

XML

XML adalah kependekan dari **Extensible Markup Language**, yakni sebuah Bahasa “markup” yang dapat diperluas. Istilah “markup” merujuk pada proses menandai text sehingga akan dibaca oleh Browser internet sebagai sebuah “perintah”. Sedangkan istilah “Extensible” merujuk pada “Tag” yang dapat dibuat sendiri oleh programmer tanpa mengikuti standar yang biasa dipergunakan pada HTML.

XML adalah format standar yang biasa dipergunakan untuk tukar menukar data di internet, contohnya adalah RSS (Really Simple Syndication) yang dapat dengan mudah dipergunakan sebagai “feeder” dari sebuah website ke website lain yang membutuhkan informasi yang disediakan.

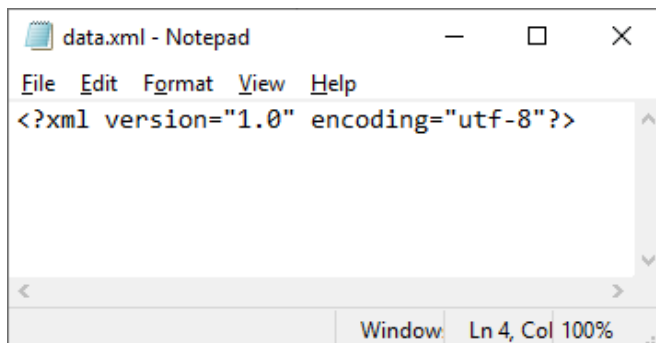
Jika Anda sudah mengenal HTML, maka istilah “markup” pada HTML biasa dilakukan dengan menandai text dengan tanda “<” dan “>”. Dengan cara tersebut, maka Browser Internet akan memperlakukannya berbeda dari text biasa tanpa tanda “markup” tersebut. Contohnya, ketika Browser Internet menemukan tanda “<h1>” dan “</h1>” maka kemudian teks yang diapit oleh kedua tanda tersebut akan ditampilkan berukuran besar dan tebal atau diberi style “Heading 1”. Istilah “markup” ini kemudian secara luas dikenal dengan istilah “Tag”.

XML sedikit berbeda dari HTML, karena menggunakan “Tag” yang diperluas sehingga tidak lagi mengikuti standar baku tersebut. Tag yang digunakan dalam XML diatur oleh programmer sehingga mudah untuk dibaca dan dimengerti disesuaikan dengan jenis data yang ada di dalamnya. XML biasanya disandingkan dengan database, sehingga penulisan tag pada dokumen XML biasanya mirip dengan struktur data pada database.

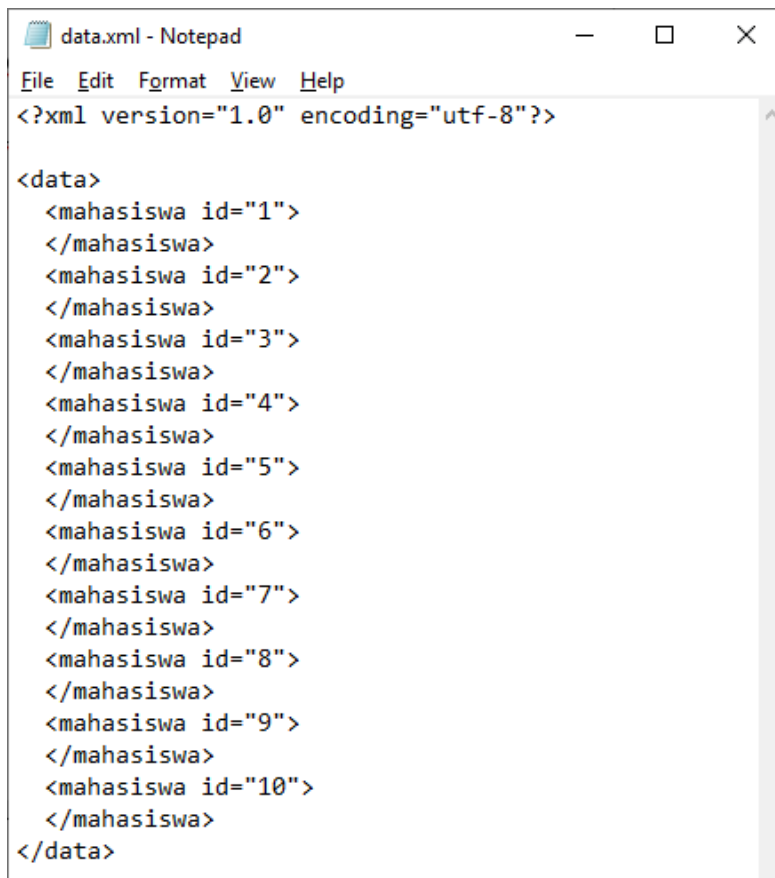
Membuat File XML

Contoh berikut ini adalah membuat file XML yang dipergunakan untuk menyimpan data mahasiswa:

