

مدیریت موزیک

فرض کنید می خواهید آهنگهای روی لپ تاپتان را مدیریت کنید. در این تمرین قصد داریم برنامه ای با این هدف طراحی کنیم که بتواند برچسب مناسب برای نام آهنگها در نظر گرفته و طبقه بندی و مرتب سازی آنها را انجام دهد.

کلاسها و توابعی که برای هر کلاس در نظر گرفته شده اند، در ادامه عنوان شده است. طبق این توضیحات کلاسهای خواسته شده را پیاد سازی کنید. غیر از `setter` و `getter` و توابعی که صریحا به عنوان `public` مشخص شده اند، بقیه توابع باید `private` باشند.

کلاس `playlist`

توابع:

```
public Playlist(String name);
```

این سازنده نام `playlist` را دریافت می کند و شی می سازد.

```
public Playlist();
```

این سازنده اولین بار که فراخوانی میشود یک `playlist` به نام `playlist1` می سازد و یک شمارنده را هر بار افزایش داده و دفعات بعد با توجه به این شمارنده نام `playlist` را مشخص می کند. به عنوان مثال در پنجمین بار فراخوانی `playlist5` ساخته می شود.

```
public void addSong(Song song) ;
```

این تابع یک آهنگ را به playlist اضافه می کند.

```
public void addAlbum(Album album) ;
```

این تابع یک آلبوم را می گیرد و به همان ترتیب که در آلبوم ظاهر شده اند، به playlist اضافه می کند.

```
public boolean removeSong(Song song) ;
```

این تابع یک آهنگ را گرفته و از playlist حذف می کند.

```
public Song getCurrentSong() ;
```

آهنگی را که در حال پخش هست برمی گرداند، اگر هنوز تابع play فراخوانی نشده و یا stop فراخوانی شده باشد، null برمی گرداند.

```
public void goNext() ;
```

آهنگ در حال اجرا را یکی جلو می برد و اگر انتهای playlist باشد، تابع stop فراخوانی می شود.

```
public void goPrevious() ;
```

این تابع آهنگ قبلی را برمی گرداند، اگر آهنگ اول باشد، تابع stop فراخوانی می شود.

```
public void play() ;
```

همیشه اولین آهنگ از لیست را به عنوان آهنگ در حال اجرا قرار می دهد.

```
public void stop() ;
```

آهنگ در حال اجرا را null می کند.

کلاس Album

```
public Album(String name, Artist artist, String genre);
```

یک آلبوم خالی با دریافت اطلاعات ورودی می سازد.

```
public static Album getSingle();
```

این تابع هر بار که فراخوانی شود یک آلبوم یکسان با نام `single` را برمی گرداند که نام خواننده و سبک آن `null` است.

```
public Artist getArtist();
```

خواننده آلبوم را برمی گرداند.

```
public String getGenre();
```

سبک آلبوم را برمی گرداند.

```
public String getName();
```

نام آلبوم را بر می گرداند.

```
public Boolean addSong(Song song);
```

یک آهنگ را می گیرد و در صورتی که خواننده آن با خواننده آلبوم یکی بود، آن را به آلبوم اضافه می کند و `true` برمی گرداند و در غیر این صورت `false` بر می گرداند.

```
public void removeSong(Song song);
```

یک آهنگ را می گیرد و در صورت وجود در آلبوم آن را حذف می کند.

```
public Song[] getSongs();
```

همه آهنگهای یک آلبوم را بر می گرداند.

```
public String toString();
```

مشخصات یک آلبوم را به صورت یک رشته شامل نام آلبوم، نام خواننده، نام سبک و نام همه آهنگها (به ترتیب الفبا بر می گرداند).

کلاس song

```
public Song(String name, Artist artist);
```

این سازنده یک آهنگ با نام داده شده و نام خواننده می سازد.

```
public String getName();
```

```
public Artist getArtist();
```

```
public Album getAlbum();
```

این توابع به ترتیب نام آهنگ، نام خواننده و نام آلبوم را بر می گردانند. اگر آهنگ در هیچ آلبومی اضافه نشده باشد، تابع آخر single را بر می گرداند.

```
public String toString();
```

اطلاعات آهنگ شامل نام، نام خواننده و نام آلبوم را به صورت یک رشته بر می گرداند.

کلاس Artist

```
public Artist(String name);
```

یک خواننده را با دریافت نام آن ایجاد می کند.

```
public String getName();
```

نام خواننده را بر می گرداند.

```
public Song[] getSongs();
```

لیست همه آهنگهای خواننده را بر می گرداند.

```
public Album[] getAlbums();
```

همه آلبومهای خواننده شده توسط یک خواننده را بر می گرداند.

```
public String toString();
```

نام خواننده و نام همه آلبومهایی را که خوانده است برمی گرداند.