

برنامه سازی سیستم چیست؟ نوع خاصی از برنامه نویسی است که نیازمند اطلاع از جزئیات فنی سیستم می باشد.

کلاس directory :

کلاس system.io.directory دارای متدهایی برای ایجاد، حذف، انتقال و انجام اعمال مختلف بر روی پوشه ها است. متدهای این کلاس استاتیک هستند و در نتیجه برای استفاده از آنها لازم به ایجاد نمونه از کلاس نیست.

در جدول زیر متدهای مفید کلاس directory آمده است.

متد	توضیح
CreateDirectory	ایجاد یک پوشه جدید
Delete	حذف یک پوشه موجود
Exists	مشخص می کند که آیا یک پوشه در یک مسیر مشخص وجود دارد یا نه.
GetCurrentDirectory	پوشه فعلی را بر می گرداند.
GetDirectories	نام پوشه هایی را که در یک پوشه مشخص وجود دارند را بر می گرداند.
GetFiles	نام فایل هایی را که در یک پوشه مشخص وجود دارند را بر می گرداند.
GetSystemFileEntries	یک آرایه رشته ای که شامل نام فایلها و پوشه های موجود در یک پوشه مشخص است را بر می گرداند
GetParent	پوشه والد یک مسیر را بر می گرداند.
Move	پوشه را به مکان جدیدی منتقل می کند

ایجاد یک پوشه

برای ایجاد یک پوشه از متد createdirectory استفاده میشود. به تکه کد زیر توجه کنید:

```
Directory.CreateDirectory(@"C:\foldername");
```

این متد یک آرگومان از نوع رشته که همان مسیر ایجاد پوشه است را دریافت می کند. متد CreateDirectory همچنین همه پوشه ها و زیر پوشه ها را با استفاده از مسیری که به آن داده می شود ایجاد می کند.

حذف یک پوشه

اگر قصد حذف کردن یک پوشه را داشته باشید می توانید از متد Delete استفاده کنید. این متد نیز یک آرگومان از نوع رشته دریافت می کند که همان مسیری است که پوشه در آن قرار دارد.

```
Directory.Delete(@"C:\foldername");
```

در حالت پیشفرض متد Delete اگر یک پوشه خالی نباشد یک استثناء (exception) را تولید میکند. اگر بخواهید همه فایلها و زیر پوشه های یک پوشه را حذف کنید باید از یکی دیگر از سربارگذاری های متد Delete که داری پارامتر دومی از نوع بولی است استفاده کنید. اگر مقدار آن را true قرار دهید هر چیزی که داخل پوشه مشخص شده است حذف می شود.

```
Directory.Delete(@"C:\foldername",true);
```

تست وجود یک پوشه

مانند کلاس File، کلاس Directory هم دارای متد Exists است که به کاربر اعلام میکند که آیا در یک مسیر مشخص پوشه مورد نظر وجود دارد یا نه؟

```
if(Directory.Exists(@"C:\foldername") == true)
{
    MessageBox.Show("پوشه وجود دارد");
}
```

به دست آوردن لیست زیر پوشه ها و پوشه ها

متد GetDirectories همه زیر پوشه های یک مسیر خاص را در اختیار ما قرار میدهد. برای روشن شدن مطلب اجازه بدهید

یک پوشه در درایو C ایجاد کنیم و پوشه هایی را در داخل این پوشه با نام های folder1، folder2، folder3 قرار دهیم.

برای به دست آوردن لیست زیر پوشه ها به صورت زیر عمل میکنیم:

ابتدا در فرم یک listbox و button اضافه میکنیم.

کد زیر را در button که ایجاد کردیم مینویسیم

```
string[] list = Directory.GetFileSystemEntries(@"C:\");
```

```
foreach (string str in list)  
{  
    listBox1.Items.Add(Path.GetFileName(str));  
}
```

مثال های سکن عملی:

؟. برنامه ای بنویسید که مسیر و نام پوشه مورد نظر را از کاربر دریافت کرده سپس با کلیک روی دکمه ای پوشه مورد نظر را ایجاد نماید.

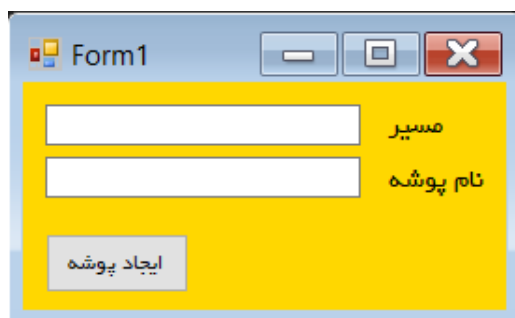
طراحی فرم:

از بخش toolbox ابزار های زیر را اضافه میکنیم:

دو textbox برای دریافت مسیر و نام پوشه

دو label برای نوشتن توضیحات

یک button برای ساخت پوشه جدید.



