

به نام خدا



درس برنامه نویسی پیشرفته

تمرین چهارم

دانشکده مهندسی کامپیوتر

دانشگاه علم و صنعت ایران

استاد مرضیه ملکی مجد

نیم سال دوم ۱۴۰۰-۱۴۰۱

مهلت ارسال :

۱۴۰۱ / ۱ / ۲۳

مبحث:

استراکچر - پراپرتی

مسئول تمارین :

آریا شهنسوار

فهرست

- | | |
|---|-----------------------------|
| ۳ | • آداب نامه تمرینات |
| ۴ | • نکات تمرین سری چهارم |
| ۵ | • تمرین ۱ . مدیریت پرواز |
| ۹ | • تمرین ۲ . فروشگاه محصولات |

آداب نامه تمرینات

- پاسخ تمامی سوالات تنها به زبان C# قابل قبول می باشد
- علیرغم اعتماد کامل تیم تی ای به شما دانشجویان عزیز ، تمامی کدهای شما با سایر دانشجویان بصورت خودکار و توسط برنامه مقایسه خواهند شد . همچنین در طول ترم ، از تمامی پاسخ های شما ارائه گرفته خواهد شد و نحوه کار تمامی بخش های هر سوال از شما پرسیده خواهد شد ، لذا از کپی نمودن کد دوستانتان خودداری کنید و تمامی پاسخ ها ، کد خودتان باشد . همچنین از آنجایی که مشورت و هم فکری با سایر دوستان بسیار کار پسندیده و مفیدی است - برخلاف کپی کردن کد (: - در صورت هم فکری با دانشجوی (دانشجویان) ، نام وی را بصورت کامنت شده در ابتدای کد خود بنویسید .
- برای ارسال تمرین در طول ترم ، در مجموع ۷ روز می توانید تاخیر داشته باشید و در صورتی که جمع تاخیر دانشجویی بیشتر از ۷ روز شود ، تمرین وی قابل قبول نخواهد بود لذا تلاش کنید تمرینات را در زمان مقرر در سامانه آپلود کنید
- برای هر تمرین در زمان ددلاین دانشجو باید درخواست خود را مبنی بر ارسال با تاخیر به تی ای ها اعلام نماید
- در تمامی تمرینات سعی شده است که سوالات ساده تر در ابتدا و سوالات دشوار تر در انتهای فایل قرار گیرند (از ساده به دشوار مرتب شده اند)
- در صورت وجود هرگونه سوال در مورد تمرینات ، سعی کنید تا جایی که امکان دارد سوال خود را در گروه پرسید چرا که شاید سوال شما ، سوال دوستان نیز باشد و دوستانتان نیز بتوانند از پاسخ سوال شما بهره ببرند .

نکات تمرین سری چهارم

- سوالات را در سامانه کوئرا و در قسمت تمرین سری چهارم آپلود نمایید .
- در تمامی سوالات باید مدیریت خطا بطور کامل و دقیق صورت گیرد و بخشی از نمره هر سوال به exception handling درست و مناسب تعلق می گیرد .
- در این سری از تمرین نیازی به نوشتن Unit test برای هیچ یک از سوالات نیست .
- در پیاده سازی کلاس ها ، دقت کنید که تمام مواردی که در سوال از شما خواسته شده است با دقت پیاده سازی شده باشد
- از آنجایی که هر سوال توسط یک تی ای طرح شده است ، تنها تی ای طراح آن سوال می تواند شما را بصورت دقیق راهنمایی کند به همین منظور طراح هر سوال در زیر نوشته شده است تا در صورت ابهام و پرسش در مورد هر سوال ، در صورتی که نیاز به پرسش سوال بصورت انفرادی در پیوی هست ، به تی ای مربوطه مراجعه بفرمایید
 - سوال ۱ : خانم سبارضی
 - سوال ۲ : خانم ریحانه شاهرخیان

تمرین ۱ . مدیریت پرواز ها

سینا برای کارش مجبور است به شهرها و کشورهای مختلف سفر کند و خرید بلیط هواپیما با توجه به شرایطش برایش سخت است. بنابراین تصمیم گرفت برنامه ای برای سامان دهی پرواز ها بنویسد.

