

RSS

- **How to use?**
- **How to implement?**

۳	مقدمه
۳	RSS (Really Simple Syndication) چیست ؟
۳	تاریخچه RSS
	RSS 0.90
	۳
	RSS 0.91
	۳
	RSS 1.0
	۴
	RSS 0.92
	۴
	RSS 2.0
	۴
۴	از چه نسخه ای از RSS استفاده کنیم ؟
۴	نحوه استفاده از RSS برای افراد عادی و مدیران پایگاههای اطلاع رسانی
۴	افراد عادی
۵	مدیران پایگاههای اطلاع رسانی
۵	نرم افزارهای گردآورنده خبر
۶	نقل اطلاعات RSS دیگران در پایگاه اطلاع رسانی خود
۷	استفاده از Web UserControl
۹	چگونگی استفاده از UserControl
۹	استفاده از caching
۱۱	ارائه RSS در پایگاه اطلاع رسانی
۱۱	نوشتن برنامه ای با استفاده از .Net
۱۱	NameSpace های مورد استفاده
۱۲	ایجاد و یا به روز رسانی فایل xml
۱۳	خواندن از روی پایگاه داده
۱۴	بستن تگ ها و پایان کار بروی فایل
۱۵	نسخه کامل برنامه
۱۶	تعیین درستی Feed تولید شده از پایگاه اطلاعاتی
۱۷	استانداردها و قواعد مورد استفاده در XML و RSS 2.0

۱۷----- XML به استانداردهای پایه ای

۱۷----- RSS 2.0 مبتنی بر Feed یک ایجاد برای موجود

۱۹----- RSS 2.0 استاندارد در Feed یک ساده از

امروزه با توجه به استفاده روز افزون از اینترنت و همچنین افزایش تعداد پایگاههای اطلاع رسانی که ارائه دهنده اخبار، اطلاعات تخصصی و... می باشند، نیاز به اشتراک گذاشتن اطلاعات بیش از پیش احساس می گردد. اما به اشتراک گذاشتن اطلاعات، بدون داشتن بستر فنی واحد و در عین حال استاندارد یکسان قابل انجام نخواهد بود. یکی از معمول ترین راههای استفاده مشترک از اطلاعات، قالب RSS می باشد که خود امروزه به عنوان یک استاندارد معرفی می گردد. برای پیاده سازی RSS، از زبان XML استفاده می شود که هم بوسیله بسیاری از زبانهای برنامه سازی پشتیبانی می گردد و هم سکوهای مختلف (Platform) برداشت یکسانی از آن دارند.

❏ RSS چیست ؟

قالبی برای دریافت آخرین اخبار از پایگاههای اطلاع رسانی. اما می توان تعریف دقیق تری به صورت زیر ارائه داد:

قالبی برای دریافت هر مطلبی که بتوان آن را به نوعی در قالب یک جدول جای داد. این قالب توسط مدیر پایگاه اطلاع رسانی تعریف شده و در یک فایل ذخیره می گردد و شما به عنوان کاربر می توانید از طریق برنامه هایی که برای خواندن این قالب طراحی شده اند، به فایل پایگاه اطلاع رسانی مورد نظر خود مراجعه کرده و آخرین اطلاعات و تغییرات آن را دریافت نمایید.

❏ تاریخچه RSS

از شرکت Netscape^۲ به عنوان مبتکر RSS یاد می شود. بعد ها شرکت Userland^۳ توسعه و کنترل نسخه های RSS را بر عهده گرفت. اما همزمان با این شرکت، یک گروه غیر تجاری نیز که مبنای کار خود را تعاریف اصلی ارائه شده از سوی Netscape قرار داده بود، نسخه جدیدی برای RSS ارائه داد که این نسخه با استقبال Userland روبه رو نگردید. Userland به توسعه نسخه اصلی خود ادامه داد تا RSS V2.0 را ارائه کرد.

RSS 0.90

اولین بار Netscape برای پخش اخبار، مطالب و اطلاعات جدید خود، RSS را پایه گذاری کرد. این نسخه که RSS 0.90 نامیده می شد، در پایگاه my.netscape.com مورد استفاده قرار گرفت. نکته مهم در مورد این نسخه، پشتیبانی آن از scriptingNews (قدیمی ترین قالبی که بنگاههای خبری برای به اشتراک گذاشتن اخبار از آن استفاده می کردند) می باشد. (۳/۱۵/۱۹۹۹)

