



# Πανεπιστήμιο Πειραιώς Τμήμα Ψηφιακών Συστημάτων

## Διαδικτυακά και Φορητά Πληροφοριακά Συστήματα (Δικτυακές Υπηρεσίες)

2<sup>η</sup> Διάλεξη

Δημοσθένης Κυριαζής

Δευτέρα 5 Μαρτίου 2018



# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# Δημιουργία HTML αρχείου

## □ Βασικός σκελετός είναι

```
<!DOCTYPE html>
<html>
    <head>
        <title> A Basic Document Template </title>
    </head>
    <body>
        Body Element
    </body>
</html>
```

## □ Παρατηρήσεις

- Ορισμένα από τα tags είναι δυνατόν να παραλειφθούν
- Οι browsers θεωρούν όλα τα λευκά διαστήματα ως απλό κενό
- Πολλαπλά κενά συμπυκνώνονται σε ένα



# Βασική δομή HTML5 σελίδας

---

```
<header>
```

```
<nav>
```

```
<article>
```

```
    <section>
```

```
    <section>
```

```
    <section>
```

```
<aside>
```

```
<footer>
```



# Φόρμες

---

- Το tag form οριοθετεί την αρχή της φόρμας
  - Ορίζει το script ή το URL στο οποίο αποστέλλονται τα εισαχθέντα δεδομένα και την μέθοδο μετάδοσης τους
  - Δύο βασικά attributes
    - Action: Ορίζει την URL που θα τα λάβει
    - Method: Λαμβάνει δύο δυνατές τιμές
      - post στέλνει όλη την πληροφορία χωριστά από την URL
      - get συνδέει την πληροφορία στο τέλος της URL
      - Ενδείκνυται η χρήση της πρώτης μεθόδου επειδή τα URLs έχουν συγκεκριμένο μήκος που δεν μπορούν να υπερβούν
- Παράδειγμα
  - `<form method="post" action="/cgi-bin/my_script">`
  - `...</form>`



# <input>

---

- Επιτρέπει διάφορες επιλογές εισόδου περιεχομένου
  - Simple text fields → text
  - Password fields → password
  - Radio buttons → radio
  - Check boxes → checkbox
  - Submit buttons → submit
  - Reset buttons → reset



# Audio (/ video) elements

---

- Το audio element χρησιμοποιείται για την εισαγωγή ηχητικών αποσπασμάτων στις σελίδες
  - Οι browsers επιτρέπουν την αναπαραγωγή συγκεκριμένων μορφών
- Υποστηρίζει διάφορα attributes
  - src: Ορίζει το αρχείο αναπαραγωγής
  - controls: Ορίζει αν θα εμφανίζονται στοιχεία ελέγχου (προεπιλογή να μην εμφανίζονται). Μπορούν να οριστούν συγκεκριμένα στοιχεία μέσω JavaScript
  - autoplay: Ορίζει αν θα αρχίσει αυτόματα η αναπαραγωγή
  - preload: Ορίζει τι να κάνει ο browser αν δεν έχει επιλεγεί autoplay. Μπορεί να λάβει τιμές:
    - none: Ο browser δεν αναπαράγει το αρχείο μέχρι ο χρήστης να το επιλέξει
    - auto: Ο browser «κατεβάζει» το αρχείο όταν «φορτώσει» η σελίδα
    - metadata: Ο browser απλά αποθηκεύει πληροφορία αναφορικά με το μέγεθος, την πρώτη εικόνα και τη διάρκεια
  - loop: Ορίζει αν θα επαναλαμβάνεται η αναπαραγωγή όταν ολοκληρωθεί



# Meter element

- Χρησιμοποιείται για να προσδιορίσει την ποσοστιαία μεταβολή μιας τιμής
- Υποστηρίζει διάφορα attributes
  - Value: Η μοναδική που απαιτείται καθώς ορίζει την τιμή που θα εμφανιστεί στο meter
  - Min / max: Προεπιλεγμένες τιμές στο 0 και 1.0
  - Low / high / optimum: Χρησιμοποιούνται για να ορίσουν κομμάτια για τα συγκεκριμένα σημεία του meter
- Παραδείγματα
  - <p>Course completion status: <meter value="0.80">80%</meter></p>
  - <p>Lectures completed: <meter min="0" max="10" value="8">8 out of 10</meter></p>