1. Jalankan Notepad.exe. Anda dapat menggunakan software apapun untuk membuat teks sederhana ini, misalnya menggunakan Adobe Dreamweaver, Notepad++, Sublime Text, dan sebagainya.
2. Ketik header file XML sebagai berikut:



3. Baris pertama tadi adalah untuk memberi tahu Browser bahwa file ini adalah file XML yang menggunakan format versi 1.0 dengan jenis font standar tulisan Latin (utf-8).
4. Buat struktur data untuk Node pertama sebagai berikut:

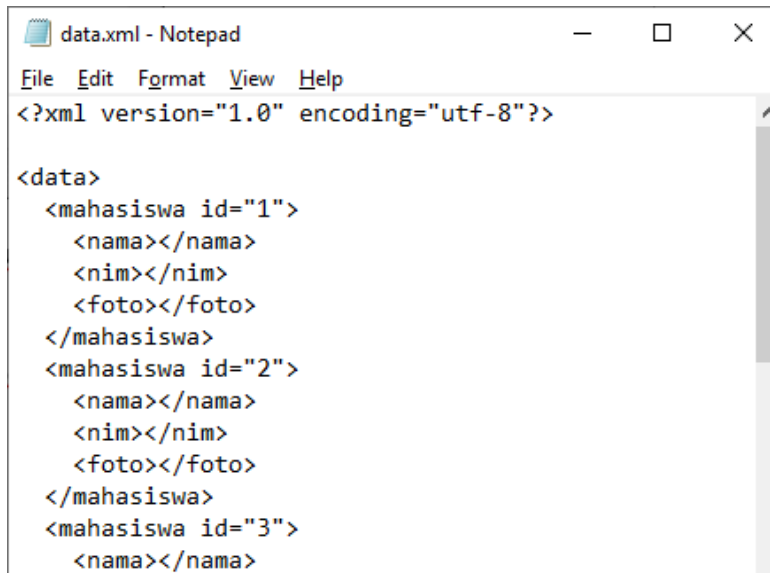


```
<?xml version="1.0" encoding="utf-8"?>

<data>
  <mahasiswa id="1">
  </mahasiswa>
  <mahasiswa id="2">
  </mahasiswa>
  <mahasiswa id="3">
  </mahasiswa>
  <mahasiswa id="4">
  </mahasiswa>
  <mahasiswa id="5">
  </mahasiswa>
  <mahasiswa id="6">
  </mahasiswa>
  <mahasiswa id="7">
  </mahasiswa>
  <mahasiswa id="8">
  </mahasiswa>
  <mahasiswa id="9">
  </mahasiswa>
  <mahasiswa id="10">
  </mahasiswa>
</data>
```

Jumlah data tidak dibatasi, sehingga kita bisa membuat puluhan, ratusan, atau bahkan ribuan data.

