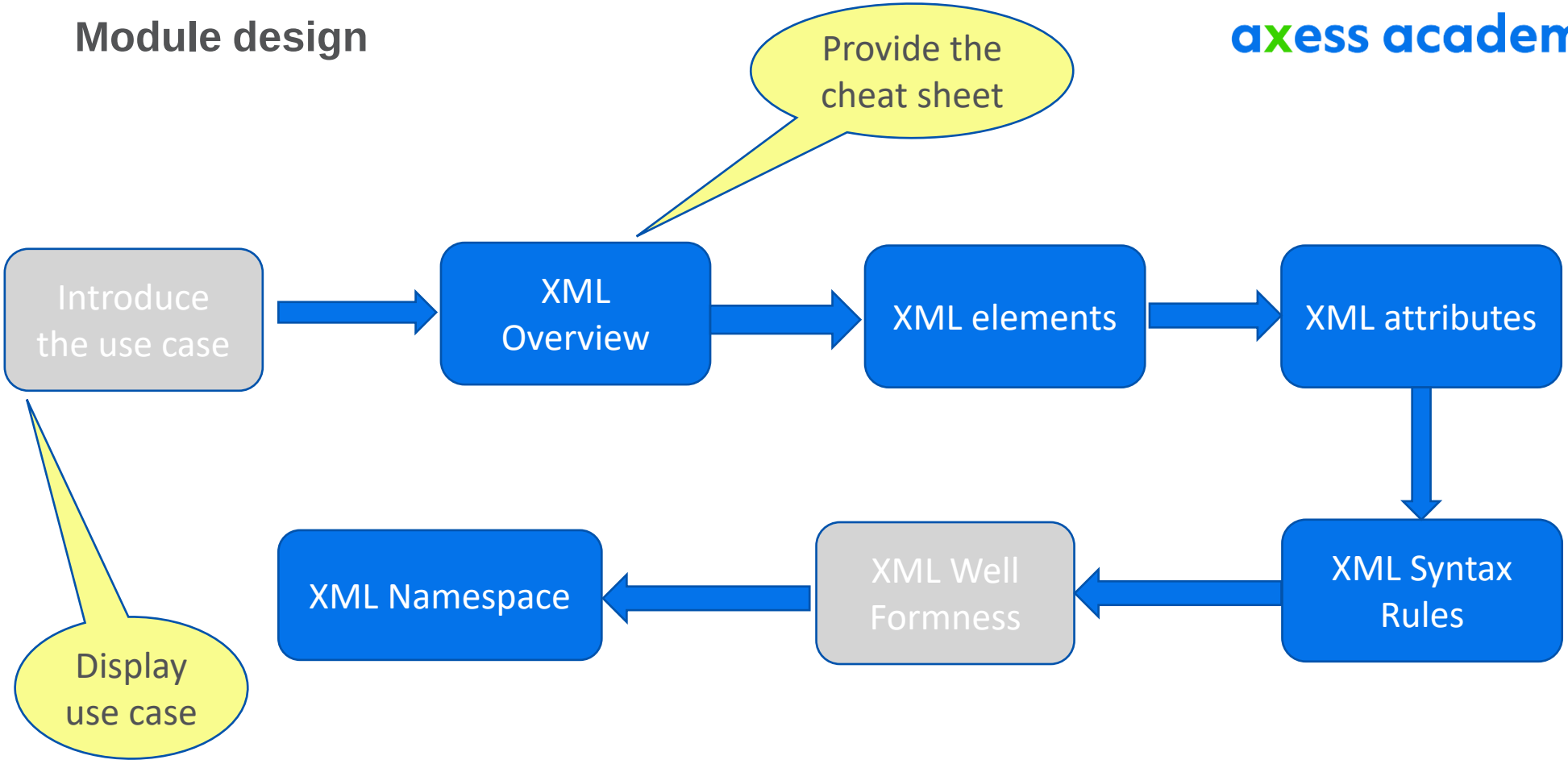
axess academy

XML

standard chartered

Module design

axess academy



S/W Requirements
IDE :Visual Studio Code Browser: Chrome

axess academy

Introduce the Use Case on XML

Presenting the Use case on XML

axess academy

- Scenario: We have bank customers holding savings or current account and bank is providing them discounts in amazon shopping. Finally we need to display the data in xml format which shows the customer details with shopping items.

```xml
<customers xmlns:scb="http://www.w3.org/customers/scb/">
    <customer>
        <custid>scb132</custid>
        <firstname>Sandra</firstname>
        <lastname>Rogers</lastname>
        <gender>Female</gender>
        <type>Savings</type>
        <balance>100000</balance>
    </customer>
    <customer>
        <custid>scb133</custid>
        <firstname>Steve</firstname>
        <lastname>Casey</lastname>
        <gender>Male</gender>
        <type>Current</type>
        <balance>300000</balance>
    </customer>
    <customer>
        <custid>scb134</custid>
        <firstname>Michelle</firstname>
        <lastname>Michaels</lastname>
        <gender>Male</gender>
        <type>Savings</type>
        <balance>200000</balance>
    </customer>
</customers>
```

```xml
<customers xmlns:shop="http://www.w3.org/customers/shopping/">
    <customer>
        <custid>scb132</custid>
        <itmes>Fridge</items>
        <amount>50000</amount>
