دانشکده مهندسی کامپیوتر

برنامه‌سازی پیشرفته
تمرین های سری دوم

مدرس: سید صالح اعتمادی
طرح تمرین: امید میرزاجانی

مهلت ارسال:
دوشنبه ۱۲ اسفند ۹۸

فهرست مطالب

۱ مقدمه

۱.۱ موارد مورد توجه

- توجه داشته باشید که برای کسب نمره‌ی قبولی درس کسب حداقل نصف نمره‌ی هر سری تمرین الزامی می‌باشد.
- مهلت ارسال پاسخ تمرین تا ساعت ۲۳:۵۹ روز اعلام‌شده است. توصیه می‌شود نوشتن تمرین را به روزهای نهایی موکول نکنید.
- همکاری و هم‌فکری شما در حل تمرین مانعی ندارد، اما پاسخ ارسالی هر کس حتما باید توسط خود او نوشته شده باشد.
- مبنای درس، اعتماد بر پاسخ ارسالی از سوی شماست؛ بنابراین ارسال پاسخ در ریپازیتوری گیت شما به این معناست که پاسخ آن تمرین، توسط شما نوشته شده است. در صورت تقلب یا اثبات عدم نوشتار پاسخ حتی یک سوال از تمرین، برای هر دو طرف تقلب‌گیرنده و تقلب‌دهنده نمره‌ی **مردود** برای درس در نظر گرفته خواهد شد.
- توجه داشته باشید که پاسخ‌ها و کدهای مربوط به هر مرحله را بایستی تا قبل از پایان زمان مربوط به آن مرحله، در سایت Azure DevOps (طبق توضیحات کارگاه‌ها و کلاس‌ها) بفرستید. درست کردن `Pull request` و `Complete` کردن `Pull request` و انتقال به شاخه‌ی `master` پس از تکمیل تمرین فراموش نشود!
- پس از پایان مهلت ارسال تا ۲ روز به ازای هر روز تاخیر ۱۰ درصد از نمره مربوط به تمرین کسر خواهد شد و پس از ۲ روز نمره‌ای به تمرین تعلق نخواهد گرفت.
- بعضی از قسمت های تمرین نیاز به پیاده سازی بر روی هر چهار زبان `C#` ، `Python` ، `C++` و `Java` را دارند بعضی هم خیر. بنابراین روبروی هر سوال زبان های مورد نیاز برای پیاده سازی مشخص شده است.

۲ آماده سازی های اولیه

۱.۲ ساخت پروژه ی C#

برای ایجاد پروژه C# کافی است کد زیر را در ترمینال خود اجرا کنید:

```
mkdir cs
cd cs
dotnet new sln
mkdir cs
cd cs
dotnet new console
cd ..
dotnet sln add cs\cs.csproj
mkdir cs.Tests
cd cs.Tests
dotnet new mstest
dotnet add reference ..\cs\cs.csproj
cd ..
dotnet sln add cs.Tests\cs.Tests.csproj
```

۲.۲ قواعد نام گذاری

قواعد نام‌گذاری تمرین را از جدول ۱ مطالعه کنید.

جدول ۱: قراردادهای نام‌گذاری تمرین

Naming conventions		
Branch	Directory	Pull Request
`fb_A2`	A2	A2

* در کل یک دیرکتوری در شاخه Assignments به نام A۲ بسازید و داخل آن چهار دیرکتوری به نام های cs ، java ، python ، cpp داشته باشید و فایل های مربوط به هر زبان را داخل دیرکتوری مربوطه بگذارید.