5. Berikutnya adalah menambahkan data pada Node kedua, misalnya sebagai berikut:



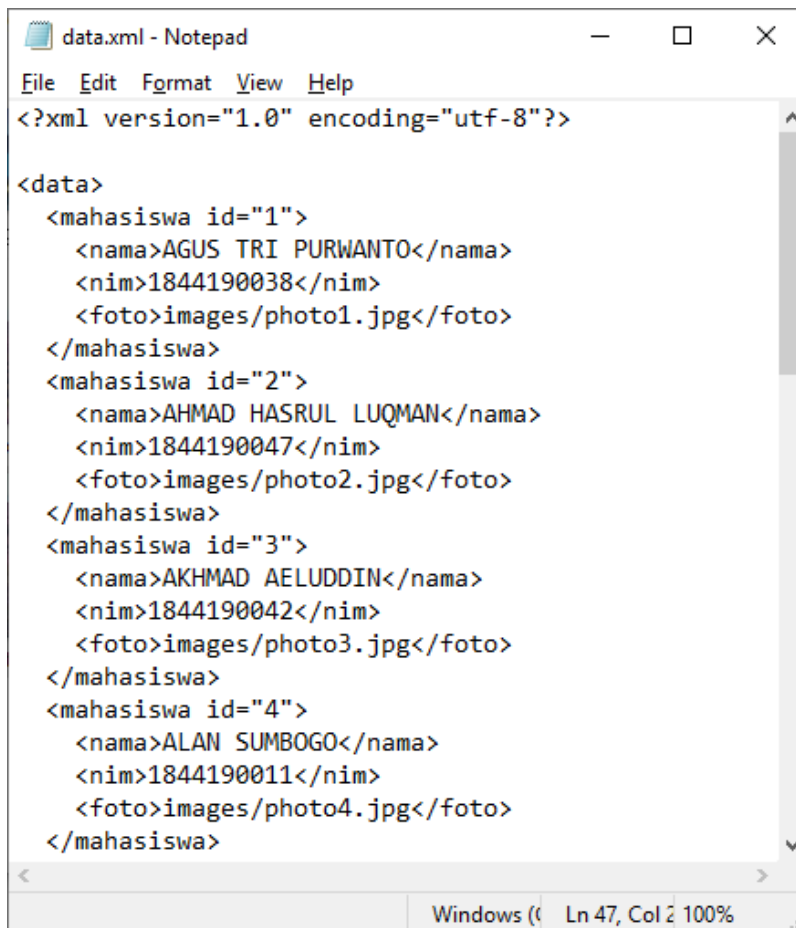
```

data.xml - Notepad
File Edit Format View Help
<?xml version="1.0" encoding="utf-8"?>

<data>
  <mahasiswa id="1">
    <nama></nama>
    <nim></nim>
    <foto></foto>
  </mahasiswa>
  <mahasiswa id="2">
    <nama></nama>
    <nim></nim>
    <foto></foto>
  </mahasiswa>
  <mahasiswa id="3">
    <nama></nama>

```

6. Selanjutnya, isi Node kedua tersebut dengan data yang dibutuhkan misalnya 10 data mahasiswa. Khusus untuk data yang berisi media (foto, suara, video), harus diisi dengan nama file dan folder tempat file tersebut berada. Misalnya sebagai berikut:



```

data.xml - Notepad
File Edit Format View Help
<?xml version="1.0" encoding="utf-8"?>

<data>
  <mahasiswa id="1">
    <nama>AGUS TRI PURWANTO</nama>
    <nim>1844190038</nim>
    <foto>images/photo1.jpg</foto>
  </mahasiswa>
  <mahasiswa id="2">
    <nama>AHMAD HASRUL LUQMAN</nama>
    <nim>1844190047</nim>
    <foto>images/photo2.jpg</foto>
  </mahasiswa>
  <mahasiswa id="3">
    <nama>AKHMAD AELUDDIN</nama>
    <nim>1844190042</nim>
    <foto>images/photo3.jpg</foto>
  </mahasiswa>
  <mahasiswa id="4">
    <nama>ALAN SUMBOGO</nama>
    <nim>1844190011</nim>
    <foto>images/photo4.jpg</foto>
  </mahasiswa>

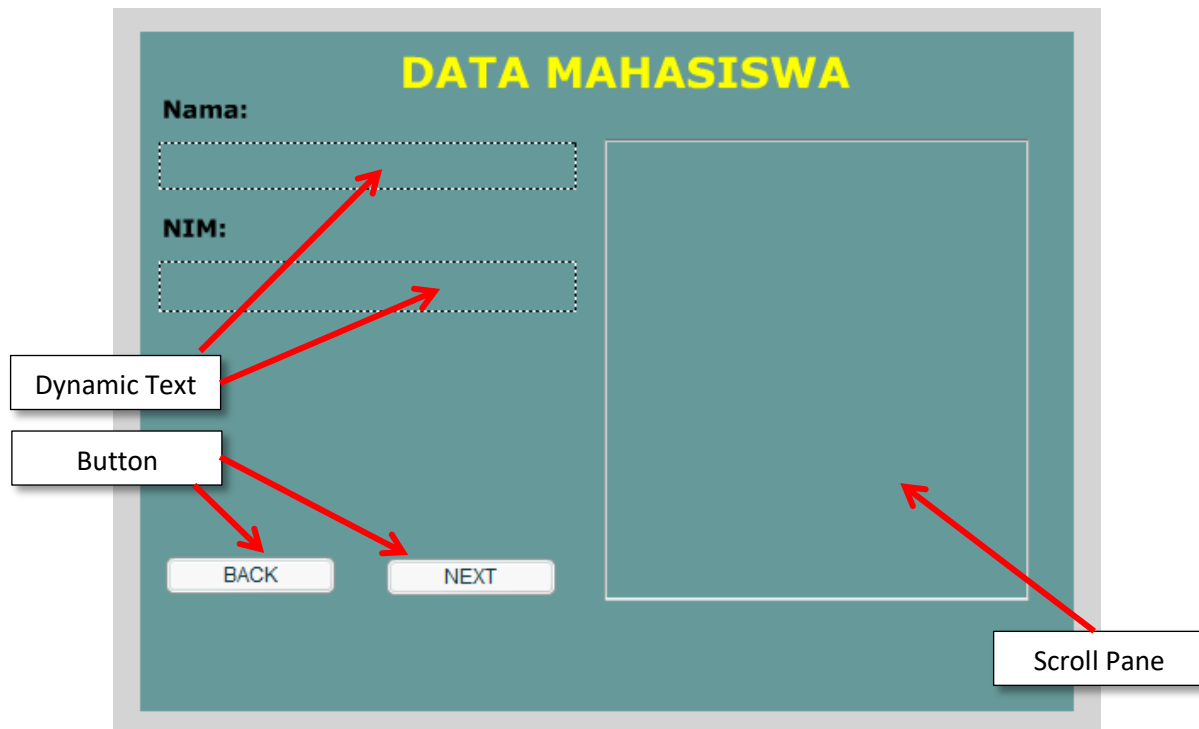
```

