

در این آموزش قصد داریم چگونگی ایجاد یک سیستم اعلام وضعیت آب و هوا را مشابه آنچه که در سایت [گوگل](#) می‌بینید برای شما توضیح دهیم. باید توجه داشت من این آموزش را با ASP.NET MVC نوشتم ولی شما می‌توانید با اندک تغییراتی در کدها، آنرا در ASP.NET وب فرمز نیز استفاده کنید. برای گرفتن آب و هوای هر شهر از Rss های اعلام وضعیت آب و هوای یاهو استفاده می‌کنم و توضیح خواهیم داد که چگونه با Rss آن کار کنید.

Rss آب و هوای هر شهر در یاهو به صورت یک لینک یکتا می‌باشد؛ به شکل زیر :

```
http://weather.yahooapis.com/forecastrss?w=WOEID&u=c
```

حال می‌خواهیم کوثری استرینگ‌های این لینک را برای شما توضیح دهیم. هر شهری بر روی کره‌ی زمین یک WOEID یکتا و منحصر بفرد دارد که شما به پارامتر w عدد WOEID شهر موردنظر خود را می‌دهید. بعد از مقداردهی پارامتر w، وقتی این لینک را در آدرس بار مرورگر خود می‌زنید، RSS مربوط به آب و هوای آن شهر را به شما می‌دهد. مثلاً WOEID تهران عدد 28350859 می‌باشد. و این لینک <http://weather.yahooapis.com/forecastrss?w=28350859&u=c> اطلاعات آب و هوای تهران را در قالب یک RSS به شما نمایش خواهد داد.

خوب، حالا پارامتر دوم یعنی پارامتر u چکاری را انجام می‌دهد؟

- \* چنانچه مقدار پارامتر u برابر c باشد، یعنی شما دمای آب و هوای شهر مد نظر را بر اساس سانتیگراد می‌خواهید.
- \* اگر مقدار پارامتر u برابر f باشد، یعنی شما دمای آب و هوای آن شهر مورد نظر را بر اساس فارنهایت می‌خواهید.

برای گرفتن WOEID شهرها هم به این سایت بروید <http://woeid.rosselliot.co.nz> و اسم هر شهری که می‌خواهید بزنید تا WOEID را به شما نمایش دهد.

در این مثال من از یک DropDown استفاده کردم که کاربر با انتخاب هر شهر از DropDown، آب و هوای آن شهر را مشاهده می‌کند. Action مربوط به صفحه‌ی Index به صورت زیر می‌باشد :

```
[HttpGet]
public ActionResult Index()
{
    ViewBag.ProvinceList = _RPosition.Positions;
    ShowWeatherProvince(8);
    return View();
}
```

در اینجا من لیست شهرها را از جدول می‌خوانم که البته این جدول را چون بخش مهمی نبود و فقط شامل ID و نام شهرها بود در فایل ضمیمه قرار ندادم و نام شهرها و ID آنها را بر عهده‌ی خودتان گذاشتم.

حال تابعی را که آب و هوای مربوط به هر شهر را نمایش می‌دهد، به شرح زیر است:

```
public ActionResult ShowWeatherProvince(int dpProvince)
{
    XmlDocument rssXml=null;
    CountryName CountryName = new CountryName();
    if (dpProvince != 0)
    {
        switch (dpProvince)
        {
            case 1:
            {
                rssXml =
                XmlDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345768&u=c");
```

```

CountryName = new CountryName() { Country = "Iran", City = "Azarbayejan-e
Sharqhi" };
break;
}
case 2:
{
    rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345767&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Azarbayejan-e
Qarbi" };
break;
}
case 3:
{
    rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2254335&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Ardebil" };
break;
}
case 4:
{
    rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=28350859&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Alborz" };
break;
}
case 5:
{
    rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345787&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Esfahan" };
break;
}
case 6:
{
    rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345775&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Ilam" };
break;
}
case 7:
{
    rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2254463&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Bushehr" };
break;
}
case 8:
{
    rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=28350859&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Tehran" };
break;
}
case 9:
{
    rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345769&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Chahar Mahall
va Bakhtiari" };
break;
}
case 10:
{
    rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=56189824&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Razavi
Khorasan" };
break;
}
case 11:
{
    rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345789&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Shomali
Khorasan" };
break;
}
case 12:
{
    rssXml =

