

## شایستگی کار با ساختار شرطی

از دوران کودکی تاکنون سؤالات زیادی داشتیم که با **اگر** شروع می‌شد و مسیر زندگی انسان را مشخص می‌کرد.

**اگر** دوچرخه داشتیم، چه می‌شد؟

**اگر** دیپلم گرفته بودم؛

**اگر** دانشجو بودم؛

بعضی از این **اگرها** دور از دسترس و برخی دیگر دست‌یافتنی است و هنگامی که اتفاق می‌افتد؛ اکثر انسان‌ها در تعقیب **اگرهای** دیگرند.

مثلاً **اگر** دوچرخه داشتیم، چه می‌شد؟ **اگر** دوچرخه داشتیم می‌گفتم **اگر** می‌توانستم برای دوچرخه‌ام زنگ بخرم چه می‌شد؟ بعد از آن در این فکر بودم که **اگر** در تیم دوچرخه سواری منطقه عضو می‌شدم و ..... در اینجا می‌خواهیم برای **اگرها** برنامه بنویسیم.

به کمک هم گروهی خود **اگرهای** کتاب دانش فنی پایه را در جدول زیر وارد کنید.

فعالیت  
گروهی

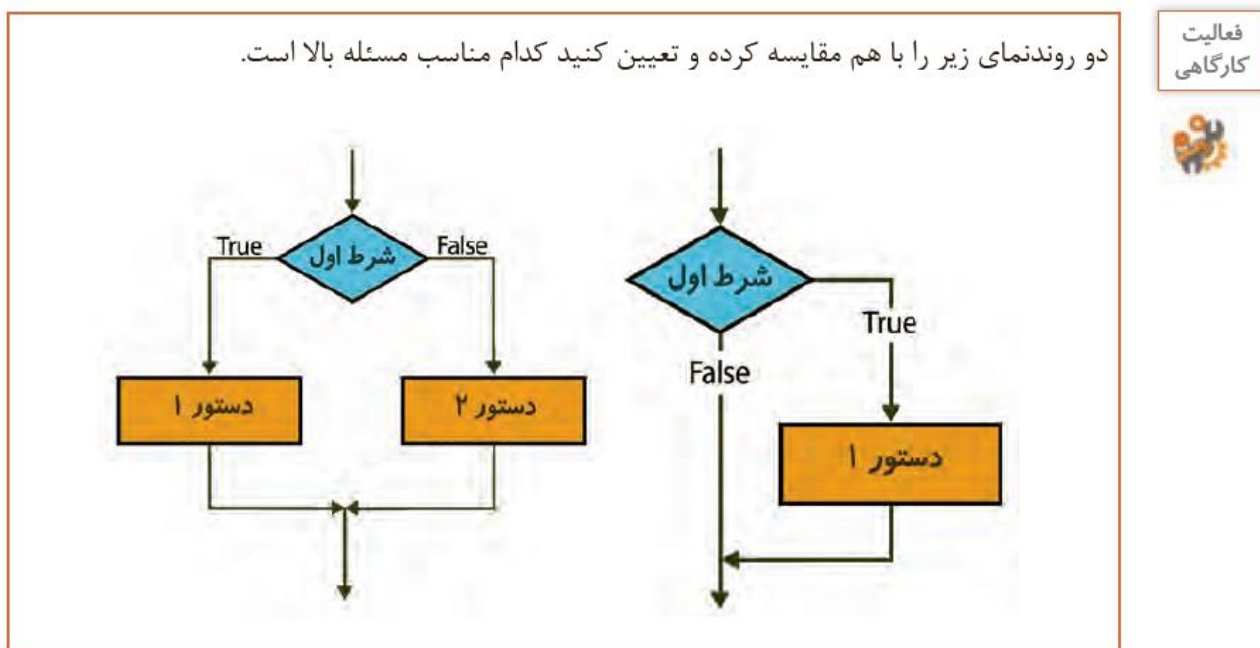


ردیف	عنوان	شرط	درست بودن شرط	نادرست بودن شرط
۱	تفکیک زباله	<b>اگر</b> زباله خشک باشد	در سطل مخصوص زباله خشک قرار بده	—
۲	خرید کالا			خرید نکردن
۳	بازی سنگ، کاغذ، قیچی			
۴				
۵				

در برخی از مسائل اگرها ساده نیست و شامل چندین شرط است. در این بخش می‌خواهیم یکی از اگرهای اولیه کودکی را تبدیل به یک قطعه کد کنیم. «اگر دوچرخه داشتم بازی می‌کردم».

```
if (x=="bicycle")  
    Console.WriteLine("Playing");
```

در قطعه کد بالا متغیر x نشان‌دهنده نوع وسیله‌ای است که داریم. با همین اگر ساده توانستیم یک قطعه کد کوچک بنویسیم. همان‌طور که مشاهده می‌کنید واژه اگر در زبان برنامه نویسی به if تبدیل شد.



