

BUỔI 8- BÀI THỰC HÀNH SỐ 8

1. Lý thuyết
2. Thực hành

I - LÝ THUYẾT - DOM – DOCUMENT OBJECT MODEL (Yêu cầu sinh viên chuẩn bị trước)

1 - Giới thiệu

- DOM - MÔ HÌNH ĐỐI TƯỢNG VĂN BẢN
- DOM – phần quan trọng nhất của tài liệu XML
- DOM là phương tiện cho phép truy xuất (đưa ra) và chỉnh sửa (Thêm xóa sửa) trên tài liệu XML
- DOM cho phép xử lý tài liệu như 1 đối tượng, có thể rút trích thông tin, thay đổi và truy vấn tài liệu thông qua mã lệnh.
- Mô hình đối tượng văn bản:
 - Trong mô hình đối tượng, mỗi phần của cuốn sách được xem là 1 đối tượng, Cuốn sách là 1 đối tượng với các thuộc tính, một chương cũng là 1 đối tượng với các thuộc tính và mỗi đoạn cũng là 1 đối tượng với các thuộc tính của chính nó
 - Mô hình đối tượng cho phép truy xuất 1 đối tượng thông qua tên mà không quan tâm tới việc đang làm việc với đối tượng là 1 chương hay phần của cuốn sách
 - Tài liệu XML được mô hình hóa như 1 cây trong đó mỗi phần tử trong cây được gọi là node, các đối tượng với các thuộc tính và phương thức được sử dụng để biểu diễn cây và node đó

2 - Cấu trúc cây của 1 tài liệu XML

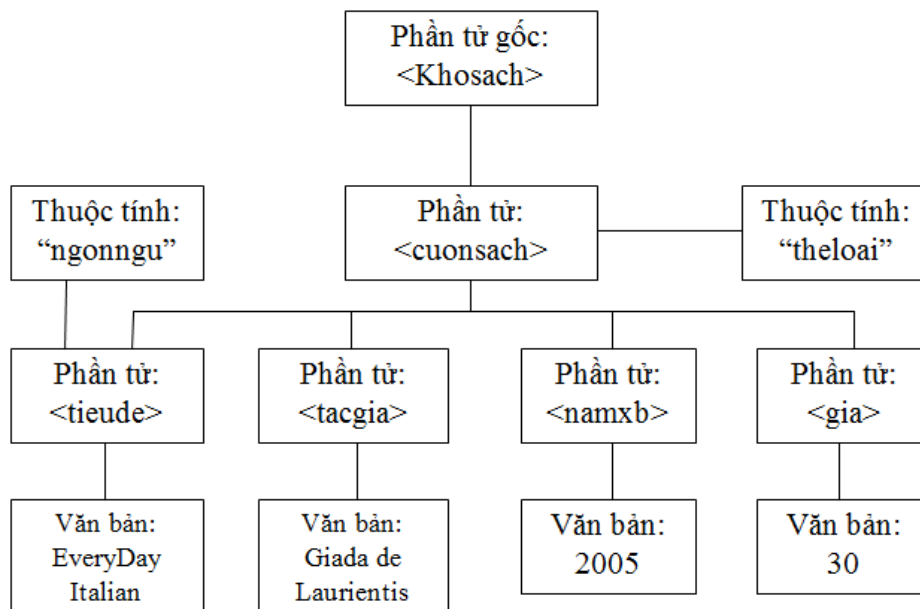
- Để hiểu 1 DOM trước hết ta tìm hiểu 1 mã của tài liệu XML và xét nó dưới góc độ DOM
- Ví dụ file cuonsach.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<khosach>
  <cuonsach theloai="cooking">
    <tieude ngonngu="en">EveryDay Italian</tieude>
    <tacgia>Giada de Laurentis</tacgia>
    <namxb>2005</namxb>
    <gia>30</gia>
  </cuonsach>
  <cuonsach theloai="children">
    <tieude ngonngu="en">Harry potter</tieude>
    <tacgia>JK Rolling</tacgia>
    <namxb>2005</namxb>
    <gia>299</gia>
  </cuonsach>
  <cuonsach theloai="web">
    <tieude ngonngu="en">XQuery Kick Start</tieude>
    <tacgia>James McGoven</tacgia>
    <tacgia>Per Bothner</tacgia>
    <tacgia>Kurt Cagle</tacgia>
    <tacgia>James Linn</tacgia>
    <tacgia>Vaidynathan Nagarajan</tacgia>
    <namxb>2005</namxb>
    <gia>67</gia>
  </cuonsach>
  <cuonsach theloai="web" biasach="paperback">
    <tieude ngonngu="en">Learning XML</tieude>
    <tacgia>Erik T Ray</tacgia>
    <namxb>2003</namxb>
    <gia>399</gia>
  </cuonsach>
</khosach>

```

- Cấu trúc cây của tài liệu “cuonsach.xml”



3 - Các đối tượng trong 1 tài liệu XML

- Các thuộc tính của DOM

- x.nodeName – Tên của node x
- x.nodeValue – Giá trị của node x
- x.parentNode – Node cha của node x
- x.childNodes – Các node con của node x
- x.attributes – các thuộc tính của node x

- Các phương thức

- x.getElementsByTagName(name) – Trả về tất cả các phần tử được xác định bởi thẻ name
- x.appendChild(node) – Thêm 1 node con vào node x
- x.removeChild(node) – Loại 1 node con khỏi node x

II – THỰC HÀNH

1- Xây dựng 1 file XML có nội dung như sau