اینام های مورد نیاز :

Currency :

شامل مقادیر : 29 = franc , 7 = dirham , 35 = pound , 30 = Dollar می باشد .

Airline :

شامل مقادیر 5 = Pars , 4 = Saha , 3 = Kish , 2 = Turkish , 1 = Mahan می باشد .

استراکچر های مورد نیاز :

Location :

این استراکت شامل کشور، استان، شهر، خیابان ، پلاک است.

Time :

این استراکت شامل سال ، ماه، روز، ساعت ، دقیقه است.

PriceInForeignCountry :

این استراکت شامل یک قیمت (از نوع double) و یک واحد پول می باشد . همچنین بوسیله سربارگذاری عملگر ~ (operator overloading) معادل آن را به واحد تومان خروجی دهد

(خروجی از جنس PriceInToman می باشد)

<https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/operators/operator-overloading>

https://www.tutorialspoint.com/csharp/csharp_operator_overloading.htm

PriceInToman :

این استراکت شامل یک فیلد قیمت از نوع double است .

کلاس های مورد نیاز :

Airplane :

این کلاس شامل تعداد ردیف ها و تعداد هر صندوقی در هر ردیف و شرکت هواپیمایی می باشد .

Flight :

این کلاس شامل فیلد های : شماره پرواز ، مبدا (location)، مقصد (location)،

، ساعت شروع پرواز و ساعت رسیدن به مقصد (time)، آیدی ، هواپیما (airplane) و قیمت پرواز (priceInToman) است . با استفاده از RegEx هنگام مقدار دهی شماره پرواز بررسی کنید که در این شماره ابتدا دو حرف انگلیسی و سپس یک عدد 4 رقمی و پس از آن یکی از علامت های @ # \$ وجود داشته باشد .

مثال :

@hw2371 یک شماره پرواز valid و hw2628 یک شماره پرواز invalid است.

نکته ۱ : تمام این اطلاعات باید به صورت property ذخیره شوند .

توجه ۱ : برای هر کدام از فیلد های کلاس باید از استراکچری که در پرانتز مربوط به آن نوشته شده است استفاده شود .

منوی برنامه :

برنامه باید دارای یک منو باشد که شامل گزینه های زیر است :

Add_flight

Sort_flight

Change_data_flight

exit

دستورهای مورد نیاز :

Add_flight :

با این دستور شماره پرواز ، مبدأ، مقصد ، ساعت شروع پرواز و ساعت رسیدن به مقصد ، آیدی ، مشخصات هواپیما (شامل تعداد ردیف ، تعداد صندلی در هر ردیف ، شرکت هواپیمایی) ، یک قیمت و یک واحد پول که بیانگر قیمت پرواز است بعنوان ورودی داده می شود . این تابع ابتدا بوسیله قیمت و واحد پول داده شده یک شی از استراکت PriceInForeignCountry می سازد و سپس با استفاده از متد موجود در این استراکت ، قیمت پرواز را به تومان بدست آورد .

حال برای هر یک از فیلد های پرواز با استفاده از اطلاعاتی که از ورودی گرفته شده یک شی از استراکت و کلاس های مورد نظر ساخته و سپس با استفاده از مفاهیم property آنها را مقداردهی میکند و یک پرواز به لیست پرواز ها اضافه میشود.

در این قسمت با استفاده از سربارگذاری ، عملگر == را به گونه ای پیاده سازی کنید که بتوان برابری دو شی از کلاس flight با نوع داده یکسان را بررسی کرد . یعنی پس از ساخت شی جدید پرواز با همه ی شی های از پیش ساخته شده چک کند و در صورت یکسان بودن خروجی مناسب نشان دهد.