شکل ۱- ساختار دستور if

شکل کلی if ساده:

(عبارت شرطی) if  
دستور;

می‌خواهیم برنامه‌ای بنویسیم که مزایای دو روش پرداخت قبوض آب و برق را مشخص کند:

✓ مراجعه به بانک: پرداخت قبوض

✓ استفاده از پرداخت الکترونیکی: پرداخت قبوض، کاهش ترافیک

۱ پروژه‌ای با نام EPayment ایجاد کنید.

۲ کد زیر را در متد Main وارد کنید.

```
Console.WriteLine("Enter your choice // 1 or 2");
byte x = byte.Parse(Console.ReadLine());
if (x==1)
    Console.WriteLine("Pay bills");
if (x==2)
{
    Console.WriteLine("Pay bills");
    Console.WriteLine("reduction of traffic ");
}
```

۳ برنامه را اجرا کرده و نتایج آن را مشاهده کنید.

۴ بعد از پرانتز **if** دوم یک **;** قرار داده، نتیجه را با نتیجه مرحله قبل مقایسه کنید.

مجموعه دستوراتی را که داخل **{}** قرار می‌گیرد. یک بلاک می‌گویند.

- می‌خواهیم قسمتی از برنامه هدایت یک خودروی بدون سرنشین را برای عبور از چراغ قرمز طراحی کنیم.
- ۱ پروژه جدیدی با نام VCondition ایجاد کنید.
  - ۲ برنامه زیر را کامل کرده، در متد Main بنویسید.

```
if ( light == "red")
    Vehicle_Condition = "stop";
else
    Vehicle_Condition = "move";
Console.WriteLine(Vehicle_Condition );
```

- ۳ کدام یک از روندنماهای شکل ۱-۱۰ مناسب این برنامه است؟
  - ۴ روندنمای این برنامه را رسم کنید.
  - ۵ برنامه را به صورت زیر تغییر دهید.
- اگر چراغ قرمز بود، ابتدا ماشین ترمز (break) کرده، سپس متوقف شود؛ در غیراین صورت با زدن یک بوق حرکت کند.

می‌خواهیم برنامه‌ای طراحی کنیم تا با دریافت یک رشته برای زوج یا فرد بودن روزهای هفته، برنامه بعدازظهرهای علی را تعیین کند.

روز	۱۵ تا ۱۶:۳۰	۱۷ تا ۱۸:۳۰	۱۸:۴۰ تا ۱۹	۱۹:۱۵ تا ۲۰:۴۵
روزهای شنبه - دوشنبه - چهارشنبه	انجام تکالیف روزانه	باشگاه ورزشی	فریضه نماز	کلاس C#
روزهای یکشنبه - سه شنبه - پنج شنبه	انجام تکالیف روزانه	کلاس زبان انگلیسی	فریضه نماز	سرزدن به پدربزرگ

۱ پروژه جدیدی با نام Week ایجاد کنید.

۲ کد زیر را در متد Main بنویسید.

آیا برنامه خطا دارد؟

```
string day = Console.ReadLine();
if (day == "odd">//condition true
{
    Console.WriteLine("Studying");
    Console.WriteLine("English class");
    Console.WriteLine("Say praying");
    Console.WriteLine("Visit Grand");
}
if (day == "Even") //condition true
    Console.WriteLine("Studying");
    Console.WriteLine("gym");
    Console.WriteLine("Say praying");
    Console.WriteLine("c# programming Class ");
```

۳ خروجی برنامه در صورتی که ورودی «even» باشد، چیست؟

۴ خروجی برنامه در صورتی که ورودی «odd» باشد، چیست؟

دلیل اشتراک پاسخ در موارد ۴ و ۵ چیست؟

۵ خطای برنامه را رفع کنید.

