

اعداد گویا

هدف

تو این سوال قصد داریم که کلاسی برای پیاده سازی اعداد کسری طراحی کنیم می دونیم که اعداد کسری شامل دو قسمت صورت و مخرج ه که هردو عدد صحیح هستن پس نیاز داریم که کلاس مون شامل این دو ویژگی باشن. توجه داشته باشید که این ویژگی ها باید بصورت private تعریف شوند.

توابع جهت پیاده سازی

تابع constructor

قراره که دو تابع سازنده داشته باشیم که هرکدوم رو توضیح میدیم :

```
Rational();
```

این تابع سازنده ورودی نداره و فقط قراره چاپ کنه که کلاس مون ساخته شد و مقدار متغیر صورت رو 0 و متغیر مخرج رو 1 قرار بده.

```
Rational(int _x , int _y);
```

این تابع سازنده دو عدد صحیح به عنوان صورت و مخرج از مون میگیره و قراره متغیر صورت و مخرج مون رو مقداردهی کنه و چاپ کنه که کلاس ساخته شد.

اگر دقت کرده باشید ابتدای سوال گفتیم که متغیر های صورت و مخرج رو بصورت private باید تعریف بشه دلیل اینکار اینه که خودمون با توابعی که تعریف میکنیم بتونیم دسترسی یک فرد از خارج رو محدود نگه داریم و کنترل کنیم. در ادامه بیشتر با این موضوع آشنا می شیم.

تابع های getter

اگر متغیری که مقدار صورت ما توی اون ذخیره شده x و متغیر مخرج اسمش y باشه نیاز به تعریف چنین توابعی برای دریافت محتوای این توابع داریم.

```
int getX();  
int getY();
```

تابع های setter

همینطور که گفتیم درواقع ما با تعریف توابع برای دسترسی به متغیر هامون میخوایم دسترسی مستقیم یک فرد خارجی رو به متغیر هامون کم کنیم و جلوی یک سری مشکلات رو بگیریم. برای مثال می دونیم که کسر با مقدار صفر نداریم پس توی تابع های setter مون باید به این مورد دقت کنید و اون رو کنترل کنید. ولی اگر متغیر هامون بصورت public ساخته می شدند هرکس میتونست محتوای متغیر رو هرچی دلش میخاد وارد کنه و باعث به خطا خوردن برنامه ما بشه!

توابعی که نیاز داریم پیاده سازی کنیم :

```
void setX(int _x);
```

- این تابع مقدار صورت رو می گیره و در متغیر صورت کلاس مون ذخیره می کنه.

```
void setY(int _y);
```

- این تابع مقدار مخرج رو می گیره و در متغیر مخرج کلاس مون ذخیره می کنه.

```
void setRational(int _x ,int _y);  
void setRational(string s);
```

- دو تا تابع بالا اسم شون یکی هست ولی مقادیر مختلفی به عنوان ورودی می گیرند. به این مفهوم

Function overloading میگویند:

در واقع وقتی که ما میخوایم چند نسخه از یک تابع داشته باشیم این مفهوم خودشو نشون میده که میتونیم اسم تابع هامون یکی باشه ولی ورودی مون متفاوت باشه!
اگر توجه کنید توی تابع constructor مون هم مثالی از Function overloading داشتیم. الان توی هر دو تابع بالا قراره صورت و مخرج کلاس مون رو مقدار دهی کنیم که تو تابع اول به صورت دو پارامتر جدا به تابع داده میشن ولی تو تابع دوم صورت و مخرج به صورت یک رشته مثل :

x/y

داده همیشه که باید پردازش کنیم و صورت و مخرج رو مقدار دهی کنیم.

نکته مهم: فراموش نکنید که ورودی هارو توی توابع setter کنترل کنید به طور مثال اجازه ورود مقدار صفر برای مخرج رو ندید!