# Ημερολόγιο και ώρα

- Δεν απαιτείται πλέον η χρήση JavaScript λόγω νέων μεθόδων εισαγωγής
  - Input type date
    - Επιλογή ημέρας από ημερολόγιο
    - Π.χ. `<input type="date" name="birthday">`
  - Input type datetime
    - Επιλογή ημερομηνίας και ώρας (βάσει UTC)
    - Π.χ. `<input type="datetime" name="bdaytime">`
  - Input type datetime-local
    - Επιλογή ημερομηνίας και ώρας (βάσει τοπικής ώρας)
  - Input type month / week
    - Επιλογή μήνα / εβδομάδας
    - Π.χ. `<input type="month" name="bdaymonth">`
- Χρήση attributes min και max για τον ορισμό ελάχιστων / μέγιστων τιμών (στη μορφή YYYY-MM-DD)
  - Π.χ. `<input type="date" name="birthday" min="2014-01-10">`



# Επιλογή αριθμών

---

- Νέο input type number για την εισαγωγή / επιλογή αριθμού
  - Δεν επιτρέπεται η εισαγωγή μη αριθμητικών χαρακτήρων
  - Με τα min και max attributes μπορεί να ορισθεί συγκεκριμένο εύρος τιμών
  - Με το step attribute μπορεί να ορισθεί το βήμα
  - Π.χ. `<input type="number" id="numberitem" name="numberitem" min="1" max="10" step="2" />`



# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# Web storage

## □ sessionStorage / localStorage

```
<p><button onclick="butClicks()" type="button">Test</button></p>  
<div id="sesStor"></div>
```

```
function butClicks() {  
  if(typeof(Storage)!=="undefined") {  
    if (sessionStorage.clicks) {  
      sessionStorage.clicks=Number(sessionStorage.clicks)+1; }  
    else {  
      sessionStorage.clicks=1; }  
    document.getElementById("sesStor").innerHTML="Number of clicks: " +  
    sessionStorage.clicks; }  
    else { document.getElementById("sesStor").innerHTML="Not supported"; } }
```



# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# Web workers (1/3)

## □ Διαδικασία

- Δημιουργία αρχείου web worker (ως αρχείο Javascript)

```
var i=11;  
function countdown () {  
    i=i-1;  
    postMessage(i);  
    setTimeout("countdown()",1000); }  
countdown();
```

Η μέθοδος `postMessage()` χρησιμοποιείται για να αποστείλει ένα μήνυμα στη σελίδα

- Δημιουργία αντικειμένου web worker στη σελίδα

```
if(typeof(ww)=="undefined") {  
    ww=new Worker("xx.js"); }
```

- Προσθήκη event στη σελίδα για ανταλλαγή μηνυμάτων

```
ww.onmessage=function(event){  
    document.getElementById("result").innerHTML=event.data; };
```

Όταν ο web worker αποστέλλει ένα μήνυμα εκτελείται ο παραπάνω κώδικας και τα δεδομένα από τον worker αποθηκεύονται στο `event.data`

- Διακοπή λειτουργίας: `ww.terminate();`



# Web workers (2/3)

---

## □ Παράδειγμα

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p>Countdown...<output id="result"></output></p>
```

```
<button onclick="startWorker()">Start</button>
```

```
<button onclick="stopWorker()">Stop</button>
```

```
<script>
```

```
var ww;
```

```
function startWorker()
```

```
{
```

```
  ww=new Worker("xx.js");
```



# Web workers (3/3)

---

```
ww.onmessage = function (event) {  
    if (event.data=="0")  
    {  
        document.getElementById("result").innerHTML="READY";  
        ww.terminate();  
    }  
    else  
    {  
        document.getElementById("result").innerHTML=event.data;  
    }  
};  
}  
function stopWorker()  
{  
    ww.terminate();  
}  
</script>
```

</body>

</html>





# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# Geolocation (1/2)

---

## □ Παράδειγμα