```
<?xml version="1.0" encoding="utf-8" ?>
<thongtincuocgoi>
  <cuocgoi sodien="0433384848" chinhanh="Dong Da">
    <sogoiden>038727272</sogoiden>
    <ngaygoi>1998-12-01</ngaygoi>
    <sophut>2</sophut>
  </cuocgoi>
  .....
  <cuocgoi sodien="0456565656" chinhanh="Hoàn kiếm">
    <sogoiden>032456667</sogoiden>
    <ngaygoi>1998-12-01</ngaygoi>
    <sophut>5</sophut>
  </cuocgoi>
  ...
</thongtincuocgoi>
```

- Tạo form để cập nhật thông các cuộc gọi như sau trong đó dữ liệu về các chi nhánh và các số gọi đi được nhập sẵn trong ComboBox

- Hiển thị thông tin của các cuộc gọi lên lưới
- Thêm thông tin về các cuộc gọi
- Chọn 1 dòng trên lưới rồi hiển thị thông tin từ dòng được chọn lên các điều khiển tương ứng
- Sửa thông tin về 1 cuộc gọi có sogoiden trùng với giá trị trên ô Số gọi đến
- Xóa thông tin về cuộc gọi có sogoiden trùng với giá trị trên ô Số gọi đến

- Sử dụng thư viện
 - **using System.xml;**
- Khởi tạo đối tượng văn bản XML trỏ tới file XML được lưu trữ

```
namespace DOM_VD1
{
    public partial class Form1 : Form
    {
        XmlDocument doc = new XmlDocument();
        public static string path = "../..//thongtincuocgoi.xml";
    }
}
```

- Lấy danh sách các chnhanh và sodien cho vào combobox

```
private void load_combobox()
{
    DataSet dts = new DataSet();
    dts.ReadXml(path); //đọc file XML lưu vào dts
    chinhanh_cbb.DataSource=dts.Tables["cuocgoi"];
    chinhanh_cbb.DisplayMember="chinhanh"; //CHINHANH LÀ THUOCTINH CỦA CUOCGOI
    sogoidi_cbb.DataSource=dts.Tables["cuocgoi"];
    sogoidi_cbb.DisplayMember = "sodien"; //SODIEN LÀ THUOCTINH CỦA CUOCGOI
}
```

- Hiển thị thông tin nội dung các cuộc gọi lên listview

```
private void hienthi()
{
    listView1.Items.Clear();
    DataSet dts = new DataSet();
    DataTable dtl = new DataTable();
    dts.ReadXml(path);
    dtl = dts.Tables["cuocgoi"];
    if (dtl.Rows.Count > 0)
    {
        int i = 0;
        foreach (DataRow dr in dtl.Rows)
        {
            ListViewItem lvi = new ListViewItem(dtl.Rows[i]["chinhanh"].ToString());
            lvi.SubItems.Add(dtl.Rows[i]["sodien"].ToString());
            lvi.SubItems.Add(dtl.Rows[i]["sogoiden"].ToString());
            lvi.SubItems.Add(dtl.Rows[i]["ngaygoi"].ToString());
            lvi.SubItems.Add(dtl.Rows[i]["sophut"].ToString());
            i++;
            listView1.Items.Add(lvi);
        }
    }
    else
    {
        MessageBox.Show("Không có dữ liệu để hiển thị!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}
```

- Sự kiện form_load: hiển thị combobox và listview