Sort_flight :

این دستور شامل 2 قسمت است :

- **سورت بر اساس قیمت :** باید از سینا یک قیمت پیشنهادی دریافت کند و پرواز هایی که با کمترین تفاوت قیمت است را بالاتر از پرواز هایی با تفاوت قیمت بیشتر، نشان دهد.
- **سورت بر اساس مبدأ و مقصد :** باید پرواز هایی که هم مبدأ و هم مقصد یکسانی با ورودی سینا دارند، را نشان دهد.

توجه ۲ : برای نشان دادن خروجی های این دو قسمت از رنگ بندی های مناسب استفاده کنید تا اطلاعات خوانا تر شوند.

Change_data_flight :

در این دستور سینا میتواند یکی از فیلد های پرواز با آیدی مورد نظرش را تغییر دهد.

یعنی پس از انتخاب این گزینه باید سینا یکی از گزینه ها را انتخاب کرده سپس اطلاعات جدید را وارد کند و با استفاده از مفاهیم property آن را تغییر دهد.

Exit :

با استفاده از این قسمت برنامه به پایان برسد.

نکته ۲ : اطلاعات مکان ها در فایلی به نام place.txt ذخیره شود.

نکته ۳ : اطلاعات زمان در فایلی به نام time.txt ذخیره شود.

نکته ۴ : اطلاعات هواپیماها در فایلی به نام airplane.txt ذخیره شود.

نکته ۵ : اطلاعات پروازها در فایلی به نام flight.txt ذخیره شود در صورتی که پس از قطع برنامه و دوباره ران گرفتن از آن ، اطلاعات قبلی از بین نرود.

راهنمایی : برای انجام اینکار میتوانید به تمام استراکت ها و کلاس های یک پرواز آیدی مشترک بدهید

تمرین ۲ . فروشگاه محصولات

در این سوال هدف این است که شما فروشگاه محصولات با ویژگی های داده شده طراحی کنید. سامانه این فروشگاه به طور کلی از سه بخش محصولات، سبد خرید و دسته بندی تشکیل شده است که در ادامه توضیحات مربوط به هر کدام از این بخشها آورده شده است.

کلاس Product :

این کلاس دارای ویژگی های آیدی، نام، قیمت، امتیاز از لحاظ محبوبیت، کارخانه سازنده است. تمامی این ویژگی ها باید به صورت property پیاده سازی شوند. متغیر مربوط به کارخانه سازنده باید به گونه ای پیاده سازی شود که در خارج از محیط کلاس نتوان آن را مقداردهی کرد و تنها بتوانیم مقدار آنرا بخوانیم و توانایی تغییر آن را نداشته باشیم. همچنین این متغیر باید در تابع سازنده باتوجه به مقدار آیدی مقداردهی شود. در این کلاس تمامی ویژگیها به جز کارخانه سازنده باید توسط تابع سازنده دریافت و مقداردهی شوند. همچنین ویژگی کارخانه هم باتوجه به مقدار آیدی مقداردهی میشود. در صورتی که مقدار آیدی عددی بین 1 تا 5 باشد، کارخانه سازنده محصول a، در صورتی که آیدی بین 5 تا 10 باشد، کارخانه سازنده محصول b و در صورتی که آیدی بیشتر از 10 باشد کارخانه سازنده محصول باید c مقداردهی شود.

توجه ۱ : برای هر property باید getter, setter مناسب نوشته شود.

توجه ۲ : آیدی هر محصول باید منحصر به فرد باشد.

کلاس Category :

این کلاس دارای ویژگی های آیدی و نام و لیست محصولات است . محصول دادهای از نوع Product است . برای بخش آیدی و نام محصول شما باید داده ای از نوع enum را پیاده سازی کنید .