توابع ضرب، جمع و تفریق

```
Rational add(Rational r);
```

- این تابع یک متغیر از جنس کلاس مون میگیره و با محتوای فعلی کلاس مون عملیات جمع رو انجام میده و در نهایت کلاس نتیجه رو به عنوان خروجی return می کنه.

```
Rational subtract(Rational r);
```

- این تابع یک متغیر از جنس کلاس مون میگیره و با محتوای فعلی کلاس مون عملیات تفریق رو انجام میده و در نهایت کلاس نتیجه رو به عنوان خروجی return می کنه.

```
Rational multiply(Rational r);
```

- این تابع یک متغیر از جنس کلاس مون میگیره و با محتوای فعلی کلاس مون عملیات ضرب رو انجام میده و در نهایت کلاس نتیجه رو به عنوان خروجی return می کنه.

- با توجه به setter هایی که بالا طراحی کردین باید حواستون باشه موقع این عملیات ها به خطایی نخوریم ;)

نکات قابل توجه:

- پیاده سازی توابع بالا ضروری ست.
- پیاده سازی main برنامه اختیاری هست ولی سعی کنید توابع های خودتون رو حداقل یک بار توی main استفاده کنید تا از کارکرد صحیح اون ها برای کیس های مختلف مطمئن باشید.

زمان شیء است

سپهر تصمیم گرفته که از شیء گرابی که یاد گرفته است کلاسی برای زمان بسازد ولی خب چون تازه کار است، نیاز به کمک شما دارد به او کمک کنید.

هدف

طراحی کلاسی به نام Time برای ذخیره کردن زمان است.

توابع جهت پیاده سازی

- فیلد های مناسب برای ذخیره سازی ساعت و دقیقه انتخاب کنید
- یک سازنده که ورودی آن دقیقه و ساعت باشد
- یک سازنده که ورودی آن تنها ساعت باشد(در این حالت دقیقه باید صفر در نظر گرفته شود)
- متد های getter برای ساعت و دقیقه
- متد های setter برای ساعت و دقیقه
- یک متد برای مقایسه که ورودی آن از جنس کلاس Time است
- متدی که بر اساس زمان رشته ای برگرداند که بگوید چه وقتی از روز است.(از ابتدای یک روز تا ساعت 11:59 صبح، 12:00 ظهر، از 12:01 تا 16:59 بعد از ظهر، از 17:00 تا 19:59 عصر و بعد از آن شب محسوب می شود)

نکات

هر جا قرار است زمان ذخیره شده تغییر کند، باید بررسی شود که اعداد وارد شده قابل قبول باشد و در غیر این صورت اجرای برنامه خاتمه یابد. برای آزمودن امکانات این کلاس یک برنامه ساده در تابع main بنویسید. دقت کنید که فیلدهای کلاس نباید به صورت عمومی (public) قابل دسترسی باشند.

موبایل شاپ جیبی

حقیقتی که در ذهن شما شکل می گیرد روزی تبدیل به واقعیت خواهد شد. پس این یک داستان واقعی است

مهتی چند وقتی بود که باتری گوشیش خراب شده بود و به فکر خرید گوشی جدید بود. سپهر هم که اینو میدونست رفیق خوبی بود ولی پول نداشت تصمیم گرفت که یک برنامه برای مهتی بنویسد که با آن بتواند اطلاعات گوشی های مختلف را در آن ذخیره کند و حتی بر اساس موارد خاصی آن لیست را فیلتر کند. سپهر هم به این نتیجه رسید که بهترین کار برای نوشتن این برنامه استفاده از شی گرای است و خب نشست و کلاس های مورد نیاز خود را روی کاغذ نوشت ولی حالا از شما می خواهد که در پیاده سازی این برنامه برای مهتی به او کمک کنید.

برنامه ما از 2 کلاس تشکیل شده است یکی کلاس موبایل و دیگری کلاس فروشگاه موبایل

کلاس موبایل:

این کلاس دارای فیلد های مختلفی مانند:

- مدل موبایل , مثال: Galaxy Note 10
- برند موبایل , مثال: Samsung
- قیمت , مثال: 1300000
- رزولوشن دوربین عکاسی , مثال: 48
- رنگ , مثال: black

کلاس فروشگاه موبایل:

- کلاس دارای یک وکتور از جنس کلاس موبایل هست که اطلاعات موبایل ها را در خود نگه می دارد.
- و توابعی برای جست و جو و فیلتر کردن گوشی ها
- همچنین تابعی برای اضافه کردن موبایل به این فروشگاه که ورودی آن از جنس کلاس موبایل می باشد.