    </customer>
    <customer>
        <custid>scb133</custid>
        <itmes>TV</items>
        <amount>80000</amount>
    </customer>
    <customer>
        <custid>scb134</custid>
        <itmes>Mobile</items>
        <amount>20000</amount>
    </customer>
</customers>
```

Presenting the Use case on XML…

axess academy

```
<customerlist  xmlns:scb="http://www.w3.org/customers/scb/"
 xmlns:amazon="http://www.w3.org/customers/shopping/">
    <scb:customer>
        <scb:custid>scb132</scb:custid>
        <scb:firstname>Sandra</scb:firstname>
        <scb:lastname>Rogers</scb:lastname>
        <scb:gender>Female</scb:gender>
    </scb:customer>
    <scb:customer>
        <scb:custid>scb133</scb:custid>
        <scb:firstname>Steve</scb:firstname>
        <scb:lastname>Casey</scb:lastname>
        <scb:gender>Male</scb:gender>
    </scb:customer>
    <scb:customer>
        <scb:custid>scb134</scb:custid>
        <scb:firstname>Michelle</scb:firstname>
        <scb:lastname>Michaels</scb:lastname>
        <scb:gender>Male</scb:gender>
    </scb:customer>
    <amazon:customer>
        <amazon:custid>scb132</amazon:custid>
        <amazon:items>Fridge</amazon:items>
        <amazon:amount>50000</amazon:amount>
    </amazon:customer>
    <amazon:customer>
        <amazon:custid>scb133</amazon:custid>
        <amazon:items>TV</amazon:items>
        <amazon:amount>80000</amazon:amount>
     </amazon:customer>
```

