

Web Programming Step by Step

Lecture 18

More Events and Validation

Reading: 9.2 - 9.3

Except where otherwise noted, the contents of this presentation are Copyright 2009 Marty Stepp and Jessica Miller.



Page/window events (9.2.5)

name	description
load	the browser loads the page
unload	the browser exits the page
resize	the browser window is resized
contextmenu	the user right-clicks to pop up a context menu
error	an error occurs when loading a document or an image

- The above events can be handled on the global window object. Also:

```
// best way to attach event handlers on page load
window.onload = function() { ... };
document.observe("dom:loaded", function() {
    $("orderform").observe("submit", verify);
});
```

JS

Form events (9.2.4)

event name	description
<code>submit</code>	form is being submitted
<code>reset</code>	form is being reset
<code>change</code>	the text or state of a form control has changed

```
window.observe("load", function() {
  $("orderform").observe("submit", verify);
});

function verify(event) {
  if ($("#zipcode").value.length < 5) {
    event.stop();           // cancel form submission unless
  }                         // zip code is 5 chars long
}
```

JS

Prototype and forms (9.1.6)

```
$("#id")["name"]
```

JS

- gets parameter with given **name** from form with given **id**

```
$F("id")
```

JS

- `$F` returns the value of a form control with the given **id**

```
var name = $F("username");
if (name.length < 4) {
  $("#username").clear();
  $("#login").disable();
}
```

JS

- other form control methods:

<code>activate</code>	<code>clear</code>	<code>disable</code>	<code>enable</code>
<code>focus</code>	<code>getValue</code>	<code>present</code>	<code>select</code>

Client-side validation code

```
<form id="exampleform" action="http://foo.com/foo.php">
```

HTML

```
window.onload = function() {  
  $("#exampleform").onsubmit = checkData;  
};  
  
function checkData(event) {  
  if ($("#city").value == "" || ($("#state").value.length != 2) {  
    Event.stop(event);  
    alert("Error, invalid city/state."); // show error message  
  }  
}
```

JS

- forms expose `onsubmit` and `onreset` events
- to abort a form submission, call Prototype's `Event.stop` on the event

Regular expressions in JavaScript

- `string.match(regex)`
 - if string fits the pattern, returns the matching text; else returns `null`
 - can be used as a Boolean truthy/falsey test:

```
var name = $("#name").value;  
if (name.match(/[a-z]+/)) { ... }
```
- an `i` can be placed after the regex for a case-insensitive match
 - `name.match(/Marty/i)` will match "marty", "MaRtY", ...

Replacing text with regular expressions

- `string.replace(regex, "text")`
 - replaces the first occurrence of given pattern with the given text
 - `var str = "Marty Stepp";`
`str.replace(/[a-z]/, "x")` returns "M**x**rty Stepp"
 - returns the modified string as its result; must be stored
str = `str.replace(/[a-z]/, "x")`
- a `g` can be placed after the regex for a global match (replace all occurrences)
 - `str.replace(/[a-z]/g, "x")` returns "Mxxxx Sxxxx"
- using a regex as a filter
 - `str = str.replace(/^[A-Z]+/g, "")` turns `str` into "MS"

Keyboard/text events (9.2.3)

name	description
<code>keydown</code>	user presses a key while this element has keyboard focus
<code>keyup</code>	user releases a key while this element has keyboard focus
<code>keypress</code>	user presses and releases a key while this element has keyboard focus
<code>focus</code>	this element gains keyboard focus
<code>blur</code>	this element loses keyboard focus
<code>select</code>	this element's text is selected or deselected)

- **focus**: the attention of the user's keyboard (given to one element at a time)

Key event objects

property name	description
keyCode	ASCII integer value of key that was pressed (convert to char with <code>String.fromCharCode</code>)
altKey, ctrlKey, shiftKey	true if Alt/Ctrl/Shift key is being held

Event.KEY_BACKSPACE	Event.KEY_DELETE	Event.KEY_DOWN	Event.KEY_END
Event.KEY_ESC	Event.KEY_HOME	Event.KEY_LEFT	Event.KEY_PAGEDOWN
Event.KEY_PAGEUP	Event.KEY_RETURN	Event.KEY_RIGHT	Event.KEY_TAB
Event.KEY_UP			

Prototype's key code constants

- issue: if the event you attach your listener to doesn't have the focus, you won't hear the event
 - possible solution: attach key listener to entire page body, outer element, etc.