۳ APKALA

۱.۳ کلاس Product

در اینجا کمی تبدیل فضای مسئله به فضای راه‌حل با استفاده از زبان شی‌گرای سی‌شارپ را تمرین می‌کنیم:
کلاس محصول دارای ویژگی های مولفه مشخصه(Id) ، نام(Name) ، قیمت(Price) و نمره از لحاظ محبوبیت(Rate) است که یک محصول را از دیگری تمایز میدهد.
ممکن است موجودیتی که در نظر می‌گیریم ویژگی(های) دیگری هم داشته باشد اما ما فقط ویژگی‌هایی را در نظر می‌گیریم که به حل مسئله کمک می‌کند.
گام اول: برای هر property ای که در کلاس وجود دارد getter و setter مناسب بنویسید.
گام دوم: شما باید سازنده (constructor) این کلاس را تکمیل کنید تا شیای که از کلاس ساخته می‌شود معتبر باشد یعنی هر Product ای که ساخته می‌شود لزوما دارای ویژگی های ذکر شده باشد.
پس از پیاده‌سازی صحیح سازنده‌ی این کلاس تستِ ProductConstructorTest پاس خواهد شد.
بعد از پیاده‌سازی این کلاس و پاس شدن تست‌های آن کار شما با این کلاس و فایل آن تمام شده است. دیگر نیازی به تغییر این کد نخواهید داشت.

۲.۳ کلاس Categoty

کلاس دسته بندی دارای ویژگی های مولفه مشخصه(Id) و محصولات(Products) است که یک دسته را از دیگری تمایز میدهد. برای مثال تمام گوشی های موبایل در یک دسته بندی قرار گیرند.
گام اول: برای هر property ای که در کلاس وجود دارد getter و setter مناسب بنویسید.
گام دوم: شما باید سازنده (constructor) این کلاس را تکمیل کنید تا شیای که از کلاس ساخته می‌شود معتبر باشد یعنی هر Category ای که ساخته می‌شود لزوما دارای ویژگی های ذکر شده باشد.
پس از پیاده‌سازی صحیح سازنده‌ی این کلاس تستِ CategoryConstructorTest پاس خواهد شد.
گام سوم: شما باید متد AddProduct را بگونه ای پیاده سازی کنید که ورودی آن یک محصول باشد و با فراخوانی متد آن محصول را به محصولات خود اضافه کند.
پس از پیاده‌سازی صحیح تستِ CategoryAddProductTest پاس خواهد شد.
گام چهارم: شما باید متد FilterByPrice را بگونه ای پیاده سازی کنید که دو ورودی از جنس int بگیرد و تمامی محصولات آن دسته بندی، که قیمتشان بین دو عدد ورودی است را به عنوان خروجی بازگرداند.
پس از پیاده‌سازی صحیح تستِ CategoryFilterByPriceTest پاس خواهد شد.

۳.۳ کلاس Cart

کلاس سبد خرید دارای ویژگی های نام صاحب سبد(Owner) و محصولات(Products) است که یک دسته را از دیگری تمایز میدهد.
گام اول: برای هر property ای که در کلاس وجود دارد getter و setter مناسب بنویسید.
گام دوم: شما باید سازنده (constructor) این کلاس را تکمیل کنید تا شیای که از کلاس ساخته می‌شود معتبر باشد یعنی هر Cart ای که ساخته می‌شود لزوما دارای ویژگی های ذکر شده باشد.
پس از پیاده‌سازی صحیح سازنده‌ی این کلاس تستِ CartConstructorTest پاس خواهد شد.
گام سوم: شما باید متد AddProduct را بگونه ای پیاده سازی کنید که ورودی آن یک محصول باشد و با فراخوانی متد آن محصول را به محصولات خود اضافه کند.
پس از پیاده‌سازی صحیح تستِ CartAddProductTest پاس خواهد شد.
گام چهارم: شما باید متد CalculatePrice را بگونه ای پیاده سازی کنید که با فراخوانی آن قیمت کل آن سبد خرید را بازگرداند.
پس از پیاده‌سازی صحیح تستِ CartCalculatePriceTest پاس خواهد شد.

