

به نام خداوند یکتا

پروژه پایان ترم - کارگاه برنامه نویسی پیشرفته

نیمسال اول ۹۷-۹۸ دانشگاه صنعتی سجاد

مدرس: اشکان رضازاده

عنوان پروژه : طراحی سامانه فروش بلیط سینما، تئاتر و کنسرت

عناصر مختلفی که در سامانه موجود است عبارت اند از:

-فیلم

-تئاتر

-کنسرت

-مدیر سیستم

-کاربر

-نویسنده

-کارگردان

-خواننده

-سالن (سالن اجرای هر یک از سه برنامه)

-بلیط

-سانس

...

توجه کنید این موارد با کلاس های لازم برای پیاده سازی یکسان نیست و شما با توجه به نیازمندی های سیستم، طبق تشخیص خودتان و با توجه به هر آنچه که در کلاس آموخته اید، کلاس های مختلف را ایجاد و ارتباط میان آن ها را برقرار می نمایید.

ویژگی های هر یک از این عناصر نامبرده عبارت اند از:

فیلم:

- کد فیلم
- نام فیلم
- ژانر
- مدت زمان
- خلاصه داستان
- سال ساخت
- نام کارگردان
- نام نویسنده
- نام بازیگران (به صورت یک رشته که هر نام با '،' از دیگری جدا می شود)
- قیمت بلیط

تئاتر:

- کد تئاتر
- نام تئاتر
- ژانر
- نویسنده
- کارگردان
- قیمت بلیط

کنسرت:

- کد کنسرت
- نام گروه

- نام خواننده
- سبک موسیقی
- نام اعضای گروه (به صورت یک رشته که هر نام با '،' از دیگری جدا می شود)
- قیمت بلیط

موجودیت فرد اعم از مدیر سیستم، کاربر، خواننده، نویسنده و کارگردان

- نام
- نام خانوادگی
- سن
- جنسیت
- نام مستعار

مدیر سیستم:

- نام کاربری
- کلمه عبور

کاربر

- نام کاربری
- کلمه عبور

سالن اجرا (برای هر سه برنامه یکسان است و تفاوتی در نوع سالن نداریم)

- کد سالن
- نام سالن
- حداکثر ظرفیت
- درجه کیفی
- آدرس
- لیستی از صندلی ها

صندلی

- کد سالن

- عنوان ردیف
- شماره صندلی

سانس:

- کد سانس
- کد برنامه
- نام برنامه
- تاریخ و ساعت اجرای برنامه
- کد سالن
- ...

بلیط:

- سریال بلیط
- قیمت
- سانس
- شماره صندلی
- نام و نام خانوادگی کاربر خریدار
- کد خریدار
- ...

بدیهی است که ممکن است موجودیت های دیگر یا فیلد های دیگری برای موجودیت های نامبرده مورد نیاز باشد که باید خودتان تشخیص داده و اضافه نمایید.

سعی کنید تمامی مطالب گفته شده در کلاس را به کار گیرید و تا حد ممکن از شیء گرایی استفاده کنید و ساختار مطلوب را ایجاد نمایید.

(راهنمایی) : مثلاً برای سه موجودیت فیلم، کنسرت و تئاتر، می توان یک کلاس پایه انتزاعی با نام برنامه در نظر گرفت که فیلدهای مشترک این سه موجودیت را در خود جای داده است و مثلاً یک تابع **show** دارد که مجازی خالص است و در کلاس های فرزند این تابع وابسته به اطلاعات موجود دوباره تعریف می شود.

می توان برای فیلم و تئاتر یک کلاس پدر در نظر گرفت که فیلدهای مشترک مانند ژانر و نام کارگردان و نام نویسنده و... را در خود جای داده است. بنا براین ابتدا موجودیت برنامه را به صورت کلاس انتزاعی ایجاد کرده و سپس یک کلاس برای کنسرت و یک کلاس پدر برای فیلم یا تئاتر می توان در نظر گرفت و حال کلاس فیلم و تئاتر را از آن ارث بری کرد.

به عنوان مثال دیگر می توان یک کلاس با یک فیلد **id** ایجاد کرد و همه کلاس هایی که این فیلد را دارند از آن ارث برد.

به عنوان مثال دیگر صندلی رزرو شده باید یک موجودیت باشد که مشخص می کند در چه سانس صندلی رزرو و در چه سانس آزاد است و...

برای سایر موجودیت ها نیز ساختار شیءگرایی را مشابه نمونه ذکر شده می توان ایجاد کرد که تحلیل و پیاده سازی آن بر عهده شماست.

منوی اصلی برنامه:

با اجرای برنامه ابتدا منویی جهت ورود به عنوان کاربر یا مدیر سیستم نمایان می شود که در صورت انتخاب هر یک باید یوزرنیم و پسورد را وارد کنیم و عملیات چک کردن صحت و لاگین انجام شود، سپس وارد سیستم شویم.

گزینه های منوی مدیر سیستم:

۱- ایجاد و حذف کاربر جدید

۲- ایجاد و حذف برنامه جدید (کنسرت، فیلم، تئاتر) : که در این قسمت باید اطلاعات مربوط به برنامه وارد شود.

