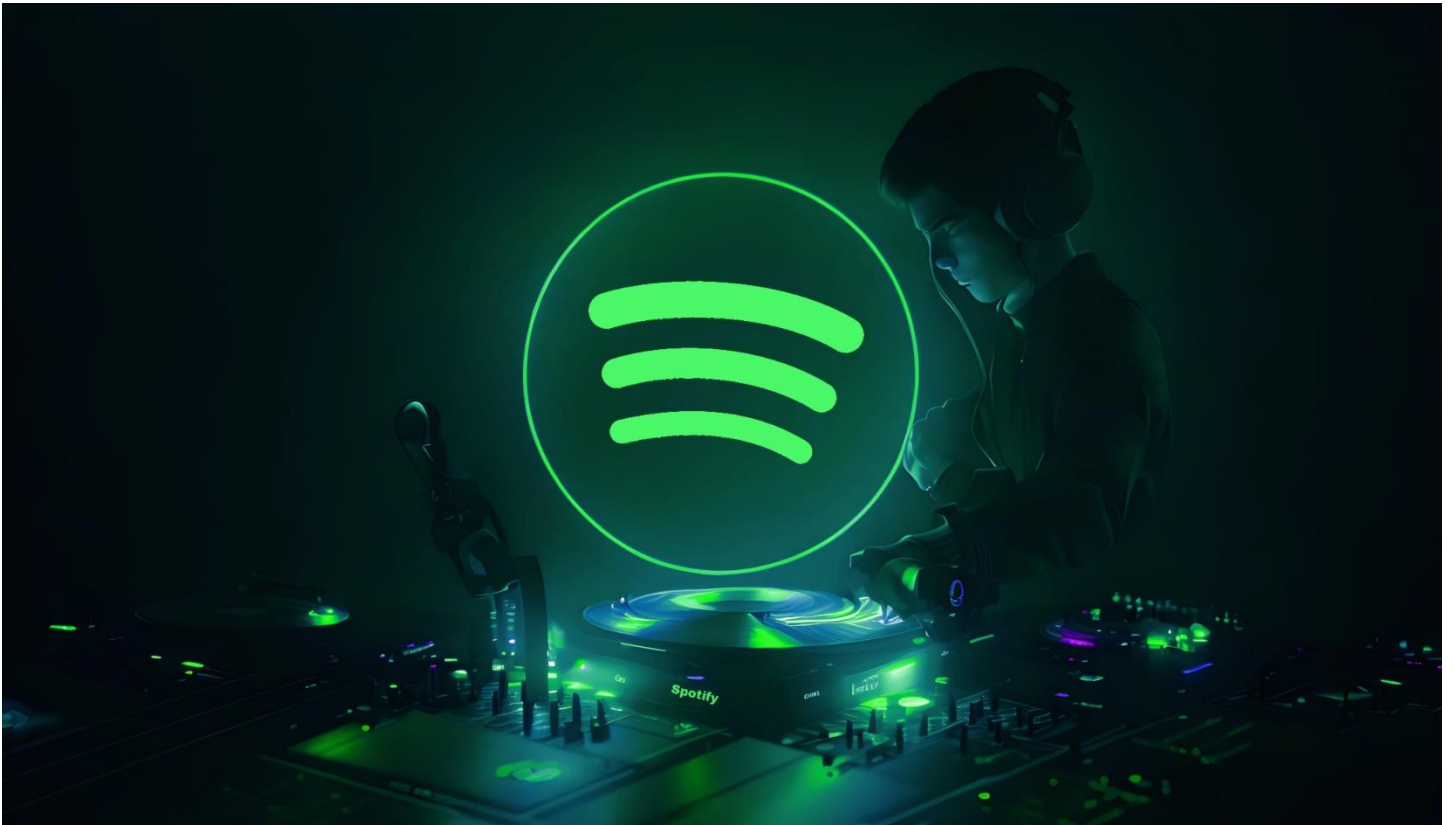


دانشگاه گازیان

پروژه دوم برنامه سازی پیشرفته

نرم 14021

Spotify



## مقدمه

موزیک پلیر یکی از ابزارهایی است که بسیاری از ما از آن برای گوش دادن به موسیقی لذت می‌بریم. در پروژه دوم برنامه‌نویسی پیشرفته، هدف شما از طریق ایجاد یک موزیک پلیر، یادگیری مباحث پایه شی گرای و کار با فایل در جاوا می‌باشد. شما با تعریف کلاس‌ها، اشیاء، سازنده‌ها، مفهوم **this** و سطوح دسترسی با اصول برنامه نویسی شی گرا آشنا می‌شوید. این پروژه به شما فرصت می‌دهد تا مفاهیم اساسی برنامه‌نویسی شی گرا را در عمل تجربه کنید و مهارت‌های برنامه‌نویسی خود را افزایش دهید.