۴.۳ کلاس Store

کلاس فروشگاه دارای ویژگی های نام (Name) ، دسته بندی ها(Categories) و سبد های خرید(Carts) است که یک فروشگاه را از دیگری تمایز میدهد.
گام اول: برای هر property ای که در کلاس وجود دارد getter و setter مناسب بنویسید.
گام دوم: شما باید سازنده (constructor) این کلاس را تکمیل کنید تا شیای که از کلاس ساخته می‌شود معتبر باشد یعنی هر Store ای که ساخته می‌شود لزوما دارای ویژگی های ذکر شده باشد.

پس از پیاده‌سازی صحیح سازنده‌ی این کلاس تستِ `StoreConstructorTest` پاس خواهد شد.

گام سوم: شما باید متد `AddCart` را بگونه ای پیاده سازی کنید که ورودی آن یک سبد خرید باشد و با فراخوانی متد آن سبد خرید را به لیست سبد های خود اضافه کند.

پس از پیاده‌سازی صحیح تستِ `StoreAddCartTest` پاس خواهد شد.

گام چهارم: شما باید متد `AddCategory` را بگونه ای پیاده سازی کنید که ورودی آن یک دسته بندی باشد و با فراخوانی متد، آن دسته بندی را به لیست سبد های خود اضافه کند.

پس از پیاده‌سازی صحیح تستِ `StoreAddCategoryTest` پاس خواهد شد.

گام پنجم: شما باید متد `Bestselling` را به گونه ای پیاده سازی کنید که با فراخوانی آن، محصولی که از همه بیشتر به فروش رفته را بازگرداند.

پس از پیاده‌سازی صحیح تستِ `StoreBestsellingTest` پاس خواهد شد.

گام ششم: شما باید متد `MostPopular` را به گونه ای پیاده سازی کنید که با فراخوانی آن، محصولی که از همه محبوب تر است، را بازگرداند.

پس از پیاده‌سازی صحیح تستِ `StoreMostPopularTest` پاس خواهد شد.

۴ Telegram

۱.۴ کلاس Person

این کلاس دارای ویژگی های مولفه مشخصه (ID) ، نام(Name) ، مخاطبین(Contacts) و چت(Chats) است. لازم به ذکر است که هر عضو، مولفه مشخصه ی منحصر به فردی دارد که او را با بقیه اعضا تمایز میدهد.

گام اول: شما باید سازنده (constructor) این کلاس را تکمیل کنید تا شیای که از کلاس ساخته می‌شود معتبر باشد یعنی هر Person ای که ساخته می‌شود لزوما دارای ویژگی های ذکر شده باشد.

پس از پیاده‌سازی صحیح سازنده‌ی این کلاس تستِ test_PersonConstructor پاس خواهد شد.

گام دوم: متد AddContact را به گونه ای پیاده سازی کنید که یک عضو(Person) بگیرد، و اگر با آن عضو قبلاً مخاطب نبوده،آن عصو را به مخاطبین خود اضافه کند.

پس از پیاده‌سازی صحیح تستِ test_PersonAddContact پاس خواهد شد.

۲.۴ کلاس Message

این کلاس دارای ویژگی های مبدا (source) ، مقصد(destination) و متن پیام(Context) است.

گام اول: شما باید سازنده (constructor) این کلاس را تکمیل کنید تا شیای که از کلاس ساخته می‌شود معتبر باشد یعنی هر Message ای که ساخته می‌شود لزوما دارای ویژگی های ذکر شده باشد.

پس از پیاده‌سازی صحیح سازنده‌ی این کلاس تستِ test_MessageConstructor پاس خواهد شد.

۳.۴ ارتباط دو کلاس

شما باید متد SendMessage را برای کلاس Person را بگونه ای پیاده سازی کنید که با گرفتن یک پیام(Message) به چت هر دو نفر آن پیام را اضافه کند. همچنین اگر دو نفر با یکدیگر مخاطب نبودن ، یکدیگر را به مخاطب خود اضافه کند.

پس از پیاده‌سازی صحیح تستِ test_SendMessage پاس خواهد شد.