

با توجه به دستورات جدیدی که به کار گرفتید، جدول زیر را تکمیل کنید.

ردیف	دستور	کاربرد دستور
۱	<code>Console.ForegroundColor</code>	رنگ قلم نوشته
۲	<code>ConsoleColor.Blue</code>	
۳	<code>Console.BackgroundColor</code>	
۴	<code>Console.Clear();</code>	

- به کمک هم گروهی خود برنامه‌ای بنویسید که نام شما را با رنگ قرمز روی زمینه آبی نمایش دهد و پس از شنیدن صدای بوق به مدت یک ثانیه رنگ صفحه نمایش را به قرمز تغییر داده، نام هم گروهی شما را روی آن به رنگ آبی نمایش دهد.
- برنامه را تغییر دهید که به جای مکث یک ثانیه با فشار کلیدی از صفحه کلید، نام هم گروهی شما را نمایش دهد. از متد `ReadKey` در بین کدهای برنامه استفاده کنید.

فعالیت گروهی



`ConsoleColor` جعبه رنگ ۱۶ تایی در `C#` است.

یادداشت

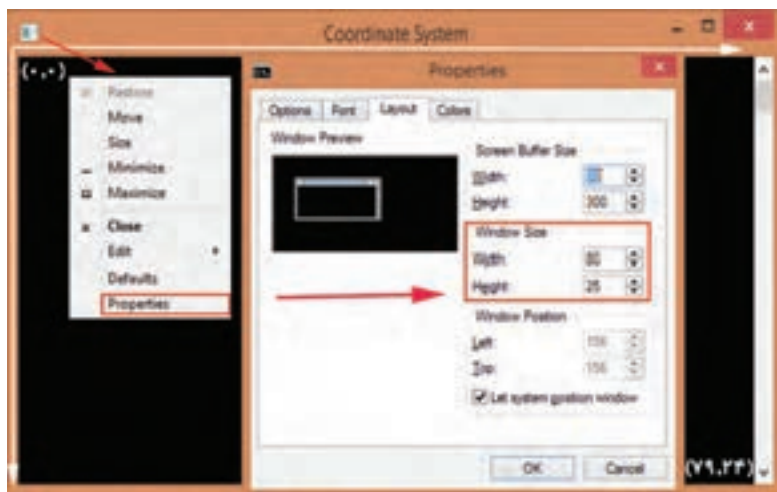


برنامه‌ای بنویسید که جدولی از همه رنگ‌های سی شارپ به همراه نام رنگ در کنسول نمایش دهد.

فعالیت منزل



ویژگی‌های پنجره کنسول



شکل ۱۷- تغییر ویژگی‌های پنجره کنسول

پنجره کنسول دارای ویژگی‌هایی است که برخی از آنها را می‌توان تغییر داد. با راست کلیک روی نوار عنوان این پنجره، تغییرات دلخواهی ایجاد کنید (شکل ۱۷).

چگونه بدون برنامه‌نویسی می‌توان رنگ زمینه، رنگ قلم، اندازه قلم و نام قلم را تغییر داد؟

کنجکاوی



کارگاه ۶ | مختصات مکان نما در کنسول

- ۱ پروژه جدید با نام `ConsoleFlag` ایجاد کنید.
- ۲ قطعه کد زیر را در متد `Main` وارد کنید.

```
Console.BackgroundColor = ConsoleColor.Green;
Console.SetCursorPosition(29, 9);
Console.WriteLine("                ");

Console.SetCursorPosition(29, 10);
Console.WriteLine("                ");

Console.BackgroundColor = ConsoleColor.White;
Console.SetCursorPosition(29, 11);
Console.WriteLine("                ");

Console.SetCursorPosition(29, 12);
Console.WriteLine("                ");
```



شکل ۱۸

۳ برنامه را اجرا کنید.

۴ در ابتدای خطی که زمینه را سفید می‌کند، نویسه‌های // قرار داده، برنامه را اجرا کنید. خروجی را با خروجی مرحله ۲ مقایسه کنید.

۵ در ابتدای یک خط برنامه نویسه‌های /* قرار داده، در انتهای چند خط پایین‌تر نویسه‌های /* را قرار دهید و برنامه را اجرا کرده، خروجی را با مرحله ۲ مقایسه کنید.

۶ از برنامه الگو بگیرید و برنامه‌ای بنویسید که شکل ۱۸ را تولید کند.

با استفاده از دستور `SetCursorPosition(left, top)` با دادن شماره ستون و سطر (فاصله از چپ و بالا) می‌توان مکان نما را به محل دلخواه در پنجره کنسول منتقل کرد.

یادداشت



برنامه‌ای بنویسید که پله‌های رنگی مطابق شکل را ترسیم کند.

فعالیت منزل



کارگاه ۷ رسم در محیط کنسول

۱ پروژه جدیدی با نام `ConsoleDraw` ایجاد کنید.

۲ قطعه کد صفحه بعد را در `Main` وارد کنید.

برای ترسیم از جدول زیر استفاده کنید به این ترتیب که پس از انتخاب کد مناسب با استفاده از کلید `Alt`، کد مورد نظر را وارد کنید از شماره‌های بخش ماشین حساب صفحه کلید باید استفاده شود.

کد	علامت	کد	علامت	کد	علامت
Alt+219	■	Alt+222	▮	Alt+176	░░░░
Alt+220	■	Alt+223	■	Alt+177	░░░░
Alt+221	▮	Alt+248	○	Alt+178	░░░░

با استفاده از راهنمای میکروسافت جدول این کدها را مشاهده کرده، برای انجام فعالیت منزل استفاده کنید.