RSS 0.91

^۱ Really Simple Syndication

^۲ <http://netscape.com>

^۳ <http://userland.com>

Netscape نسخه جدید RSS را بر مبنای scriptingNews 2.0b1 ارائه داد. در این نسخه گامهای اصلی استاندارد سازی این قالب برداشته شد و بسیاری از تگ (tag) های قالب <scriptingNews> مورد استفاده قرار گرفت (7/10/1999). پس از این Netscape کار بر روی RSS را رها کرد و Userland با اعمال تغییراتی بر روی RSS 0.91، این نسخه را ارائه کرد (6/4/2000)

RSS 1.0

این نسخه که ارتباط چندانی با نسخه های قبلی RSS نداشت، توسط یک گروه خصوصی و با هدایت Real Dornfest ارائه شد. این نسخه که به نسبت نسخه های قبلی از پیچیدگی های بیشتری برخوردار بود، توسط Userland به عنوان نسخه جدید RSS پذیرفته نشد (8/14/2000).

RSS 0.92

در حقیقت این نسخه همان RSS 0.91 می باشد که تعدادی عناصر اختیاری به آن افزوده شده است. طراح آن Dave Winer طراح اصلی ScriptingNews از Userland می باشد (12/25/2000).

RSS 2.0

Dave Winer پس از ترک Userland این نسخه را طراحی کرد (9/18/2002). سپس مشخصات RSS 2.0 توسط دانشگاه هاروارد و تحت مجوز پدید آورندگان آن منتشر گردید (7/15/03).

■ از چه نسخه ای از RSS استفاده کنیم؟

پایگاه اطلاع رسانی syndic8.com اعلام کرده است که بیش از ۵۰ درصد از feed های موجود از RSS 0.91 استفاده می کنند، تنها ۲۵ درصد، RSS 1.0 را مورد استفاده قرار می دهند و ۲۵ درصد باقی مانده از قالب RSS 9.x و یا RSS 2.0 استفاده می کنند.

با این حال، گفتن این مطلب که کدامیک از این نسخه ها به نسبت دیگری در وضعیت بهتری قرار می گیرد، کار چندان ساده ای نمی باشد و با توجه به اینکه امروزه نرم افزار های موجود، تمامی قالب ها را پشتیبانی می کنند، این لحاظ مشکلی برای کاربران ایجاد نخواهد کرد. اما به واسطه اینکه RSS 2.0 آخرین نسخه ارائه شده می باشد و RSS 1.0 تنها نسخه ای است که از استانداردهای دیگر نسخه ها پیروی نمی کند، ما از RSS 2.0 استفاده و آن را توصیه می نمایم.

■ نحوه استفاده از RSS برای افراد عادی و مدیران پایگاههای اطلاع رسانی

• افراد عادی

همانطور که در تعریف RSS ذکر شده است، برای مطالعه اطلاعات و اخبار از طریق RSS، احتیاج به نرم افزارهای خاصی می باشد که اصطلاحاً گردآور خبر (News aggregators) نامیده می شوند. این نرم افزار این امکان را به کاربران می دهد که کانال های مورد علاقه خود را (پایگاههای اطلاع رسانی که مطالب

خود را به صورت RSS نیز ارائه می دهند (اضافه کرده و در یک بازه زمانی دلخواه ، آخرین تغییرات را مشاهده نمایند .

به همین منظور نرم افزار های مختلفی ارائه شده اند که در بخش نرم افزار می توانید نام آن ها را مشاهده نمایید .

• مدیران پایگاههای اطلاع رسانی

مدیران پایگاههای اطلاع رسانی به دو صورت می توانند از RSS استفاده نمایند :

۱- برای نقل اطلاعات و اخبار از پایگاههای دیگر

با استفاده از بخش " انتقال اطلاعات RSS دیگران در پایگاه اطلاع رسانی " مدیر سیستم می تواند از اطلاعات و اخبار سایتهای دیگر در پایگاه خود استفاده نماید .

۲- برای ارائه اطلاعات و اخبار پایگاه خود در قالب RSS

با استفاده از بخش " ارائه RSS در پایگاه اطلاع رسانی " ، مدیر سیستم می تواند اطلاعات و اخبار پایگاه خود را در این قالب در دسترس سایرین قرار دهد.

■ نرم افزار های گردآورنده خبر

نرم افزار های مختلفی برای جمع آوری و خواندن اطلاعات پایگاههای ارائه دهنده RSS وجود دارد که ما ۳ نمونه از آنها را در اینجا ذکر می کنیم :

▪ NewsDesk 1.0

▪ RssReader 1.0.72

▪ Active Web Reader 2.14

تمامی برنامه های فوق از طریق پایگاه <http://download.com> قابل دریافت می باشند

نقل اطلاعات RSS دیگران در پایگاه اطلاع رسانی خود

ساده ترین نوع استفاده از RSS در پایگاه اطلاع رسانی ، نمایش آخرین اخبار پایگاههای دیگر می باشد . در این حالت کافی است که محلی برای نمایش اطلاعات پایگاه مورد نشر در نظر بگیرید و بسته به اینکه بخواهید فایل مورد نظر چگونه نمایش داده شود ، آن را سفارشی نمایید .