کد نویسی:

ابتدا در بالا و ابتدای کد نویسی فضای نام system.io را اضافه میکنیم

```
using System.IO;
```

سپس کد های زیر را در button1 یا همان کلید "ایجاد پوشه" مینویسیم.

```
string path = textBox1.Text;
```

مسیر را از textbox1 گرفته و در متغیر path میریزد.

```
string folder_name = TextBox2.Text;
```

نام پوشه را از textbox2 گرفته و در متغیر folder_name میریزد.

```
Directory.CreateDirectory(path + folder_name);
```

مسیر و نام پوشه را بهم اتصال میدهد و با استفاده از کلاس directory و متد createDirectory پوشه جدید می سازد.

؟. برنامه ای بنویسید که مسیر و نام پوشه مورد نظر را از کاربر دریافت کرده سپس با کلیک روی دکمه ای پوشه مورد نظر را حذف نماید.

طراحی فرم:

از بخش toolbox ابزار های زیر را اضافه میکنیم:

دو textbox برای دریافت مسیر و نام پوشه

دو label برای نوشتن توضیحات

یک button برای حذف پوشه.



کد نویسی:

ابتدا در بالا و ابتدای کد نویسی فضای نام system.io را اضافه میکنیم (در صورتی که قبلا اضافه نکرده باشیم)

```
using System.IO;
```

سپس کد های زیر را در button1 یا همان کلید "حذف پوشه" مینویسیم.

```
string path = textBox1.Text;
```

مسیر را از textBox1 گرفته و در متغیر path میریزد.

```
string folder_name = TextBox2.Text;
```

نام پوشه را از textbox2 گرفته و در متغیر folder_name میریزد.

```
if (Directory.Exists(path + folder_name) == true)
```

اگر پوشه ای با نام و آدرس داده شده وجود داشت

```
{
```

```
Directory.Delete(path + folder_name,true);
```

در پارامتر اول مسیر و نام پوشه را بهم اتصال میدهد و پارامتر دوم که از نوع Boolean است اگر مقدار true داشته باشد محتویات درون پوشه نیز حذف میشود سپس با استفاده از کلاس directory و متد delete پوشه را حذف می کند.

```
}
```

```
Else
```

در غیر این صورت

```
{  
    MessageBox.Show("نشد یافت نظر مورد پوشه");  
    پیام "پوشه مورد نظر یافت نشد" را به کاربر نمایش می دهد.  
}
```

؟. برنامه ای بنویسید که با دریافت مسیر مورد نظر تمام پوشه های آن مسیر را درون `listbox` نمایش دهد.

طراحی فرم:

از بخش `toolbox` ابزار های زیر را اضافه میکنیم:

یک `textbox` برای نمایش مسیر و نام پوشه

یک `label` برای نوشتن توضیحات

دو `button` یکی برای دریافت آدرس از کاربر یکی برای نمایش پوشه های مسیر

یک `folderbrowserdialog` برای نمایش پنجره انتخاب پوشه

یک `listbox` برای نمایش نام پوشه ها



کد نویسی:

ابتدا در بالا و ابتدای کد نویسی فضای نام `system.io` را اضافه میکنیم (در صورتی که قبلا اضافه نکرده باشیم)

```
using System.IO;
```

سپس کد های زیر را در `button1` یا همان کلید "Browse" مینویسیم.

```
folderBrowserDialog1.ShowDialog();
```

پنجره `folderbrowserdialog` را برای انتخاب مسیر نمایش می دهد.

```
textBox1.Text = folderBrowserDialog1.SelectedPath + @"\";
```

مسیر دریافت شده را با یک / اتصال داده و در textBox1 نمایش می دهد.

سپس کد های زیر را در button2 یا همان کلید "نمایش پوشه های مسیر" مینویسیم.

```
listBox1.Items.Clear();
```

تمام آیتم های listBox1 را پاک می کند

```
string path = textBox1.Text;
```

مسیر را از textBox1 گرفته و در متغیر path میریزد.

```
string[] list = Directory.GetDirectories(path);
```

زیر پوشه های مسیر (که در متغیر path قرار دارد) را در آرایه ای رشته ای به نام list می ریزد.

```
foreach (string str in list)
```

با استفاده از دستور foreach با متغیر str در آرایه list پیمایش می کند.

```
{
```

```
listBox1.Items.Add(Path.GetFileName(str));
```

نام زیرپوشه های مسیر را در listBox1 اضافه می کند (درون پرانتز با استفاده از کلاس path و متد GetFileName

مسیر را از نام پوشه جدا می کند)

```
}
```


برنامه ای بنویسید که با دریافت مسیر مورد نظر تمام فایل های آن مسیر را درون listBox نمایش دهد.