پژوهش



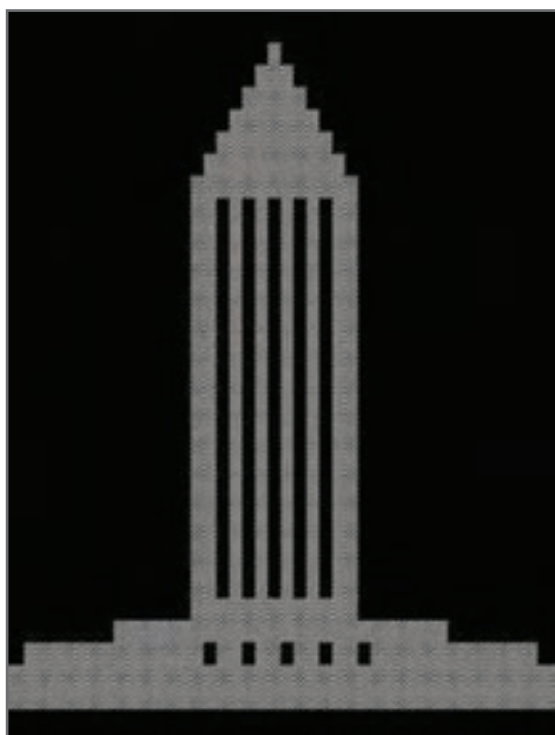
پودمان چهارم: حل مسائل ساده

۳ برنامه را اجرا کنید.

۴ با استفاده از دستورات رنگ، تصویر را رنگ آمیزی کنید.

رسم مقبره ابن سینا دانشمند بزرگ ایرانی را در محیط کنسول کدنویسی کنید.

فعالیت منزل



شکل روبه‌رو را در محیط کنسول کدنویسی کرده، با سلیقه و خلاقیت خود از رنگ‌های متنوعی استفاده کنید.

فعالیت منزل





آنچه آموختم:

- ۱.....
- ۲.....
- ۳.....

ارزشیابی مرحله ۴

مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص ها/داوری /نمره دهی)	نمره
ایجاد جذابیت در برنامه های ساده	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه ای که نرم افزار IDE برنامه نویسی روی آن نصب باشد زمان: ۲۰ دقیقه	بالاتر از حد انتظار	ایجاد برنامه با خروجی های رنگی در مختصات دلخواه	۳
		در حد انتظار	ایجاد یک برنامه با خروجی های رنگی	۲
		پایین تر از حد انتظار	ایجاد یک برنامه با خروجی ساده	۱
<p>معیار شایستگی انجام کار:</p> <p>کسب حداقل نمره ۲ از مراحل حل مسئله و به کارگیری اجزای IDE</p> <p>کسب حداقل نمره ۲ از بخش شایستگی های غیر فنی، ایمنی، بهداشت، توجهات زیست محیطی و نگرش</p> <p>کسب حداقل میانگین ۲ از مراحل کار</p>				

شرح کار:

- ۱ حل مسئله
 ۲ نوشتن برنامه‌های ساده
 ۳ به کارگیری اجزای IDE
 ۴ ایجاد جذابیت در برنامه‌های ساده

استاندارد عملکرد:

با دانش حل مسئله، راه حل و الگوریتم ارائه دهد و از محیط IDE برای ایجاد برنامه‌های ساده کنسول برای تولید خروجی‌های جذاب استفاده کند.
 شاخص‌ها:

شماره مرحله کار	شاخص‌های مرحله کار
۱	تعیین داده، اطلاعات، ورودی و خروجی در مسئله - نوشتن الگوریتم راه حل مسئله و رسم روند نمای آن - پیاده کردن یک روند نما در نرم‌افزار رسم روندنما
۲	ایجاد پروژه در مسیر تعیین شده - باز کردن پروژه موجود - توقف اجرای برنامه جهت مشاهده خروجی - به کارگیری اجزای IDE برای انجام عملیات تعیین شده
۳	ایجاد خروجی تصویری و صوتی مورد نظر
۴	نمایش خروجی تصویری رنگی مورد نظر در مختصات تعیین شده صفحه نمایش

شرایط انجام کار و ابزار و تجهیزات:

مکان: کارگاه رایانه مطابق استاندارد تجهیزات هنرستان‌ها
 تجهیزات: رایانه مجهز به کارت صدا که نرم‌افزار رسم روندنما و نرم‌افزار IDE برنامه‌نویسی روی آن نصب باشد - کاغذ - نوشت افزار
 زمان: ۶۰ دقیقه (حل مسئله ۱۵ دقیقه - به کارگیری اجزای IDE ۱۰ دقیقه - نوشتن برنامه‌های ساده ۱۵ دقیقه - ایجاد جذابیت در برنامه‌های ساده ۲۰ دقیقه)

معیار شایستگی:

ردیف	مرحله کار	حداقل نمره قبولی از ۳	نمره هنرجو
۱	حل مسئله	۲	
۲	به کارگیری اجزای IDE	۲	
۳	نوشتن برنامه‌های ساده	۱	
۴	ایجاد جذابیت در برنامه‌های ساده	۱	
	شایستگی‌های غیر فنی، ایمنی، بهداشت، توجهات زیست محیطی و نگرش:		
	حل مسئله - زبان فنی - رعایت ارگونومی - رسم روندنمای الکترونیکی کاهش مصرف کاغذ - کاربرد حل مسئله در مسائل روزمره زندگی و کار		۲
	میانگین نمرات		*

* حداقل میانگین نمرات هنرجو برای قبولی و کسب شایستگی، ۲ است.

واحد یادگیری ۸

شایستگی کار با انواع داده‌ها، دریافت و نمایش آنها

آیا تا به حال پی برده اید

- برای ورود داده‌ها به برنامه چه باید کرد؟
- چگونه داده‌ها در برنامه نگهداری می‌شوند؟
- چگونه می‌توان نتایج و خروجی برنامه را نمایش داد؟
- متداول‌ترین انواع داده‌ها چیست؟

هدف از این واحد شایستگی، به‌کارگیری داده و متغیر در برنامه‌نویسی و استفاده از دستورات ورودی و خروجی در محیط کنسول است.

استاندارد عملکرد

با استفاده از دانش متغیر و داده و دستورات ورودی و خروجی، داده و متغیر را در برنامه به‌کار گیرد و داده‌ها را به انواع دیگر داده تبدیل کند.

پودمان چهارم: حل مسائل ساده

برنامه‌هایی که تاکنون نوشتید، به نمایش رشته‌های مختلف بر روی صفحه نمایش محدود می‌شد. در بیشتر برنامه‌ها، با داده‌ها و مقادیر مختلف سر و کار داریم. این داده‌ها ممکن است در زمان اجرای برنامه از ورودی خوانده شده، در حافظه رایانه ذخیره شوند و در برنامه برای رسیدن به نتایج مطلوب از آنها استفاده شود. سرانجام داده‌ها و نتایج در خروجی نمایش داده شوند.