```

```

XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345789&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Jonubi
Khorasan" };
        break;
    }
    case 13:
    {
        rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345778&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Khuzestan" };
        break;
    }
    case 14:
    {
        rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2255311&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Zanjan" };
        break;
    }
    case 15:
    {
        rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345784&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Semnan" };
        break;
    }
    case 16:
    {
        rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345770&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Sistan va
Baluchestan" };
        break;
    }
    case 17:
    {
        rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345772&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Fars" };
        break;
    }
    case 18:
    {
        rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=20070200&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Qazvin" };
        break;
    }
    case 19:
    {
        rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2255062&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Qom" };
        break;
    }
    case 20:
    {
        rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345779&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Kordestan" };
        break;
    }
    case 21:
    {
        rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2254796&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Kerman" };
        break;
    }
    case 22:
    {
        rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2254797&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Kermanshah" };
        break;
    }
    case 23:
    {
        rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345771&u=c");
CountryName = new CountryName() { Country = "Iran", City = "Kohgiluyeh va

```

```

Buyer Ahmad" };

                break;
            }
            case 24:
            {
                rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=20070201&u=c");
                CountryName = new CountryName() { Country = "Iran", City = "Golestan" };
                break;
            }
            case 25:
            {
                rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345773&u=c");
                CountryName = new CountryName() { Country = "Iran", City = "Gilan" };
                break;
            }
            case 26:
            {
                rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345782&u=c");
                CountryName = new CountryName() { Country = "Iran", City = "Lorestan" };
                break;
            }
            case 27:
            {
                rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345783&u=c");
                CountryName = new CountryName() { Country = "Iran", City = "Markazi" };
                break;
            }
            case 28:
            {
                rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345780&u=c");
                CountryName = new CountryName() { Country = "Iran", City = "Mazandaran" };
                break;
            }
            case 29:
            {
                rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2254664&u=c");
                CountryName = new CountryName() { Country = "Iran", City = "Hamedan" };
                break;
            }
            case 30:
            {
                rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345776&u=c");
                CountryName = new CountryName() { Country = "Iran", City = "Hormozgan" };
                break;
            }
            case 31:
            {
                rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2253355&u=c");
                CountryName = new CountryName() { Country = "Iran", City = "Yazd" };
                break;
            }
        }
        ViewBag.Location = CountryName;
        XNamespace yWeatherNS = "http://xml.weather.yahoo.com/ns/rss/1.0";
        List<YahooWeatherRssItem> WeatherList = new List<YahooWeatherRssItem>();
        for (int i = 0; i < 4; i++)
        {
            YahooWeatherRssItem YahooWeatherRssItem = new YahooWeatherRssItem()
            {
                Code = Convert.ToInt32(rssXml.Descendants("item").Elements(yWeatherNS +
"forecast").ElementAt(i).Attribute("code").Value),
                Day = rssXml.Descendants("item").Elements(yWeatherNS +
"forecast").ElementAt(i).Attribute("day").Value,
                Low = rssXml.Descendants("item").Elements(yWeatherNS +
"forecast").ElementAt(i).Attribute("low").Value,
                High = rssXml.Descendants("item").Elements(yWeatherNS +
"forecast").ElementAt(i).Attribute("high").Value,
                Text = rssXml.Descendants("item").Elements(yWeatherNS +
"forecast").ElementAt(i).Attribute("text").Value,
            };

            WeatherList.Add(YahooWeatherRssItem);
        }
    }

```

```

        ViewBag.FeedList = WeatherList;
    }

    return PartialView("_Weather");
}

```

قسمت SwitchCase، مقدار و Value مربوط به هر آیتم DropDown را که شامل یک اسم شهر است، میگیرد و RSS مربوط به آن شهر را بر میگرداند.

حالا کد مربوط به خواندن فایل Rss را برایتان توضیح می‌دهم: حلقه‌ی 0 تا 4 (که در کد بالا مشاهده می‌کنید) یعنی اطلاعات 4 روز آینده را برابیم برگردان.

من تگ‌های Code، Day، Low، High و text فایل RSS را در این حلقه For می‌خوانم که البته مقادیر این 4 روز را در لیستی اضافه می‌کنم که نوع این لیست هم از نوع YahooWeatherRssItem می‌باشد. من این کلاس را در فایل ضمیمه قرار دادم. اکنون هر کدام از این تگ‌ها را برایتان توضیح می‌دهم:

code: هر آب و هوا کدی دارد. مثلاً آب و هوای نیمه ابری یک کد، آب و هوای آفتابی کدی دیگر و ...

