경하세요 (접근제한자는 public)

실행 화면

```
public class ArrayPrint {  
    void printArray(String[] arr) {  
        for(int i = 0; i < arr.length; i++) {  
            System.out.print(arr[i] + " ");  
        }  
        System.out.println();  
    }  
    void printArray(int[] arr) {  
        for(int i = 0; i < arr.length; i++) {  
            System.out.print(arr[i] + " ");  
        }  
        System.out.println();  
    }  
    void printArray(char[] arr) {  
        for(int i = 0; i < arr.length; i++) {  
            System.out.print(arr[i] + " ");  
        }  
        System.out.println();  
    }  
}
```

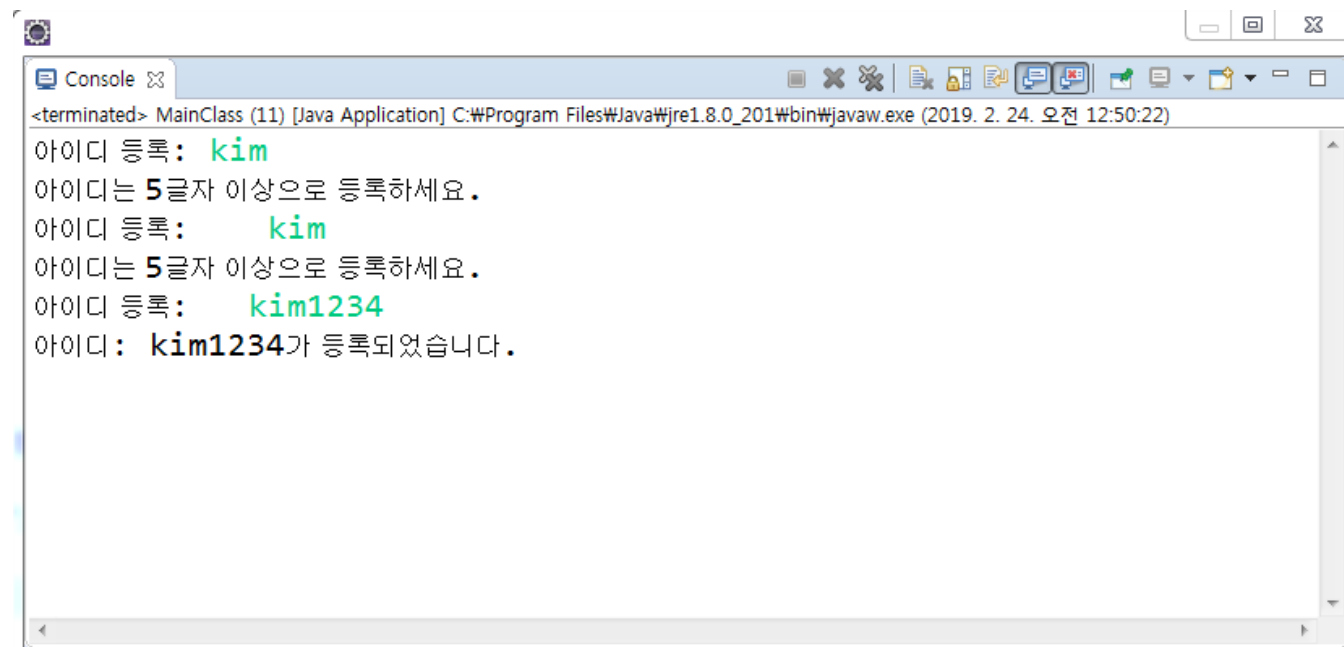
```
public class mainClass {  
    public static void main(String[] args) {  
  
        String[] sArr = {"홍길동", "이순신", "김길동"};  
        int[] iArr = {1,2,3,4,5};  
        char[] cArr = {'가', '나'};  
  
        ArrayPrint ap = new ArrayPrint(); //객체 생성 불가  
  
        ArrayPrint.printArray(sArr);  
        ArrayPrint.printArray(iArr);  
        ArrayPrint.printArray(cArr);  
    }  
}
```



문제

1. 무한 루프에서 아이디를 입력 받으세요.
2. 5글자 미만 이라면 "5글자 이상으로 등록하세요" 를 출력합니다.
3. 입력 글자가 5글자 이상이라면 "공백은 제거" 하고 반복문을 종료합니다.
4. String API를 이용하세요.