axess academy

XML

XML

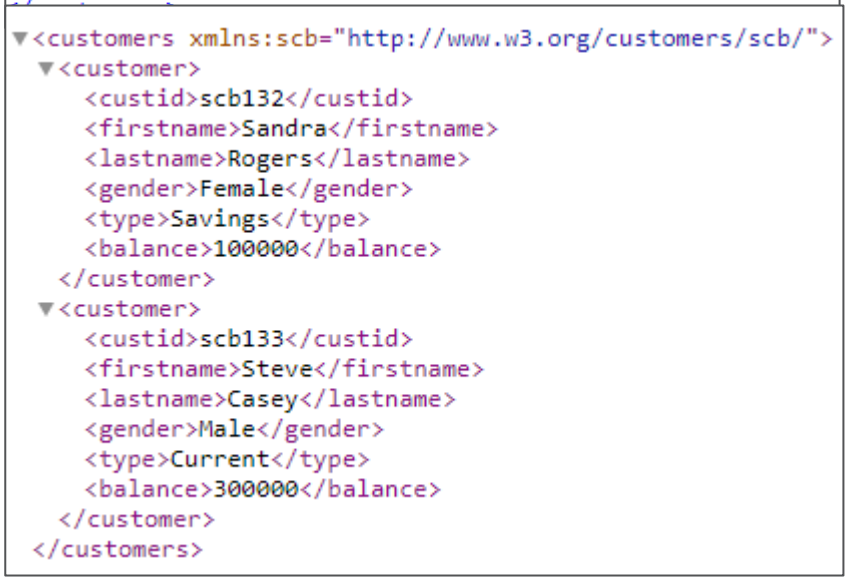
axess academy

- XML is a software- and hardware-independent tool for storing and transporting data.
- XML is not a replacement for HTML.
- XML is designed to be self-descriptive.
- XML is designed to carry data, not to display data.
- XML tags are not predefined. You must define your own tags.
- XML is platform independent and language independent.

customers.xml

```
<customers xmlns:scb="http://www.w3.org/customers/scb/">
    <customer>
        <custid>scb132</custid>
        <firstname>Sandra</firstname>
        <lastname>Rogers</lastname>
        <gender>Female</gender>
        <type>Savings</type>
        <balance>100000</balance>
    </customer>
    <customer>
        <custid>scb133</custid>
        <firstname>Steve</firstname>
        <lastname>Casey</lastname>
        <gender>Male</gender>
        <type>Current</type>
        <balance>300000</balance>
    </customer>
```

XML document in browser



XML Element

axess academy

- An XML doc starts at a **root element** and branches from the root to **child elements**.
- All elements can have sub elements (child elements). The terms parent, child, and sibling are used to describe the relationships between elements.
- An XML element is everything from (including) the element's start tag to (including) the element's end tag. In this example custid, firstname and lastname are elements.
- Element can contain:
  - text
  - attributes
  - other elements
  - or a mix of the above
- Element with no content is said to be empty and it can have attributes
- Element names are case-sensitive
- Element names must start with a letter or underscore
- Element names cannot contain spaces

```
<customer>
    <custid>scb132</custid>
    <firstname>Sandra</firstname>
    <lastname>Rogers</lastname>
    <gender>Female</gender>
    <type>Savings</type>
    <balance>100000</balance>
</customer>
```

XML Attributes

axess academy

- Attributes are designed to contain data related to a specific element.
- Attribute values must always be quoted. Either single or double quotes can be used.
- Single element can have multiple attributes but that's not a good practice.
- XML elements vs attributes: In the first example gender is an attribute. In the last, gender is an element. Both examples provide the same information.
- There are no rules about when to use attributes or when to use elements in XML.

```
<person gender="female">
    <firstname>Anna</firstname>
    <lastname>Smith</lastname>
</person>
```

```
<person>
    <gender>female</gender>
    <firstname>Anna</firstname>
    <lastname>Smith</lastname>
</person>
```

XML Syntax Rules

axess academy

- XML Documents Must Have a Root Element
- The XML prolog is optional. If it exists, it must come first in the document.

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

- XML tags are case sensitive. The tag <Book> is different from the tag <book>.
- In XML, all elements **must** be properly nested within each other.
- In XML, the attribute values must always be quoted.
- XML does not truncate multiple white-spaces (HTML truncates multiple white-spaces to one single white-space).

Well Formness of XML

axess academy

- Well Formness of XML is something if XML follows the syntax rules.
- To check whether XML is well formed or not, please open XML in the browser. If you are able to open successfully XML is well formed else you will get the error.

```
<customers xmlns:scb="http://www.w3.org/customers/scb/">
    <customer>
        <custid>scb132</custid>
        <firstname>Sandra</firstname>
        <lastname>Rogers</lastname>
        <gender>Female</gender>
        <type>Savings</type>
        <balance>100000</balance>
    </customer>
    <customer>
        <custid>scb133</custid>
        <firstname>Steve</firstname>
        <lastname>Casey</lastname>
        <gender>Male</gender>
        <type>Current</type>
        <balanc>300000</balance>
    </customer>
</customers>
```

balance is mis-spelled

Output in browser

XML Namespace

axess academy

- XML Namespaces provide a method to avoid element name conflicts.
- In XML, element names are defined by the developer. This often results in a conflict when trying to mix XML documents from different XML applications.
- Suppose we are getting customer data in xmls from different places like bank customers and shopping customers then both XML's will be having the common set of elements like: firstname, lastname. To differentiate both XML data we do need namepspace.
- Syntax:

<element xmlns:prefix="namespaceidentifier"

- A **Uniform Resource Identifier** (URI) is a string of characters which identifies an Resource.

