

GUI Widgets

James Baldo Jr.

SWE 432

Design and Implementation of Software for the Web

Widget Terms

- Screen : Entire display area
- Pixel : Each dot on the screen
- Bitmapped : Each pixel can be controlled
 - Typical: 256 X 256
 - Workstation 1000 X 1000
- Windows : Areas of display.
Each window is usually a process
- Widget : Window with specialized functions

Widget History

- **First generation** : Put characters on screen
- **Second generation** : Put text and pictures on screen
- **Now** : Put widgets on screen

**Widget: A building block for an interface.
Includes characters, text, pictures,
and other elements.**

Widgets

- **Label**
- **Events**
- **Form**
- **List**
- **Scroll Bar**
- **Push Button**
- **Radio Box**
- **Dialog**
- **Text Box**
- **Pull-down Menu**
- **Menu Bars**

WIDGET GUIDELINES

Label

- **Use for simple feedback**
- **Often combined with other widgets**
- **Simple text**
- **No events**

Widget Guidelines

Events

An event is an interaction with a GUI that can create a function call (*callback*)

- Moving onto a widget
- Moving off of a widget
- Clicking on a widget
(or even “button down” and “button up”)

Widget Guidelines

Form

- **A parent of other widgets**
- **Use when a set of widgets needs to be aggregated**
- **No callbacks**

Widget Guidelines

List

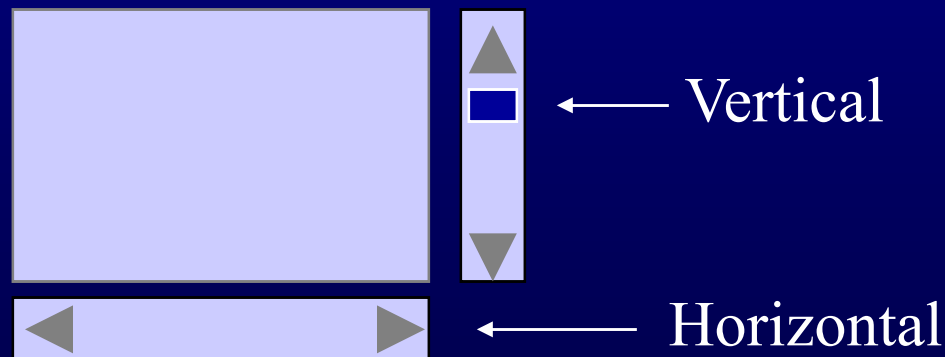
- Use as a menu
- Sometimes called *picklists*, *list boxes* and *listviews*
- All the menu guidelines should apply here
- Sometimes text only, sometimes text + icons



Widget Guidelines

Scroll Bar

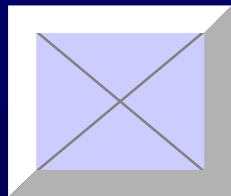
- Use when list or text is longer than will fit in the window
- Better to expand the window, if possible
- Scrollbars require mixing fine motor control (holding a button on a tiny icon) with large motor control (moving your arm)



Widget Guidelines

Push Button

- **Used to activate a particular action**
- **This is usually unrecoverable ... do not use in dangerous situations**
- **Usually offers no feedback**



Label
Parent
Callback

Widget Guidelines

Radio Box

- Selects one from of a set of mutually exclusive options
- Actually a specialized menu (single-selection)
 - Collection of checkboxes
- Uses
 - set some state in system
 - set options for customization
- 3 – 8 options
- Originally diamonds, MS changed to circles
- Very fast, but uses a lot of screen space

Widget Guidelines

Radio Box (2)

- Small group of mutually exclusive choices
- If crowded, use a drop-down list (slower)
- Same as single selection lists



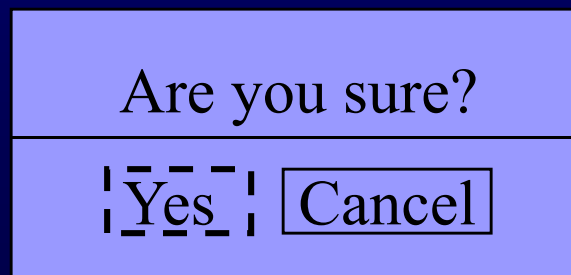
Pull-down Menus Vs. Radio Buttons

- They both accomplish the same thing
- Radio buttons are fixed on the screen,
Pull-down menus show up on-demand
- Radio buttons are faster and more convenient
- Use pull-down instead of radio buttons when:
 - More than 5 choices
 - Screen is crowded
 - Choices available depend on other selections (the choices change dynamically)