کارگاه ۱ طراحی نمون برگ اطلاعات هنرجو

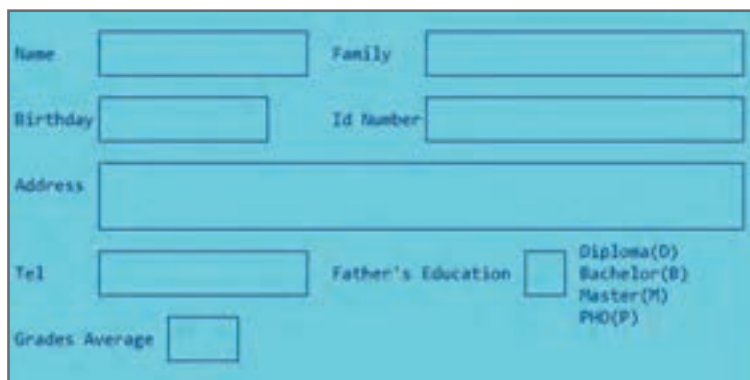
۱ یک پروژه جدید به نام ConsoleInfo1 در VS ایجاد کنید.

۲ کدهای زیر را برای رسم نمون برگ Name و Family وارد کنید. هر جزء اطلاعاتی در نمون برگ را یک فیلد می‌گویند.

```
Console.ForegroundColor = ConsoleColor.DarkBlue;
Console.BackgroundColor = ConsoleColor.Cyan;
Console.WriteLine("");
Console.WriteLine("Name: ");
Console.WriteLine("Family: ");
Console.WriteLine(" ");
```



۳ کدهای لازم برای رسم کامل نمون برگ اطلاعات هنرجو (شکل ۱۹) را وارد کنید.



شکل ۱۹- نمون برگ اطلاعات هنرجو

۴ برنامه را به صورتی تغییر دهید که اطلاعات شما را در نمون برگ نمایش دهد.

۵ برای نمایش اطلاعات هم‌گروهی خود، چه می‌کنید؟

برای نمایش اطلاعات افراد مختلف، باید هر بار بعد از ایجاد تغییرات، دوباره برنامه را ترجمه و اجرا کرد.

آیا تغییر **کد برنامه** برای نمایش اطلاعات افراد مختلف مناسب است؟ چرا؟

کنجکاو



روش دیگر، ورود اطلاعات فرد در زمان اجرای برنامه است. در این حالت به تغییر کد و ترجمه دوباره برنامه نیازی نیست. دستور خواندن، اطلاعات را از ورودی خوانده در مکان مشخصی از حافظه اصلی ذخیره می‌کند. سپس با استفاده از دستور نوشتن می‌توان اطلاعات را از حافظه برداشته، روی صفحه نمایش نشان داد. این مکان مشخص از حافظه، **متغیر (Variable)** نامیده می‌شود. **متغیر** مکانی در حافظه (RAM) رایانه است که مقداری (Value) را به طور موقتی نگهداری می‌کند.

کنجکاو



متغیر باید چه ویژگی‌هایی داشته باشد؟ چرا؟

نام متغیر

هر متغیر دارای یک نام است. از این نام برای دسترسی به متغیر استفاده می‌شود. برای بالا بردن خوانایی برنامه، بهتر است نام مناسبی برای متغیر انتخاب شود.

کنجکاو



آیا برای مترجم زبان سی شارپ، با معنا بودن نام متغیر مهم است؟ چرا؟

در هر زبان برنامه‌نویسی، برای نام‌گذاری متغیرها، قوانین و قراردادهای خاصی وجود دارد. در زبان سی شارپ بعضی از این قوانین به صورت زیر است:

۱ استفاده از حروف الفبا، ارقام و کاراکتر زیرخط (Underscore) مجاز است.

۲ استفاده از ارقام در ابتدای نام متغیر غیرمجاز است.

۳ از کلید واژگان زبان برنامه‌نویسی (Keywords) برای نام یک متغیر استفاده نشود. هر زبان برنامه‌نویسی شامل یک سری واژگان از پیش تعریف شده است که به آنها **واژگان کلیدی** می‌گویند. این واژگان معنی خاصی برای مترجم دارند و نباید از آنها به عنوان نام متغیر استفاده کرد.

واژگان کلیدی C# را که تاکنون آموخته‌اید، بنویسید.

به کمک هم گروهی خود، نام‌های نامعتبر را مشخص کرده، با ذکر دلیل در جدول زیر یادداشت کنید:

فعالیت گروهی



نام متغیر نامعتبر	دلیل عدم اعتبار
First name	وجود فاصله
Iname	
name -Last	
Last.name	
Blood type	
if	کلید واژه

پودمان چهارم: حل مسائل ساده

شرکت مایکروسافت برای کمک به استفاده کنندگان محصولات نرم‌افزاری خود، مجموعه‌ای از منابع، مطالب مفید، راهنمایی‌ها، مقاله‌ها و مثال‌ها را با عنوان MSDN (The Microsoft Developer Network) گردآوری کرده است.

فیلم ۱۰۲۱۹ : کار با راهنما

فیلم



فیلم کار با راهنمای مایکروسافت را مشاهده کنید و فعالیت منزل را انجام دهید.

با استفاده از راهنمای مایکروسافت، قوانین نام‌گذاری دیگری پیدا کنید.

فعالیت منزل



برای هر فیلد نمونه برگ شکل ۱۹ یک متغیر در نظر گرفته، چند نام مناسب برای آن انتخاب کنید. بررسی کنید آیا نام‌های پیشنهادی هم‌گروهی شما مطابق قوانین زبان سی شارپ است؟

فعالیت گروهی



عنوان فیلد	نام متغیر	عنوان فیلد	نام متغیر
Name		Tel	
Family		Father's Education	
Birthday		Grades Average	
Id Number		Address	

برای نام‌گذاری متغیرها، هر برنامه‌نویس روش خاصی را دنبال می‌کند. دو روش از متداول‌ترین روش‌های نام‌گذاری، روش کوهان شتری (camelCase) و پاسکال (PascalCase) است. در روش کوهان شتری، اولین حرف نام متغیر با حرف کوچک آغاز می‌شود. اگر نام متغیر از چند بخش تشکیل شده باشد، حرف اول بخش‌های بعدی با حروف بزرگ نوشته می‌شود. مانند:

fileName , userName , localId

در این کتاب برای نام‌گذاری متغیرها از روش کوهان شتری استفاده شده است.