Any arbitrary string

URI

```
<customers xmlns:scb="http://www.w3.org/customers/scb/">
    <customer>
        <custid>scb132</custid>
        <firstname>Sandra</firstname>
        <lastname>Rogers</lastname>
        <gender>Female</gender>
        <type>Savings</type>
        <balance>100000</balance>
    </customer>
    <customer>
        <custid>scb133</custid>
        <firstname>Steve</firstname>
        <lastname>Casey</lastname>
        <gender>Male</gender>
        <type>Current</type>
        <balanc>300000</balance>
    </customer>
</customers>
```

axess academy

Let us work on XML use case

Presenting the Use case on XML

axess academy

- Scenario: We have bank customers holding savings or current account and bank is providing them discounts in amazon shopping. Finally we need to display the data in xml format which shows the customer details with shopping items.

```xml
<customers xmlns:scb="http://www.w3.org/customers/scb/">
    <customer>
        <custid>scb132</custid>
        <firstname>Sandra</firstname>
        <lastname>Rogers</lastname>
        <gender>Female</gender>
        <type>Savings</type>
        <balance>100000</balance>
    </customer>
    <customer>
        <custid>scb133</custid>
        <firstname>Steve</firstname>
        <lastname>Casey</lastname>
        <gender>Male</gender>
        <type>Current</type>
        <balance>300000</balance>
    </customer>
    <customer>
        <custid>scb134</custid>
        <firstname>Michelle</firstname>
        <lastname>Michaels</lastname>
        <gender>Male</gender>
        <type>Savings</type>
        <balance>200000</balance>
    </customer>
</customers>
```

```xml
<customers xmlns:shop="http://www.w3.org/customers/shopping/">
    <customer>
        <custid>scb132</custid>
        <itmes>Fridge</items>
        <amount>50000</amount>
    </customer>
    <customer>
        <custid>scb133</custid>
        <itmes>TV</items>
        <amount>80000</amount>
    </customer>
    <customer>
        <custid>scb134</custid>
        <itmes>Mobile</items>
        <amount>20000</amount>
    </customer>
</customers>
```

Presenting the Use case on XML…

axess academy

```
<customerlist  xmlns:scb="http://www.w3.org/customers/scb/"
 xmlns:amazon="http://www.w3.org/customers/shopping/">
    <scb:customer>
        <scb:custid>scb132</scb:custid>
        <scb:firstname>Sandra</scb:firstname>
        <scb:lastname>Rogers</scb:lastname>
        <scb:gender>Female</scb:gender>
    </scb:customer>
    <scb:customer>
        <scb:custid>scb133</scb:custid>
        <scb:firstname>Steve</scb:firstname>
        <scb:lastname>Casey</scb:lastname>
        <scb:gender>Male</scb:gender>
    </scb:customer>
    <scb:customer>
        <scb:custid>scb134</scb:custid>
        <scb:firstname>Michelle</scb:firstname>
        <scb:lastname>Michaels</scb:lastname>
        <scb:gender>Male</scb:gender>
    </scb:customer>
    <amazon:customer>
        <amazon:custid>scb132</amazon:custid>
        <amazon:items>Fridge</amazon:items>
        <amazon:amount>50000</amazon:amount>
    </amazon:customer>
    <amazon:customer>
        <amazon:custid>scb133</amazon:custid>
        <amazon:items>TV</amazon:items>
        <amazon:amount>80000</amazon:amount>
     </amazon:customer>
```

Cheat Sheet references

axess academy

Cheat Sheet for XML : XML Cheat Sheet

axess academy

Thank You