Low: حداقل دمای آن روز را به ما می‌دهد.

High: حداکثر دمای آن روز را به ما می‌دهد.

day: نام روز از هفته را بر می‌گرداند مثلاً شنبه، یکشنبه و ....

text: که توضیحاتی می‌دهد مثلاً اگر هوا آفتابی باشد مقدار sunny را بر می‌گرداند و ...

خوب، تا اینجا ما Rss مربوط به هر شهر را خواندیم حالا در قسمت Design باید چکار کنیم.

کدهای html صفحه‌ی Index ما شامل کدهای زیر است:

```

@{
    ViewBag.Title = "Weather";
}
<link href="~/Content/User/Weather/Weather.css" rel="stylesheet" />
@section scripts{
    <script src="@Url.Content("~/Scripts/jquery-1.6.2.min.js")" type="text/javascript"></script>
    <script src="@Url.Content("~/Scripts/jquery.unobtrusive-ajax.min.js")"
type="text/javascript"></script>
    <script type="text/javascript">
        $("#dpProvince").change(function () {
            $(this).parents("form").submit();
        });
    </script>
}
<h2>Weather</h2>
<div id="Progress">
    
</div>
<div id="BoxContent"> @Html.Partial("_Weather")</div>

    @using (Ajax.BeginForm(actionName: "ShowWeatherProvince", ajaxOptions: new AjaxOptions {
UpdateTargetId = "BoxContent", LoadingElementId = "Progress", InsertionMode = InsertionMode.Replace }))
    {
<div style="padding-top:15px;">
    <div style="float:left; width:133px;">Select Your Province</div>
    <div style="float:left"> @Html.DropDownList("dpProvince", new
SelectList(ViewBag.ProvinceList, "Id", "Name"),"Select Your Province", new { @class =
"webUserDropDown", @style = "width:172px" })</div>
    </div>
}

```

و کدهای Weather\_ که Partial است به صورت زیر است:

```

@{
    List<Weather.YahooWeatherRssItem> Feeds = ViewBag.FeedList;
}
<div>