## شروع برنامه

با شروع برنامه انتظار می‌رود کاربر با انتخاب یکی از دو گزینه **Login** یا **Sign up** و وارد کردن نام کاربری و رمز خود یا ساخت اکانت به پروفایل خود وارد شود.

### Sign up

Username

Password

Gender

Male

Date of birth

January

1

2000

Type of User

Listener / Artist

Cancel

Submit Information

Sign Up Page

### Login

Username

Password

Type of User

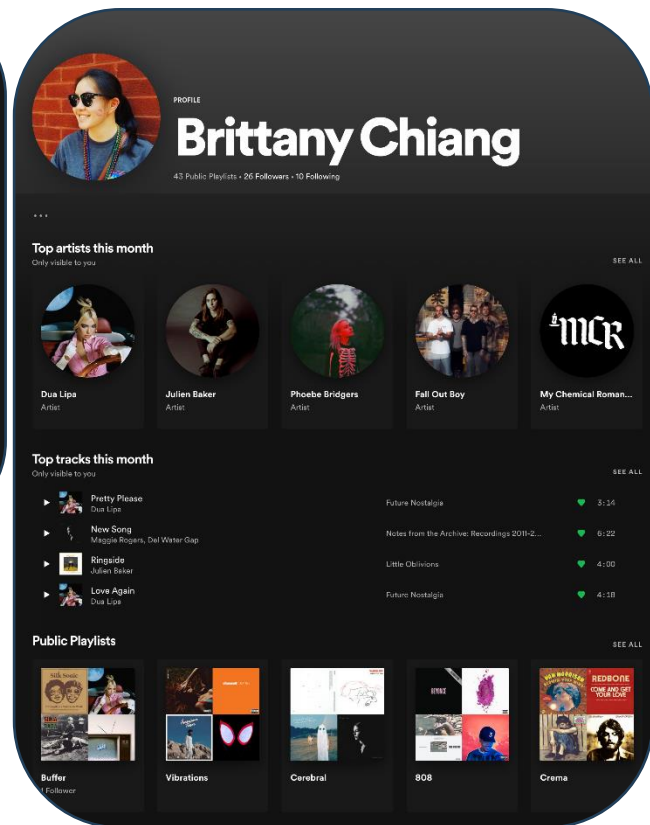
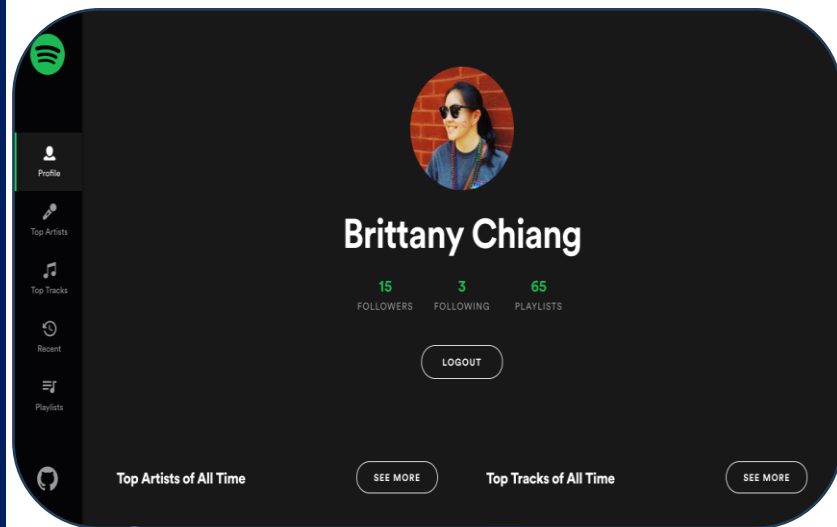
Artist

Cancel

Login

Login Page

**نکته:** در صورتی که تمایلی به پیاده سازی GUI ندارید، باید اطلاعاتی که در تصویر بالا از کاربر گرفته می‌شود را در ترمینال از کاربر در قسمت ثبت نام دریافت و روی فایل به شیوه صحیح ذخیره کنید.



## Examples of the page after sign up or login

## ساختار کلی

شما باید کلاس های ذکر شده در پایین را به درستی پیاده سازی کنید. می توانید به سلیقه خودتان کلاس هایی را برای کامل تر شدن پروژه اضافه کنید و برنامه ی خود را به برنامه اصلی شبیه تر کنید. حتی می توانید کلاس های اصلی را بر اساس طراحی موردنظرتان تغییر دهید اما در این مورد باید توجه بسیار مناسبی داشته باشید. در زمان ارائه ، این مورد سختگیرانه بررسی می شود.