اگر خروجی برنامه نادرست است با توجه به آنچه قبلاً آموختید نوع خطا را مشخص کنید و مشکل برنامه را برطرف کنید.

آیا می‌توان برنامه را با ساختار if - else نوشت؟

## کارگاه ۴ اجرای گام به گام if \_ else

در یک اداره ۲۴ طبقه، ۲ دستگاه آسانسور، مخصوص طبقات زوج و فرد و یک نفر جهت راهنمایی مراجعان وجود دارد که با پرسش از افراد، آسانسور مناسب را مشخص می‌کند. می‌خواهیم با هدف مکانیزه کردن قسمتی از تشکیلات ساختمانی از یک برنامه کمک بگیریم تا با وارد کردن شماره طبقه، آسانسور مناسب را مشخص کند. (آسانسور A مخصوص طبقات فرد و آسانسور B مخصوص طبقات زوج است).

۱ پروژه جدیدی با نام ChElevator ایجاد کنید.

۲ کد زیر را در Main وارد کنید.

```
Console.WriteLine("Enter the floor");
int floor = int.Parse(Console.ReadLine());
if (floor % 2 == 0)
    Console.WriteLine("Elevator A");
else
    Console.WriteLine("Elevator B");
```

۳ اگر مراجعه کننده به طور سهوی عدد ۳۴ را وارد کند خروجی برنامه چیست؟

۴ برنامه را طوری تغییر دهید تا در صورتی که کاربر عددی خارج از محدوده تعداد طبقات وارد کرد پیام مناسبی چاپ شود.

۵ خروجی برنامه را با مقادیر روبه‌رو Trace کنید.

خروجی	floor
	10
	25
	-6
	5.2
	floor13

## کارگاه ۵ عملگر سه تایی

می‌خواهیم برنامه‌ای بنویسیم تا با دریافت سن فرد تعیین کند آیا صلاحیت دریافت گواهی نامه رانندگی دارد؟

۱ پروژه جدیدی با نام DLicence ایجاد کنید.

اگر سن فرد مساوی یا بزرگ‌تر از ۱۸ سال باشد می‌تواند در آزمون رانندگی شرکت کند و در غیر این صورت این امکان وجود ندارد. در نتیجه باید با استفاده از یک دستور شرطی، سن بررسی شود.

۲ کد زیر را تکمیل کنید و در متد Main وارد کنید.

```
if (age >= 18)
    status = "Allowed";
else
    status = "Not Allowed";
Console.WriteLine(status);
```

۳ خروجی برنامه را با مقادیر زیر Trace کنید.

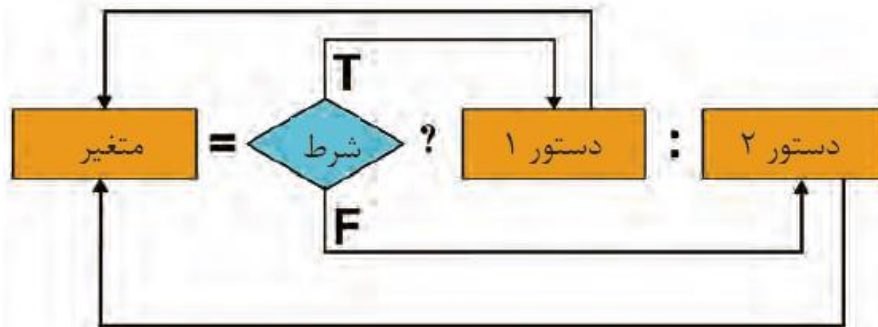
خروجی	age
	20
	18
	10

۴ متغیرهای age و status را اعلان کنید.

۵ به جای کد قبلی، کد زیر را وارد کنید.

```
Console.WriteLine("How old are you?");  
int age = int.Parse(Console.ReadLine());  
string s;  
s = (age >= 18 ? "Allowed": " Not Allowed");  
Console.WriteLine (s);
```

۶ خروجی برنامه را در دو حالت مقایسه کنید.



شکل ۲- عملگر سه تایی

در برخی موارد که دستور شرطی، از یک دستور، برای بخش if و else استفاده می کند؛ می توان از عملگر سه تایی (ternary) به صورت زیر استفاده کرد. هنگامی که شرط درست باشد خروجی این عملگر حاصل دستور 1 و هنگامی که شرط نادرست باشد، حاصل دستور 2 است (شکل ۲).

