



بسم الله الرحمن الرحيم

برنامه سازی پیشرفته C#

دکتر ملکی مجد

تمرین شماره دوم

مهلت ارسال: 3 آبان

پس از پایان مهلت ارسال، تا ۲ روز به ازای هر روز تاخیر ۱۰ درصد از نمره تمرین کسر خواهد شد و پس از ۲ روز نمره ای به تمرین تعلق نخواهد گرفت.

**توجه:**

اسامی تمامی پارامترها می بایست قابل فهم و خوانا باشد، در غیر این صورت از نمره شما کسر می شود.

حتما از کامنت گذاری برای توضیح کد استفاده کنید، در غیر این صورت از نمره شما کسر خواهد شد.

در صورت هر گونه مشکل در این تمرین حتما به مسئولین حل تمرین مراجعه کنید.

برای سوال دوم متد **CalculateBMI** نمره امتیازی برای ایجاد **Unit Test** در نظر گرفته شده است.

آشنایی با **Unit Test** در لینک زیر:

<https://www.c-sharpcorner.com/article/a-basic-introduction-of-unit-test-for-beginners/>

**اهداف** این تمرین یادگیری موارد زیر می باشد:

- Defining and using a class
- Making objects
- Constructor (default constructor, constructor overloading)
- Boxing and unboxing
- Deep copy
- Static data and method

**مسئله 1)** برنامه ای بنویسید که ابتدا زمان را در قالب 3 عدد (که شامل ساعت و دقیقه و ثانیه است) از کاربر گرفته و با تشکیل دادن کلاس Time آن را به دو فرمت مانند زیر نمایش دهد:

20 : 47

8 : 47 : 00 PM

**توجه 1:** نمایش ساعت باید با استفاده از شی ساخته شده و صدا زدن متد بر روی شی باشد.

**توجه 2:** در صورتیکه ساعت غیر استاندارد تعریف شود (عدد ساعت بیشتر از 24، و عدد دقیقه و ثانیه هرکدام بیشتر از 60 باشند) باید مجددا ورودی گرفته شود.

نمونه ورودی:

20,47,0

خروجی:

20 : 47

8 : 47 : 00 PM

نمونه:

14,60,8

→ Time is not correct, please enter another number.

7,50,30

→ 7 : 50

7 : 50 : 30 AM

---

**مسئله 2** bmi هر شخص نسبت وزن(kg) به مجذور قد(m) تعریف می شود. کلاسی با نام BMI با مشخصات زیر تعریف کنید:

### Property ها:

دارای property های Height و Weight از نوع داده double که سطح دسترسی هر دوی آنها private می باشد.

هم چنین این کلاس دارای دو property از نوع داده static double است که در آن ها مقادیر زیر تعریف شده است:

```
static double Low = 22.5;  
static double High = 30;
```

### سازنده ها:

این کلاس که سازنده ای شامل دو پارامتر قد با حرف اختصاری h از نوع double و وزن با حرف اختصاری w است که موارد زیر را در آن بنویسید (از واژه کلیدی this استفاده کنید)

```
this.Height = h;
```

```
this.Weight = w;
```

هم چنین این کلاس دارای یک سازنده بدون پارامتر است که وزن و قد خود را در آن می نویسید. ☺

### متدها:

1) برای محاسبه bmi، متد calculateBMI را تعریف کرده تا bmi را طبق رابطه گفته شده محاسبه کند. ورودی این متد از نوع void و خروجی از نوع object بوده و سطح دسترسی آن public است. (در واقع bmi از نوع داده double است، اما به کمک unboxing آن را در قالب object، return می کنیم).