```
<p id="geoTest">Coordinates?</p>
<button onclick="getLocation()">Submit</button>
<div id="myMap"></div>
<script>
    var x=document.getElementById("geoTest");
    function getLocation() {
    if (navigator.geolocation) {
        navigator.geolocation.getCurrentPosition(showPosition); }
    else {
        x.innerHTML="Geolocation not supported";} }
```



# Geolocation (2/2)

---

(συνέχεια)

```
function showPosition(position) {  
    x.innerHTML="Geografiko platos: " + position.coords.latitude +  
    "<br>Mikos: " + position.coords.longitude;  
  
    var latlon=position.coords.latitude+","+position.coords.longitude;  
    var  
    img_url="http://maps.googleapis.com/maps/api/staticmap?center=" +  
    latlon+"&zoom=14&size=500x500";  
    document.getElementById("myMap").innerHTML="<img  
    src='"+img_url+"'>";  
}  
</script>
```



# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# Εισαγωγή

---

- Βιβλιοθήκη Javascript
- Επιτρέπει
  - Επιλογή elements, προσθήκη περιεχομένου, απόκρυψη / εμφάνιση περιεχομένου, τροποποίηση tag attributes, καθορισμό τιμών πεδίων φόρμας, ...
- Ουσιαστικά πρόκειται για μια συλλογή συναρτήσεων JavaScript που μπορούν να προστεθούν σε δικτυακές εφαρμογές






# Χρήση jQuery

---

- Πρώτο βήμα: Download jQuery library
- Η jQuery library είναι ένα αρχείο JavaScript που μπορεί να προσπελαστεί με 2 τρόπους:
  - Download jquery.js και χρήση τοπικά από τον υπολογιστή
  - Χρήση έκδοσης (hosted version) που βρίσκεται διαθέσιμη στο δίκτυο - CDN (content delivery network)
- Για προγραμματισμό προτείνεται ο πρώτος τρόπος (δεν απαιτεί σύνδεση στο δίκτυο)

# www.jquery.com






write less, do more.

PluginsContributeEventsSupportjQuery Foundation


DownloadAPI DocumentationBlogPluginsBrowser Support

Search




**Lightweight Footprint**

Only 32kB minified and gzipped.  
Can also be included as an AMD module




**CSS3 Compliant**

Supports CSS3 selectors to find elements as well as in style property manipulation



**Cross-Browser**

Chrome, Edge, Firefox, IE, Safari, Android, iOS, and more




**Download jQuery**  
v3.1.1

[View Source on GitHub →](#)  
[How jQuery Works →](#)


## What is jQuery?

jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript.


## Corporate Members




famo.us



WORDPRESS



neobux




maxcdn

Support from our corporate members makes it possible for the jQuery Foundation to continue our work on our JavaScript libraries and pushing the open web forward with events and participation in the standards process. View our [members page](#) for a full list of corporate and individual members.

## Other jQuery Foundation Projects

## Resources

- [jQuery Core API Documentation](#)
- [jQuery Learning Center](#)
- [jQuery Blog](#)
- [Contribute to jQuery](#)
- [About the jQuery Foundation](#)
- [Browse or Submit jQuery Bugs](#)



TRY jQuery

Learn jQuery in your browser!

# Προσθήκη jQuery στον κώδικα (1/2)

---



## □ Στο πεδίο <head>

### ■ Τοπικά