7. Simpan teks tersebut dengan nama file: **data.xml** (Save as type: All files)
8. Buat folder dengan nama “images” di tempat data.xml tersebut berada.
9. Isi folder tersebut dengan foto-foto yang diperlukan, yakni: **photo1.jpg, photo2.jpg, photo3.jpg**, sampai dengan **photo10.jpg**.

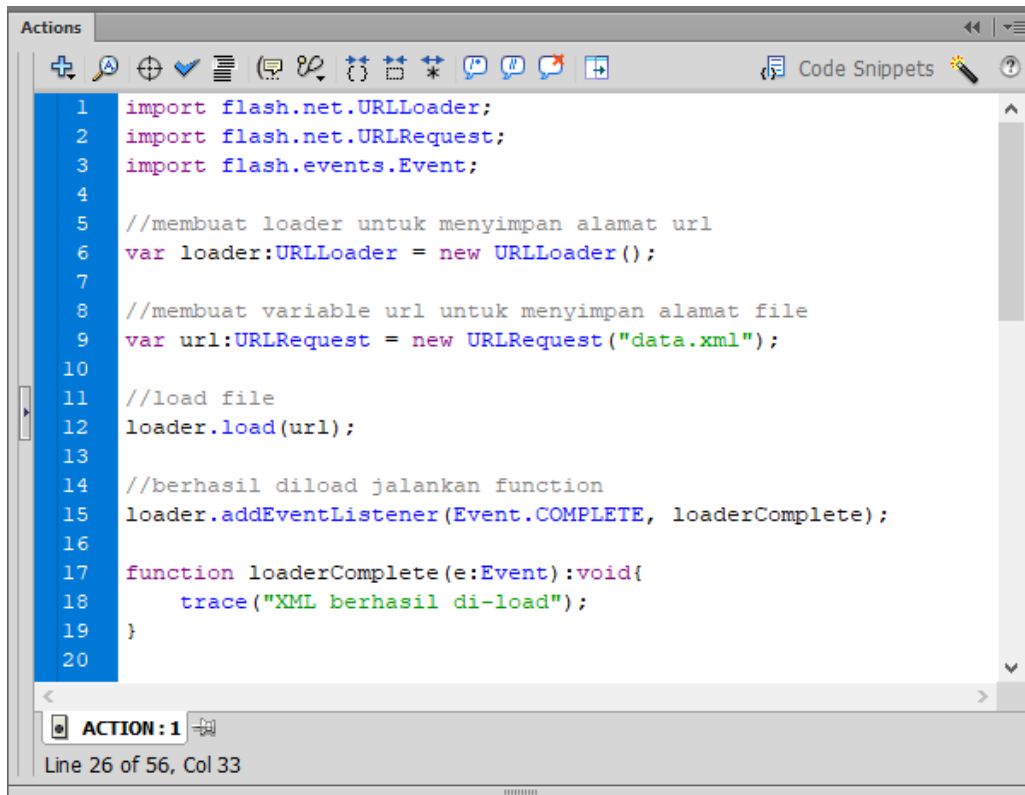
Membuat Multimedia untuk Menampilkan Data

Sekarang, kita akan membuat antarmuka Multimedia Interaktif dengan menggunakan Adobe Flash. Antarmuka ini akan kita pergunakan untuk menampilkan data dalam bentuk teks dan foto.

