Handling User Input

Outline of the Lecture
- Creating Selection Lists (Radio Buttons)
- Input fields
- Setting Variable Values in WML

WML supports two kinds of user input: entering text and selecting from a list. The following sections describe how to both prompt the user and handle the response.

Creating Selection Lists (<select> <option> elements)

- A selection list is a list of options that a user can select.
  - The <select></select> WML tags are used to define a selection list
  - The <option></option> tags are used to define an item in a selection list. Items are presented as radio buttons in some WAP browsers.
- The <option></option> tag pair should be enclosed within the <select></select> tags.

Example 8.1

```xml
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3//EN" "http://www.wapforum.org/DTD/wml13.dtd">
<wml>
  <card id="card1" title="Selection List">
    <p>This is a selection list:<br/></p>
    <select name="selection_list">
      <option value="WML">WML Tutorial</option>
      <option value="WAP">WAP Tutorial</option>
      <option value="HTML">HTML Tutorial</option>
    </select>
  </card>
</wml>
```
In WML, a selection list is associated with a **variable**, which stores the value of the item selected. The **variable name** is specified with the name attribute of the `<select>` element and the **value of an item** is specified with the value attribute of the `<option>` element.

- The **value** attribute of the `<select>` tag can be used to set the default option that will be selected initially.

```xml
<select name="selection_list" value="WAP">
  <option value="WML">WML Tutorial</option>
  <option value="WAP">WAP Tutorial</option>
  <option value="HTML">HTML Tutorial</option>
</select>
```

- The `<option>` element specifies a particular choice within a `<select>` element.

**Syntax**

```xml
<option title="label" value="value" onpick="url">content</option>
```

- Where content represents the text the device will display to represent the particular selection item and the action to perform if the user selects it.

```xml
<option value="1">
  <img src="/img1" localsrc="" alt="Choice 1"/>
</option>
```

### Attributes

<table>
<thead>
<tr>
<th>attribute</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>title</td>
<td>A label that identifies the option.</td>
</tr>
<tr>
<td>value</td>
<td>Specifies the value to assign to the variable defined in the <code>&lt;select&gt;</code> element name attribute</td>
</tr>
<tr>
<td>onpick</td>
<td>Specifies the URL to open if the user selects the option</td>
</tr>
</tbody>
</table>

**Example**

```xml
<wml><card>
<p>Please select your favorite color:
<select name="color">
  <option value="1">red</option>
  <option value="2">blue</option>
  <option value="3">green</option>
</select>
</p></card>
```
<select>

- The `<select>` element specifies a list of options from which the user can choose. You can specify either single- or multiple-choice `<select>` elements.

- **Syntax**

  ```xml
  <select title="label" multiple="boolean" name="variable"
          value="default" tabindex="n">content</select>
  ```

  **Attributes**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>title</td>
<td>Specifies a brief label for the <code>&lt;select&gt;</code> list.</td>
</tr>
<tr>
<td>multiple</td>
<td>true</td>
</tr>
<tr>
<td></td>
<td>Specifies whether the user can select multiple items.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the variable in which the device stores the value(s) associated with the option(s) chosen by the user.</td>
</tr>
<tr>
<td>value</td>
<td>A string specifying the default value(s) for the variable specified by the name attribute.</td>
</tr>
<tr>
<td>tabindex</td>
<td>The UP.Browser software does not currently support this attribute.</td>
</tr>
</tbody>
</table>

- **Example**

  ```xml
  <wml><card>
  <p>Please choose your favorite animals:
  <select multiple="true" name="x">
  <option value="D">Dog</option>
  <option value="C">Cat</option>
  <option value="H">Horse</option>
  </select>
  </p>
  <do type="accept"><noop/></do>
  </card></wml>
  ```
<optgroup>
  
  • The <optgroup> element allows you to group multiple <option> elements within a card.
  
  • Syntax

  
  \[
  \text{<optgroup title="label">content</optgroup>}
  \]

</optgroup>

<fieldset>
  
  • The <fieldset> element allows you to group multiple text or input items within a card. Specifying one or more <fieldset> elements lets you control how the device presents card content in order to simplify user navigation.
  
  • Syntax

  
  \[
  \text{<fieldset title="label">content</fieldset>}
  \]

  \text{Input fields}

  • Input fields are used to obtain alphanumeric data from users. The <input/> tag is used to create input fields

  \[
  \text{<input name="myname" maxlength="16"/>}
  \]

  • Like a selection list, an input field is associated with a variable, which stores the data entered by the user. The variable name is specified with the name attribute of the <input> element.