```
<script src="jquery-1.11.2.min.js"></script>
```

### ■ Για Google CDN

```
<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js">  
</script>
```



# Προσθήκη jQuery στον κώδικα (2/2)

---



- Συχνά υπάρχουν αναφορές στο stylesheet
  - Προσθήκη του script στο τέλος του head (μετά στο style)
- Προσθήκη επιπλέον μεθόδων
  - jQuery UI project (<http://jqueryui.com>)
    - ▣ Π.χ. Datepicker, alert boxes, κλπ



# jQuery & DOM

---

- Η jQuery επιτρέπει την αλλαγή στοιχείων του DOM (και προσθήκη στοιχείων) με χρήση selectors
  - Π.χ. `$(selector)`
- Στην πραγματικότητα δεν προσφέρει νέες μεθόδους αλλά χρησιμοποιεί συγκεκριμένες της JavaScript και τις κάνει εύκολα προσβάσιμες



# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# document.ready (1/3)

---

- Η πιο σημαντική μέθοδος (event handler) της jQuery!
  - Εκτελείται μόνο όταν το DOM έχει «φορτωθεί» πλήρως
  - Η πρόσθετη αξία της jQuery (όπως και της JavaScript) είναι η αλλαγή του DOM, άρα θα πρέπει πρώτα να είναι έτοιμο πριν γίνει η οποιαδήποτε αλλαγή!
  - Τοποθετώντας όλο τον κώδικα σε αυτή τη μέθοδο είναι βέβαιο ότι το DOM είναι έτοιμο όταν εκτελείται ο κώδικας.



# document.ready (2/3)

---

## □ Επεξήγηση

- Χρήση selector: `$(document)`
- Εφαρμογή μεθόδου `.ready(function() {..} )` η οποία εφαρμόζει τον κώδικα όταν έχει ολοκληρωθεί το DOM
  - Π.χ. `alert("DOM loaded");`
- Χρησιμοποιούνται παρενθέσεις γιατί μεταφέρεται ως όρισμα τη συνάρτηση για το ready event στον event handler

## □ Σύνοψη

```
$(document).ready(function() {  
    //jQuery DOM code goes here  
});
```



# document.ready (3/3)

- Κώδικας JavaScript (όπως μεταβλητές, πίνακες, κλπ) μπορούν να προστεθούν εκτός του event handler