1. Buat Teks dengan Text Tool. Pada bagian yang akan diisi dengan data, ganti menjadi Dynamic Text.
2. Untuk menempatkan foto, buat Scroll Pane (menu: Window > Component > User Interface > Scroll Pane), ukuran panjang dan lebar disesuaikan dengan ukuran foto (misalnya 300 x 300px).
3. Tambahkan Button dengan menggunakan Component > User Interface > Button, atau bias juga membuat Button sendiri dengan menggunakan Text Tool:



4. Beri nama instance masing-masing dynamic text dan scroll pane dengan nama: **displaynama**, **displaynim**, dan **displayfoto**.
5. Beri nama instance masing-masing Button dengan nama: **back_btn** dan **next_btn**.
6. Tambahkan layer baru, beri nama: **ACTION**
7. Pada layer ACTION, tambahkan script (menu: **Window > Action**) sebagai berikut:



```

1  import flash.net.URLLoader;
2  import flash.net.URLRequest;
3  import flash.events.Event;
4
5  //membuat loader untuk menyimpan alamat url
6  var loader:URLLoader = new ULLoader();
7
8  //membuat variable url untuk menyimpan alamat file
9  var url:URLRequest = new URLRequest("data.xml");
10
11 //load file
12 loader.load(url);
13
14 //berhasil diload jalankan function
15 loader.addEventListener(Event.COMPLETE, loaderComplete);
16
17 function loaderComplete(e:Event):void{
18     trace("XML berhasil di-load");
19 }
20

```

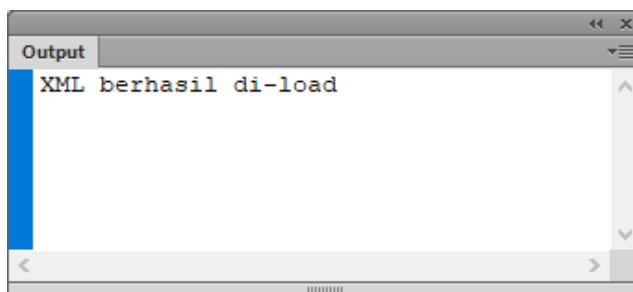
ACTION: 1
Line 26 of 56, Col 33

Catatan:

Baris 1 – 3 secara otomatis akan muncul sendiri ketika kita mengetikkan Class pada script. Akan tetapi jika tidak muncul secara otomatis, bisa diketik secara manual.