یادداشت



در روش پاسکال اولین حرف کلمات به شکل بزرگ و بقیه حروف به شکل کوچک نوشته می‌شوند. مانند:

FileName , UserName , LocalId

سی‌شارپ برای نام‌گذاری فضای نامی، کلاس و متد از روش پاسکال استفاده می‌کند. در این کتاب برای نام‌گذاری پروژه‌ها از این روش استفاده شده است.

یادداشت



هر برنامه‌نویس می‌تواند از روش دلخواهی برای نام‌گذاری استفاده کند. اما دانستن و رعایت روش‌های رایج در بین برنامه‌نویسان حرفه‌ای ضروری است.

نام‌های انتخاب شده در فعالیت کارگاهی قبل را به روش کوهان شتری بنویسید تا در برنامه به کار گرفته شوند.

فعالیت
کارگاهی



عنوان فیلد	نام متغیر	عنوان فیلد	نام متغیر
Name		Tel	
Family		Father's Education	
Birthday		Grades Average	
Id Number		Address	

در مورد سایر روش‌های متداول نام‌گذاری تحقیق کنید.

پژوهش



برداشت



آنچه آموختیم:

- ۱.....
- ۲.....
- ۳.....

نوع داده

انتخاب متغیر باید متناسب با نوع و مقدار داده‌ای باشد که در آن قرار می‌گیرد. نوع داده، میزان فضای مورد نیاز در حافظه را مشخص می‌کند. برای مثال، برای پختن یا گرم کردن غذا، معمولاً یک ظرف متناسب با غذای مورد نظر انتخاب می‌شود. اندازه ظرف انتخابی شما، معمولاً متناسب با میزان و نوع غذایی است که قصد پختن آن را دارید. زبان‌های برنامه‌نویسی نیز برای نگهداری داده‌ها، از انواع مختلفی استفاده می‌کنند. سی‌شارپ انواع گسترده‌ای از داده‌های مختلف دارد که نوع عددی، رشته‌ای، کاراکتری و منطقی از متداول‌ترین آنهاست.

داده‌های نمون برگ شکل ۱۹ را دسته‌بندی کنید.

داده غیر عددی	داده عددی
Id Number	Average



با کمک هنرآموز خود بررسی کنید چرا برای کد ملی بهتر است نوع داده غیر عددی انتخاب شود؟



کارگاه ۲ | اعلان و مقداردهی متغیر

- یک پروژه جدید به نام ConsoleInfo2 در VS ایجاد کنید.
- متغیر `firstName` را به صورت زیر اعلان کرده، در متد `Main` برنامه قرار دهید.

```
static void Main(string[] args)
{
    string firstName;
}
```

در زبان سی شارپ پیش از استفاده از متغیرها، باید آنها را اعلان کرد. شکل کلی اعلان متغیر در برنامه به روش زیر است:

`DataType Variable;`



نام متغیر نوع داده

- در برنامه سایر فیلدهای غیر عددی نمون برگ شکل ۱۹ را از نوع رشته‌ای اعلان کنید. در زبان سی شارپ در صورتی که متغیری تعریف شود ولی به کار گرفته نشود، پیام هشدار (warning) صادر می‌شود. هشدارها در ترجمه و اجرای برنامه تأثیری ندارند. اشاره‌گر ماوس را روی خط سبز قرار دهید تا پیام هشدار ظاهر شود.

```
string firstName;
```

The variable 'firstName' is declared but never used

کنجکاوی

خط سبز زیر نام متغیر چه هشدار می‌دهد؟



۴ برای نمایش مقدار متغیر روی صفحه نمایش، از متد `WriteLine` استفاده می‌شود. دستور زیر را به برنامه اضافه کرده، برنامه را اجرا کنید.

```
Console.WriteLine(firstName);
```

کنجکاوی

دلیل خطای برنامه چیست؟



۵ کدهای برنامه را به صورت زیر تغییر دهید.

```
string firstName = "Sara";
```

می‌توانید نام خود را در برنامه وارد کرده، برنامه را اجرا کنید.

کنجکاوی

چرا خط سبز زیر نام متغیر از بین رفت؟



پس از اعلان متغیر، با توجه به نوع متغیر، می‌توان مقداری را در آن ذخیره کرد. برای مقداردهی متغیر، از دستور انتساب استفاده می‌شود.

۶ خط زیر را جایگزین دستور مرحله ۵ کنید.

```
string firstname = "Sara";
```

خطایی که رخ داده را بنویسید و با استفاده از کتاب همراه هنرجو، علت وقوع این خطا را توضیح دهید. این خطا کدام خصوصیت `C#` را یادآوری می‌کند؟

۷ کلید متغیرهای رشته‌ای اعلان شده در برنامه را با مشخصات خودتان و مطابق مرحله ۵ مقداردهی کنید.

یادداشت



هشدار یک رفتار غیرقابل انتظار را گزارش می‌دهد ولی باعث توقف اجرای برنامه نمی‌شود، اما خطا (Error) ناشی از وقوع یک مشکل در برنامه است. وقوع خطا باعث توقف اجرای برنامه می‌شود.

فعالیت
کارگاهی



به کمک هنرآموز خود، تعریف متغیر را در راهنمای مایکروسافت جست‌وجو کنید و عبارت `"type - safe language"` را توضیح دهید.

