

다음은 교재 8장(page 294)에서 제공하는 코드로 여러 개의 Webtoon 객체를 배열로 관리하고 있는 예시코드입니다.

이코드를 제시조건에 맞게 코드를 변경하여 WebtoonTest.java 소스 코드를 제출합니다.

// 작성자 정보를 주석으로 처리하세요

```
public class WebtoonTest {  
    public static void main(String[] args) {  
        Webtoon levelUpUp = new Webtoon("나 혼자만 레벨업", "판타지", "추공님");  
        Webtoon sweetHouse = new Webtoon("스위트 집", "스릴러", "칸비님");  
        Webtoon itaewonClassic = new Webtoon("이태원 클래식", "드라마", "광진님");  
  
        Webtoon[] webtoons = {levelUpUp, sweetHouse, itaewonClassic};  
        for (int i = 0; i < webtoons.length; i++) {  
            System.out.println(webtoons[i].toStr());  
        }  
        System.out.printf("웹툰 객체의 총 수: %d", Webtoon.getCount());  
    }  
}
```

```
class Webtoon {  
    private String title; // 제목  
    private String genre; // 장르  
    private String author; // 저자  
    private static int count = 0; // 생성된 웹툰 객체의 수  
  
    public Webtoon(String t, String g, String a) {
```

```
    title = t;  
    genre = g;  
    author = a;  
    Webtoon.count++;  
}
```

```
public String toStr() {  
    return String.format("Webtoon { title: %s, genre: %s, author: %s }",  
        title, genre, author);  
}
```

```
public static int getCount() {  
    return Webtoon.count;  
}  
}
```

제시조건

1. Webtoon[] webtoons → ArrayList<Webtoon> Webtoons 로 변경합니다.

2.

```
Webtoon levelUpUp = new Webtoon("나 혼자만 레벨업", "판타지", "추공님");  
Webtoon sweetHouse = new Webtoon("스위트 집", "스릴러", "칸비님");  
Webtoon itaewonClassic = new Webtoon("이태원 클래식", "드라마", "광진님");
```



```
String data = "나 혼자만 레벨업,판타지,추공님Wn"  
            + "스위트 집, 스릴러, 칸비님Wn"  
            + " 이태원 클래식, 드라마,광진님";
```

문자열 data에 있는 정보를 Scanner 클래스를 사용하여 읽어들이 Webtoon 객체를 생성 관리합니다.
이때 StringTokenizer 클래스를 적절히 사용합니다.
불필요한 공백은 제거합니다.

3. 다음은 배열을 순회하는 코드입니다.

```
for (int i = 0; i < webtoons.length; i++) {  
    System.out.println(webtoons[i].toStr());  
}
```



Collection을 순회하는 forEach 코드로 변경합니다.

4. 입력정보를 제공된 3건에서 10건으로 추가해 봅니다.