```
private void Form1_Load(object sender, EventArgs e)
{
    load_combobox();
    hienthi();
}
```

- Click 1 dòng trên listview sẽ hiển thị nội dung đó trên các đối tượng combobox – textbox

```

private void listView1_SelectedIndexChanged(object sender, EventArgs e)
{
    foreach(ListViewItem item in listView1.SelectedItems)
    {
        chinhanh_cbb.Text=item.SubItems[0].Text;
        sogoidi_cbb.Text=item.SubItems[1].Text;
        sogoiden_txt.Text=item.SubItems[2].Text;
        ngaygoi_txt.Text=item.SubItems[3].Text;
        sophut_txt.Text=item.SubItems[4].Text;
    }
}

```

- Hàm thêm node cuocgoi mới – thêm dữ liệu vào file xml

```

private void them()
{
    doc.Load(path);
    XmlElement cuocgoi, sogoiden, ngaygoi, sophut;
    XmlAttribute chinhanh, sodien;
    cuocgoi = doc.CreateElement("cuocgoi");
    chinhanh = doc.CreateAttribute("chinhanh");
    sodien = doc.CreateAttribute("sodien");
    sogoiden = doc.CreateElement("sogoiden");
    ngaygoi = doc.CreateElement("ngaygoi");
    sophut = doc.CreateElement("sophut");
    chinhanh.InnerText = chinhanh_cbb.Text;
    sodien.InnerText = sogoidi_cbb.Text;
    sogoiden.InnerText = sogoiden_txt.Text;
    ngaygoi.InnerText = ngaygoi_txt.Text;
    sophut.InnerText = sophut_txt.Text;
    cuocgoi.SetAttributeNode(chinhanh);
    cuocgoi.SetAttributeNode(sodien);
    cuocgoi.AppendChild(sogoiden);
    cuocgoi.AppendChild(ngaygoi);
    cuocgoi.AppendChild(sophut);
    doc.DocumentElement.AppendChild(cuocgoi);
    doc.Save(path);
    MessageBox.Show("Thêm thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
}

```

- Hàm sửa node cuocgoi – sửa dữ liệu trên file xml – kiểm tra sogoiden có tồn tại không thì mới sửa – nếu không thì thông báo

```

private void sua()
{
    doc.Load(path);
    //trở tới node sogoiden là node có giá trị sogoiden trên form
    XmlNode node = doc.SelectSingleNode("/thongtincuocgoi/cuocgoi[sogoiden='" + (sogoiden_txt.Text).Trim() + "']");
    if (node != null)
    {
        node.Attributes[0].InnerText = sogoidi_cbb.Text;
        node.Attributes[1].InnerText = chinhanh_cbb.Text;
        //cuoc goi đến không sử dụng vì nó là nơi kiểm tra
        node.ChildNodes[1].InnerText = ngaygoi_txt.Text;
        node.ChildNodes[2].InnerText = sophut_txt.Text;

        doc.Save(path);
        MessageBox.Show("Sửa thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
    else
    {
        MessageBox.Show("Số gọi đến muốn sửa không có trong CSDL!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}
}

```

- **Chú ý** Nếu cần kiểm tra sogoidi có tồn tại hay không rồi mới cho xóa – mà sogoi lại là thuộc tính không phải là phần tử như sogoiden thì ta viết lại như sau

```

XmlNode node =
doc.SelectSingleNode("/thongtincuocgoi/cuocgoi[@sodien='" +
(sogoidi_cbb.Text).Trim() + "']");

```

- Hàm xóa node cuocgoi – xóa dl trên file xml – kiểm tra sogoiden có tồn tại không thì mới xóa – nếu không thì thông báo

```

private void xoa()
{
    doc.Load(path);
    //trò tới node sogoiden là node có giá trị sogoiden trên form
    XmlNode node = doc.SelectSingleNode("/thongtincuocgoi/cuocgoi[sogoiden='" + (sogoiden_txt.Text).Trim() + "']");
    if (node != null)
    {
        doc.DocumentElement.RemoveChild(node);

        doc.Save(path);
        MessageBox.Show("Xóa thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }

    else
        MessageBox.Show("Số gọi đến muốn xóa không có trong CSDL!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
}

```

- Các button “thêm mới”, “sửa”, ”xóa” gọi các hàm them(), sua(), xoa() và hienthi()