Enum گفته شده باید شامل اعضای زیر و آیدیهای زیر باشد.

Phone = 1, car = 2, watch =3,

T-shirt = 4, Laptop = 5, Tablet = 6,

Charger = 7, Glass = 8, Robot= 9

در تابع سازنده این کلاس متغیرهای آیدی و نام، دریافت و مقداردهی میشوند.

متد AddPrductCategory: این متد به عنوان ورودی لیستی از محصولات از نوع Product را دریافت میکند . سپس این محصولات را به لیست محصولات شی ساخته شده از کلاس Category اضافه میکند.

متد FilterByPrice : این متد دو ورودی دارد . کمترین قیمت و بیشترین قیمت . شما باید به عنوان خروجی تمامی محصولاتی که قیمتشان بین دو عدد داده شده است را برگردانید و سپس در main برنامه آنها را به کاربر نمایش دهید.

متد ShowSupply : این متد ورودی ندارد . با فراخوانی این متد شما باید لیست مرتب شده ای (سورت شده) از محصولات متعلق به این Category را باتوجه به قیمتشان به کاربر نمایش دهید

استراکچر People :

این استراکچر باید دارای فیلدهای نام ، نام خانوادگی ، سن ، جنسیت و شماره تماس باشد . شماره تماس باید حتما 11 رقم باشد و حتما باید با رقم های 09 شروع شود . این کلاس ۳ متد به نام EditProfile دارد. اولین متد برای تغییر نام و نام خانوادگی است که دو ورودی از نوع string دارد. دومین متد برای تغییر سن است که ورودی آن از نوع اینت میباشد و سومین متد دارای یک ورودی از نوع string است که برای تغییر شماره تلفن است. ورودی ها در این ۳ متد درواقع مقدار جدید نام ، نام خانوادگی ، سن و شماره تماس هستند و با صدا زدن آنها باید مقادیر آپدیت شوند.

نکته ۱ : جنسیت باید از نوع enum باشد که به صورت مقابل است :

male = 1, female = 2, others = 3

کلاس Cart :

این کلاس باید دارای فیلدهای صاحب سبد از نوع استراکت People و محصولات که لیستی از Product است باشد. در تابع سازنده باید فیلد صاحب سبد دریافت و مقداردهی شود.

متد AddProductToCart:

این متد به عنوان ورودی لیستی از محصولات را دریافت خواهد کرد و آنها را به سبد خرید اضافه خواهد کرد.

متد DeleteProduct :

این متد به عنوان ورودی لیستی از محصولات را دریافت میکند و آنها را از سبد خرید حذف میکند و در انتها در هر خط اسم محصولات باقی مانده در سبد خرید به همراه قیمت آن را چاپ میکند.

متد CalculatePrice :

این متد باید در هر خط نام محصول به همراه قیمت آن را بنویسد. در انتها قیمت کل (جمع تمام قیمت های محصولات) هم نوشته شود.

در مین برنامه ابتدا باید منویی نمایش داده شود و کاربر در هر مرحله از بین گزینه های نمایش داده شده یکی را انتخاب کند. در صورتی که ورودی کاربر جزو موارد گفته شده نبود، باید این منو دوباره به کاربر نمایش داده شود و این کار تا زمانی ادامه یابد که یا کاربر گزینه Exit یا Back را بزند و یا گزینه درست را وارد کند.

منوی ابتدایی :

در ابتدا باید منویی شامل سه گزینه Category, Cart, Exit به کاربر نمایش داده شود. با انتخاب هریک از گزینه های Category, Cart باید منوی مربوط به همان بخش به کاربر نمایش داده شود و در صورتی که کاربر Exit را وارد کرد باید از محیط برنامه خارج شوید. همان طور که گفته شد تا زمانی که ورودی صحیح توسط کاربر وارد نشده باشد باید این منو به کاربر نمایش داده شود.