Widget Guidelines

Dialog

- Make sure that labels are clear and unambiguous Yes/No is sometimes not as clear as Yes/Cancel
- The user must know what will happen when a choice is made
- Built-in specialized dialog boxes are often convenient, but sometimes not exactly what is needed



Widget Guidelines

Text Box

- Appropriate size (vertically and horizontally)
- Do not use when it is possible to select
- This is the most flexible but slowest and most error-prone selection method
 - Also can present security vulnerabilities
- Operations: *insert, delete, copy, cut, paste, select*
- Text boxes must be validated
 - Active: Invalid keystrokes are ignored
 - Passive: String is checked after user enters data

Widget Guidelines

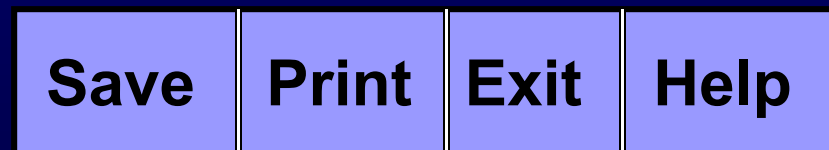
Pull-Down Menu

- Use as a menu
- Can be longer if there is a clear ordering , or searching is easy
(states, courses, ...)
- Can be longer if list is fixed and syntax must be perfect
 - That is, we can trade off searching time for errors on data entry.

Widget Guidelines

Menu Bars (or ButCons)

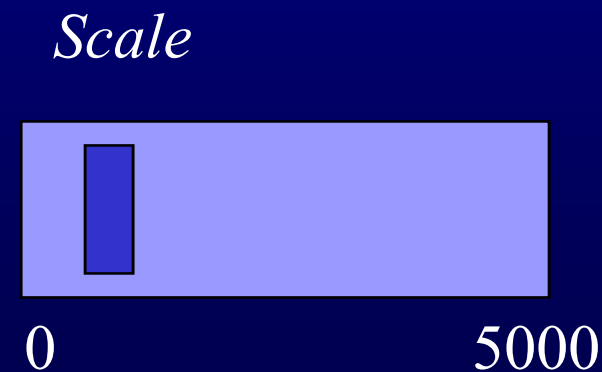
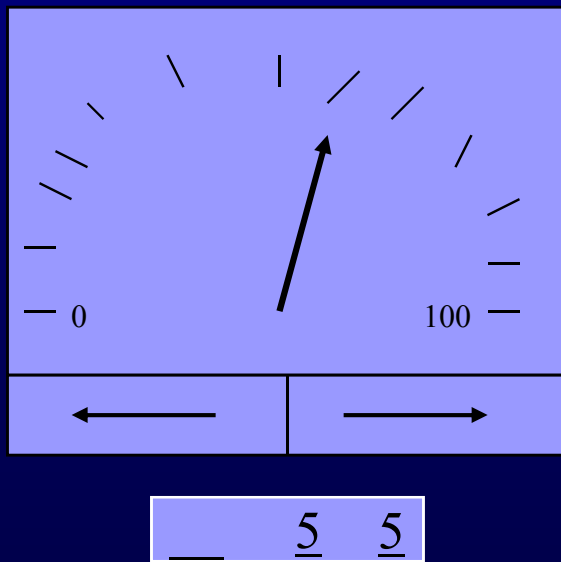
- Used for permanent menus
- Put choices that should always be available:
 - Crucial choices (Exit, Help, ...)
 - Most often used choices



Widget Guidelines

Bound Value – Scale

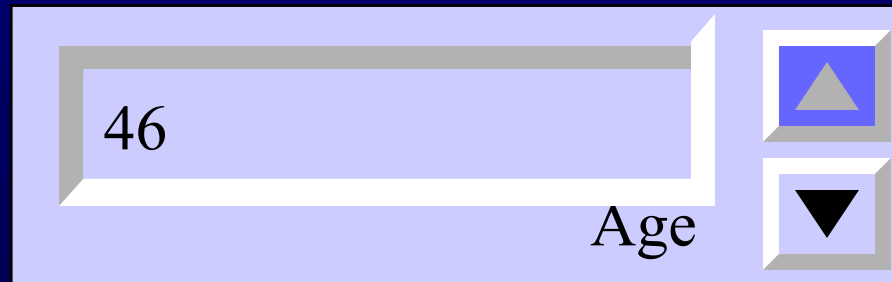
- Use when user needs to select a value from a large range
- Often combined with text selection for flexibility