2) متد PrintInfo را به گونه ای پیاده سازی کنید که ورودی و خروجی آن از نوع void و سطح دسترسی آن public باشد و اطلاعات شخص را به شکل زیر چاپ کند:

Height = person's height

Weight = person's weight

BMI = person's bmi

underweight / normal/ fat

(خط آخر بدین شکل چاپ می شود:

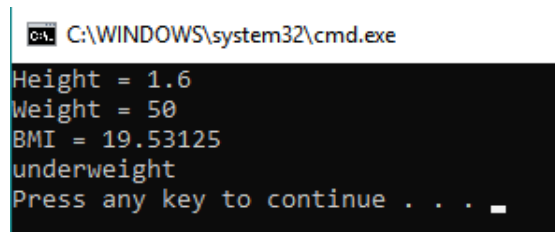
اگر bmi کمتر از Low باشد فرد کمبود وزن دارد. **underweight** چاپ شود.

اگر bmi بیشتر یا مساوی Low و کمتر یا مساوی High باشد فرد نرمال است. **normal** چاپ شود.

اگر bmi از High بیشتر باشد فرد چاق است. **fat** چاپ شود.)

مثال:

```
static void Main(string[] args) {  
    BMI person = new BMI(50, 1.6);  
    person.PrintInfo();  
}
```



C:\WINDOWS\system32\cmd.exe

```
Height = 1.6  
Weight = 50  
BMI = 19.53125  
underweight  
Press any key to continue . . .
```

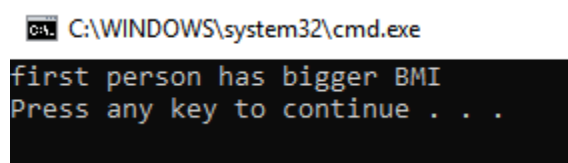
3) متد **CompareBMI** را به گونه ای پیاده سازی کنید که ورودی آن از نوع داده **BMI** و خروجی آن از نوع **void** باشد. در بدنه این متد **bmi** دو شخص باید با هم مقایسه شوند و پیامی را به شکل زیر چاپ کند:

اگر bmi شخص اول بزرگتر بود، **first person has bigger BMI** چاپ شود.

اگر bmi شخص دوم بزرگتر بود، **second person has bigger BMI** چاپ شود.

مثال:

```
static void Main(string[] args)
{
    BMI person = new BMI(50, 1.6);
    BMI personTwo = new BMI(50, 1.8);
    person.CompareBMI(personTwo);
}
```



```
C:\WINDOWS\system32\cmd.exe
first person has bigger BMI
Press any key to continue . . .
```

در انتها پس از پیاده سازی صحیح کلاس، در `main` در دو خط اطلاعات دو نفر به شما داده می شود. پس از ساختن دو شی، ابتدا متد `PrintInfo` را برای هر دو شی فراخوانی کرده و در آخرین مرحله متد `CompareBMI` را فراخوانی کنید.

نمونه ورودی:

72, 1.74

62, 1.6

خروجی:

Height = 1.74

Weight = 72

BMI = 23.7812128418549

normal

Height = 1.6

Weight = 62

BMI = 24.21875

normal

second person has bigger BMI

نمونه ورودی:

45,1.63

52, 1.48

خروجی:

Height = 1.63

Weight = 45

BMI = 16.9370318792578

underweight

Height = 1.48

Weight = 52

BMI = 23.7399561723886

normal

second person has bigger BMI

**مسئله 3)** یک آزمایشگاه خصوصی قصد دارد سیستمی داشته باشد که به اطلاعات کارمندان آنها دسترسی داشته باشد.

اطلاعات هر کارمند در یک شی جدید ذخیره می شود. پس از پایان گرفتن اطلاعات، آرایه ای از اشیاء داریم که از این به بعد برنامه با آنها کار میکنیم.

اطلاعات هر یک از این کارمندان به صورت نام، نام خانوادگی، شماره پرسنلی و تعداد ساعات کاری در هر روز است.

در صورتی که ساعت کاری وارد نشده بود، یک سازنده با سه پارامتر تعریف کنید که شامل نام، نام خانوادگی و شماره پرسنلی باشد، و در آن تعداد ساعات کاری را به صورت پیش فرض 6 قرار دهد.

