



پروژه ی چهارم برنامه سازی پیشرفته

دکتر سیدابوالقاسم میرروشندل

دفترچه تلفن

طراح: امیرحسین مهدی نژاد

داستان پروژه

پدربزرگ از محدودیت های بخش مخاطبین گوشی قدیمی اش می نالد!
از شما می خواهد برنامه ای بنویسید که دفترچه تلفنی با قابلیت های مطلوبش داشته باشد.

هدف از پروژه

هدف از انجام این پروژه آشنایی بیشتر با شی گرای است.

در این پروژه می بایست یک برنامه ی contacts را طبق ساختاری که در ادامه معلوم شده است، طراحی کرده و سپس از این ساختار طراحی شده در console استفاده کنید. لازم است کلاس ها و توابعی که در زیر آمده است، طبق ساختار مطرح شده پیاده سازی شوند و صحیح بودن پیاده سازی ساختار زیر نیز الزامی است.

یک کلاس contact ایجاد کنید که شامل موارد زیر باشد:

۱. نام، شماره تلفن، تاریخ تولد، محل سکونت، جنسیت، گروه، ایمیل
۲. شامل متدهای سازنده دارای ۲ آرگومان (نام و شماره) به بالا
۳. متدهای getter و setter مناسب
۴. متدی برای پیدا کردن سایر مخاطبین هم گروه با مخاطبی خاص
۵. متد findNeighbour برای پیدا کردن مخاطبین همسایه با مخاطبی خاص

یک کلاس AddressBook تعریف نمایید که دارای:

۱. یک آرایه یا Arraylist از contactها
۲. یک متغیر برای تعداد contactها
۳. متدهای حذف و اضافه کردن یک مخاطب بر اساس نام یا شماره
۴. متد نمایش لیست contactها

یک کلاس group درست کنید که:

۱. در بر گیرنده ی گروه های favorites, friends, family, ... قابل حذف، اضافه و ویرایش
۲. هر مخاطب متعلق به حداقل یک گروه باشد (اگر عضو گروه خاصی نیست در واقع عضو یک گروه unsigned یا پیشفرض باشد) به طوری که در صورت حذف گروه مخاطبین آن از بین نروند بلکه unsigned شوند
۳. یک آرایه یا Arraylist برای groupهای ساخته شده به همراه متغیر نشان دهنده ی تعداد contactهای هر گروه

یک کلاس location بسازید به طوری که:

۱. شامل کشور، استان، شهر، محله، کوچه باشد
۲. برای هر مکان خاص شامل مختصات به صورت x, y باشد که در همین کلاس توسط متد دیگری برای تشخیص همسایگی به کار می رود
۳. یک متغیر double برای شعاع همسایگی
۴. یک متغیر برای تعداد locationهای در حال استفاده

برنامه ی شما باید قابلیت ساخت دفترچه تلفن جدید، اضافه کردن و ویرایش و حذف مخاطبین به صورت تکی و دسته ای، پیدا کردن مخاطبین همسایه، اضافه کردن مخاطبین به صورت تکی یا دسته ای به گروه های تعریف شده و همچنین اضافه کردن، ویرایش و حذف گروه ها را داشته باشد.

در تمامی بخش ها، وجود یک متد toString برای نمایش ویژگی های کلاس ساخته شده الزامی است.

ورودی و خروجی

می بایست یک رابط کاربری مناسب برای این برنامه درست کنید که راحتی کار با آن از الزامات پروژه است.

کد نمونه

توجه کنید این کد، تنها بخشی از عملکرد توابع را پوشش می دهد و لازم است شما تمامی آنچه خواسته شده را با جزئیات پیاده سازی کنید. بدیهی است رعایت وراثت و محدودیت های دسترسی به متدها (private, protected, ...) نیز باید مورد توجه قرار بگیرد.

Sample code

```
Contact contact1 = new Contact("Jack", "+989122222222")
Contact contact2 = new Contact("Ben", "091299999999", 1998 /*birthday*/);
AddressBook addressBook = new AddressBook();
addressBook.add(contact1);
addressBook.add(contact2);
int contactCount = addressBook.getContactCount(); //return 2
addressBook.remove("Ben");
addressBook.add(new Contact("Rose", "9128888888"));
addressBook.displayAll();
```

نکات تکمیلی

- پیاده سازی یک رابط کاربری مناسب برای این برنامه و راحتی کار با آن از الزامات پروژه است اما نمایش گرافیکی رابط کاربری و دفترچه تلفن، **نمره ی مثبت** دارد.
- پیاده سازی عملکرد این ساختار، به سلیقه ی شما بستگی دارد ولی باید تمام توابع نام برده شده مورد استفاده قرار گیرند. (در صورت نیاز، کلاس های بیشتری نسبت به آنچه گفته شد ایجاد کنید)

نحوه ی ارسال پروژه

- هیچ توجیهی در استفاده از زبان های برنامه نویسی دیگر وجود نداشته، و برابر با تحویل ندادن پروژه، **نمره ی صفر** به دانشجو تعلق می گیرد.
- پروژه تنها بصورت انفرادی قابل انجام بوده و در صورت وجود هر گونه تشابه بین دو کد یا عدم تسلط به روند برنامه در زمان تحویل، **نمره ی منفی** به دانشجویمان داده می شود.
- برای تسلط بهتر و کسب نمره خوب در این درس به انجام پروژه های اضافه بر پروژه های ترم اقدام کنید.
- زودتر از deadline پروژه از طریق quera اقدام به ارسال نموده و تأیید گرفته شده را بلافاصله چک کنید.
- رعایت فاصله ی کد از سر خط و همچنین comment نویسی مناسب **الزامی** است.