10. Save file Flash tersebut pada folder yang sama dengan tempat kita menempatkan file data.xml

11. Jalankan Test Movie [**Ctrl + ENTER**], dengan hasil sebagai berikut:



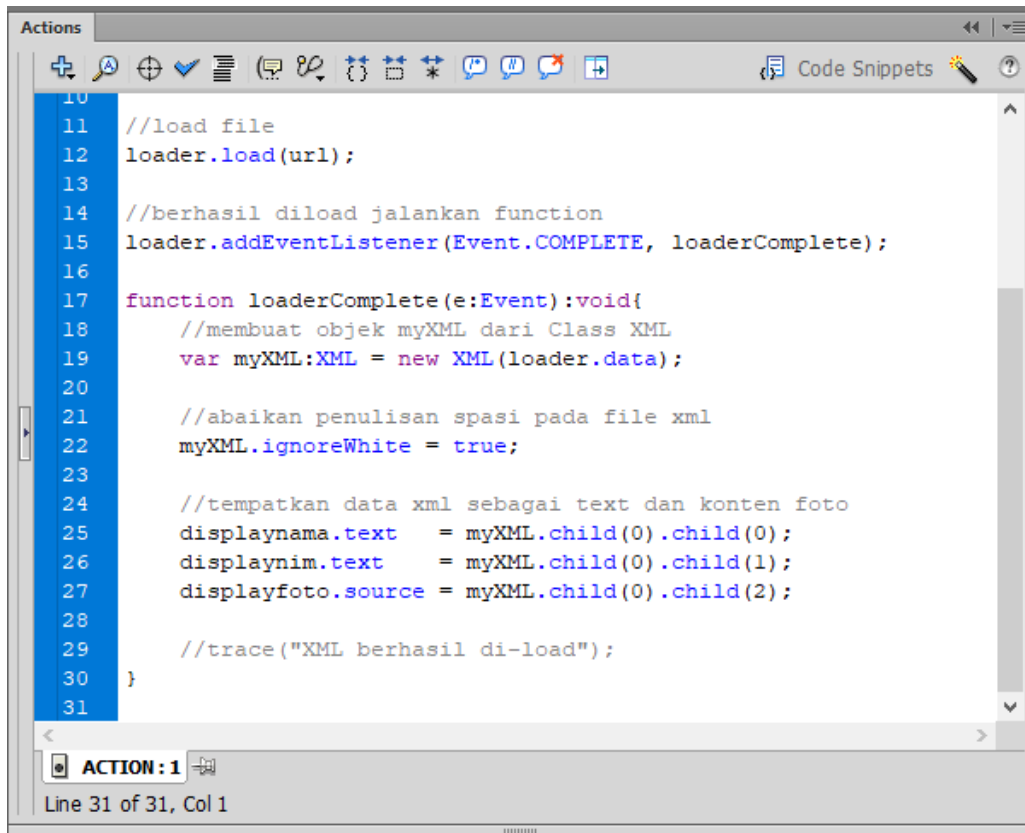
```

Output
XML berhasil di-load

```

Pada panel Output akan tertulis message seperti yang telah kita buat pada script: **"XML berhasil di-load"**.

12. Berikutnya, lengkapi function **loaderComplete** tersebut menjadi sebagai berikut:



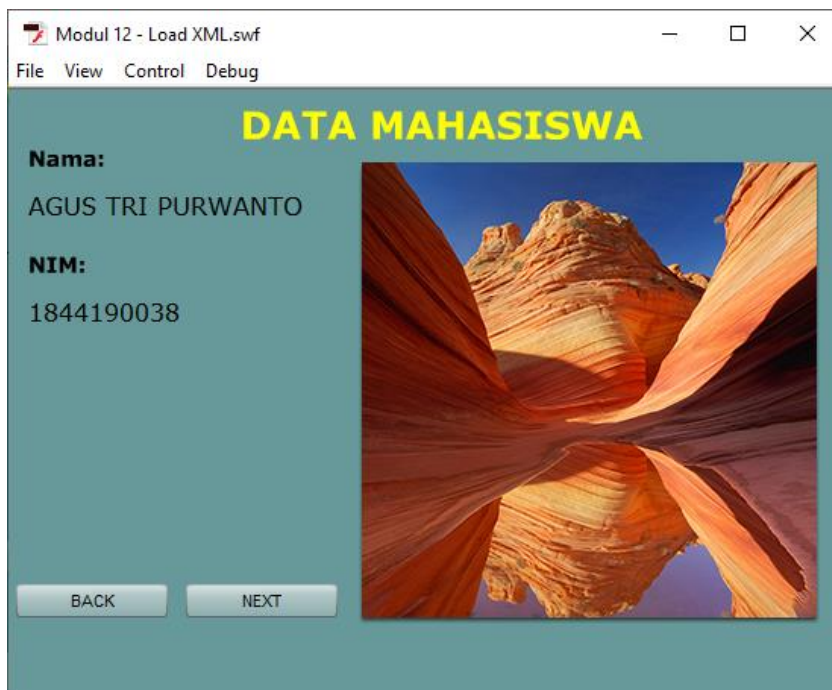
```

10
11 //load file
12 loader.load(url);
13
14 //berhasil diload jalankan function
