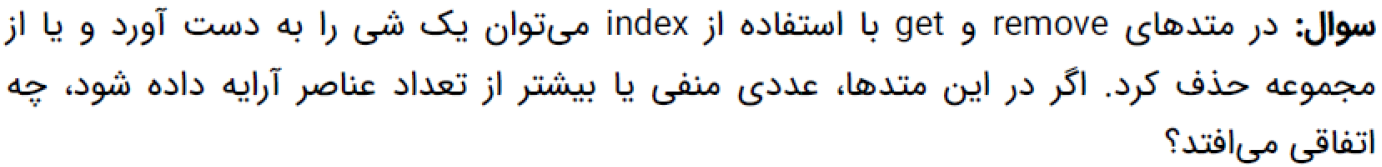
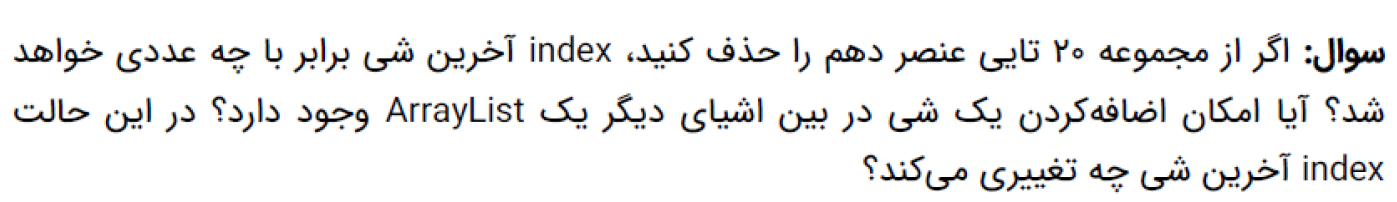
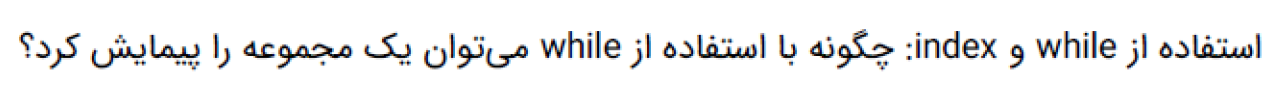
**به نام خدا**

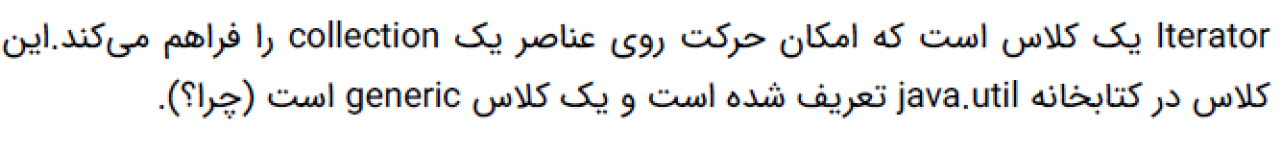
**محمد رضا حسن زاده 9731703 تمارین برنامه نویسی پیشرفته**

چون اندیس ها خارج از بازه‌ی بین صفر تا سایزشان منهای یک نیستند. در نتیجه به ارور IndexOutOfBoundsException بر می‌خوریم.

اگر از یک مجموعه‌ی 20 تایی عنصر دهم را حذف کنیم اندیس آخرین عنصر برابر با 18 می‌شود. امکان اضافه کردن در بین اشیای دیگر هم وجود دارد می‌توان با استفاده از متد add وبا وارد کردن دو پارامتر می‌توان بین آنها نیز یک شی‌ای را وارد کرد. در این حالت اندیس آخرین شی برابر با 20 می‌شود.

می‌توان به وسیله‌ی قطعه کد زیر این کار را انجام داد.

int temp = 0;  
while(temp < collection.size())  
{  
 //body of the loop  
 temp++;  
}

علت این است که به هنگام استفاده از این کلاس باید در بعد از Iterator از <> استفاده کنیم و تایپ متغیری را که می‌خواهیم iterate کنیم را مشخص کنیم. خود کامپایلر با استفاده از نوع ورودی‌ای که در داخل <> قرار می‌دهیم خودش تابع مناسب را فراخوانی می‌کند و با از آن ها استفاده می‌کنند.

**اشکال زدایی**

1- برای متغیر tracks باید یک کانستراکتور تعریف شود

2- در داخل حلقه‌ی for قسمت شرطش باید اصلاح شود چراکه متغیر i نمیتواند تا خود سایز کالکشن برود. همیشه بیشترین اندیس، یکی از تعداد سایز کالکشن ها کمتر است.

3- در اینجا پیمایش از اول به آخر ایراد دارد چراکه اگر که متد remove اجرا شود اندیس ها همگی یکی کم می‌شود و امکان دارد این بین یه سری مشکلات در داده ها پیش بیاید. برای رفع این مشکل از آخری به اول پیمایش می‌کنیم که این مشکل پیش نیاید.

import java.util.ArrayList;  
  
public class MusicOrganizer {  
 private ArrayList<String> tracks = new ArrayList<String>();  
   
 public void removeTrack(String nameLike){  
 for(int i = tracks.size() - 1; i >= 0; i-- ){  
 if(tracks.get(i).contains(nameLike)){  
 tracks.remove(i);  
 }  
 }  
 }  
}

و همچنین مجاز هستیم که از فراخوانی زنجیره‌ای استفاده کنیم.

**پاسخ دهید**

1- در اجرا این دوتا کد با یکدیگر تفاوتی ندارند اما در قطعه کد اول آبجکتی که ساخته شده داخل یک متغیر ذخیره شده است و می‌توان از آن در ادامه‌ی کد استفاده کنیم اما در قطعه کد دوم به این گونه نیست و نمی‌توان از آن در ادامه‌ی کد استفاده کرد.

2-

1. Teachers
2. Students
3. Courses
4. InformationGathering

3-

1. Hashset: این کلاس Set را پیاده سازی کرده است به وسیله‌ی یک جدول hash آن پیاده سازی شده است. این کالکشن به مانند یک مجموعه در ریاضیات عمل می‌کند.
2. Arraylist: این کلاس در جاوا به صورت داینامیک می‌تواند اطلاعات را ذخیره بکند و همچنین سایز آن ها رشد می‌کند با افزایش عناصر.
3. Hashmap: این کلاس جنریک مانند یک ArrayList می‌باشد اما این گونه که به جای اینکه اندیس ها به صورت خودکار از صفر تا سایز منهای یک بروند، ما برایش اندیس تعریف می‌کنیم که چه چیزی باشد. مثلا برای دفترچه تلفن می‌توان از این کلاس استفاده کرد، کاربرد این کلاس جاییست که می‌خواهیم برای هر داده‌ای یک اندیس خاصی تعریف کنیم و به وسیله‌ی آن اندیس به داده‌ی داخل آن عنصر دسترسی پیدا کنیم.

4- Apache POI