```

```

@{
    HtmlString StartTable = new HtmlString("<table class='WeatherTable' cellspacing='0' cellpadding='0'><tbody><tr>");
    HtmlString EndTable = new HtmlString("</tr></tbody></table>");
    HtmlString StartTD = new HtmlString("<td>");
    HtmlString EndTD = new HtmlString("</td>");
}
<div style="width: 300px;">
    @{
        @StartTable
        foreach (var item in Feeds)
        {
            @StartTD
            <div>@item.Day</div>
            <div>
                @{
                    string FileName = "";
                    switch (item.Code)
                    {
                        case 0: { FileName = "/Images/User/Weather/Tornado.png"; break; }
                        case 1: { FileName = "/Images/User/Weather/storm2.gif"; break; }
                        case 2: { FileName = "/Images/User/Weather/storm2.gif"; break; }
                        case 3: { FileName = "/Images/User/Weather/storm2.gif"; break; }
                        case 4: { FileName = "/Images/User/Weather/15.gif"; break; }
                        case 5: { FileName = "/Images/User/Weather/29.gif"; break; }
                        case 6: { FileName = "/Images/User/Weather/29.gif"; break; }
                        case 7: { FileName = "/Images/User/Weather/29.gif"; break; }
                        case 8: { FileName = "/Images/User/Weather/26.gif"; break; }
                        case 9: { FileName = "/Images/User/Weather/drizzle.png"; break; }
                        case 10: { FileName = "/Images/User/Weather/26.gif"; break; }
                        case 11: { FileName = "/Images/User/Weather/18.gif"; break; }
                        case 12: { FileName = "/Images/User/Weather/18.gif"; break; }
                        case 13: { FileName = "/Images/User/Weather/19.gif"; break; }
                        case 14: { FileName = "/Images/User/Weather/19.gif"; break; }
                        case 15: { FileName = "/Images/User/Weather/19.gif"; break; }
                        case 16: { FileName = "/Images/User/Weather/22.gif"; break; }
                        case 17: { FileName = "/Images/User/Weather/Hail.png"; break; }
                        case 18: { FileName = "/Images/User/Weather/25.gif"; break; }
                        case 19: { FileName = "/Images/User/Weather/dust.png"; break; }
                        case 20: { FileName = "/Images/User/Weather/fog_icon.png"; break; }
                        case 21: { FileName = "/Images/User/Weather/hazy_icon.png"; break; }
                        case 22: { FileName = "/Images/User/Weather/2017737395.png"; break; }
                        case 23: { FileName = "/Images/User/Weather/32.gif"; break; }
                        case 24: { FileName = "/Images/User/Weather/32.gif"; break; }
                        case 25: { FileName = "/Images/User/Weather/31.gif"; break; }
                        case 26: { FileName = "/Images/User/Weather/7.gif"; break; }
                        case 27: { FileName = "/Images/User/Weather/38.gif"; break; }
                        case 28: { FileName = "/Images/User/Weather/6.gif"; break; }
                        case 29: { FileName = "/Images/User/Weather/35.gif"; break; }
                        case 30: { FileName = "/Images/User/Weather/7.gif"; break; }
                        case 31: { FileName = "/Images/User/Weather/33.gif"; break; }
                        case 32: { FileName = "/Images/User/Weather/1.gif"; break; }
                        case 33: { FileName = "/Images/User/Weather/34.gif"; break; }
                        case 34: { FileName = "/Images/User/Weather/2.gif"; break; }
                        case 35: { FileName = "/Images/User/Weather/freezing_rain.png"; break; }
                        case 36: { FileName = "/Images/User/Weather/30.gif"; break; }
                        case 37: { FileName = "/Images/User/Weather/15.gif"; break; }
                        case 38: { FileName = "/Images/User/Weather/15.gif"; break; }
                        case 39: { FileName = "/Images/User/Weather/15.gif"; break; }
                        case 40: { FileName = "/Images/User/Weather/12.gif"; break; }
                        case 41: { FileName = "/Images/User/Weather/22.gif"; break; }
                        case 42: { FileName = "/Images/User/Weather/22.gif"; break; }
                        case 43: { FileName = "/Images/User/Weather/22.gif"; break; }
                        case 44: { FileName = "/Images/User/Weather/39.gif"; break; }
                        case 45: { FileName = "/Images/User/Weather/thundershowers.png"; break; }
                        case 46: { FileName = "/Images/User/Weather/19.gif"; break; }
                        case 47: { FileName = "/Images/User/Weather/thundershowers.png"; break; }
                        case 3200: { FileName = "/Images/User/Weather/1211810662.png"; break; }
                    }
                }
                <img alt='@item.Text' title='@item.Text' src='@FileName'>
            </div>
            <div>
                <span>@item.High°</span>
                <span>@item.Low°</span>
            </div>
            @EndTD
        }
    }
    @EndTable
</div>

```

</div>

من عکس‌های مربوط به وضعیت آب و هوا را در فایل ضمیمه قرار دادم.  
چنانچه در مورد RSS وضعیت آب و هوای یاهو اطلاعات دقیق‌تری را می‌خواهید بدانید به این [لینک](#) بروید.  
در آموزش بعدی قصد دارم برایتان این بخش را توضیح دهم که بر اساس IP بازدید کننده سایت شما، اطلاعات آب و هوایی شهر  
بازدید کننده را برایش در سایت نمایش دهد.

<Files-06bf65bac63d4dd694b15fc24d4cb074.zip>

موفق باشید

## نظرات خوانندگان

نویسنده: وحید نصیری  
تاریخ: ۱۳۹۲/۱۲/۱۱ ۰:۵

ممنون از شما. چند نکته جزئی در مورد بهبود کیفیت کدها. در View های MVC باید کدنویسی صرفا به نمایش اطلاعات View خلاصه شود. یعنی switch داخل آن بهتر است تبدیل به یک extension method شده و نهایتا استفاده شود (برای تمیز کردن View). همچنین استفاده از switch 1 و 2 و 10000 اصطلاحا به magic code مشهور هستند. یعنی مشخص نیست معنای این اعداد چی هست. این ها را عموما بهتر است تبدیل به enum کرد و بعد استفاده نهایی.

نویسنده: شقایق اشتیری  
تاریخ: ۱۳۹۲/۱۲/۱۲ ۸:۳۹

ممنونم از شما که مشکل کد نویسی من را گفتید. extension method ها منظور تون همون helper ها هست ؟ لینکی دارید که در مورد این extension method آموزش داده باشد چون من می خوام کدهامو تا جایی که می شود بهینه بنویسم .