15 loader.addEventListener(Event.COMPLETE, loaderComplete);
16
17 function loaderComplete(e:Event):void{
18     //membuat objek myXML dari Class XML
19     var myXML:XML = new XML(loader.data);
20
21     //abaikan penulisan spasi pada file xml
22     myXML.ignoreWhite = true;
23
24     //tempatkan data xml sebagai text dan konten foto
25     displaynama.text = myXML.child(0).child(0);
26     displaynim.text = myXML.child(0).child(1);
27     displayfoto.source = myXML.child(0).child(2);
28
29     //trace("XML berhasil di-load");
30 }
31

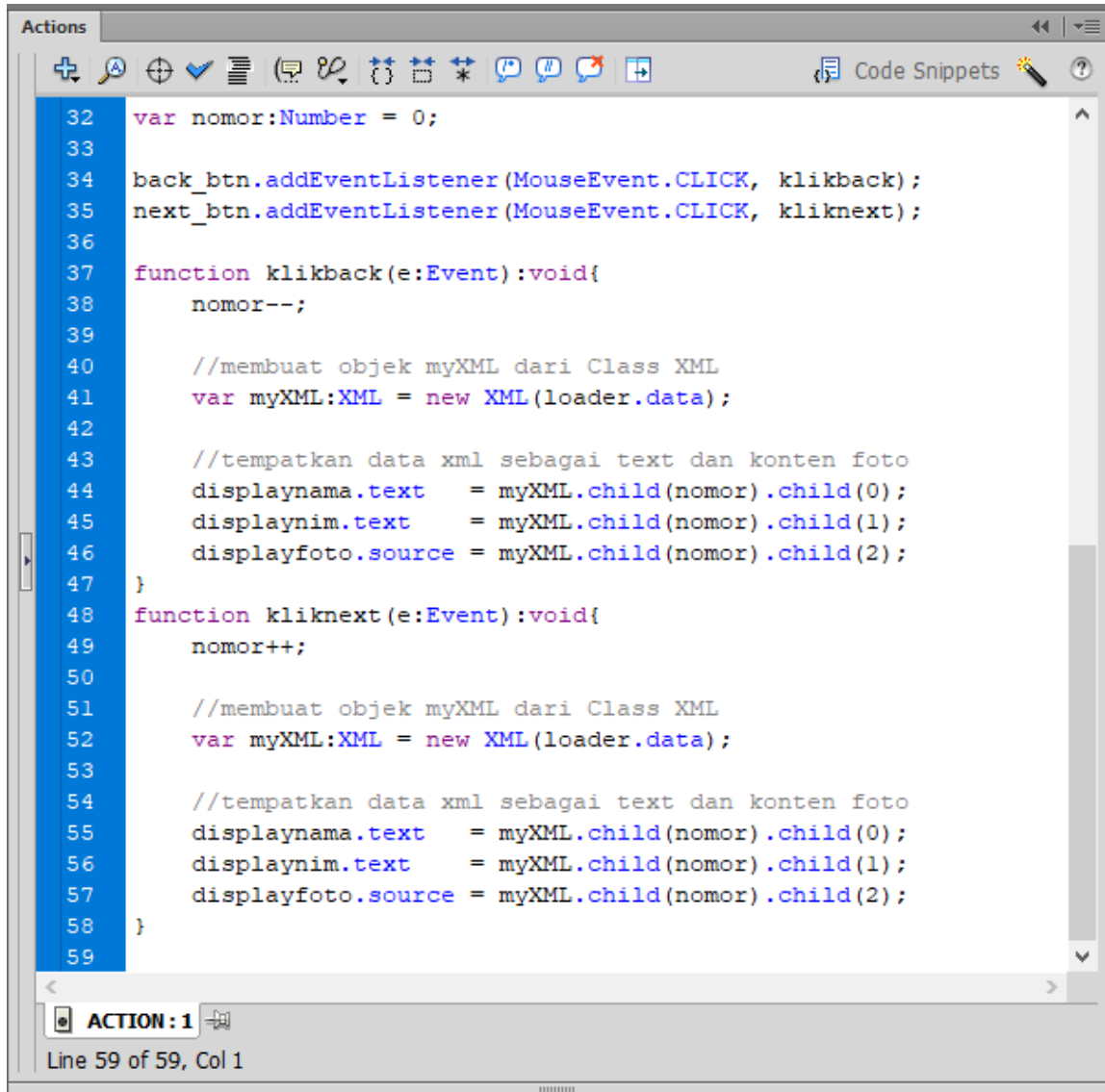
```

