

به نام خدا

تاریخ

فرض کنید با یک تیم برنامه نویسی که در حال پیاده سازی یک برنامه تقویم هستند، همکاری می کنید. و از شما خواسته شده است که یک کلاس به نام `Date` که دارای ویژگی های زیر است، طراحی کنید تا دیگر برنامه نویسان بتوانند از کلاس شما استفاده کرده و قسمت های دیگر برنامه را پیاده سازی نمایند.

کلاس `Date`:

هر شی از کلاس `Date`، یک تاریخ شمسی را به شکل روز، ماه و سال در خود ذخیره می کند.

این کلاس دارای `attribute` های زیر است:

`day`: روز تاریخ

`month`: ماه تاریخ

`year`: سال تاریخ

متدهای مناسب را به کلاس `Date` به نحوی اضافه کنید که عملکرد کلاس مطابق آنچه در ادامه می آید، باشد.

- بتوانیم یک شی تاریخ را با دادن سال و ماه و روز بسازیم. در صورتیکه سال و ماه و روز داده نشود، مقدار پیش فرض هر کدام 1 در نظر گرفته شود و در صورتیکه مقدار هر کدام غیر مجاز بود، `exception` از نوع `DateTimeError` ایجاد شود. مثلا در کد زیر تعریف `d1` درست است ولی برای `d2` باید `exception` داده شود.

```
d1 = Date(1400,12, 4)
```

```
d2 = Date(1400,8,31)
```

- عمل جمع را بین یک شی از کلاس تاریخ و یک عدد صحیح بتوانیم انجام دهیم. در این حالت، عدد صحیح تعداد روز را مشخص می کند. و حاصل جمع یک تاریخ با یک عدد صحیح بدین معنا است که به تاریخ به اندازه عدد صحیح روز اضافه کنیم. حاصل عبارت باید به صورت یک شی از نوع تاریخ باشد. مثلا با انجام عملیات زیر `d2`، یک شی تاریخ خواهد بود که تاریخ 1400/1/12 را نشان خواهد داد.

```
d1 = Date(1400,1,2)
```

```
d2 = d1 + 10
```

- عمل تفریق را بین دو شی از کلاس `Date` بتوانیم انجام دهیم. حاصل عمل تفریق بین دو شی از نوع `Date`، یک عدد صحیح خواهد بود که نشان دهنده، اختلاف روزهای دو تاریخ است. در صورتیکه تاریخ اول کوچکتر باشد، عدد حاصل منفی خواهد بود. مثلا در نمونه کد زیر، عدد ۳۰ چاپ خواهد شد.

```
d1 = Date(1402,10,5)
```

```
d2 = Date(1402,11,5)
```

```
print(d2-d1)
```

- در صورتیکه یک شی تاریخ را در تابع `print` قرار دهیم، تاریخ به صورت `year/month/day` چاپ شود. مثلاً خروجی نمونه کد زیر `1402/2/10` است.

```
d1 = Date(1402,2,10)
```

```
print(d1)
```

- متدی در کلاس `Date` باشد که تاریخ را به تاریخ میلادی تبدیل کرده و به صورت یک رشته به فرم، `month.day.year` خروجی دهد. برای مثال با اجرای کد زیر باید `3.21.2023` چاپ شود.

```
d1 = Date(1402,1,1)
```

```
print(d1.convert2miladi())
```

- می خواهیم اشیا از نوع `Date` قابل پیمایش باشند. بدین معنا که اگر یک شی از نوع تاریخ را در حلقه `for` به کار ببریم، به ترتیب تاریخ روزهای از ابتدای سال تا تاریخ روزی که شی مشخص می کند را به ما بدهد. برای واضح تر شدن نمونه کد زیر را در نظر بگیرید:

```
d1 = Date(1400,1,10)
```

```
for d in d1 :
```

```
    print(d)
```

- در این حالت، `d` یک تاریخ خواهد بود که به ترتیب مقادیر از `1400/1/1` تا `1400/1/10` را می گیرد. و در دستور `print(d)` آنرا چاپ کرده ایم.

توضیحات:

- سال های کبیسه را در نظر نگیرید و اسفند را ۲۹ روزه در نظر بگیرید.
- هر متد دیگری به غیر از متدهای بالا را که برای کمک به پیاده سازی متدهای بالا نیاز دارید، می توانید به کلاس خود اضافه کنید.
- راهنمایی: برای سادگی محاسبه اختلاف دو تاریخ یا جمع عدد و تاریخ می توانید، تاریخ را به روز تبدیل کند و سپس تفریق و جمع را انجام داده و سپس حاصل را به تاریخ تبدیل کنید. برای تبدیل تاریخ به روز، تعداد روزهای گذشته از `1/1/1` تا تاریخ مورد نظر را محاسبه کنید.