  • The max\text{length} attribute of the <input> element limits the number of characters that a user can enter in an input field.

  • The input field can contain a value by default.

  \[
  \text{<input name="FisrtName" maxlength="16" value="Ahmed"/>}
  \]

  • Syntax

  \[
  \text{<input name="variable" title="label" type="type" value="value" format="specifier" emptyok="boolean" size="n" max\text{length}="n" tab\text{index}="n"/>}
  \]
### Input Element (Attributes)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Attribute value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td><strong>Required.</strong> The name of the variable in which the device stores the text entered by the user.</td>
</tr>
<tr>
<td>title</td>
<td>Specifies a brief label for the input item. Some devices use the label as a tooltip when displaying the input field.</td>
</tr>
<tr>
<td>type</td>
<td>**text</td>
</tr>
<tr>
<td>value</td>
<td>Specifies the value of the variable named in the name attribute.</td>
</tr>
<tr>
<td>format</td>
<td>Specifies a data format that the user entry must match as the following:</td>
</tr>
<tr>
<td>emptyok</td>
<td>**true</td>
</tr>
<tr>
<td>size</td>
<td>The UP Browser software does not support this attribute.</td>
</tr>
<tr>
<td>maxlength</td>
<td>Specifies the maximum number of characters the user can enter.</td>
</tr>
<tr>
<td>tabindex</td>
<td>The UP Browser software does not support this attribute.</td>
</tr>
</tbody>
</table>

#### Tag Description

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Any symbolic or uppercase alphabetic character (no numbers)</td>
</tr>
<tr>
<td>a</td>
<td>Any symbolic or lowercase alphabetic character (no numbers)</td>
</tr>
<tr>
<td>N</td>
<td>Any numeric character (no symbols or alphabetic characters)</td>
</tr>
<tr>
<td>X</td>
<td>Any symbolic, numeric, or uppercase alphabetic character (not changeable to lowercase)</td>
</tr>
<tr>
<td>x</td>
<td>Any symbolic, numeric, or lowercase alphabetic character (not changeable to uppercase)</td>
</tr>
<tr>
<td>M</td>
<td>Any symbolic, numeric, or uppercase alphabetic character (changeable to lowercase) -- for multiple character input, defaults to uppercase first character</td>
</tr>
<tr>
<td>m</td>
<td>Any symbolic, numeric, or lowercase alphabetic character (changeable to uppercase) -- for multiple character input, defaults to lowercase first character</td>
</tr>
</tbody>
</table>
• **Example**

```xml
<wml>
  <card>
    <p>
      First Name:
      <input name="fname" maxlength="15" />
    </p>
    <p>
      Last Name:
      <input name="lname" maxlength="15" />
    </p>
    <p>
      State:
      <input name="state" maxlength="2" emptyok="true" value="CA" />
    </p>
    <p>
      Zipcode:
      <input name="zipcode" maxlength="10" />
    </p>
    <p>
      Password:
      <input name="password" maxlength="8" type="password"/>
    </p>
  </card>
</wml>
```

### Setting Variable Values in WML

- A major difference between WML and HTML is that WML has build-in **support of variables**.
- Variable names in WML are **case-sensitive**.
- All variables are stored as string. They have a global scope, which means once you have set the value of a variable, you can read it in any cards and decks.
- You can set the value of a variable in the following ways:
  1. Using the `<setvar/>` tag
  2. Using data collection tags `<select>` and `<input/>`

- **syntax**.
  - `$identifier`
  - `$(identifier)`

---

**Example 8.2**

```xml
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3//EN"
     "http://www.wapforum.org/DTD/wml13.dtd">
<wml>
  <card id="card1" title="free form input">
    <do type="accept" label="Get Res">
      <go href="#card2"></go>
    </do>
  </card>
</wml>
```
First Name:
<input type="text" name="first" value ="Fname" size ="8" maxlength="8" emptyok="no" format ="A*a" />
Last Name:
<input type="text" name="last" value ="Lname" size ="8" maxlength="8" emptyok="yes" format ="A*a" />
Age:
<input type="text" name="age" size ="2" maxlength="2" emptyok="yes" format ="2N" />
Country:
<select name="country">
  <option value ="jordan">Jordan</option>
  <option value ="Syria">Syria</option>
  <option value ="Ukraine">Ukraine</option>
  <option value ="Russia">Russia</option>
  <option value ="Italy">Italy</option>
</select>
<p>Your entry:
First Name:$first
Last Name:$last
Age: $age
country: $country
</p>