طراحی فرم:

از بخش toolbox ابزار های زیر را اضافه میکنیم

یک textbox برای نمایش مسیر و نام پوشه

یک label برای نوشتن توضیحات

دو button یکی برای دریافت آدرس از کاربر یکی برای نمایش فایل های مسیر

یک listBox برای نمایش نام فایل ها

یک folderbrowserdialog برای نمایش پنجره انتخاب پوشه



کد نویسی:

ابتدا در بالا و ابتدای کد نویسی فضای نام `system.io` را اضافه میکنیم (در صورتی که قبلا اضافه نکرده باشیم)

```
using System.IO;
```

سپس کد های زیر را در `button1` یا همان کلید "Browse" مینویسیم.

```
folderBrowserDialog1.ShowDialog();
```

پنجره `folderbrowserdialog` را برای انتخاب مسیر نمایش می دهد.

```
textBox1.Text = folderBrowserDialog1.SelectedPath+"\";
```

مسیر دریافت شده را با یک / اتصال داده و در `textBox1` نمایش می دهد.

سپس کد های زیر را در button2 یا همان کلید "نمایش فایل ها و پوشه های مسیر" مینویسیم.

```
listBox1.Items.Clear();
```

تمام آیتم های listBox1 را پاک می کند

```
string path = textBox1.Text;
```

مسیر را از textBox1 گرفته و در متغیر path میریزد.

```
string[] list = Directory.GetFiles(path);
```

فایل های مسیر (که در متغیر path قرار دارد) را در آرایه ای رشته ای به نام list می ریزد.

```
foreach (string str in list)
```

با استفاده از دستور foreach با متغیر str در آرایه list پیمایش می کند.

```
{
```

```
listBox1.Items.Add(Path.GetFileName(str));
```

نام فایل ها و زیرپوشه های مسیر را در listBox1 اضافه می کند (درون پرانتز با استفاده از کلاس path و متد

GetFileName مسیر را از نام پوشه جدا می کند)

```
}
```

؟ برنامه ای بنویسید که با دریافت مسیر مورد نظر تمام فایل ها و پوشه های آن مسیر را درون
listbox نمایش دهد.

طراحی فرم:

از بخش toolbox ابزار های زیر را اضافه میکنیم:

یک textbox برای نمایش مسیر و نام پوشه

یک label برای نوشتن توضیحات

دو button یکی برای دریافت آدرس از کاربر یکی برای نمایش پوشه ها و فایل های مسیر

یک folderbrowserdialog برای نمایش پنجره انتخاب پوشه

یک listBox برای نمایش پوشه ها و فایل های مسیر



کد نویسی:

ابتدا در بالا و ابتدای کد نویسی فضای نام `system.io` را اضافه میکنیم (در صورتی که قبلا اضافه نکرده باشیم)

```
using System.IO;
```

سپس کد های زیر را در `button1` یا همان کلید "Browse" مینویسیم.

```
folderBrowserDialog1.ShowDialog();
```

پنجره `folderbrowsedialog` را برای انتخاب مسیر نمایش می دهد.

```
textBox1.Text = folderBrowserDialog1.SelectedPath + @"\";
```

مسیر دریافت شده را با یک / اتصال داده و در `textBox1` نمایش می دهد.

سپس کد های زیر را در button2 یا همان کلید "نمایش فایل ها و پوشه های مسیر" مینویسیم.

```
listBox1.Items.Clear();
```

تمام آیتم های listBox1 را پاک می کند

```
string path = textBox1.Text;
```

مسیر را از textBox1 گرفته و در متغیر path میریزد.

```
string[] list = Directory.GetFileSystemEntries(path);
```

فایل ها و پوشه های مسیر (که در متغیر path قرار دارد) را در آرایه ای رشته ای به نام list می ریزد.

```
foreach (string str in list)
```

با استفاده از دستور foreach با متغیر str در آرایه list پیمایش می کند.

```
{
```

```
listBox1.Items.Add(Path.GetFileName(str));
```

نام زیرپوشه های مسیر را در listBox1 اضافه می کند (درون پرانتز با استفاده از کلاس path و متد GetFileName

مسیر را از نام پوشه جدا می کند)

```
}
```