JS & JQuery

Colin Dao

JS Basic

JS - Using in a browser

Option 1:

```
<script type="text/javascript">
  // nen co ; o cuoi moi cau lenh
  document.write("<h1> Hello World </h1>");

  // run on web load
  $( document ).ready(function() {
    console.log( "ready!" );
  });

</script>
```

Option 2:

```
<script scrc=""></script>
```

JS - Variables

Syntax: `var var_name = value;`

Examples:

```
var name = "Let's Javascript";
var i = 1;
var fl = 1.11;
var boo =  true;
```

JS – Primitives data types

- JavaScript has three “primitive” types:
  number, string, boolean

- Everything else is an object

1. Numbers are always stored as floating-point values
2. Strings may be enclosed in single quotes or double quotes
3. Booleans are either true or false
4. 0, "0", '', undefined, null, NaN are false , other values are true

JS - Statements

```
var thinkingOf;       // number computer has choose
var guess;            // User latest guess

thinkingOf = Math.ceil(Math.random()*1000); // initialize number

// play until user guess the number
guess = window.prompt("I'm thinking of a number between 1 and 1000." +
      " what is it?", "" );

while (guess != thinkingOf)
{
  // Evaluate the user's guess
  if ( guess < thinkingOf ) {
      guess = window.prompt(guess + " was too low. Guess Again.", "");
  } else {
      guess = window.prompt(guess + " was too high. Guess Again.", "");
  }
}

window.alert(guess + " is correct!"); // Game over, congratulate the user
```

JS – Function

```
var f = function(arg){
 // function body
};
```

Example:

```
var hello = function(){
  console.log("Hello World");
};

function hello() {
  console.log("Hello World");
};
```

JS – Object

JS → Object oriented

```
var obj = {};
```

Example 1:

```
var obj = {
    greet: function(){
        console.log('Hello');
    }
};

obj.greet();
```

JS – Object

```
var obj = {};

Example 2:

    var obj = {
        prop1: 'test1',
        prop2: 'test2',
        prop3: function(){}
    };

    // ! this loop does not act in order
    for(var prop in obj){
        console.log(obj[prop]);
    };
```

JQuery Basic

Jquery – What is?

- Focus on the **interaction between JavaScript and HTML**
- (Almost) every operation are
  + **Find something**
  + **Do something to it**

Jquery – Syntax

$(selector).action()

- A $ sign to define/access jQuery
- A (selector) to "query (or find)" HTML elements
- A jQuery action() to be performed on the element(s)

Examples:

```
$(this).hide(); // hides the current element.

$("p").hide(); // hides all <p> elements.

$(".test").hide(); // hides all elements with class="test".

$("#test").hide(); // hides the element with id="test".
```

Jquery – Document Ready Event

```
$(document).ready(function(){

    // Fires after web load
    // JS methods go here...

});
```

Jquery – Collections

```
$(document).ready(function(){
    $('div.section').length;
    $('div.section')[0];
    $('div.section').each(function(i){
        console.log("Item " + i + " is ", this);
    });
});
```

Jquery – HTML

```
$('span#msg').text('This content was updated!');

$('div#intro').html('<em>Hello World</em>');

$('a.nav').attr('href', 'http://flickr.com/');

$('a.nav').attr({
    'href': 'http://flickr.com/',
    'id': 'flickr'
});

$('#intro').removeAttr('id');
```

Jquery – CSS

```
$('#intro').addClass('highlighted');

$('#intro').removeClass('highlighted');

$('#intro').toggleClass('highlighted');

$('p').css({ 'font-size': '20px', color: 'red' });
```

Jquery – Traversing the DOM

```
$('div.section').parent();

$('div.section').next();

$('div.section').prev();

$('h1:first').parents();
```

Jquery – Handling Events

```
$('a:first').click(function(){

    $(this).css({backgroundColor: 'orange'});

});

$("#p1").hover(
  function(){
    alert("You entered p1!");
  },
  function(){
    alert("Bye! You now leave p1!");
  }
);
```

Jquery – Effects (1)

```
$("#hide").click(function(){
    $("p").hide(500);
});

$("#show").click(function(){
    $("p").show(500);
});

$("button").click(function(){
    $("p").toggle(500);
});
```

Others

- fadeIn(timeSpeed, callback), fadeOut(), fadeToggle()

- slideDown(timeSpeed, callback), slideUp(), slideToggle()

Jquery – Effects (2)

```
$("button").click(function(){
    $("div").animate({
        left: '250px',
        opacity: '0.5',
        height: '150px',
        width: '150px'
    }, 1000);
});
```

Jquery – Effects (3)

```
$("button").click(function(){
    $("p").hide("slow", function(){
        alert("The paragraph is now hidden");
    });
});
```

AJAX & JSON

Rails – Return JSON result

```
respond_to do |format|
  format.html {}
  format.json { render :json => { user: @user }, status: :ok}
end

// Example JSON result
{
    user: {
      id: 1,
      name: 'test',
      email: 'test@example.com'
    }
}
```

AJAX – HANDLE JSON

```
var menuId = $( "ul.nav" ).first().attr( "id" );
var userName = $( "input" ).first().value;

$.ajax({
    url: "/users.json",
//    method: "POST",
    type: 'POST',
//    type: 'GET',
    data: {
        id : menuId,
        name: userName
//        _method: 'PUT'
    },
    dataType: "json",
    success: function(data) {
        $( "#log" ).html( data );
    },
    error: function(jqXHR, textStatus) {
        alert( "Request failed: " + textStatus );
    }
});
```

End