در صورتی که Category وارد شد، ابتدا باید از کاربر بخواهید تا نام Category مورد نظر خود را وارد کند.

پس از آن منویی شامل گزینه های AddProductCategory و FilterByPrice و ShowSupply نمایش داده شود. با وارد کردن گزینه AddProductCategory ابتدا از کاربر تعداد کالاهایی که قصد اضافه کردن آنها را دارد گرفته شود و سپس مشخصات مربوط به هریک از کالاها دریافت شود و به لیست کالاهای مربوط به این Category اضافه شوند.

در صورت انتخاب FilterByPrice دو عدد به عنوان ورودیهای این متد گرفته شود و سپس باتوجه به این ورودیها، لیست کالاهای مربوط به این Category که قیمتشان در بازه داده شده است چاپ شود. بدین صورت که در ابتدا نام و آیدی Category چاپ شود و سپس محصولات نمایش داده شوند. همچنین این منو باید یک گزینه با عنوان Back برای برگشتن به منوی اصلی برنامه داشته باشد.

در صورت انتخاب ShowSupply همان طور که در توضیحات متد گفته شده بود، پس از چاپ نام و آیدی Category باید لیست مرتب شده ای از محصولات را باتوجه به قیمتشان چاپ کنید. همچنین این منو باید یک گزینه با عنوان Back برای برگشتن به منوی اصلی برنامه داشته باشد.

در صورتی که هیچ یک از گزینه های گفته شده و کلمه Back توسط کاربر وارد نشد، باید منو دوباره به کاربر نمایش داده شود تا زمانی که ورودی درست توسط کاربر وارد شود.

در صورتی که در منوی ابتدایی برنامه، گزینه Cart وارد شد، در ابتدا باید مشخصات فرد شامل نام، نام خانوادگی و سن و شماره تماس فرد گرفته شود. توجه شود که همان طور که گفته شد، شماره تماس فرد باید با فرمت توضیح داده شده در قسمت های قبل باشد و در صورتی که از فرمت گفته شده پیروی نمی‌کرد، باید از کاربر درخواست شود تا دوباره شماره تماس را وارد کند.

پس از دریافت مشخصات فرد، باید منویی شامل گزینه های

AddProductToCart

CalculatePrice,

Back

DeleteProduct

EditProfile

به کاربر نمایش داده شود. در صورتی که کاربر ورودی نادرستی وارد کند باید این منو دوباره به کاربر نمایش داده شود تا زمانی که کاربر یکی از گزینه های مجاز منو را انتخاب کند.

در صورتی که کاربر AddProductToCart را انتخاب کند، باید اطلاعات مربوط به کالا از کاربر گرفته شود و در انتها این محصول به سبد خرید فرد اضافه شود. سپس منوی قبل را دوباره نمایش دهد.

در صورتی که کاربر CalculatePrice را انتخاب کرد، باید لیست نام تمام کالاهایی که کاربر سفارش داده به همراه قیمت آنها نمایش داده شود. در انتها قیمت نهایی هم نمایش داده شود. با زدن گزینه Back منوی قبلی دوباره به کاربر نمایش داده شود.

در صورتی که کاربر DeleteProduct را انتخاب کرد، باید اطلاعات مربوط به کالاهایی که کاربر قصد حذف دارد از او گرفته شود و پس از حذف کالاها از سبدخرید لیست نام تمام کالاهایی که کاربر سفارش داده به همراه قیمت آنها نمایش داده شود. با زدن گزینه Back منوی قبلی دوباره به کاربر نمایش داده شود.

در صورتی که کاربر EditProfile را انتخاب کرد سه گزینه ی

Age

name and family name

phone number

به کاربر نمایش داده شود و طبق انتخاب کاربر و توضیحات داده شده در متد مربوطه اطلاعات کاربر ویرایش شود. با زدن گزینه ی Back منوی قبلی دوباره به کاربر نمایش داده میشود.

موفق باشید :