نویسنده: شاهین کیاست  
تاریخ: ۱۳۹۲/۱۲/۱۲ ۹:۱۵

یک مثال در [اینجا](#)

قسمت آخر ( If های شرطی را در View ها را در متدهای کمکی کپسوله کنید)

نویسنده: وحید نصیری  
تاریخ: ۱۳۹۲/۱۲/۱۲ ۹:۵۱

قسمت switch ایی را که در View نوشتید، تبدیل کنید به یک متد کمکی در کلاسی خارج از View: (مهم نیست [متد الحاقی](#) باشد یا خیر؛ فقط داخل View نباشد)

```
public static string GetFileName(int code)
{
    switch (code)
    {
        case 0: return "/Images/User/Weather/Tornado.png";
        //...
    }
}
```

بعد یک خاصیت محاسباتی به نام FileName به مدل مورد استفاده اضافه کنید:

```
public class YahooWeatherRssItem
{
    public int Code { get; set; }
    //...
    public string FileName
    {
        get { return Util.GetFileName(Code); }
    }
}
```

به این صورت View از کدهای محاسبات یافتن FileName خالی می شود.

نویسنده: علی  
تاریخ: ۱۳۹۲/۱۲/۱۲ ۱۰:۷

یه راه دیگه هم استفاده از Yahoo Query Language هست، می تونید با کوئری زیر اطلاعات آب و هوای شهر مورد نظر رو در قالب JSON دریافت کنید



```
select * from weather.forecast where woeid in (select woeid from geo.placefinder where text="CityName")
```

نویسنده: شقایق اشتری  
تاریخ: ۱۰:۱۳ ۱۳۹۲/۱۲/۱۲

یه سوال دیگه که داشتم اگر بین کدهایمان ، کد html وجود داشت باز هم باید آنها را از view جدا کنیم و به صورت extension method بنویسیم؟

یا اینکه جداسازی کدها از View را زمانی انجام می‌دهیم که چندین خط کد پشت سر هم نوشته باشیم و بین آنها کد html نباشد ؟

نویسنده: وحید نصیری  
تاریخ: ۱۰:۴۱ ۱۳۹۲/۱۲/۱۲

- نه الزاما. اگر چندین جا استفاده و تکرار می‌شود یا منطق طولانی دارد، روش «[نوشتن HTML Helpers ویژه، به کمک امکانات Razor](#)» می‌تواند مفید باشد.

- منطق قرار گرفته در View فقط باید کار «نهایی» نمایشی را انجام دهد. اگر در View مثلا مستقیما با دیتابیس کار می‌کنید، محاسبات مرتبط با فیلد خاصی را انجام می‌دهید و کلا هر منطقی که «نهایی» بودن نمایش اطلاعات را زیر سؤال ببرد، باید از View جدا شده و به کنترلر و زیرساخت آن منتقل شود.

نویسنده: محسن تقی پور  
تاریخ: ۰:۸ ۱۳۹۲/۱۲/۲۰

میشه راجبه این روش کمی بیشتر توضیح بدین

نویسنده: Anaritus  
تاریخ: ۰:۴۴ ۱۳۹۲/۱۲/۲۱

سلام  
ممنون از مقاله خوبتون  
امکانش هست سورتس کد webForm ش رو هم بذارید واسه دانلود؟  
ممنون

نویسنده: علی  
تاریخ: ۱۰:۸ ۱۳۹۲/۱۲/۲۱

[اینجا](#) یک مثال از نحوه کار با YQL رو توضیح داده

نویسنده: ع پارسایی  
تاریخ: ۱۶:۳۶ ۱۳۹۳/۱۱/۱۸

با سلام

امکان داره بگید در اکشن، قسمت ViewBag.ProvinceList = \_RPosition.Positions چه کار میکنه و RPosition کارش چیه؟

نویسنده: ع پارسایی  
تاریخ: ۱۹:۳۴ ۱۳۹۳/۱۱/۲۱

با سلام؛ در مطلب گفته شده برای گرفتن WOEID شهرها به فلان سایت بروید و هر کشوری رو بخواهیم باید بریم و دستی واردش اما من دیتابیس یا هر چیز دیگری رو میخوام که قابلیت این رو داشته باشه که کاربر با وارد کردن کشور و شهر بتونه آب و هوا رو

ببینه. شما میتونید راهنمایی ام کنید که چه کار باید بکنم و یا منبعی برای دیتابیس مختصات جغرافیایی کشورها وجود داره؟ ممنون میشم کمکم کنید. با تشکر