```

private void them_button_Click(object sender, EventArgs e)
{
    try
    {
        if (chinhanh_cbb.Text == "" || sogoidi_cbb.Text == "" || sogoiden_txt.Text == "" || ngaygoi_txt.Text == "" || sophut_txt.Text == "")
        {
            MessageBox.Show("Vui lòng nhập đầy đủ thông tin muốn thêm!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
        }
        else
        {
            them();
            hienthi();
        }
    }
    catch (Exception)
    {
        MessageBox.Show("Có lỗi xảy ra! không thể thêm.", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}

```

- Sửa


```

private void sua_button_Click(object sender, EventArgs e)
{
    try
    {
        if (sogoiden_txt.Text == "")
        {
            MessageBox.Show("Vui lòng nhập vào số gọi đến muốn sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
        }
        else
        {
            sua();
            hienthi();
        }
    }
    catch (Exception)
    {
        MessageBox.Show("Có lỗi xảy ra! không thể sửa.", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}

```

- Xóa + Thoát

```
private void xoa_button_Click(object sender, EventArgs e)
{
    try
    {
        if (sogoiden_txt.Text == "")
        {
            MessageBox.Show("Vui lòng nhập vào số gọi đến muốn xóa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
        }
        else
        {
            xoa();
            hienthi();
        }
    }
    catch (Exception)
    {
        MessageBox.Show("Có lỗi xảy ra! không thể xóa.", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}

private void thoat_button_Click(object sender, EventArgs e)
{
    DialogResult a = MessageBox.Show("Bạn có thực sự muốn thoát?", "Thông báo", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (a == DialogResult.Yes)
    {
        Application.Exit();
    }
}
```

BUỔI 9 - BÀI THỰC HÀNH SỐ 9

Câu 1. Tạo một Windows Form mới và đặt tên gồm họ tên sinh viên dự thi, tên lớp (ví dụ: NguyenToanThang_KHMT1) lưu tại ổ D.

Câu 2. Tạo file xml chứa kết quả học tập của các sinh viên theo mẫu sau

```
<bangdiem>
    <sinhvien masv="1111" monhoc="Lập trình Windows">
        <diemlan1>3</diemlan1>
        <diemlan2>8</diemlan2>
    </sinhvien>
    ...
</bangdiem>
```

Câu 3. Thực hiện các công việc sau

- a. Tạo form để cập nhật thông các cuộc gọi như sau trong đó dữ liệu về mã sinh viên và môn học được hiển thị sẵn trong ComboBox

STT	Họ tên SV	Môn học	Điểm lần 1	Điểm lần 2

- b. Hiển thị dữ liệu từ file XML đã tạo ở câu 2 lên lưới
- c. Khi chọn nút “Thêm” sẽ tạo một phần tử mới rồi thêm vào file XML đã tạo ở câu 2
- d. Chọn 1 dòng trên lưới rồi hiển thị thông tin từ dòng được chọn lên các điều khiển tương ứng
- e. Khi chọn nút “Sửa” sẽ cập nhật lại thông tin của phần tử có masv trong file xml trùng với nội dung của ô Mã sinh viên
- f. Khi chọn nút “Xóa” sẽ xóa phần tử trong file xml có masv trùng với nội dung của ô Mã sinh viên.

BUỔI 10- BÀI THỰC HÀNH SỐ 10

Câu 1. Tạo một Windows Form mới và đặt tên gồm họ tên sinh viên dự thi, tên lớp (ví dụ: NguyenToanThang_KHMT1) lưu tại ổ D

Câu 2. Tạo file xml chứa danh sách các khách hàng theo mẫu sau

```
<?xml version="1.0" encoding="utf-8" ?>
<danh sach khach hang>
  <khach hang makh="111111" chinh anh="Ha Noi">
    <hoten></hoten>
    <diachi></diachi>
    <sodt></sodt>
  </khach hang>
  ...
</danh sach khach hang>
```

Câu 3. Thực hiện các công việc sau

- a. Tạo form để cập nhật thông các cuộc gọi như sau trong đó dữ liệu về các chi nhánh trong ComboBox

Chi nhánh	Mã khách...	Họ tên	Địa chỉ	Số điện t...

- b. Hiển thị thông tin của các thuê bao lên lưới
- c. Thêm thông tin về các thuê bao
- d. Chọn 1 dòng trên lưới rồi hiển thị thông tin từ dòng được chọn lên các điều khiển tương ứng
- e. Sửa thông tin về 1 thuê bao đã chọn
- f. Xóa thông tin về thuê bao đã chọn