```
$(function() {
```

```
    ...
```

```
});
```

```
var numShows=3;  
var numTickets=15;  
$(document).ready(function() {  
    for(i=0; i<numTickets; i++) {  
        var numTotal = i + 1;  
        $(".container").append("<p>There are " + numTotal + " tickets available</p>");  
    } //end for loop  
} );
```



# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - `document.ready`
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# Selectors, Filters, CSS (1/2)

---

- Οι Selectors είναι το βασικό εργαλείο της jQuery
  - Απαιτούνται για την επιλογή του στοιχείου του DOM το οποίο θα τροποποιηθεί
  - Η σύνταξη είναι ακριβώς η ίδια με τους CSS selectors για ID, class, tag, κλπ (υπάρχουν κάποιοι επιπλέον)
  - Με χρήση selector, γίνεται έλεγχος όλου του DOM για τα elements που ορίζονται στο selector, και επιλέγονται
- Τα Filters χρησιμοποιούνται σε συνδυασμό με τους selectors για την επιλογή συγκεκριμένων στοιχείων (π.χ. βάσει θέσης, επιλογής, εμφάνισης, κλπ)





# Selectors, Filters, CSS (2/2)

- Αποτελούνται από τα εξής μέρη
  - jQuery (συνήθως χρησιμοποιείται το σύμβολο \$, αλλά το jQuery επίσης λειτουργεί)
  - Τα DOM elements που επιλέγονται – με χρήση ‘ ’ μέσα σε παρενθέσεις
  - Η μέθοδος jQuery method που θα εφαρμοστεί

jQuery	Selector	jQuery method / action
\$ ή jQuery	( ‘div’ )	.css( ‘border’, ‘1px solid black’ );



## \* Selector

---

- Επιλογή όλων των elements στο DOM



# ID Selector

---

- Ο jQuery selector για την επιλογή IDs συντάσσεται ως: ( '#' )
  - Πρέπει πάντα να περιλαμβάνεται το σύμβολο # - διαφορετικά δεν λειτουργεί σωστά!
- Χρησιμοποιεί τη μέθοδο της JavaScript getElementById()



# Element Selector

---

- Ο jQuery selector για την επιλογή elements συντάσσεται ως: (‘όνομα element’)
- Χρησιμοποιεί τη μέθοδο της JavaScript `getElementsByTagName()`
- Π.χ. Για όλα τα `<a>` tag μιας σελίδας:
  - `var linksList = $('a');`



# Class Selector

---

- Ο jQuery selector για την επιλογή classes συντάσσεται ως: ( `' .όνομα class'` )
  - Πρέπει πάντα να περιλαμβάνεται το σύμβολο `.` - διαφορετικά δεν λειτουργεί σωστά!
- Χρησιμοποιείται τη μέθοδο της JavaScript `getElementByClassName()`
- Π.χ. Για όλες τις κλάσεις `jq2014`
  - `$('jq2014')`



# Parent-child selectors

---

- Χρησιμοποιούνται για την επιλογή elements όταν tag, CSS, και ID elements δεν μπορούν να χρησιμοποιηθούν
- Βασική χρήση σε navigation menus
- Επιτρέπουν την επιλογή των απευθείας «παιδιών» ενός element με χρήση του >
  - Π.χ. Ο selector \$('body > p') θα επιλέξει όλα τα elements παραγράφων αρκεί να είναι παιδιά του body element



# Descendent Selectors

---

- Οι Parent-child selectors εφαρμόζονται για απευθείας παιδιά (π.χ. `<li>` tags μέσα σε `<ul>` tags). Όταν απαιτείται η επιλογή παιδιών σε μεγαλύτερο «βάθος» σε σχέση με το αρχικό element χρησιμοποιούνται descendent selectors
- Η διαφορά στη σύνταξη είναι στο `>` που σε αυτή την περίπτωση δεν χρησιμοποιείται



# Επιλογή πολλαπλών elements

---

- Για την επιλογή πολλαπλών elements (π.χ. classes, IDs, HTML tags) καθώς και σχέσεων parent-child, γίνεται χρήση jQuery selector που διαχωρίζονται με κόμμα μεταξύ τους





# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# Attribute selector (1/2)

---

- Επιτρέπει την επιλογή elements εάν έχουν ένα συγκεκριμένο attribute
- Χρησιμοποιούνται επίσης για τον έλεγχο της τιμής του attribute
- Π.χ. Για `<img>` tags που έχουν το alt attribute
  - `$('img[alt]')`



# Attribute selector (2/2)

## □ Σύνταξη

- [attribute] – για το συγκεκριμένο attribute
- [attribute="value"] – για elements με συγκεκριμένη τιμή στο attribute, π.χ. text boxes σε μια φόρμα: `$('input[type="text"]')`
- [attribute^="value"] – για elements με την τιμή του attribute να ξεκινάει σε συγκεκριμένη τιμή - π.χ. εξωτερικά links:  
`$('a[href^="http://"]')` ή mailto links: `$('a[href^="mailto:"]')`
- [attribute\$="value"] – για elements που η τιμή του attribute τελειώνει σε συγκεκριμένη τιμή - π.χ. file extensions για PDF:  
`$('a[href$=".pdf"]')`
- [attribute\*="value"] – για elements που η τιμή του attribute περιλαμβάνει το συγκεκριμένη τιμή σε οποιοδήποτε σημείο – π.χ. link προς ένα domain: `$('a[href*="unipi.gr"]')`



# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# Φιλτράρισμα (1/3)

---

- Χρησιμοποιούνται όταν ο στόχος είναι η επιλογή ενός ή συγκεκριμένων elements στο DOM
  - Βασικό για δυναμικές σελίδες που αλλάζουν βάσει request
- Το φίλτρο ορίζεται ως :filter



## Φιλτράρισμα (2/3)

Όνομα	Περιγραφή
:even and :odd	Εύρεση even και odd βάσει του index
:header	Εύρεση elements με <h1>...<h6> tags
:first (:last και :only)	Εύρεση πρώτου, τελευταίου, ή μοναδικού
:nth-child	Εύρεση ν-οστού
:parent	Εύρεση parent ενός element
:visible	Εύρεση element που είναι ορατό
:hidden	Εύρεση element που είναι κρυμμένο
:contains('κείμενο')	Εύρεση element που περιλαμβάνει συγκεκριμένο κείμενο



## Φιλτράρισμα (3/3)

---

- Χρήση :not() φίλτρου για την εύρεση elements που δεν ταιριάζουν σε κάποιο τύπο selector
  - Π.χ. Επιλογή όλων των <a> tag εκτός αυτών της κλάσης  
testButton: \$('a:not(.testButton)');
- Μπορεί να χρησιμοποιηθεί με όλα τα jQuery filters και τους αντίστοιχους selectors
  - Π.χ. Επιλογή όλων των link που δεν ξεκινούν με http://:  
\$('a:not([href^="http://"])')



# Loops

---

- Loops σε σύνολο elements επιτυγχάνονται με συγκεκριμένες μεθόδους της jQuery αυτόματα
  - Π.χ. Επιλογή όλων των εικόνων σε ένα <div> tag με ID=slideshow και στη συνέχεια απόκρυψη αυτών των εικόνων: `$('#slideshow img').hide();`





# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# Προσθήκη κειμένου & HTML (1/2)

---

- Με χρήση της `html()` είναι η αντικατάσταση του κειμένου μέσα στο συγκεκριμένο `element` με άλλο HTML
- Η χρήση της `text()` είναι ίδια με την `.html()` αλλά δεν μπορεί να περιλαμβάνει HTML tags
  - Χρήσιμη για την αλλαγή του κειμένου σε κάποιο tag
- Η `append()` προσθέτει κείμενο ως το τελευταίο `child element` για το συγκεκριμένο `element`
- Η `prepend()` λειτουργεί όπως η `append()`, αλλά προσθέτει αμέσως μετά το αρχικό tag



## Προσθήκη κειμένου & HTML (2/2)

---

- Για την προσθήκη HTML εκτός κάποια επιλογής (πριν ή μετά από το αρχικό tag ενός επιλεγμένου element)
  - before()
  - after()



# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# Διαγραφή επιλογών

- Για την αντικατάσταση ή διαγραφή ενός επιλεγμένου element γίνεται χρήση της μεθόδου `remove()`
  - Π.χ. dialog box με ID `porup`: `$('#porup').remove();`
- Η μέθοδος `remove()` δεν περιορίζεται σε ένα μόνο επιλεγμένο element αλλά μπορεί να εφαρμοστεί σε πλήθος
  - Π.χ. διαγραφή όλως των `<span>` tags που ανήκουν στην κλάση `error`: `$('.span.error').remove();`



# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS



# Επεξεργασία κλάσεων

- Με χρήση μεθόδων για την προσθήκη / διαγραφή / αντικατάστασή
  - `addClass()`: Προσθήκη κλάσης (δεν περιλαμβάνεται η .)
    - ▣ Π.χ. `$('a[href^="http://"]').addClass("intro");`
  - `removeClass()`: Διαγραφή κλάσης
  - `toggleClass()`: Αντικατάσταση κλάσης (χρησιμοποιείται για δυναμικές αλλαγές όπως προκύπτουν συνήθως από radio buttons)



# Σημερινή διάλεξη

---

- Σύνοψη προηγούμενης διάλεξης
- HTML5 Hands-on
  - Web storage
  - Web workers
  - Geolocation
- jQuery
  - Εισαγωγή
  - document.ready
  - Selectors, Filters, CSS
  - Attribute selector
  - Φιλτράρισμα
  - Loops
  - Προσθήκη κειμένου & HTML
  - Διαγραφή επιλογών
  - Επεξεργασία κλάσεων
  - Αλλαγή CSS





# Αλλαγή css (1/2)

---

- Γίνεται με χρήση της μεθόδου `css()` με 3 τρόπους
  - Εύρεση τιμής CSS για κάποιο element με χρήση του ονόματος της ιδιότητας
    - ▣ Π.χ. `var fontSize= $('body').css('font-size');`
  - Ορισμός μιας ιδιότητας CSS για κάποιο element με χρήση 2 ορισμάτων (ιδιότητα, τιμή)
    - ▣ Π.χ. `$('body').css('font-size', baseFont*2);`
    - ▣ Η τιμή μπορεί να είναι string ή αριθμός



# Αλλαγή css (2/2)

---

- Γίνεται με χρήση της μεθόδου `css()` με 3 τρόπους (συν)
  - Ορισμός πολλαπλών ιδιοτήτων CSS
    - ▣ Χρήση μεθοδολογίας για 1 ιδιότητα πολλές φορές
      - `$('#test').css('background-color','green');`
      - `$('#test').css('border','2px solid black');`
    - ▣ Χρήση object literal στη μέθοδο `css()` με σύνταξη: ιδιότητα:τιμή, ιδιότητα:τιμή
      - Π.χ. `{ 'background-color' : 'green', 'border' : '2px solid black' }`