۳- ایجاد و حذف سالن (اطلاعات در نظر گرفته شده شامل حداکثر ظرفیت و آدرس و ...)

۴- ایجاد و حذف سانس (یعنی ایجاد یک سانس برای یکی از برنامه ها اعم از فیلم، کنسرت یا تئاتر و دریافت ورودی های مورد نیاز)

۵- ایجاد و حذف صندلی (ابتدا باید سالن مورد نظر انتخاب شود و سپس صندلی ها برای آن سالن ایجاد گردد).

۶- ایجاد و حذف نویسنده، کارگردان، خواننده و ویرایش اطلاعات آنان

۷- ویرایش اطلاعات بلیط تا قبل از تاریخ اجرای برنامه

۸- گزارش آمار فروش هر یک از برنامه ها اعم از فیلم، تئاتر و کنسرت (تعداد و جمع کل مبلغ)

۹- مشاهده سانس های موجود در یک تاریخ مشخص (ورودی : تاریخ، خروجی: اطلاعات سانس های آن تاریخ) با حذف یک برنامه، تمامی بلیط ها صندلی های رزرو شده و سانس های مربوط به آن برنامه نیز باید حذف گردد(نمره اضافه)

گزینه های منوی کاربر

۱- جستجو برای برنامه بر اساس کد و نام برنامه: شامل مشاهده اطلاعات کامل آن برنامه، سانس های موجود که تاریخ آن ها برای آینده است و تعداد صندلی خالی برای سانس ها.

۲- در صورت پیدا شدن سانس و موجود بودن ظرفیت برای برنامه مورد نظر، پس از انتخاب برنامه، صندلی های قابل رزرو نمایش داده شود و کاربر یکی از آن ها را انتخاب نماید.

۳- ، خرید بلیط برای آن سانس: که این خود شامل پروسه چک کردن خالی بودن صندلی مورد نظر است و در صورت خالی نبودن اخطار نمایش داده می شود و در صورت خالی بودن آن صندلی رزرو خواهد شد و شخص دیگری نمی تواند در آن تاریخ آن صندلی را رزرو نماید.

۴- مشاهده تمام بلیط های خریداری شده

۵- جستجوی بلیط خریداری شده بر اساس سریال بلیط

۶- ابطال بلیط قبل از شروع برنامه (نمره اضافه) : در این قسمت کاربر می تواند تا قبل از شروع برنامه بلیط خود را باطل کرده و در نتیجه باید تمامی عواملی که به صورت وابسته با این بلیط ایجاد کرده اید را حذف کنید.

*نکته: برای خرید بلیط از بحث پرداخت و ... صرف نظر کنید و فرض کنید کاربر همان جا که صندلی را رزرو کرد و بلیط صادر شد پرداخت را نیز انجام داده است.

نکات تکمیلی:

۱- برای انجام جستجو ها و دیگر پردازش های رشته ای از توابع کتابخانه ای استفاده نکنید و از توابعی که خودتان برای پردازش رشته ها نوشته اید بهره بگیرید.

۲- در پیاده سازی پروژه از مفاهیم زیر بهره بگیرید:

- شیء گرایی کامل برنامه

- ارث بری

- چندریختی (مثلا برای نمایش اطلاعات هر کدام از برنامه ها، می توانید با چند ریختی پویا توابع مختلف show را پیاده سازی نمایید)

- سعی کنید تا حد امکان از مطالبی که در کلاس گفته نشده است استفاده نکنید، اما استفاده از Vector مانعی ندارد.

-بکارگیری الگو

۳- لطفا صحت اطلاعات ورودی را در هر مرحله بررسی کنید، مثلا اگر کاربر شماره صندلی که موجود نیست را وارد کرد خطا نمایش داده شود، یا اگر برای ثبت سانس، مدیر سیستم کد برنامه ای را وارد کرد که اصلا وجود ندارد خطا نمایش داده شود و...

تقسیم کار صحیح و برابر و تسلط همه اعضای گروه بر پروژه را حتما مد نظر قرار داده و تا حد امکان برنامه را ساده پیاده سازی کنید.

صحت عملکرد برنامه، تمیزی و ساختار صحیح کد ها، منطق پیاده سازی کلاس ها، تسلط تمامی اعضای گروه در هنگام ارائه و از همه مهمتر **تلاش** شما در جهت اجرای پروژه ملاک های ارزش گذاری پروژه شماست.

- پروژه را در قالب تیم های حداکثر ۳ نفره می توانید تحویل دهید.

- روز تحویل حضوری پروژه هنوز مشخص نشده است، تا اطلاع ثانوی تاریخ تحویل را ۲۰ دی ماه در نظر بگیرید.

- انجام پروژه جهت یادگیری مفاهیم تئوری و آشنایی با استفاده از کلاس ها در عمل است، لذا لطفا از کپی برداری پروژه به هر شیوه اعم از تغییر نام متغیر ها، تغییر ترتیب نوشتن توابع، کلاس ها و... خود داری ننمایید، در صورت مشاهده کپی، از هر دو گروه بخش زیادی از نمره تمرین کسر خواهد شد.

- در صورتی که در رابطه با صورت پروژه سوالی دارید لطفا فقط با ایمیل ashkan.rezazadeh@mail.um.ac.ir در ارتباط باشید.

شاد و سربلند باشید