شکل کلی عملگر سه تایی دستور 2 : دستور 1 ؟ (عبارت شرطی) = خروجی

می‌خواهیم برنامه‌ای بنویسیم که کاربر معتبر برای ورود به سیستم را مشخص کند.

نام کاربری معتبر: reza

گذرواژه صحیح: 123456

۱ پروژه جدیدی با نام UserPass ایجاد کنید.

۲ کد زیر را در متد Main وارد کنید.

```
Console.WriteLine("Enter your user name");
string user = Console.ReadLine();
Console.WriteLine("Enter your password");
string pass = Console.ReadLine();
if (user == "reza")
    if (pass == "123456")
        Console.WriteLine("Welcome " + user);
    else
        Console.WriteLine("The password is incorrect");
else
    Console.WriteLine("Unkown user");
Console.ReadKey();
```

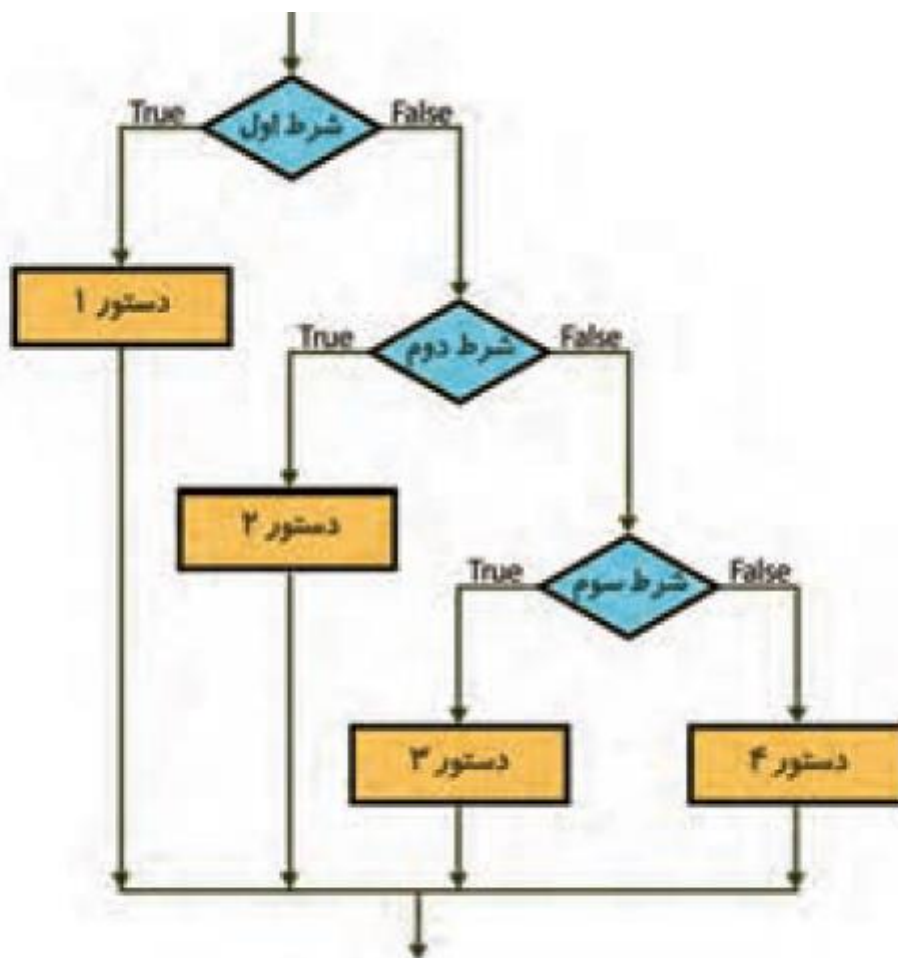
۲ با توجه به برنامه جدول زیر را تکمیل کنید.

شرط	دستور
اگر نام کاربری و گذرواژه صحیح است؛	
اگر نام کاربری صحیح است و گذرواژه صحیح نیست.	
اگر نام کاربری صحیح نیست؛	

برای تقسیم‌بندی صحیح عبارات شرطی باید از بلاک {} استفاده کنید. بلاک تعیین‌کننده محدوده عبارت شرطی است که بیش از یک دستور دارد.