## I. کلاس اصلی (Music Player (Main))

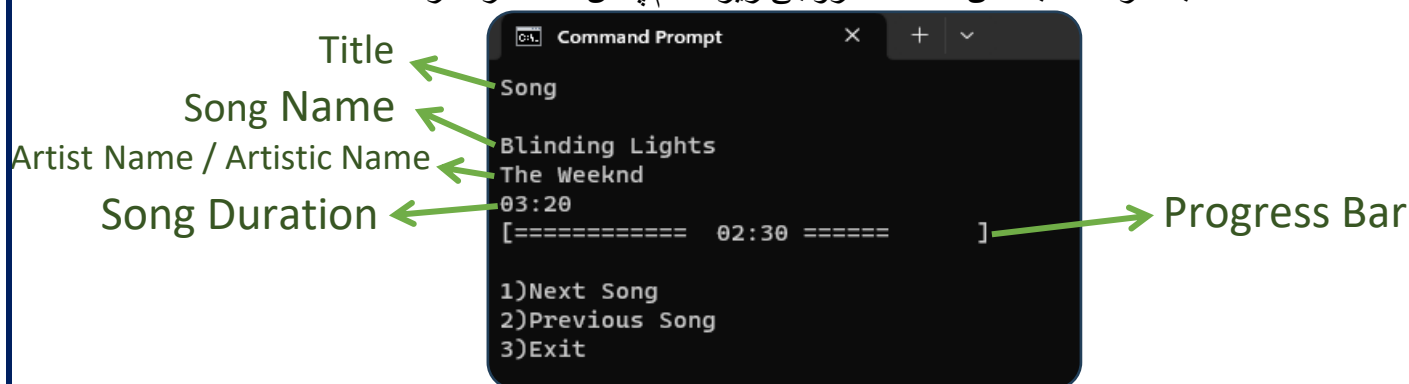
- متدی برای ورود به عنوان Artist (هنرمند)
- متدی برای ورود به عنوان Listener (شنونده)
- متدی برای ثبت نام در برنامه (Sign up)
- # بسته به پیاده سازی صفحه Login خود دو متد ورود را می توانید یکی کنید.

## II. هنرمند (Artist)

- نام هنری (Artistic Name)
- نام آلبوم ها (Album Names)
- نام آهنگ ها (Songs Names)
- متدی برای گذاشتن و تغییر نام هنری (Set Artistic Name)
- متدی برای ساخت آهنگ جدید (Create Song)
- متدی برای حذف آهنگ (Remove Song)
- متدی برای ساخت آلبوم جدید (Create Album)
- متدی برای حذف آلبوم (Remove Album)
- متدی برای تغییر آلبوم (تغییر نام آلبوم و حذف و اضافه کردن آهنگ) (Edit Album)

## III. شنونده (Listener)

- لیست های پخش (Playlists)
- شماره آخرین آهنگ در حال پخش به همراه شماره یا اسم لیست پخش که در آن قرار داشته است (فقط در صورتی که بیشتر از یک لیست پخش داشته باشد).
- متدی برای ساخت لیست پخش (Create Playlist)
- متدی برای حذف لیست پخش (Remove Playlist)
- متدی برای تغییر لیست پخش (تغییر نام لیست پخش و حذف و اضافه کردن آهنگ و تغییر ترتیب آهنگ ها) (پیاده سازی مرتب سازی آهنگ ها به ترتیب الفبا به صورت صعودی و نزولی) (Edit Playlist)
- متدی برای پخش آهنگ (Play)
- متد پخش آهنگ شامل 2 حالت است:
  - پخش آخرین آهنگی که در حال پخش بوده (Play the last song that was playing)
  - انتخاب لیست پخش (وارد کردن نام لیست پخش یا وارد کردن شماره آن):
 بعد از انتخاب لیست پخش باید نام آهنگی که می خواهید را وارد کند تا آهنگ پخش شود. (Search Song)  
 بعد از انتخاب شدن آهنگ، خروجی زیر حکم پخش آهنگ را دارد.



- ✓ پیاده سازی (Seek Bar / Playback Bar) Progress Bar به صورتی که پر شود و پخش آهنگ را شبیه سازی کند. **نمره اضافی** است. (برای پر شدن آن زمانی بین 5 تا 15 ثانیه به صورت رندوم در نظر بگیرید.)
- **یاد آوری:** همانطور که در صفحه ثبت نام مشاهده کردید هر دو کلاس هنرمند و شنونده باید شامل ویژگی های: Username و Password و Gender و Date of Birth باشند.