به عنوان مثال :

برای نمایش اخبار منتشر شده در پورتال استان سیستان و بلوچستان در پایگاه خود، در ساده ترین حالت ، کافی است که یک صفحه .aspx ایجاد نمایید . سپس یک Datagrid در صفحه قرار دهید :

```
<asp:DataGrid id="dgRss" runat="server" AutoGenerateColumns="False"
BorderColor="#DEBA84" BorderStyle="None" CellSpacing="2" BorderWidth="1px"
BackColor="#DEBA84" CellPadding="3" dir=rtl >
<SelectedItemStyle Font-Bold="True" ForeColor="White"
BackColor="#738A9C"></SelectedItemStyle>
<ItemStyle ForeColor="#8C4510" BackColor="#FFF7E7"></ItemStyle>
<HeaderStyle Font-Bold="True" ForeColor="White"
BackColor="#A55129"></HeaderStyle>
<FooterStyle ForeColor="#8C4510" BackColor="#F7DFB5"></FooterStyle>
<Columns>
<asp:TemplateColumn HeaderText="title">
<ItemTemplate>
<a href=<%=# DataBinder.Eval(Container, "DataItem.link") %>><%=#
DataBinder.Eval(Container, "DataItem.title") %></a>
</ItemTemplate>
</asp:TemplateColumn>
<asp:BoundColumn DataField="description"
HeaderText="Description"></asp:BoundColumn>
</Columns>
<PagerStyle HorizontalAlign="Center" ForeColor="#8C4510"
Mode="NumericPages"></PagerStyle>
</asp:DataGrid>
```

جدول 1

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;
using System.Xml;
namespace rss
{
```

```

/// <summary>
/// Summary description for WebForm1.
/// </summary>
public class WebForm1 : System.Web.UI.Page
{
    protected System.Web.UI.WebControls.DataGrid dgPost;

    private void Page_Load(object sender, System.EventArgs e)
    {
        dgPost.DataSource=GetRss("http://sbportal.ir/rss.xml");
        dgPost.DataBind();
    }
    public DataTable GetRss(string str)
    {
        XmlTextReader xmlr = new XmlTextReader(str);
        DataSet ds = new DataSet();
        ds.ReadXml(xmlr);
        return ds.Tables[2];
    }
}

```

جدول 2

همانطور که مشخص است ، Datagrid تعریف شده برای نمایش عنوان و توضیح خبر مورد استفاده قرار می گیرد . این ساده ترین حالت ممکن برای استفاده از xml پایگاههای دیگر می باشد.

در این حالت می توان به شی " XmlTextReader " نشانی فایل مورد نظر را ارسال کرد . سپس فایل را از طریق متد ReadXml در DataSet قرار داد. نکته مهم این است که در هنگام خواندن xml ، DataSet به ازای هر سطح از xml یک DataTable ایجاد می کند . بنابراین عناصر <rss> و <channel> هر کدام به عنوان یک جدول در DataSet شناخته خواهند شد . می توان نتیجه گرفت که برای نمایش اطلاعات <channel> (شامل زبان ، ttl ، copyright و ...) کافی است که اطلاعات جدول شماره ۱ را استخراج نمایید .

استفاده از Web UserControl

با استفاده از روش بالا می توان اطلاعات را به درستی نمایش داد . اما استفاده از UserControl این امکان را می دهد که بتوانیم بدون تغییر محسوس در صفحات، کدی ایجاد نماییم که هم قابلیت استفاده مجدد ، بدون تکرار کردن کد ، را داشته باشد و هم تغییر و کنترل آن ساده و از یک فایل، برای تمامی صفحات باشد .

برای اینکار ابتدا با استفاده از Visual Studio فایلی با عنوان rssFeed.ascx می سازیم و مجدداً روند ذکر شده در روش قبل را تکرار می کنیم .