실행 화면

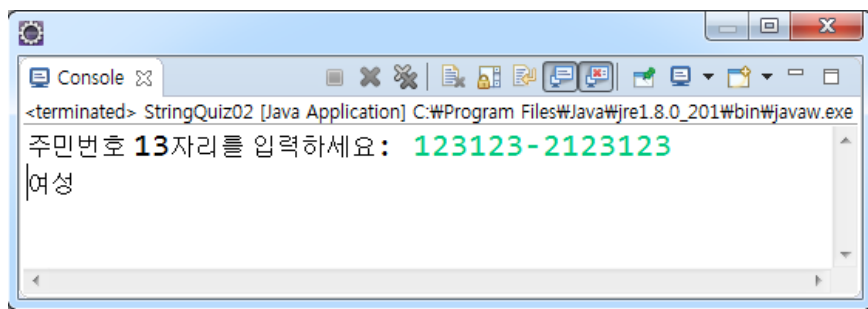
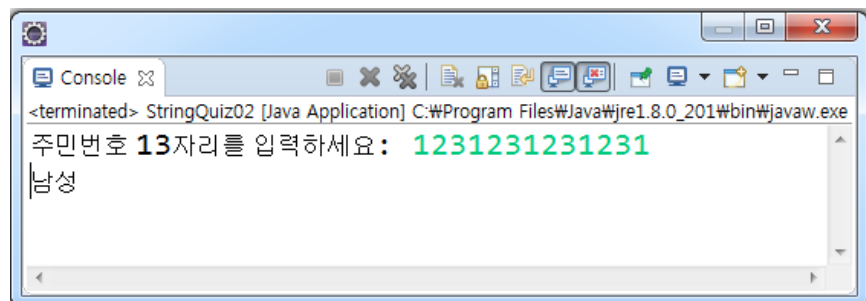
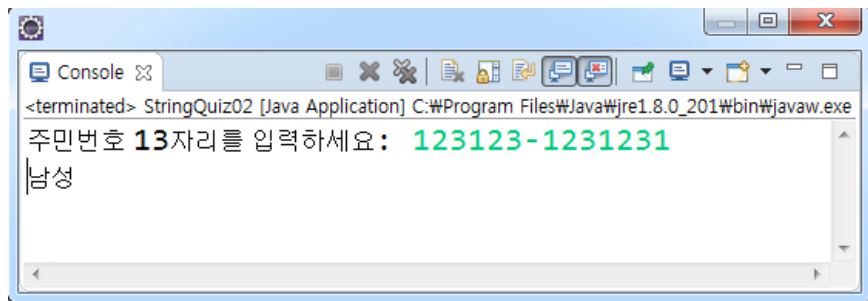


```
<terminated> MainClass (11) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (2019. 2. 24. 오전 12:50:22)
아이디 등록: kim
아이디는 5글자 이상으로 등록하세요.
아이디 등록: kim
아이디는 5글자 이상으로 등록하세요.
아이디 등록: kim1234
아이디: kim1234가 등록되었습니다.
```

문제

- 다음과 같은 결과가 나올 수 있도록 String API를 이용해서 처리하세요.

실행 화면



문제

1. Palindrome클래스에 palindromeCheck()함수는 매개변수로 String 값을 받아, 회문 여부를 검사하는 static 메서드입니다.
2. 회문: 똑바로 읽으나 거꾸로 읽으나 똑같은 단어나 문장.
3. ex) 다시 합창 합시다, 다 같은 것은 같다, 아 좋다 좋아 등
4. 문장은 공백을 포함하여 받을 수 있습니다.
5. 회문이라면 "회문입니다", 아니라면 "회문이 아닙니다" 를 리턴합니다.
6. 힌트)
 - * 함수를 이용하여 공백을 제거하세요. 힌트: replace()
 - * 문장의 길이의 중간값을 구하세요.
 - * 중간을 중심으로 앞, 뒤 문자열을 자르세요. 힌트 :substring()
 - * for문에서 charAt()을 사용해서 반대로 잘라 붙이세요. (or 빌더를 이용 하여 문자를 뒤집어 비교)

실행 화면

```
public class MainClass {  
    public static void main(String[] args) {  
  
        System.out.println(Palindrome.palindromeCheck("다시다"));  
        System.out.println(Palindrome.palindromeCheck("다시 합창 합시다"));  
        System.out.println(Palindrome.palindromeCheck("아 좋다 좋아"));  
        System.out.println(Palindrome.palindromeCheck("안녕하세요"));  
    }  
}
```