Widget Guidelines

Spinner

- Use when user needs to select a value from a range
- The value has to be precise, but the value is fairly large
 - Age
 - Weight
 - Day of year
- Also can be combined with text selection for flexibility
- Edit window can choose to allow arbitrary (*invalid*) inputs



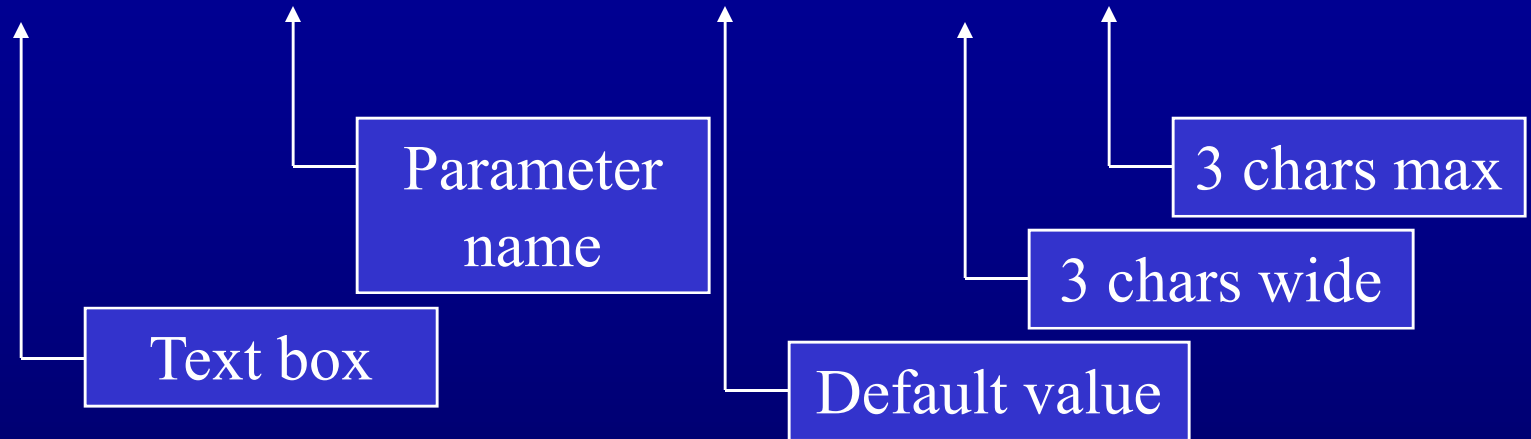
HTML Widgets – Form Elements

1. Textboxes
 1. Password boxes
 2. Text areas
2. Radio buttons
3. Checkboxes
4. Menus
5. Buttons
 1. Submit
 2. Reset
6. Tab order
7. Keyboard shortcuts

Next 7 slides ...

HTML Text Boxes

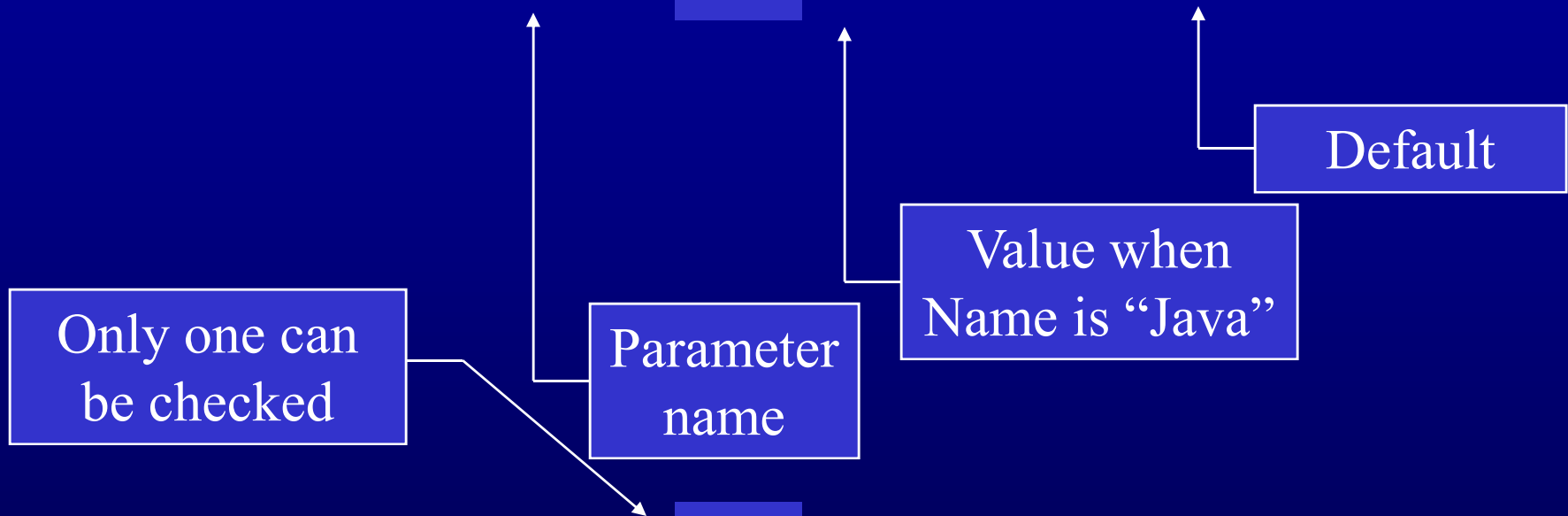
- `<INPUT Type="text" Name="Age" Value=0 Size=3 Maxlength=3>`