ACTION: 1
Line 31 of 31, Col 1

13. Test Movie dengan shortcut **Ctrl + ENTER** atau menu: **Control > Test Movie > Test**.



14. Setelah berhasil menampilkan data pertama, maka kita bisa lanjutkan dengan menambahkan script untuk kedua Button sebagai berikut:



```

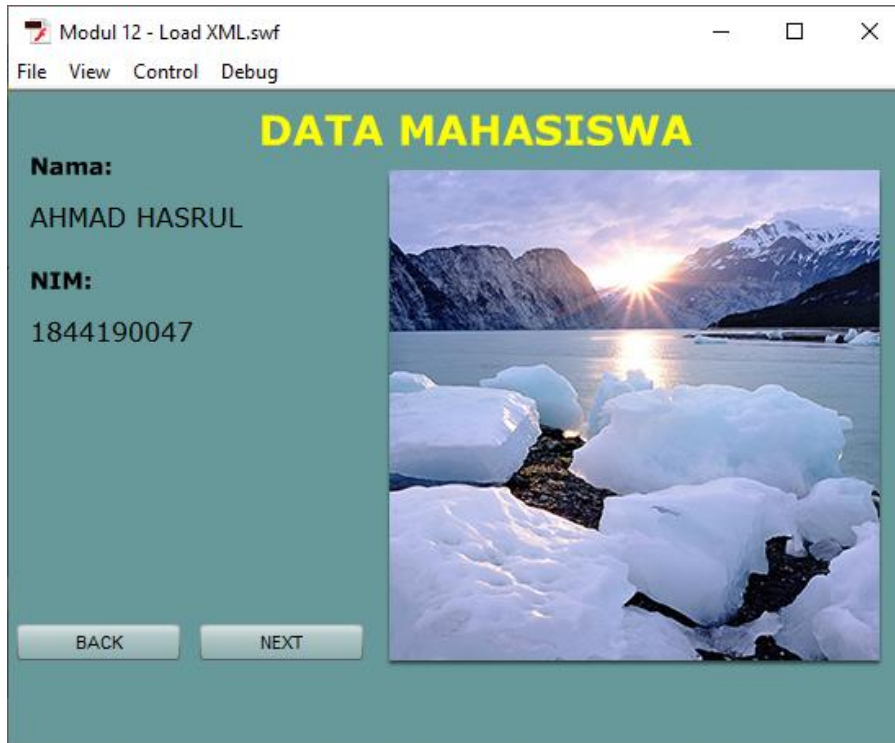
32  var nomor:Number = 0;
33
34  back_btn.addEventListener(MouseEvent.CLICK, klikback);
35  next_btn.addEventListener(MouseEvent.CLICK, kliknext);
36
37  function klikback(e:Event):void{
38      nomor--;
39
40      //membuat objek myXML dari Class XML
41      var myXML:XML = new XML(loader.data);
42
43      //tempatkan data xml sebagai text dan konten foto
44      displaynama.text    = myXML.child(nomor).child(0);
45      displaynim.text     = myXML.child(nomor).child(1);
46      displayfoto.source  = myXML.child(nomor).child(2);
47  }
48  function kliknext(e:Event):void{
49      nomor++;
50
51      //membuat objek myXML dari Class XML
52      var myXML:XML = new XML(loader.data);
53
54      //tempatkan data xml sebagai text dan konten foto
55      displaynama.text    = myXML.child(nomor).child(0);
56      displaynim.text     = myXML.child(nomor).child(1);
57      displayfoto.source  = myXML.child(nomor).child(2);
58  }
59

```

ACTION : 1
Line 59 of 59, Col 1

15. Test Movie dengan shortcut **Ctrl + ENTER** atau menu: **Control > Test Movie > Test**.

Klik tombol NEXT dan tombol BACK untuk melihat semua data ditampilkan dengan baik pada layar aplikasi yang telah kita buat.



16. Jika tombol NEXT di-klik hingga data terakhir, maka layar akan menampilkan data yang kosong. Demikian halnya jika kita klik tombol BACK hingga data sebelum yang paling awal, maka layar akan menampilkan data yang kosong.
17. Perbaiki lagi Action Script dengan menambahkan IF ... ELSE sebagai berikut:


```

37 function klikback(e:Event):void{
38     nomor--;
39
40     if(nomor>=0){
41         //membuat objek myXML dari Class XML
42         var myXML:XML = new XML(loader.data);
43
44         //tempatkan data xml sebagai text dan konten foto
45         displaynama.text = myXML.child(nomor).child(0);
46         displaynim.text = myXML.child(nomor).child(1);
47         displayfoto.source = myXML.child(nomor).child(2);
48     } else {
49         nomor=0;
50     }
51 }
52 function kliknext(e:Event):void{
53     nomor++;
54
55     if(nomor<10){
56         //membuat objek myXML dari Class XML
57         var myXML:XML = new XML(loader.data);
58
59         //tempatkan data xml sebagai text dan konten foto
60         displaynama.text = myXML.child(nomor).child(0);
61         displaynim.text = myXML.child(nomor).child(1);
62         displayfoto.source = myXML.child(nomor).child(2);
63     } else{
64         nomor=10;
65     }
66 }
67

```

Layer 3:1
Line 67 of 67, Col 1

18. Test Movie dengan shortcut **Ctrl + ENTER** atau menu: **Control > Test Movie > Test**.

Klik tombol NEXT dan tombol BACK untuk melihat semua data ditampilkan dengan baik pada layar aplikasi yang telah kita buat.

Sekarang, jika tombol NEXT dilanjutkan hingga paling ujung, maka layar akan berhenti pada data terakhir.

Demikian halnya, jika tombol BACK dilanjutkan hingga paling awal, maka layar akan berhenti pada data paling awal.



Apabila ingin dikembangkan lebih jauh, maka Multimedia Interaktif ini bisa kita kembangkan untuk menyajikan data dalam bentuk media lain selain foto di atas, yakni media suara (file MP3) atau media video (file MP4) baik yang berbentuk file ataupun streaming.