```

<%@ Control Language="c#" AutoEventWireup="false" Codebehind="rssFeed.ascx.cs"
Inherits="rss.RssFeed" TargetSchema="http://schemas.microsoft.com/intellisense/ie5"%>

```

```

<asp:DataGrid id="dgRss" runat="server" AutoGenerateColumns="False"
BorderColor="#DEBA84" BorderStyle="None" CellSpacing="2" BorderWidth="1px"

```

```

BackColor="#DEBA84" CellPadding="3" dir=rtl >
<SelectedItemStyle Font-Bold="True" ForeColor="White"
BackColor="#738A9C"></SelectedItemStyle>
<ItemStyle ForeColor="#8C4510" BackColor="#FFF7E7"></ItemStyle>
<HeaderStyle Font-Bold="True" ForeColor="White"
BackColor="#A55129"></HeaderStyle>
<FooterStyle ForeColor="#8C4510" BackColor="#F7DFB5"></FooterStyle>
<Columns>
<asp:TemplateColumn HeaderText="title">
<ItemTemplate>
<a href=<%# DataBinder.Eval(Container, "DataItem.link") %>><%#
DataBinder.Eval(Container, "DataItem.title") %></a>
</ItemTemplate>
</asp:TemplateColumn>
<asp:BoundColumn DataField="description"
HeaderText="Description"></asp:BoundColumn>
</Columns>
<PagerStyle HorizontalAlign="Center" ForeColor="#8C4510"
Mode="NumericPages"></PagerStyle>
</asp:DataGrid>

```

جدول 3

```

namespace rss
{
    using System;
    using System.Data;
    using System.Drawing;
    using System.Web;
    using System.Web.UI.WebControls;
    using System.Web.UI.HtmlControls;
    using System.Xml;

    /// <summary>
    /// Summary description for RssFeed.
    /// </summary>
    public class RssFeed : System.Web.UI.UserControl
    {

        private void Page_Load(object sender, System.EventArgs e)
        {
            dgPost.DataSource=GetRss("http://sbportal.ir/rss.xml");
            dgPost.DataBind();
        }
        public DataTable GetRss(string str)
        {
            XmlTextReader xmlr = new XmlTextReader(str);
            DataSet ds = new DataSet();
            ds.ReadXml(xmlr);
        }
    }
}

```

```

return ds.Tables[2];
    }
}
}

```

جدول 4

چگونگی استفاده از UserControl

نکته مهم در مورد UserControl نحوه استفاده از این فایل می باشد. برای استفاده از UserControl در صفحات دیگر، احتیاج به معرفی UserControl ایجاد شده داریم که از طریق کد زیر قابل انجام می باشد.

```
<%@ Register TagPrefix="rss" TagName="rssFeed" Src="ssFeed.ascx" %>
```

با استفاده از این کد می توان کنترل جدید را در هر صفحه.aspx که می خواهید از آن استفاده کنید، معرفی نمایید. به عنوان مثال:

```

<%@ Page language="c#" Codebehind="View.aspx.cs" AutoEventWireup="false"
Inherits="rss.WebForm1" %>
<%@ Register TagPrefix="rss" TagName="rssFeed" Src="rssFeed.ascx" %>
<HTML>
  <HEAD>
  </HEAD>
  <body>
    <form id="frm" method="post" runat="server">
      <rss:rssFeed id="ucRssFeed" runat="Server" />
    </form>
  </body>
</HTML>

```

جدول 5

استفاده از caching

مطمئناً در هنگام نمایش اطلاعات سایرین در پایگاه خود، بهتر است که به ازای هر بار، بارگذاری صفحه، به پایگاه ارائه دهنده RSS مراجعه نکنیم. بلکه در یک مدت زمان مشخص تغییر اطلاعات را چک نماییم. یکی از ساده ترین روشها Cache کردن اطلاعات RSS برای یک مدت زمان دلخواه می باشد تا هر زمان که این بازه زمانی به پایان می رسد بتوانیم اطلاعات Feed را مجدداً از پایگاه ارائه دهنده بخوانیم. برای این کار می توانیم به صورت زیر عمل نماییم:

```

namespace rss
{
    using System;

```

```

using System.Data;
using System.Drawing;
using System.Web;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;
using System.Xml;

/// <summary>
///         Summary description for RssFeed.
/// </summary>
public class RssFeed : System.Web.UI.UserControl
{

    private void Page_Load(object sender, System.EventArgs e)
    {
        if (Cache ["RSSFeed"]==null)
        {
            DataTable dt=GetRss("http://localhost/rss/rss.xml");

Cache.Insert ("RSSFeed",dt,null,DateTime.Now.AddMinutes (20) ,TimeSpan.Zero)
;

        }
        dgPost.DataSource = Cache["RSSFeed"];
        dgPost.DataBind();
    }
    public DataTable GetRss(string str)
    {
        XmlTextReader xmlr = new XmlTextReader(str);
        DataSet ds = new DataSet();
        ds.ReadXml(xmlr);
        return ds.Tables[2];
    }