#### IV. آلبوم (Album)

- ❖ نام آلبوم (Album Name)
- ❖ نام هنرمند (Artist Name)
- ❖ لیست آهنگ ها (List of Songs)
- متدی برای اضافه کردن آهنگ (Add Song)
- متدی برای حذف آهنگ (Remove Song)

#### V. آهنگ (Song)

- ❖ نام آهنگ (Song Name)
- ❖ نام هنرمند (Artist Name)
- ❖ نام آلبوم (Album Name)
- ❖ مدت زمان پخش آهنگ (Song Duration)

#### VI. لیست پخش (Playlist)

- ❖ نام لیست پخش (Playlist Name)
- ❖ نام شنونده (Listener Name)
- ❖ لیست آهنگ ها (List of Songs)
- متدی برای اضافه کردن آهنگ (Add Song)
- متدی برای حذف آهنگ (Remove Song)
- متدی برای پیدا کردن آهنگ (Search Song)

### نکات تکمیلی

- از ابتدای ورود به برنامه هر صفحه باید شامل یک تیتل (Titel) با توجه به کلاس یا متدی که در آن قرار دارید باشد.
- بعد از وارد کردن هر دستور که شما را به صفحه جدید وارد می کند یا ورودی جدید از شما می خواهد باید صفحه برنامه رفرش شود. (کد در ترمینال ارائه گرفته می شود و وارد کردن تعداد زیادی اینتر پذیرفته نمی شود.)
- در هر صفحه ای که قرار دارید باید متد Exit ای داشته باشید که شما را به صفحه قبلی یا پروفایلتان برگرداند.
- برای خروج از کاربر فعلی باید متد Log out در پروفایل پیاده سازی شود.
- پیاده سازی جزئیاتی مثل دکمه Pause / Play ، عوض کردن آهنگ ، انتخاب آهنگ به صورت شافل (رندوم) ، قرار دادن پخش آهنگ روی تکرار بعد از اتمام تک آهنگ یا تکرار از اول پلی لیست ، تاریخ انتشار آلبوم و هر قابلیت دیگری که یک برنامه پخش آهنگ دارد به عهده دانشجو می باشد و جزء **خلاقیت** در پروژه به حساب می آید و دارای **نمره اضافی** است.
- ذخیره اطلاعات کاربران به شیوه فرمت دهی شده و مرتب روی فایل جزء بخش های اصلی پروژه می باشد.

- برای Login کردن و چک کردن Password شما به هیچ وجه حق ندارید از get Password یا هر فانکشنی که مقدار Password را return کند استفاده کنید!

Sample:

```
Command Prompt

Profile

Username: ali
Gender: Male
Date of birth: 2000/1/1
Type of User : Listener

PlayLists:
1_Remixes:[song1 ,song2, song3 , ...]
2_Happy songs:[song1 ,song2, song3 , ...]

1)Create PlayList
2)Remove PlayList
3)Edit PlayList
4)Play
5)Log out
```

## نمره مثبت

1. پیاده سازی Follower و Following به طور کامل
2. پیاده سازی بخشی برای پیدا کردن پروفایل دیگر شنوندگان و اضافه کردن لیست پخش آنها به پروفایل خود.
3. طراحی رابط گرافیکی (GUI)
4. بعد از پیاده سازی GUI برنامه قادر به انتخاب یک فایل آهنگ و پخش آن باشد.
5. ذخیره اطلاعات به کمک دیتابیس (SQL)
6. رعایت هر چه بیشتر اصول چهارگانه شی گرای: (Abstraction, Encapsulation, Polymorphism , Inheritance)

## توجه:

پیشنهاد می‌شود با توجه به زمانی که برای این پروژه دارید پروژه را با GUI پیاده سازی کنید تا بتوانید از آن در پروژه های بعدی خود استفاده کنید.

هر گونه ایده و خلاقیت جدید در پروژه با هماهنگی با دستیاران آموزشی **نمره مثبت** به همراه دارد.

هیچ توجیهی در استفاده از زبان های برنامه نویسی دیگر وجود نداشته و برابر با تحویل ندادن پروژه ، **نمره صفر** به دانشجو تعلق میگیرد.

پروژه تنها به صورت انفرادی قابل انجام بوده و در صورت وجود هرگونه تشابه بین دو کد یا عدم تسلط به روند، **نمره منفی** به دانشجویان داده میشود.

به کدی که قابل اجرا نباشد و از شی گزایی استفاده نشده باشد نمره ای بین 0 تا 20 با توجه به کد زده شده تعلق میگیرد.

در صورت مشاهده باگ در حین ارائه اگر در زمان ارائه قادر به دیباگ آن باشید نمره منفی به شما تعلق نمی گیرد.

کامنت نویسی مناسب و رعایت دندانه گذاری الزامی است.