

پروژه اول درس ساختمان داده‌ها

سامان‌دهی نمایندگی‌های خودروسازی

هدف از انجام این پروژه، ایجاد سامانه‌ای به منظور سامان‌دهی نمایندگی‌های یک شرکت خودروسازی است. این سامانه برای استفاده مشتریان از خدمات و نیز استفاده مدیران شرکت برای گزارش‌گیری از نمایندگی‌ها پیاده‌سازی می‌شود. در این سامانه، مدیران شرکت می‌توانند خدماتی را برای نمایندگی‌ها تعریف کنند. هر خدمت شامل نام خدمت، مدل خودرو، توضیحات برای مشتری، توضیحات فنی برای نمایندگی‌ها و هزینه‌های شرکت خواهد بود. ساختمان داده پیشنهادی برای نگهداری این خدمات، لیست پیوندی تعمیم‌یافته است و تمام نمایندگی‌ها به این لیست پیوندی دسترسی دارند. هر نمایندگی یک یا چند خدمت اصلی را به مشتریان ارائه می‌دهد. از طرفی ممکن است شرکت بخواهد خدمات را تا هر تعداد سطح ممکن دسته‌بندی نماید. در نتیجه هر خدمتی ممکن است شامل زیرخدمات دیگری نیز باشد. برای مثال، ممکن است خدمت تعمیرات بدنه شامل خدمات تعویض سپر، صاف‌کاری، خدمات لاستیک و خدمات د/خلی باشد. به طور مشابه، خدمات لاستیک، خود شامل تعویض لاستیک و پنچرگیری باشد.

همچنین این امکان وجود دارد که یک خدمت اصلی مشخص توسط بیش از یک نمایندگی ارائه شود. بنابراین در صورتی که نمایندگی یکی از خدمات اصلی ارائه شده خود را حذف نماید، فقط در صورتی که هیچ نمایندگی دیگری آن خدمت را ارائه ندهد، خدمت مورد نظر و تمامی زیرخدمات‌های آن حذف می‌شوند و در غیر این صورت نباید حذف شود.

درخواست افزودن و حذف خدمات توسط دستورات موجود در جدول زیر قابل انجام است:

دستور	درخواست
add service <Service_Name>	اضافه کردن خدمت جدید به لیست خدمات اصلی
add subservice <Subservice_Name> to <Service_Name>	اضافه کردن زیرخدمت جدید به یکی از خدمات موجود (اصلی یا زیرخدمت)
add offer <Service_Name> to <Agency_Name>	ارائه یک خدمت اصلی توسط یک نمایندگی
delete <Service_Name> from <Agency_Name>	حذف یک خدمت (و زیرخدمات‌ها در صورت لزوم) از یک نمایندگی

در این سامانه، علاوه بر لیست پیوندی تعریف شده برای انواع خدمات، لیست پیوندی دیگری شامل نام نمایندگی‌ها وجود دارد. مشتریان می‌توانند با انتخاب یک نمایندگی از این لیست و یک خدمت یا زیرخدمتی از لیست پیوندی خدمات ارائه شده توسط همان نمایندگی، سفارش استفاده از خدمت برای نمایندگی انتخاب شده را به همراه میزان فوریت انجام آن (به ترتیب: ضروری، مورد نیاز، عادی) به سامانه ارسال نمایند.

هر نمایندگی برای نوبت‌دهی سفارش‌های خود از یک صف اولویت‌دار استفاده می‌کند و اولویت‌ها نیز بر اساس فوریت انجام و زمان ارسال سفارش مشخص می‌شوند. از بین دو سفارش با فوریت یکسان، سفارشی که زودتر ارسال شده باشد، دارای اولویت بیشتری است. نمایندگی باید صف سفارش‌ها را بر اساس اولویت‌های تعیین شده مشاهده نمایند. برای پیاده‌سازی صف سفارش‌های هر نمایندگی، از یک **Max Heap** استفاده کنید. در هر لحظه و پس از اضافه شدن هر سفارش، ریشه سفارش با بالاترین اولویت است. برای نمایش لیست سفارش‌ها نیز در هر مرحله به ترتیب سفارش با بالاترین اولویت از ساختار حذف و نمایش داده می‌شود تا زمانی که کل صف خالی شود. با توجه به این توضیحات، سامانه دارای قابلیت‌های زیر نیز خواهد بود:

درخواست	دستور
اضافه کردن نمایندگی	add agency <Agency_Name>
مشاهده لیست نمایندگی‌ها	list agencies
مشاهده لیست خدمات اصلی همراه با تمامی زیرخدمات	list services
مشاهده لیست زیرخدمات یک خدمت مشخص	list services from <Service_Name>
سفارش استفاده از یک خدمت به یک نمایندگی مشخص	order <Service_Name> to <Agency_Name> by <Customer_Name> with <Immediacy_Level>
مشاهده صف سفارش‌ها و انجام هر یک (به ترتیب اولویت)	list orders <Agency_Name>

دقت داشته باشید که نیازی به تعریف و بررسی دسترسی‌های مشتری، نمایندگی و مدیر نیست؛ بلکه صرف وارد کردن دستورات به سیستم بیانگر تفاوت نقش‌های کاربران سامانه است. همچنین نیازی به ذخیره‌سازی نام مشتریان در لیست جداگانه‌ای وجود ندارد.

نکات پیاده‌سازی

در پیاده‌سازی این پروژه به موارد زیر پیاده‌سازی شود:

۱. همه ساختمان داده‌های مورد نیاز با توجه به تعریف پروژه باید پیاده‌سازی شوند. استفاده از ساختمان داده‌های آماده مجاز نیست. فقط استفاده از آرایه (ساده، **ArrayList** یا **Vector**) مجاز است. صف، پشته،

لیست پیوندی، درخت یا هر ساختمان داده دیگر مورد استفاده باید پیاده‌سازی شود. (به عنوان درس توجه کنید!)

۲. پیاده‌سازی باید به صورت **تک نفره** باشد و محدودیتی برای زبان پیاده‌سازی وجود ندارد. اما دقت کنید که استفاده از ساختمان داده‌های آماده و به صورت کتابخانه مجاز نیست.

۳. در حین انجام پروژه، بحث و بررسی بین دانشجویان آزاد است اما هر دانشجو موظف است به تنهایی پروژه را انجام دهد و در هنگام تحویل حضوری، دانشجو باید به تمام جزئیات پیاده‌سازی کد کاملاً مسلط باشد. در مورد قسمت‌هایی از کد و نحوه عملکرد برنامه نیز از دانشجو سوال خواهد شد. همچنین با مواردی که تقلب و کپی‌کردن تشخیص داده شوند، برخورد جدی خواهد شد (برای تشخیص درصد شباهت کدها از [سامانه Moss](#) استفاده می‌شود).