}
}

```

ارائه RSS در پایگاه اطلاع رسانی

در این قسمت شما به عنوان مدیر یک پایگاه اطلاع رسانی که می خواهید بخش RSS اخبار پایگاه خود را در اختیار سایرین (دیگر پایگاههای اطلاع رسانی و یا خوانندگان خبر از طریق برنامه گردآور خبر) News aggregator (قرار دهد .

پیش از شروع ، لازم است که قوانین پایه ای مربوط به xml و RSS 2.0 را مطالعه نماید. حال جدولی از پایگاه داده (Data base) که می خواهید تغییرات آن را نمایش دهید انتخاب نمایید برای مثال ما در اینجا از جدولی به نام SMP_NEWS استفاده می کنیم که شامل فیلدهای زیر است :

```
NEWS_ID
NEWS_TITLE
NEWS_LEAD
NEWS_IMAGE_PATH
NEWS_DATE
```

جدول 7

در بخش بعدی با استفاده از .NET. برنامه ای خواهیم نوشت تا فایل RSS جدول بالا را تولید نماید .

نوشتن برنامه ای با استفاده از .Net. برای ایجاد Feed(Feed یک فایل xml است که از پایگاه داده خود می سازید تا از آن برای ارائه RSS استفاده شود .

توضیح : نکات تکمیلی در مورد هر کد در برابر آن ارائه شده است .

Namespace های مورد استفاده

مشخص است که برای کار با داده های xml احتیاج به System.Xml خواهیم داشت و چون به کلاس Encoding نیز احتیاج داریم باید از Namespace مربوط به آن System.Text استفاده کنیم .

```
using System;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.Text;
using System.Web;
using System.Xml;
```

جدول 8

آنچه در Page_load لازم داریم

- ۱- ایجاد و یا به روز رسانی فایل xml
- ۲- اتصال به پایگاه داده برای خواندن اطلاعات مورد نظر
- ۳- بستن تگها و اتمام کار بر روی فایل

۱- ایجاد و یا به روز رسانی فایل xml

```

XmlTextWriter xmlWriter = new XmlTextWriter(Server.MapPath("rss.xml"),
,Encoding.UTF8);
تعریف شی از نوع xmlWriter این امکان را داریم که یک فایل xml ایجاد کرده و یا اطلاعات آن را به روز نماییم. با
استفاده از این شی می توان Encoding فایل را نیز تعیین کرد.

xmlWriter.WriteStartDocument();
شروع فایل و افزودن مشخصات لازم برای فایل xml نظیر نسخه 1.0 برای تگ xml

xmlWriter.WriteStartElement("rss");
افزودن تگ شروع کننده عنصر <rss> به عنوان اولین عنصر در فایل

xmlWriter.WriteAttributeString("version", "2.0");
افزودن مشخصه "نسخه" برای عنصر <rss>

xmlWriter.WriteStartElement("channel");
افزودن تگ شروع کننده عنصر <channel> برای معرفی پایگاه ارائه دهنده RSS

xmlWriter.WriteElementString("title", "Sbportal NewsWriter");
افزودن عنصر <title> به عنوان فرزند <channel> برای نمایش عنوان پایگاه ارائه دهنده RSS. به عنوان مثال
sbportal نمایان گر این مطلب است که این feed متعلق به سایت sbportal.ir می باشد.

xmlWriter.WriteElementString("link", "http://sbportal.ir");
افزودن عنصر <link> به عنوان فرزند <channel> برای نمایش آدرس پایگاه ارائه دهنده RSS

xmlWriter.WriteElementString("description", "بلوچستان و سیستان استان پورتال");
افزودن عنصر <description> به عنوان فرزند <channel> برای نمایش توضیحات پایگاه ارائه دهنده RSS
    
```

```
xmlWriter.WriteElementString("copyright","(c) 2004, SBPORTAL.IR. All rights reserved.");
```

افزودن عنصر <copyright> به عنوان فرزند <channel> برای نمایش حقوق پدید آورندگان پایگاه ارائه دهنده RSS

```
xmlWriter.WriteElementString("ttl","25");
```

افزودن عنصر <ttl> (Time To Live) به عنوان فرزند <channel> برای اینکه نرم افزار گردآور خبر بداند پس از چه مدت زمانی باید فایل مورد نظر را برای اطلاعات جدید چک نماید. به عبارت دیگر پس از چه مدت زمانی اطلاعات موجود "قدیمی" به حساب می آیند و دوباره باید به روز شوند. <ttl> بر حسب دقیقه می باشد.

جدول 9

۲- خواندن از روی پایگاه داده