همونطور که متوجه شدید وظیفه کلاس فروشگاه ما جست و جو بین لیست موبایل هاست ما باید چند تابع خصوصی داشته باشیم:

- فیلتر کردن بر اساس برند موبایل
- فیلتر کردن بر اساس رنگ موبایل
- فیلتر کردن بر اساس رزولوشن دوربین موبایل
- فیلتر کردن بر اساس قیمت موبایل

و توابع عمومی به شکل زیر باید داشته باشیم:

```
void search(string type , vector<string> q);
```

- این تابع یک ورودی نوع جست و جو از ما میگیرد و یک ورودی وکتوری از مقادیری که باید فیلتر بشه. به طور مثال اگر ما داریم روی برند گوشی ها جست و جو انجام میدیم وقتی وکتوری از برند های SAMSUNG,APPLE به تابع مون بدیم انتظار داریم همه موبایل هایی که با این برند ها توی فروشگاه ما موجوده رو چاپ کنه برامون.

```
void search(string type , int q);
```

این تابع یک ورودی نوع جست و جو از ما میگیرد و یک ورودی عدد صحیح.

- بدین صورت که وقتی که ما روی قیمت درحال جست و جو هستیم، موبایل هایی به مبلغ کوچکتر مساوی مبلغ ما داریم چاپ بشه
- اگر جست و جو براساس رزولوشن دوربین بود، گوشی هایی که رزولوشن انتخابی ما رو دارند چاپ بشه.

نکته: برای آزمودن امکانات این کلاس یک برنامه ساده در تابع main بنویسید.

گیت باگ (:)

مهتی جدیدا رزومه ش رو برای کار توی یک شرکت فرستاده، سرپرست منابع انسانی شرکت با توجه به رزومه اون قبول کرده که یک هفته مهتی براشون به صورت آزمایشی کار کنه تا بتونه متوجه بشه آیا مهتی مهارت های لازم برای بودن تو کار تیمی و کار شرکتی رو داره یا نه.

برنامه نویس ارشد شرکت برای بررسی کردن مهارت های مهتی راجب گیت و گیت هاب براش چالش هایی گذاشته:

چالش اول:

باید که کد های یک مخزن توی گیت هاب رو بررسی و دیباگ کنه و کد اصلاح شده خودش رو با استفاده از مهارت هاش توی گیت برای صاحب مخزن ارسال کنه. [لینک ریپازیتوری](#)

چالش دوم:

باید که هر فرد با توجه به این فایل

قسمت مربوط به خودش رو ترجمه کنه و برای صاحب مخزن با استفاده از مهارت هاش ارسال کنه. فایلی که قراره این تغییرات رو روش انجام بدین از لینک زیر قابل دسترس هست. این فایل درواقع فایل مربوط به قسمت هشت کتاب کد تمیز هست.

[لینک فایل](#)

[لینک ریپازیتوری](#)

نکات قابل توجه:

- کد هایی که قراره دیباگ کنید درواقع کد های تمرین دیباگ! هفته پیش هستش و هدف استفاده از مهارت هاتون توی گیت و گیت هاب هست.
- برای چالش اول حتما طبق سوال جلسه قبل یکی از نکاتش رو هنگام دیباگ کردن در نظر بگیرید و استفاده کنید. (دیباگ با assertion یا debug_print و یا نوشتن unit_test)
- ریپو ها رو برای خودتون فورک کنید و اونجا کامیت هاتون رو انجام بدین و درنهایت pull request بفرسین و بعد از انجام این کار ها، نام و نام خنوادگی تون به همراه نام کاربری گیت هاب تون رو

توی یک pdf بزارید و آپلود کنید.