- `<INPUT Type="password" Name="Pword" Size=7 Maxlength=7>`
- `<TEXTAREA Name="Opinion" Rows=5 Cols=65>default</TEXTAREA>`

HTML Radio Buttons

- `<INPUT Type="radio" Name="Java" Value="none" Checked>`



- `<INPUT Type="radio" Name="Java" Value="beginner">`
- `<INPUT Type="radio" Name="Java" Value="inter">`
- `<INPUT Type="radio" Name="Java" Value="expert">`

HTML Checkboxes

- `<INPUT Type="checkbox" Name="Languages" Value="Java">`



- `<INPUT Type="checkbox" Name="Languages" Value="C++">`
- `<INPUT Type="checkbox" Name="Languages" Value="C">`
- `<INPUT Type="checkbox" Name="Languages" Value="Fortran">`

HTML Menus

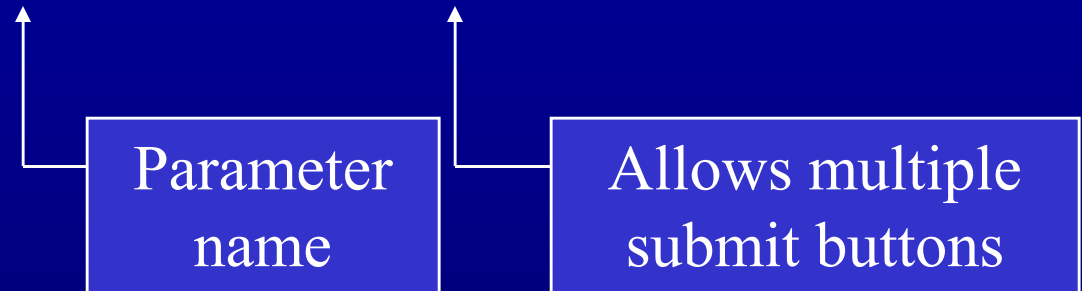
- `<SELECT Name="Major">`
 - `<OPTION Selected Value="CS">CS</OPTION >`
 - `<OPTION Value="CPE">CPE</OPTION>`
 - `<OPTION Value="ECE">ECE</OPTION>`
 - `<OPTION Value="EE">EE</OPTION>`
 - `<OPTION Value="SWE">SWE</OPTION>`
 - `<OPTION Value="SYST">SYST</OPTION>`
 - `<OPTION Value="Other">Other</OPTION>`
- `</SELECT>`

Parameter
name

Default

HTML Buttons

- `<BUTTON Type="submit" Name="submit" Value="Submit">`



- `<BUTTON Type="submit" Name="submit" Value="Retrieve">`
- `<BUTTON Type="submit" Name="submit" Value="Update">`
- `<BUTTON Type="reset" Name="reset">`

HTML Tab Order and Keyboard Shortcuts

- `<BUTTON Type="submit" Name="submit" Value="Retrieve" Tabindex=1 Accesskey="R" >`
- `<BUTTON Type="submit" Name="submit" Value="Update" Tabindex=2 Accesskey="U" >`
- `<BUTTON Type="reset" Name="reset" Tabindex=3 Accesskey="S" >`

Setting Focus

- **HTML does not have any facilities for setting the initial focus into a field in a form**
- **Javascript can be used ...**

```
<SCRIPT type="text/javascript">  
function setFocus()  
{  
    document.focus.firstName.focus();  
}  
</SCRIPT>
```

```
<BODY onload="setFocus()>
```

```
<FORM method="post" name="focus" action="">
```

```
First name: <INPUT type="text" name="firstName">
```

```
Last name: <INPUT type="text" name="lastName">
```