```
SqlConnection sqlCn = new SqlConnection(ConfigurationSettings.AppSettings["ConnectionString"]);
```

خواندن رشته حاوی اطلاعات اتصال به پایگاه داده از web.config

```
string sqlNews = "SELECT TOP 15 NEWS_ID, NEWS_LEAD, NEWS_TITLE, NEWS_DATE FROM SMP_NEWS WHERE STATUS_ID=2 ORDER BY NEWS_DATE DESC";
```

تعریف پرس وجو برای نمایش ردیفهای مورد نظر

```
sqlCn.Open();
SqlCommand sqlcmdNews = new SqlCommand(sqlNews, sqlCn);
SqlDataReader drNews = sqlcmdNews.ExecuteReader();
```

برقراری اتصال با پایگاه داده و شروع خواندن اطلاعات

```
while (drNews.Read())
{
    xmlWriter.WriteStartElement("item");
```

افزودن تگ شروع کننده عنصر <item> و مطالب آن به ازای هر خبر خوانده شده از پایگاه

```
    xmlWriter.WriteElementString("title", drNews.GetString(drNews.GetOrdinal("NEWS_TITLE")));
```

افزودن عنصر <title> به عنوان فرزند عنصر <item> به ازای هر خبر خوانده شده از پایگاه که حاوی عنوان خبر خوانده

```

        شده می باشد .

xmlWriter.WriteElementString("description",drNews.GetString
        (drNews.GetOrdinal("NEWS_LEAD")));
افزودن عنصر <description> به عنوان فرزند عنصر <item> به ازای هر خبر خوانده شده از پایگاه که حاوی خلاصه
        خبر خوانده شده می باشد .

xmlWriter.WriteElementString("link",
        "http://sbportal.ir/newsdetail_afa_ni_"+drNews.GetInt32
        (drNews.GetOrdinal("NEWS_ID")).ToString()+".html");
افزودن عنصر <link> به عنوان فرزند عنصر <item> به ازای هر خبر خوانده شده از پایگاه که حاوی آدرس خبر بوده و
        برای نمایش متن کامل خبر در مرور گر مورد استفاده قرار می گیرد.

xmlWriter.WriteElementString("pubDate", drNews.GetDateTime
        (drNews.GetOrdinal("NEWS_DATE")).ToString("R"));

افزودن عنصر <pubDate> به عنوان فرزند عنصر <item> به ازای هر خبر خوانده شده از پایگاه حاوی تاریخ خبر خوانده
        شده می باشد . این تاریخ باید به صورت استاندارد 822 RTC نوشته شده باشد.

xmlWriter.WriteEndElement();
        افزودن تگ پایانی </item> در پایان هر خبر
    }
    
```

جدول 10

و در انتها : بستن تگ ها و پایان کار بروی فایل

```

xmlWriter.WriteEndElement();
        افزودن تگ پایانی </channel>

xmlWriter.WriteEndElement();
        افزودن تگ پایانی </rss>

xmlWriter.WriteEndDocument();
        بستن تمامی تگ ها و یا مشخصه ها که بسته نشده اند .

xmlWriter.Close();
        پایان کار با فایل
    
```

جدول 11

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;
using System.Xml;
using System.IO;
using System.Text;
using System.Configuration;
namespace rss
{
    /// <summary>
    /// Summary description for createXml.
    /// </summary>
    public class createXml : System.Web.UI.Page
    {
        private void Page_Load(object sender, System.EventArgs e)
        {
            XmlTextWriter xmlWriter = new XmlTextWriter(
                Server.MapPath("rss.xml"), Encoding.UTF8);
            xmlWriter.WriteStartDocument();
            xmlWriter.WriteStartElement("rss");
            xmlWriter.WriteAttributeString("version", "2.0");
            xmlWriter.WriteStartElement("channel");
            xmlWriter.WriteElementString("title", "Sbportal News");
            xmlWriter.WriteElementString("link", "http://sbportal.ir");

            xmlWriter.WriteElementString("description",
                "بلوچستان و سیستان استان پورتال");
            xmlWriter.WriteElementString("copyright",
                "(c) 2004, SBPORTLA.IR. All rights reserved.");
            xmlWriter.WriteElementString("ttl", "25");
            SqlConnection sqlCn = new SqlConnection(ConfigurationSettings
                .AppSettings["ConnectionString"]);
            sqlCn.Open();
            string sqlNews = "SELECT TOP 15 NEWS_ID, NEWS_LEAD, NEWS_TITLE,
                NEWS_DATE FROM SMP_NEWS WHERE STATUS_ID=2 ORDER BY NEWS_DATE
                DESC";
            SqlCommand sqlcmdNews = new SqlCommand(sqlNews, sqlCn);
            SqlDataReader drNews = sqlcmdNews.ExecuteReader();
            while (drNews.Read())
            {
```

```

        xmlWriter.WriteStartElement("item");
        xmlWriter.WriteElementString("title", drNews.GetString
            (drNews.GetOrdinal("NEWS_TITLE")));
        xmlWriter.WriteElementString("description", drNews.GetString
            (drNews.GetOrdinal("NEWS_LEAD")));
        xmlWriter.WriteElementString("link",
            "http://sbportal.ir/newsdetail_afa_ni_"+drNews.GetInt32
            (drNews.GetOrdinal("NEWS_ID")).ToString()+".html");

        xmlWriter.WriteElementString("pubDate", drNews.GetDateTime
            (drNews.GetOrdinal("NEWS_DATE")).ToString("R"));
        xmlWriter.WriteEndElement();
    }

    drNews.Close();
    sqlCn.Close();

    xmlWriter.WriteEndElement();
    xmlWriter.WriteEndElement();
    xmlWriter.WriteEndDocument();
    xmlWriter.Close();

}
}
}