دقت داشته باشید که شماره پرسنلی هر کارمند **منحصر به فرد** بوده و در صورت تکراری بودن آن شماره مجدد از کاربر گرفته می شود.

#### متدها:

متد Salary را به گونه ای تشکیل دهید که ورودی آن تعداد ساعت کاری از نوع **int** باشد و در خروجی حقوق هر کارمند را به صورت **int** محاسبه کند و برگرداند:

اگر زیر 3 ساعت کار کرده باشد **ساعتی 10 دلار** دریافت می کند.

اگر بیشتر یا مساوی 3 و کمتر از 7 ساعت کار کرده باشد **ساعتی 20 دلار** دریافت می کند.

اگر 7 یا 8 ساعت کار کرده باشد **ساعتی 50 دلار** دریافت می کند.

متدی **static** به نام **OpenOrClose** را به گونه ای تعریف کنید به گونه ای که ساعت سیستم را به عنوان ورودی دریافت کند در صورتیکه ساعت سیستم با ساعت کاری آزمایشگاه تطابق داشت پیام **"Lab is open"** و در غیر این صورت **"Lab is closed"** را به عنوان خروجی برگرداند.

**توجه 1:** ساعت کاری این آزمایشگاه از ساعت 8 الی 16 میباشد.

**توجه 2:** جهت آشنایی با چگونگی استفاده از ساعت سیستم در **C#** می توانید لینک زیر را مطالعه کنید:

<https://docs.microsoft.com/en-us/dotnet/api/system.datetime.hour?view=netframework-4.8>

در **main**، در خط اول ورودی تعداد کارمندان داده می شود (**n**). در **n** خط بعد اطلاعات کارمندان را به ورودی می دهیم. شما باید نام خانوادگی و حقوق هر فرد را چاپ کنید.



نمونه ورودی:

2

Sara,Hamidi,136902,4

Ali,Kabiri,135011

خروجی:

Hamidi: 80\$

Kabiri: 120\$

نمونه ورودی:

3

Farbod,Habibi,134803,7

Sheyda,Hakimi,134803,3

→Number not valid, enter another number

Sheyda,Hakimi,134804,3

Samin,Ahmadi,137210,2

→ Habibi: 350\$

Hakimi: 60\$

Ahmadi: 20\$

---

**مسئله 4)** کلاسی با نام Student تشکیل دهید به گونه ای که دارای سه property شماره دانشجویی، معدل و bmi است.

هر دانشجو یک دوقلو همسان دارد که رقم آخر شماره دانشجویی اش یکی بیشتر و معدلش نیم نمره کمتر است. این کلاس دارای یک متد کپی است که شی متناظر قُل را با استفاده از متد کپی می سازد.

در خط اول ورودی، تعداد دانشجویان (n) داده می شود و در n خط بعد آرایه ای از دانشجوها با اطلاعات شان داده می شود. برای هر دانشجو یک شی ساخته می شود. و در خروجی لیست مرتب شده ی دانشجویان براساس معدل آنها چاپ می شود.

```
static void Main(string[] args)
{
    int count = int.Parse(Console.ReadLine());
    Student[] students = new Student[2 * count];

    for (int i = 0; i < count * 2; i+=2)
    {
        string input = Console.ReadLine();

        students[i] = new Student(int.Parse(input.Split(',')[0]),
                                   double.Parse(input.Split(',')[1]),
                                   new BMI(double.Parse(input.Split(',')[2]),

                                   double.Parse(input.Split(',')[3])));

        students[i + 1] = students[i].Copy();
    }

    //...
    //at last your code should sort students by their averages
    //and print the averages
}
```

نمونه ورودی:

2

972345,19.5,50,1.67

962372,16,70,1.73

خروجی:

15.5 16 19 19.5