تمرین نهایی :

- مقدمه :

مدیرعامل اوبر اقای داراخسروشاهی پس از سفر به ایران از دیدن استارت آپ هایی مانند باما ، کارنامه و دیوار و ... به وجد آمده و قصد دارد که این سیستم را برای اوبر نیز راه اندازی کند.
در واقع سرویس مورد نظر او آگهی فروش ماشین است . پس شما باید این کار را به نحو احسن برای او انجام دهید .

اندپوینت های این سوپراپ! نوشته شده اند . شما باید تست این اندپوینت ها را ارائه دهید تا محصول اماده ی لانچ در production باشد . در این مرحله تست نویسی باید از کتابخونه ی Pytest استفاده کنید .

- بخش اول : تست نویسی

تست هایی که باید بنویسید در فایل تست app cars موجود اند . وباید انها را تکمیل نمایید .
پیش نیاد های تست:
شما موظف به پیاده سازی fixture ای برای cars هستید که این fixture باید ۱۰ تا ماشین را بسازد و به شما برگرداند .
شما موظف به استفاده از fixture api_client هستید که برای شما اماده شده است .

تست ها:
تست اول :
تست کنید که api create ماشین ها درست کار میکند ! (از fixture api_client استفاده کنید)
تست دوم :
در فایل karname/api/v1/mixins/ ما مجبور به override کردن میکسین create شده ایم ،
مطمئن شوید که این میکسین به درستی کار میکند و اگر دیتا با شرایط این میکیسن سازگار نباشد ، ماشینی ساخته نمیشود !
تست سوم :
اندپوینت list cars را تست نمایید (در این تست حتما از fixture cars و SnapShot استفاده کنید در غیر این صورت این تست پذیرفته نیست)
تست چهارم :
یک فیلترست روی برند های ماشین ساخته شده است ، تست کنید که این فیلترست به درستی کار میکند(حتما از snapshot استفاده کنید)
تست پنجم :
اندپوینت delete را مورد تست قرار دهید.
تست ششم :
اندپوینت update را تست کنید
در این تست باید متد patch رو تست کنید (نیازی به تست متد put نیست)
در کامنتی در این تست فرق میان متد های patch , put را در rest شرح دهید .

- بخش دوم : استفاده از مکانیسم caching

با توجه به درخواست های بسیار مکرر در این سایت ، متدی در مدل ماشین ها ساخته شده به نام get_price .
برای ساده سازی مسئله فرض میکنیم که ماشین های با برند یکسان و سال ساخت یکسان قیمت برابری دارند . یعنی مثلا ماشین های بنز سال ۲۰۲۰ همگی یک قیمت مشخص دارند . یا مثلا تمام ماشین های پورشه ی سال ۱۹۹۰ قیمتی یکسان باهم دارند .
این متد وظیفه دارد که برند و سال ساخت ماشین را بگیرد و قیمت را محاسبه کند . قیمت ها بر اساس فرمول زیر محاسبه میشوند :

قیمت = سال ساخت * تعداد حروف برند (به انگلیسی) * ۱۰۰
(مثلا ماشین بنز سال ۲۰۲۴ به قیمت ۲۰۲۴ * ۳ * ۱۰۰ به فروش میرسد)
(یا مثلا پورشه سال ۱۹۹۹ به قیمت ۱۹۹۹ * ۶ *۱۰۰ به فروش میرسد)

حالا با توجه به این که تقاضا برای محاسبه ی قیمت های خودرو بسیار بسیار بالاست ، تصمیم گرفتیم این محاسبات را cache کنیم ، به این منظور از property های موجود در کتابخانه functool استفاده میکنیم .
این متد را به همراه کش پیاده سازی کنید . همچنین راهی برای این پیدا کنید که متوجه شوید که چند بار این cache در این متد miss , hit شده است .

- بخش امتیازی : روش Given-When-Then در نامگذاری تست ها :

امتیازی :
این پست اقای fowler از پیشگامان صنعت نرم افزار را مطالعه کنید و نام تست های برنامه را به فرمتی که در این پست گفته شده در بیاورید (فراموش نکنید که در pytest لازم است حتما نام متد تست شما با عبارت test_ شروع شود)
پست : https://martinfowler.com/bliki/GivenWhenThen.html