```

جدول 12

پس از اجرای این برنامه کوتاه می توانید فایل xml ایجاد شده که در ریشه (Root) پایگاه اطلاع رسانی قرار دارد ، در اختیار دیگران قرار دهید .

تعیین درستی Feed تولید شده از پایگاه اطلاعاتی :

هرگز فراموش نکنید که در انتهای کار، اطلاعات تولیدی خود را از جهت درستی و استاندارد بودن Feed ایجاد شده ، بیازمایید. به همین دلیل نرم افزارها و پایگاههایی وجود دارند که عملیات تعیین درستی فایل xml شما را انجام می دهند. به عنوان مثال از پایگاه RSS Validator⁴ می توانید استفاده کنید. برای استفاده از خدمات این پایگاه ، کافی است که آدرس فایل xml پایگاه خود را وارد کرده و نتیجه نهایی را دریافت کنید. از دیگر پایگاههای ارائه دهنده این سرویس Feed Validator⁵ می باشد .

⁴ <http://rss.scripting.com/>

⁵ <http://feeds.archive.org/validator/>

استانداردها و قواعد مورد استفاده در XML و RSS 2.0

عناوین

- نگاهی به استانداردهای پایه ای XML
- نگاهی به استانداردهای موجود برای ایجاد یک Feed مبتنی بر RSS 2.0
- نمونه ای ساده از یک Feed استاندارد در RSS 2.0

📌 نگاهی به استانداردهای پایه ای XML

با توجه به اینکه RSS از فایل‌های XML استفاده می کند ، پیش از شروع ، چند قانون ساده در مورد فایل‌های XML را به خاطر بسپارید :

- ۱- محتویات فایل‌های XML به بزرگی و کوچکی حروف حساس می باشند. بنابراین <item> با <Item> یکسان نمی باشد.
- ۲- عناصر در هر فایل xml ، باید به درستی در داخل یکدیگر قرار داده شوند. در نظر داشته باشید که چگونه قرار گرفتن عناصر در فایل xml ، تاثیر مستقیمی بر استاندارد بودن Feed بر حسب نسخه آن دارد .
- ۳- کاراکترهایی نظیر < ، > ، & ، " و ' در صورتی که در داخل متن و مقادیر صفات باشند باید ، با < ، > ، " و ' جایگزین گردند .
- ۴- تمامی مقادیر مشخصه ها ، در داخل " " قرار داده می شوند . به عنوان مثال <rss version="2.0"> یک نمونه صحیح می باشد.

📌 نگاهی به قواعد موجود برای ایجاد یک Feed مبتنی بر RSS 2.0

- ۱- فایل RSS با عنصر <rss> آغاز شده و با </rss> پایان می یابد .

```
<rss version="2.0">
...
</rss>
```

جدول 13

- ۲- تنها عنصر فرزند <rss> ، <channel> نامیده می شود که دارای ۳ عنصر اجباری می باشد . در نظر داشته باشید که <channel> ، در حقیقت همان عنصر اصلی است که اطلاعات پایگاه ما را در بر می گیرد. در نرم افزار های گردآورنده خبر (News aggregator) نیز، از هر پایگاه جدید به عنوان یک channel

جدید یاد می شود . عناصر <channel> شامل عبارتند از title ، link ، description که در جدول زیر موارد استفاده آنها بیان شده است .