اگر درون یک ساختار شرطی، ساختار شرطی دیگری باشد؛ به آن عبارات شرطی تودرتو (Nested conditional statements) می‌گویند.



شکل ۵- ساختار شرطی تودرتو

## عبارات شرطی تودرتو و عملگرهای منطقی

عبارات شرطی تودرتو را می‌توان با استفاده از عملگرهای منطقی ساده‌تر و خواناتر نوشت. اما به کار بردن آنها دقت زیادی لازم دارد.

### کارگاه ۷ عبارات شرطی با عملگرهای منطقی

می‌خواهیم با استفاده از نرم‌افزار پیش‌بینی هوا امکان بازدید هنرجویان از نمایشگاه کتاب را در فصل بهار بررسی کنیم.

امکان برگزاری	هوا	دما	ردیف
عالی		بیشتر از ۱۹ درجه	۱
خوب		بیشتر از ۱۹ درجه	۲
مناسب		کمتر از ۱۹ درجه	۳
غیر ممکن		کمتر از ۱۹ درجه	۴

۱ پروژه جدیدی با نام Weather ایجاد کنید.

۲ متغیرهای مورد نیاز برنامه را اعلان کنید.

```
Console.WriteLine("Enter the air temperature");
int temp = int.Parse(Console.ReadLine());
Console.WriteLine("Enter the weather conditions: sunny / cloudy / partly cloudy/ rainy");
string condition = Console.ReadLine();
```

۲ برای ردیف ۱ جدول، قطعه کد زیر را وارد کنید.

```
if (temp > 19)
    if (condition == "sunny")
        Console.WriteLine("Excellent");
```

۴ برنامه را اجرا کرده، ورودی‌هایی به برنامه بدهید که پیام **Excellent** را نمایش دهد.

۵ کد زیر را جایگزین خطوط بالا کنید.

```
if ((temp > 19) && (condition == "sunny"))  
    Console.WriteLine("Excellent");
```

۶ برنامه را دوباره اجرا کرده، نتایج را مقایسه کنید.

۷ کد ردیف ۲ جدول را با استفاده از عملگرهای منطقی بنویسید.

۸ کد زیر را برای ردیف ۳ جدول به برنامه اضافه کرده، سپس برنامه را اجرا و نتیجه را یادداشت کنید.

```
if (temp < 19)  
{  
    if (condition == "sunny")  
        Console.WriteLine("suitable");  
    if (condition == "partly cloudy")  
        Console.WriteLine("suitable");  
}
```

## کارگاه ۹ ترکیب عبارات منطقی، محاسباتی و شرطی

می‌خواهیم برنامه‌ای بنویسیم که با دریافت سه عدد  $a$ ،  $b$ ،  $c$  مشخص کند آیا این سه عدد می‌توانند اضلاع یک مثلث باشند؟

**روش اول:** برای حل مسئله، از قضیهٔ همار در هندسه استفاده می‌کنیم. این قضیه بیان می‌کند که در هر مثلث مجموع هر دو ضلع از ضلع دیگر بزرگ‌تر است.<sup>۱</sup> برای مثال سعی کنید مثلثی با ابعاد ۱ و ۲ و ۲۰ ترسیم کنید. چنین مثلثی وجود ندارد. زیرا هرگز دو ضلع ۱ و ۲ به هم نخواهند رسید. مطابق قضیه همار، مجموع دو ضلع باید از ضلع سوم بزرگ‌تر باشد که در این مثال  $۲+۱ > ۲۰$  نیست.

زمانی که داده‌های ورودی از سه شرط ( $a + b > c$ ,  $a + c > b$ ,  $b + c > a$ ) به درستی عبور کنند، به خروجی true می‌رسیم و اگر در هر نقطه از مسیر منحرف شوند؛ خروجی false خواهد بود. برای این کار به متغیری (count) نیاز داریم که در مسیر حرکت به سوی مقصد، تعداد شرط‌هایی را که به درستی از آن عبور می‌کنیم؛ بشمارد و هر وقت تعداد شمارش شده به عدد ۳ رسید؛ خروجی true شود.

۱ پروژه جدیدی با نام Triangle ایجاد کنید.