جدول ارزشیابی شایستگی‌های غیر فنی، ایمنی و بهداشت و توجهات زیست محیطی

نمره	استاندارد (شاخص‌ها/داوری/نمره دهی)	نتایج ممکن	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	شایستگی‌ها
۲	انتخاب نام متغیر مناسب با عملکرد آن - توجه به شیوه قراردادی نام‌گذاری متغیر در زبان برنامه‌نویسی	قابل قبول	مسئولیت‌پذیری، توجه به جزئیات کار - زبان فنی	شایستگی‌های غیر فنی
			رعایت ارگونومی	ایمنی و بهداشت
۱	توجه به ایمنی و بهداشت محیط کارگاه	غیر قابل قبول	دقت در انتخاب نوع داده هنگام اعلان متغیر با توجه به محدوده مقداری که قرار است در متغیر نگهداری شود - دقت در انتخاب نام مجاز و متناسب با عملکرد متغیر	توجهات زیست محیطی
				نگرش

ارزشیابی مرحله ۱

نمره	استاندارد (شاخص‌ها/داوری/نمره دهی)	نتایج ممکن	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	مراحل کار
۳	نام‌گذاری صحیح متغیر به روش‌های مختلف - اعلان متغیر - مقداردهی متغیر - رفع خطای برنامه	بالتر از حد انتظار	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب باشد زمان: ۱۰ دقیقه	اعلان و مقداردهی متغیر
۲	نام‌گذاری صحیح متغیر به روش‌های مختلف - اعلان متغیر - مقداردهی متغیر	در حد انتظار		
۱	نام‌گذاری صحیح متغیر بدون توجه به روش‌های نام‌گذاری - اعلان متغیر	پایین‌تر از حد انتظار		

دریافت داده

تاکنون مقداردهی متغیرها در برنامه انجام شد. در این حالت با هر تغییر مقدار، باید دوباره برنامه را ترجمه کرد. اما می‌توان مقداردهی متغیرها را به وسیله دستورات ورود داده انجام داد. به دستور زیر توجه کنید:

```
string firstName;
```

```
firstName = Console.ReadLine();
```

در این قطعه کد، با اجرای متد `ReadLine()` مقدار وارد شده در متغیر `firstName` قرار می‌گیرد.



`ReadLine()` برای دریافت داده رشته‌ای از ورودی به کار می‌رود. ورود نویسه‌ها در رشته تا زدن کلید `Enter` ادامه پیدا می‌کند.

کارگاه ۴ دریافت متغیر رشته‌ای

۱ پروژه `ConsoleInfo2` را باز کنید.

۲ به جای مقداردهی مستقیم متغیر `firstName`، دستورات زیر را در متد `Main` بنویسید و برنامه را اجرا کنید.

```
string firstName;
firstName = Console.ReadLine();
```

۳ برای بالا بردن خوانایی برنامه، بهتر است قبل از `ReadLine()` پیام `Please enter your first name :` نمایش داده شود. نمایش پیام مناسب در خروجی، باعث ارتباط کاربر با برنامه، هنگام ورود داده‌ها می‌شود.

```
string firstName;
Console.WriteLine("Please enter your first name:");
firstName = Console.ReadLine();
```

۴ دستورات لازم برای خواندن سایر متغیرهای رشته‌ای نمون برگ اطلاعات هنرجو را به برنامه اضافه کنید و در خروجی نمایش دهید.

۵ فیلد سال تولد در گروه داده‌های عددی قرار دارد. آیا می‌توان آن را به صورت رشته‌ای دریافت و چاپ کرد؟ قطعه کد لازم برای این کار را به برنامه اضافه کنید.

براساس نمون برگ شکل ۱۹، کدام داده عددی صحیح و کدام اعشاری است؟



نوع داده عددی صحیح

در زبان‌های برنامه‌سازی مقادیر عددی به دو صورت صحیح یا اعشاری در نظر گرفته می‌شوند.

جدول ۱ برخی از انواع داده‌های عددی صحیح را نشان می‌دهد.

جدول ۱- برخی انواع داده‌های عددی صحیح

نوع داده	اندازه بر حسب بیت
byte	۸ بیت
short	۱۶ بیت
int	۳۲ بیت
long	۶۴ بیت

توجه داشته باشید که انواع داده‌های عددی، با توجه به میزان حافظه‌ای که به آنها تخصیص داده می‌شود، محدوده خاصی از اعداد را می‌پذیرند. با توجه به آنچه در کتاب دانش فنی آموخته‌اید، بیشترین و کمترین مقدار هر نوع داده با توجه به اندازه نوع داده بر حسب بیت مشخص می‌شود.



برای داده از نوع byte چه محدوده‌ای از اعداد مجاز است؟ با استفاده از راهنمای میکروسافت پاسخ خود را بررسی کرده، جدول زیر را کامل کنید:

جدول ۲- انواع داده‌های صحیح

نوع داده	محدوده
sbyte	
byte	
short	
ushort	
int	

نوع داده‌ها با پیشوندهای u, s (ushort و sbyte) را در کتاب همراه هنرجو پیدا کنید. محدوده مجاز byte و sbyte را با یکدیگر مقایسه کنید.



با کمک هم گروهی خود، برای هر یک از متغیرهای زیر نوع داده مناسب پیشنهاد دهید. دلیل انتخاب خود را در جدول بنویسید.



توضیح	مقدار	نام متغیر	نوع داده	دلیل انتخاب
سن (سال)	۱۶	age	byte	
سال تولد	۱۳۸۰		ushort	
مسافت تهران تا مشهد (کیلومتر)	۸۹۷			
سردترین دمای ثبت شده (سانتی‌گراد)	-۸۹			
قد دانش‌آموز (سانتی‌متر)	۱۶۵			
جمعیت جهان (نفر)	۷,۵۰۰,۰۰۰,۰۰۰			

۱ برنامه ConsoleInfo2 را باز کنید.

۲ برای دریافت سال تولد از ورودی، کدهای زیر را به برنامه اضافه کنید:

```
ushort birthday;
Console.WriteLine("Enter your Birthday's year:");
birthday = Console.ReadLine();
```

چه خطایی مشاهده می‌شود؟ علت وقوع خطا چیست؟

ReadLine() عبارت ورودی را به صورت رشته دریافت می‌کند. اگر کاربر مقدار عددی هم وارد کند، ورودی کاربر به صورت رشته‌ای دریافت شده، می‌توان آن را در یک متغیر از نوع رشته‌ای ذخیره کرد. اما برای انجام محاسبات یا تصمیم‌گیری بر روی اعداد، باید آنها را به صورت عددی مورد استفاده قرار داد. بنابراین باید متغیر رشته‌ای را به مقدار عددی تبدیل کنیم. متد Parse رشته ورودی را به نوع داده مورد نظر تبدیل می‌کند.

۳ تغییرات زیر را در کدهای مربوط به دریافت سال تولد اعمال کرده و برنامه را اجرا کنید:

```
birthday = ushort.Parse(Console.ReadLine());
```

دریافت رشته متد تبدیل نوع

دستور بالا با استفاده از ReadLine() مقدار سال تولد را به صورت رشته از ورودی می‌گیرد و با استفاده از متد Parse آن را به نوع داده ushort تبدیل می‌کند.