title :	عنوان feed که اغلب شامل نام پایگاه اطلاع رسانی می باشد .
link :	آدرس پایگاه اطلاع رسانی
description:	توضیحات مربوط به پایگاه اطلاع رسانی

جدول 14

در مشخصات ارائه شده برای RSS 2.0⁶ عناصر اختیاری دیگری نیز وجود دارد نظیر :

عنوان	توضیح
language	زبانی که کانال به آن زبان نوشته شده است. دانستن زبان این امکان را به نرم افزارهای گردآورنده خبر می دهد تا کانال هایی که زبان یکسانی دارند را در یک صفحه نمایش دهد و یا کاراکترهای مخصوص آن زبان را لحاظ کند.
copyright	نمایش دهنده پدید آورنده کانال
managingEditor	آدرس پست الکترونیکی فردی که مسوولیت اخبار و مقالات را بر عهده دارد .
webMaster	آدرس پست الکترونیکی فردی که مسوولیت مسائل فنی کانال را بر عهده دارد .
pubDate	زمانی که کانال به روز می گردد این تاریخ با استفاده از RFC 822 نوشته می شود.
lastBuildDate	تاریخ آخرین بار مرتبه ای که اطلاعات کانال به روز شده است . این تاریخ تاریخ با استفاده از RFC 822 نوشته می شود .
category	طبقه بندی محتوایی کانال. اینکه کانال حاوی چه مطالبی می باشد.
generator	بیان کننده نرم افزاری که کانال را تولید کرده است
docs	آدرسی که به مستندات نسخه ای از RSS اشاره می کند که در کانال مورد استفاده قرار گرفته است.
ttl	Time To Live یا Ttl بیان کننده مدت زمانی است که گردآورنده خبر پس از ان زمان مجددا فایل مربوط به کانال را برای به روز رسانی فایل خود چک کرده و اطلاعات فعلی را "قدیمی" به شمار می آورد .
image	فایلی از نوع GIF، JPEG، و یا PNG که به همراه کانال نمایش داده می شود .
rating	PICS rating(Platform for Internet Content Selection) مربوط به کانال
textInput	نمایش دهنده یک Text Box به همراه کانال که می تواند کاربری های متفاوتی داشته باشد از جمله دریافت نظر مشاهده کنندگان کانال .

⁶ <http://blogs.law.harvard.edu/tech/rss>

skipHours	با این عنصر به نرم افزار گردآورنده خبر گفته می شود که در چه ساعتی کانال مورد نظر را برای اطلاعات جدید چک نکند.
skipDays	با این عنصر به نرم افزار گردآورنده خبر گفته می شود که در چه روزی کانال مورد نظر را برای اطلاعات جدید چک نکند .

جدول 15

۳- هر اطلاع و یا خبر داخل، یک عنصر <item> قرار گرفته می شود. این عنصر در داخل <channel> قرار داده می شود. در نظر داشته باشید، تعداد عناصر <item> در هر <channel> دلخواه می باشد . به جز یکی از دو عنصر title و یا ، description بقیه موارد اختیاری می باشند .
عناصر معمول در <item> شامل موارد زیر می باشند :

title :	عنوان
link :	آدرس
description:	توضیحات
author :	نویسنده
pubDate	تاریخ انتشار

جدول 16

نکته: pubDate از قالب RFC 822 پیروی می کند. ۳ حرف معادل روز (اختیاری) سپس "،" ، تاریخ روز، سپس ، ۳ حرف به عنوان نماد ماه سپس سال و در انتها ساعت و Time Zone به عنوان مثال Sat, 07 Sep 2002 00:00:01 GMT

نمونه ای ساده از یک Feed استاندارد در RSS 2.0

```
<rss version="2.0">
  <channel>
    <title>Latest DataWebControls.com FAQs</title>
    <link>http://datawebcontrols.com</link>
    <description>
      This is the syndication feed for the FAQs
      at DataWebControls.com
    </description>
    <item>
      <title>Working with the DataGrid</title>
      <link>http://datawebcontrols.com/faqs/DataGrid.aspx</link>
      <pubDate>Mon, 07 Jul 2003 21:00:00 GMT</pubDate>
    </item>
    <item>
      <title>Working with the Repeater</title>
      <description>
        This article examines how to work with the Repeater control.
      </description>
      <link>http://datawebcontrols.com/faqs/Repeater.aspx</link>
      <pubDate>Tue 08 Jul 2003 12:00:00 GMT</pubDate>
    </item>
  </channel>
</rss>
```

```
</item>  
</channel>  
</rss>
```