۲ برنامه زیر را تکمیل کرده و در متد Main بنویسید.

```
int count = 0;
if (b + c > a) count++;
if (a + c > b) count++;
if (a + b > c) count++;
if (count == 3)
    Console.WriteLine(true);
else
    Console.WriteLine(false);
```

```
if (a + b <= c)
    Console.WriteLine(false);
else if (a + c <= b)
    Console.WriteLine(false);
else if (b + c <= a)
    Console.WriteLine(false);
else
    Console.WriteLine(true);
```

۳ برنامه را با ۳ مقدار ورودی اجرا کنید.

روش دوم: می‌توانیم از شرط تودرتو استفاده کنیم.

در این حالت true را در چهارمین دستور قرار می‌دهیم، زیرا تنها زمانی دستور چهارم اجرا می‌شود که دستورهای یک تا سه اجرا نشده باشند. در این ساختار شرطی، دیگر نیازی به شمارنده نیست.

۴ برنامه را اجرا کنید و نتایج را با کد قبلی

مقایسه کنید.

روش سوم: در این روش از عملگرهای منطقی استفاده می‌کنیم.

```
Console.WriteLine(a + b > c && a + c > b && b + c > a);
```

اگر به عملگر && دقت کنید؛ متوجه می‌شوید که خروجی این عملگر زمانی true است که همه عملوندهای آن true باشد.

۵ برنامه را اجرا کرده، نتایج را دوباره با مراحل قبل مقایسه کنید.

- می‌خواهیم برنامه‌ای بنویسیم که با دریافت یک فصل، ماه‌های آن را تعیین کند.
- ۱ پروژه جدیدی با نام Season ایجاد کنید.
  - ۲ قطعه کد زیر را تکمیل کرده، در متد Main بنویسید.

```

if (season=="Bahar")
    Console.WriteLine("Farvardin, Ordibehesht, Khordad");
else if (season=="Tabestan")
    Console.WriteLine("Tir, Mordad, Shahrivar");
else if (season=="Paeiz")
    Console.WriteLine("Mehr, Aban, Azar");
else if (season=="Zemestan")
    Console.WriteLine("Dey, Bahman, Esfand");
else
    Console.WriteLine("Season Wrong");

```

### ۳ برنامه را اجرا و خروجی را مشاهده کنید.

برنامه ساده است اما نکته قابل توجه در این برنامه بررسی عبارت season در چندین حالت است. در زبان سی شارپ می‌توان برای بررسی حالت‌های مختلف یک عبارت از دستور switch استفاده کرد. ساختار این دستور به صورت زیر است :

### شکل کلی دستور switch

<pre> switch(عبارت) {     case مقدار ۱:         دستور ۱;         break;     case مقدار ۲:         دستور ۲;         break;     case مقدار ۳:         دستور ۳;         break;     default:         دستور ۴;         break; } </pre>	<p>اگر حاصل یا مقدار عبارت برابر مقدار ۱ بود دستور ۱ انجام شده و با دستور break از ساختار خارج می‌شود.</p> <p>اگر برابر مقدار ۲ بود دستور ۲ انجام شده، با دستور break از ساختار خارج می‌شود (برای بقیه مقادیر نیز همین روش اجرا می‌شود).</p> <p>در نهایت، در صورتی که حاصل برابر هیچ یک از مقادیر نباشد دستور ۴ (default) انجام می‌شود و با دستور break از ساختار خارج می‌شود.</p>
---	--

۴ برنامه زیر را جایگزین برنامه قبلی کرده، برخی دستورات ناقص را تکمیل کنید.

```
string season ;
Console.WriteLine("Enter Season:");
season = Console.ReadLine();
switch (season)
{
    case "Bahar":
        Console.WriteLine("Farvardin, Ordibehesht, Khordad");
        break;
    case "Tabestan":
        Console.WriteLine("Tir, Mordad, Shahrivar");
        break;
    case " Paeiz ":
        _____
        _____
    case "Zemestan":
        _____
        break;
    default:
        Console.WriteLine("Season Wrong");
        _____
}
```

۵ برنامه را اجرا و نتایج را با کد قبلی مقایسه کنید.

در دستور switch برای قرار دادن مجموعه‌ای از دستورات در یک case یا default، نیازی به بلاک نیست.

پایان پودمان ۵

موفق باشید

میرحسینی