۴ برنامه را دوباره اجرا کنید و رشته‌ای از حروف وارد کنید. چه اتفاقی رخ می‌دهد؟

۵ برنامه را با ورود عدد منفی اجرا کنید. علت خطا را بررسی کنید.

نوع داده عددی اعشاری

جدول ۳ انواع داده‌های اعشاری را نشان می‌دهد.

جدول ۳- برخی انواع داده‌های عددی اعشاری

نوع داده	اندازه برحسب بیت	دقت
float	۳۲	۷ رقم
double	۶۴	۱۶-۱۵ رقم

برای کار با داده‌های اعشاری می‌توان از روش معمول یا روش نقطه شناور (شبه نماد علمی) استفاده کرد. سی‌شارپ برای نمایش اعداد خیلی بزرگ یا خیلی کوچک از روش نقطه شناور استفاده می‌کند.

```
float average = 17.25f;
double weight = 57.5;
double electricalCharge = 1.602E - 19
```

در روش نقطه شناور، قبل از نقطه اعشار، فقط یک رقم قرار داده می‌شود و بقیه ارقام به سمت راست نقطه اعشار انتقال می‌یابد. در نتیجه عدد اعشار با توان مناسبی از 10^0 نمایش داده می‌شود. برای استفاده از نقطه شناور در سی‌شارپ، به جای ضرب و پایه 10^0 ، از حرف e یا E استفاده می‌شود. در سی‌شارپ عدد اعشاری به صورت پیش فرض از نوع `double` در نظر گرفته می‌شود.

عدد	نماد علمی	نماد علمی در C#
165.2345	$1.652345 \times 10^2 \rightarrow$	1.652345E2

در عبارت زیر دلیل استفاده از نویسه f بعد از عدد چیست؟ اگر این نویسه حذف شود، چه رخ خواهد داد؟

کنجکاوی



`float` average = 17.25 `f` ;

برای محاسبه تعداد الکترون در یک کولن بار الکتریکی ($6.24150975E18$)، جمعیت کره زمین و عدد پی از کدام نوع متغیر استفاده می‌شود؟

فعالیت کارگاهی



عدد پی یک عدد گنگ است که دارای ارقام اعشاری زیادی است. برای جلوگیری از خطای تایپ، می‌توان عدد پی را به صورت یک ثابت (`Constant`) اعلان کرد. استفاده از ثابت‌ها اگر با نام مناسب انجام گیرد، باعث افزایش خوانایی برنامه می‌شود. ثابت، مقداری است که در طول اجرای برنامه تغییر نمی‌کند. برای تعریف ثابت، از کلید واژه `const` استفاده می‌شود. در قطعه کد زیر هر جا که نیاز به استفاده از عدد پی است، ثابت `pi` به کار برده می‌شود.

```
const double pi = 3.14159;
Console.WriteLine(pi);
```

یادداشت



برای نام‌گذاری ثابت، از قوانین نام‌گذاری متغیر استفاده می‌شود.

برداشت



آنچه آموختم:

- ۱.....
- ۲.....
- ۳.....

ارزشیابی مرحله ۲

مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
دریافت داده از ورودی	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب باشد زمان: ۱۵ دقیقه	بالاتر از حد انتظار	دریافت داده رشته‌ای و عددی - تبدیل داده رشته‌ای به عددی - به‌کارگیری اعداد با روش نقطه شناور در برنامه - استفاده از ثابت در برنامه - تعیین مناسب‌ترین نوع داده برای متغیر	۳
		در حد انتظار	دریافت داده رشته‌ای و عددی - تبدیل داده رشته‌ای به عددی - استفاده از ثابت در برنامه	۲
		پایین‌تر از حد انتظار	دریافت داده رشته‌ای و عددی	۱

تبدیل نوع داده

انواع عددی به راحتی به یکدیگر قابل تبدیل هستند. همیشه تبدیل از ظرفیت پایین به بالا امکان‌پذیر است. در ساده‌ترین حالت این تبدیل در یک دستور انتساب انجام می‌شود. تبدیل ضمنی (Implicite) یعنی می‌توان متغیری از نوع `int` را با متغیری از نوع `byte` مقداردهی کرد. به همین ترتیب می‌توان متغیری از نوع `float` را در متغیری از نوع `double` قرار داد.

کارگاه ۶ تبدیل عدد صحیح به اعشاری

- ۱ پروژه جدیدی به نام `ConsoleYear` ایجاد کنید.
- ۲ متغیرهای زیر را تعریف و مقداردهی کنید.

```
ushort birthday = 1380;
float year = 0;
```

```
year = birthday;
Console.WriteLine(year);
```

- ۳ دستورات زیر را در متد `Main` بنویسید.

- ۴ برنامه را اجرا کنید. آیا برنامه خطا دارد؟

کارگاه ۷ تبدیل نوع داده

در این برنامه می‌خواهیم براساس معدل هنرجو، جایزه‌ای به او اختصاص دهیم. کد جایزه، با بخش صحیح معدل مطابقت دارد.

- ۱ برنامه `ConsoleInfo2` را باز کنید.

- ۲ برای فیلد `Grades average` در نمونه برگ شکل ۱۹، چه نوع داده‌ای در نظر می‌گیرید؟
- ۳ کدهای لازم را برای ورود داده معدل به برنامه اضافه کنید و خروجی را روی نمونه برگ اطلاعات هنرجو نمایش دهید.

۴ در نمون برگ اطلاعات هنرجو، فیلدی را اضافه کنید که بخش صحیح معدل هنرجو را نمایش دهد.

برای به دست آوردن کد جایزه، باید از بخش صحیح معدل هنرجو استفاده کرد. بنابراین باید عمل تبدیل متغیر اعشاری به عدد صحیح انجام شود.

۵ دستورات زیر را به برنامه اضافه کنید و برنامه را اجرا کنید.

```
double average = 19.5;  
int num = 0;  
num = average;
```

چه خطایی مشاهده می‌شود؟ علت وقوع خطا چیست؟

انتساب و مقداردهی یک عدد یا متغیر اعشاری به متغیر عددی صحیح امکان‌پذیر نیست و باید از تبدیل نوع داده یا تبدیل صریح (Explicite) استفاده کرد. بنابراین لازم است نوعی که قرار است داده به آن تبدیل شود در یک پرانتز در کنار متغیر سمت راست قرار گیرد.

۶ دستور زیر را جایگزین کنید و برنامه را اجرا کنید.
به این ترتیب بخش اعشار معدل حذف شده و متغیر num با بخش صحیح مقداردهی می‌شود.

```
num = (int) average;
```

نوع داده کاراکتری(char)

اگر بخواهیم داده‌ای داشته باشیم که فقط یک نویسه در آن قرار گیرد، بهتر است از نوع داده char استفاده کنیم. نویسه یک علامت یا حرف است که بین دو تک کوتیشن " قرار می‌گیرد.

```
char ch = 'A';
```

برای فیلد Father's Education در نمون برگ اطلاعات هنرجو چه پیشنهادی برای نام متغیر و نوع داده دارید؟

فعالیت
کارگاهی



کارگاه ۸ دریافت متغیر کاراکتری

۱ برنامه ConsoleInfo2 را باز کنید.

۲ برای دریافت مدرک تحصیلی پدر، کدهای زیر را به برنامه اضافه کنید.

```
char fatherEdu;
```

```
Console.WriteLine("Enter you Father's Education(D/B/M/P):");
```

```
fatherEdu = char.Parse(Console.ReadLine());
```

۳ مقدار دریافت شده را در نمون برگ نمایش دهید.

۴ به جای یک نویسه یک رشته وارد کنید. چه اتفاقی می‌افتد؟ چرا؟

۵ آیا می‌توان نوع مدرک تحصیلی را به صورت یک عدد دریافت کرده، به صورت یک نویسه نشان داد؟ کدهای لازم برای این منظور را به برنامه اضافه کنید. یک راه پیشنهادی استفاده از تبدیل نوع داده عددی به کاراکتر است.

```
Degree: Diploma(100) Bachelor(98) Master(109) PHD(112)
```

پودمان چهارم: حل مسائل ساده

برای تبدیل کد اسکی به نویسه معادل آن، می توان از تبدیل نوع داده استفاده کرد.

یادداشت



حاصل اجرای دستور; (65 (char)) Console.WriteLine چیست؟

کنجکاوی



نوع داده منطقی (bool)

اگر داده‌ای فقط دو مقدار درست (true) یا نادرست (false) داشته باشد، از نوع bool استفاده می‌کنیم. متغیر از نوع منطقی فقط می‌تواند با یکی از دو مقدار true و یا false مقداردهی شود.

- در نمون برگ اطلاعات هنرجو، برای کدام متغیر از داده منطقی استفاده می‌شود؟ برای هم گروهی خود دلیل بیاورید.
- با کمک هم گروهی خود، دو داده در زندگی روزمره نام ببرید که نیازمند تعریف متغیری از نوع منطقی است.

فعالیت گروهی



برای اعلان و مقداردهی و نمایش متغیر flag به ترتیب زیر عمل می‌کنیم:

```
bool flag;  
flag = true;  
Console.WriteLine(flag);
```

آنچه آموختیم:

برداشت



- ۱
- ۲
- ۳

ارزشیابی مرحله ۳

مراحل کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
تبدیل داده‌ها	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب باشد. زمان: ۱۵ دقیقه	بالاتر از حد انتظار	اعلان و مقداردهی متغیرهای کاراکتری و منطقی - تبدیل صریح داده‌ها به یکدیگر - تشخیص و رفع خطاهای تبدیل نوع	۳
		در حد انتظار	اعلان و مقداردهی متغیرهای کاراکتری و منطقی - تبدیل صریح داده‌ها به یکدیگر	۲
		پایین‌تر از حد انتظار	اعلان و مقداردهی متغیرهای کاراکتری و منطقی	۱

کارگاه ۹ الگوی جای‌گذاری ساده (Formatting)

۱ برنامه ConsoleInfo2 را در VS باز کنید.

۲ متغیر رشته‌ای برای دریافت و نگهداری نام هنرستان تعریف کنید.

```
string schoolName;
```

۳ با نمایش پیام مناسب، نام هنرستان را دریافت کنید.

۴ برای نمایش مقدار ورودی دستور زیر را بنویسید:

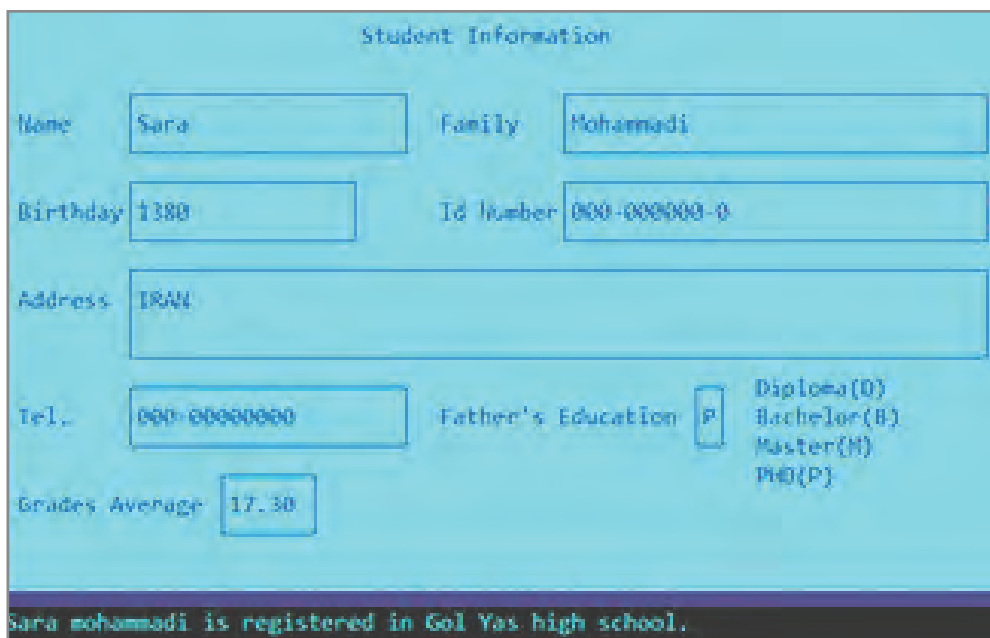
```
Console.WriteLine(schoolName);
```

۵ دستور زیر را جایگزین دستور مرحله ۴ کرده، خروجی را با قبل مقایسه کنید.

```
Console.Write("name of school: {0}", schoolName);
```

با استفاده از الگوی جای‌گذاری (Formatting) در رشته‌ها، می‌توان محل قرارگیری متغیر درون رشته را تعیین کرد. در رشته خروجی، به جای {0} مقدار متغیر schoolName قرار می‌گیرد. اگر بخواهیم مقدار متغیر یا عبارت دیگری را نیز داخل رشته جای‌گذاری کنیم، شماره‌های ۱، ۲، ۳ و... را به ترتیب بین علامت { } قرار می‌دهیم.

۶ با استفاده از الگوی جای‌گذاری کد برنامه را به صورتی تغییر دهید که خروجی بر اساس اطلاعات ورودی به صورت زیر باشد:



The screenshot shows a console application window titled "Student Information" with a light blue background. It contains several input fields and a dropdown menu. The fields are: Name (Sara), Family (Mohammadi), Birthday (1380), Id Number (000-000000-0), Address (IRAN), Tel. (000-000000000), Grades Average (17.30), and Father's Education (P). The dropdown menu for Father's Education has options: Diploma(D), Bachelor(B), Master(M), and PhD(P). At the bottom of the window, a dark blue bar displays the output: "Sara mohammadi is registered in Gol Yas high school."



در الگوی جای گذاری می توان محل قرارگیری یک متغیر یا عبارت در رشته را مشخص کرد.

کارگاه ۱۰ الگوی جای گذاری

شکل کلی الگوی جای گذاری به صورت زیر است:

{الگوی نمایش: عدد تراز ، شماره}

نحوه نمایش داده → فضای اختصاص داده شده برای نمایش مقدار متغیر یا عبارت

۱ پروژه جدیدی به نام `ConsolePrint` ایجاد کنید.

۲ دو متغیر زیر را تعریف و مقداردهی کنید.

```
string firstName = "sara";
```

```
float average = 17.25f;
```

۳ دستورات زیر را در متد `Main` بنویسید:

```
Console.WriteLine(" I am ");
```

```
Console.WriteLine(firstName);
```

```
Console.WriteLine(" and my score is: ");
```

```
Console.WriteLine(17.25);
```

۴ برنامه را اجرا کرده، خروجی را یادداشت کنید:

```
I am sara and my score is: 17.25
```

۵ دستور زیر را جایگزین دستورات مرحله ۳ کنید. برنامه را اجرا کرده، خروجی را مشاهده کنید.

```
Console.WriteLine("I am {0} and my score is: {1,8}",firstName ,average);
```

۶ در دستور بالا ۸- را جایگزین عدد ۸ کرده، `{1,-8}` خروجی را مقایسه کنید.

برنامه ای بنویسید که نام و معدل سه هنرجو را گرفته، به شکل مناسب نمایش دهد. برای نام هنرجو فضای ۲۰ ستونی و برای معدل فضای ۵ ستونی تراز از سمت چپ در نظر بگیرید.



```
1> Name: sara          score: 17.25
2> Name: naryan       score: 16.5
3> Name: zahra        score: 20
```



آنچه آموختم:

..... ۱

..... ۲

..... ۳

ارزشیابی مرحله ۴

مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
قالب بندی خروجی	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب باشد. زمان: ۱۰ دقیقه	بالاتر از حد انتظار	نمایش داده با پیام مناسب با استفاده از قالب بندی - تعیین فضای مناسب و محل نمایش داده در خروجی	۳
		در حد انتظار	نمایش داده با پیام مناسب با استفاده از قالب بندی	۲
		پایین تر از حد انتظار	نمایش پیام با چند دستور بدون قالب بندی	۱
<p>معیار شایستگی انجام کار:</p> <p>کسب حداقل نمره ۲ از مراحل اعلان و مقداردهی متغیر و دریافت داده از ورودی</p> <p>کسب حداقل نمره ۲ از بخش شایستگی‌های غیر فنی، ایمنی، بهداشت، توجهات زیست محیطی و نگرش</p> <p>کسب حداقل میانگین ۲ از مراحل کار</p>				

جدول ارزشیابی پایانی

شرح کار:	
۱ اعلان و مقداردهی متغیر	۲ دریافت داده از ورودی
۲ تبدیل داده‌ها	۲ قالب‌بندی خروجی
استاندارد عملکرد:	
با استفاده از دانش متغیر و داده و دستورات ورودی و خروجی، داده و متغیر را در برنامه به کار گیرد و داده‌ها را به انواع دیگر داده تبدیل کند.	
شاخص‌ها:	
شماره مرحله کار	شاخص‌های مرحله کار
۱	اعلان متغیر با نام مجاز و متناسب با عملکرد متغیر به روش نامگذاری تعیین شده - مقداردهی متغیر
۲	دریافت داده
۳	تبدیل داده دریافت‌شده به نوع داده مورد نیاز - رفع خطای تبدیل نوع صریح
۴	نمایش داده و پیام با استفاده از قالب‌بندی مورد نیاز
شرایط انجام کار و ابزار و تجهیزات:	
مکان: کارگاه رایانه مطابق استاندارد تجهیزات هنرستان‌ها	
تجهیزات: رایانه‌ای که نرم‌افزار برنامه‌نویسی روی آن نصب باشد.	
زمان: ۵۰ دقیقه (اعلان و مقداردهی متغیر ۱۰ دقیقه - دریافت داده از ورودی ۱۵ دقیقه - تبدیل داده‌ها ۱۵ دقیقه - استفاده از الگوی جایگذاری ۱۰ دقیقه)	
معیار شایستگی:	
ردیف	مرحله کار
۱	اعلان و مقداردهی متغیر
۲	دریافت داده از ورودی
۳	تبدیل داده‌ها
۴	قالب‌بندی خروجی
۲	شایستگی‌های غیرفنی، ایمنی، بهداشت، توجهات زیست محیطی و نگرش: مسئولیت‌پذیری - زبان فنی - رعایت ارگونومی - دقت در انتخاب نوع داده هنگام اعلان متغیر با توجه به محدوده مقداری که قرار است در متغیر نگهداری شود - دقت در انتخاب نام مجاز و متناسب یا عملکرد متغیر
*	میانگین نمرات
* حداقل میانگین نمرات هنرجو برای قبولی و کسب شایستگی، ۲ است.	