# ΔΙΑΔΙΚΤΥΟ ΚΑΙ ΕΦΑΡΜΟΓΕΣ Εαρινό εξάμηνο 2019

**Άσκηση 1η :  Δυναμικές Σελίδες**

**Eυρίσκεται και στο site του μαθήματος** https://courses.cn.ntua.gr/

**Προθεσμία παράδοσης σε γραπτή μορφή Παρασκευή 22.03.19**

Να παραδοθούν σε έντυπη μορφή τα σχετικά προγράμματα και τα αποτελέσματά τους. Ίδιες απαντήσεις από διαφορετικούς σπουδαστές δεν θα λαμβάνονται υπ’ όψιν.

**Επίδειξη στο Εργαστήριο οποιαδήποτε Πέμπτη η Παρασκευή (12.45 - 14.30) μετά την γραπτή παράδοση**

Το παράδειγμα των σημειώσεων CookieServlet (όπως και το SessionServlet) είναι αρκετά πρωτόγονο:

1. Οι φόρμες DestinationSelection.html, CitiesRecommendation.html

είναι προκατασκευασμένες και δεν κατεβαίνουν από τον server.

1. Εξυπηρετούμε με ένα servlet (CookieServlet) δύο φόρμες. Για την εξυπηρέτηση της καθε μίας χρησιμοποιούμε τις δύο δυνατότητες αποστολής GET και POST, και ο κώδικας που κατασκευάζει την απόκριση έρχεται αντίστοιχα στις μεθόδους doGet και doPost. Τι θα κάναμε με περισσότερες φόρμες;
2. Το ονόμα του cookie προέρχεται από την επιλογή του χρήστη (ανορθόδοξο). Συνήθως το όνομα είναι σταθερό, αντιστοιχεί σε ένα control της φόρμας, όπου κάτι επιλέγει ο χρήστης του cookie (<input> tag, βλ. http://www.w3schools.com/html/html\_forms.asp) και η τιμή του cookie είναι η επιλογή.

(μερικές) **Απαντήσεις:**

**1.** Η πρώτη φόρμα είναι π.χ. η index.html, κάθε επόμενη κατασκευάζεται (μέρος του κώδικα απόκρισης στην προηγούμενη) από το servlet με διαδοχικά output.println(html\_code\_as\_string), κλπ.

**2.** **Με ένα και μόνο servlet**

Θα χρησιμοποιήσομε κάποιο σημάδι πάνω σε κάθε φόρμα, ώστε αυτή να ξεχωρίζεται. Το καλύτερο είναι ένα αόρατο τέτοιο σημάδι, δηλ. ένα <input> tag που είναι κρυμένο και ψεύτικο. Δεν παρουσιάζεται (type="hidden"), του δύνομε ένα δικό μας όνομα (π.χ. form\_ID) και κάθε φόρμα έχει εκ κατασκευής την δική της τιμή (π.χ. form\_1). Δηλ.

<input type="hidden" name="form\_ID" value="form\_1" />

Ο κώδικας (μία doGet ή μια doPost για όλες τις φόρμες) δεν έχει παρά να διαχωρίζει, δηλ.

if (request.getParameter("form\_ID")=="form\_1") {what to do with form 1 request arrival};

σελ. 1/3

if (request.getParameter("form\_ID")=="form\_2") {what to do with form 2 request arrival}; //etc

Αν η αλληλουχία των διαδοχικών φορμών είναι σταθερή και γνωστή θα μπορούσαμε να έχομε μέσα στο servlet μία τοπική μεταβλητή (private int i=1;) που αυτή θα αυξανόταν σε κάθε νέα άφιξη request και στο τέλος θα γύριζε στο 1. Μας καλύπτει αυτό;

**2. Με ένα servlet για κάθε φόρμα**

Είναι κομψότερο για πιό πολύπλοκες περιπτώσεις. Πώς μας ρίχνει ο κώδικας του ενός servlet στον κώδικα του επομένου;

**3**. **Προφανές με την εξής βοήθεια**. Η **[getCookies](http://java.sun.com/j2ee/1.4/docs/api/javax/servlet/http/HttpServletRequest.html" \l "getCookies())**() επιστρέφει το [Cookie](http://java.sun.com/j2ee/1.4/docs/api/javax/servlet/http/Cookie.html)[] που είναι όμως array της class Cookie (βρείτε την στο

<http://docs.oracle.com/javaee/1.4/api/index.html>

μέσα στο package javax.servlet.http). Θέλομε να βρούμε εκεί μέσα ένα Cookie με το όνομά του. Αυτό δεν γίνεται παρά να διατρέξομε όλα τα στοιχεία του (βλ. κώδικα στοις σημειώσεις ρωτώντας στο καθένα άν έχει το επιθυμητό όνομα

###### Στην επόμενη σελίδα επαναλαμβάνονται οι αλληλοσυνδέσεις (όπως στις σημειώσεις)

###### 

###### ΕΚΦΩΝΗΣΗ

Μελτήστε τα παραπάνω. Επιλέξτε μία περιοχή ενδιαφέροντός σας (π.χ. αυτοκίνητα, τρόφιμα, διακοπές, κλπ) και ένα σενάριο, που εμπλέκει τουλάχιστον τέσσερεις φὀρμες. Όταν επισκέπτεται το site, ο χρήστης λαμβάνει την πρώτη –στατική- φόρμα, η οποία παρουσιάζει το θέμα (π.χ. προϊόντα) περιέχοντας μερικές επιλογές. Κάθε επόμενη φόρμα πρέπει να εμφανίζει ή να επηρεάζεται από στοιχεία που δόθηκαν σε ΟΛΕΣ τις προηγούμενες, αρα απαιτούνται cookies και διάβασμα των στοιχείων που έδωσε ο χρήστης στην αμέσως προηγούμενη φόρμα. Επίσης η ακολουθία των φορμών ΔΕΝ πρέπει να είναι πάντα η ίδια καί δυνατόν ανα αλλάζει από την επιλογή του χρήστη. Π.χ. σε μία τουριστική εφαρμογή όπου στην πρώτη φόρμα καλείται ο χρήστης να επιλέξει την μεταφορα και την διανυκτέρευση, αφήνομε την δυνατότητα να επιλέξει σαν επόμενο στάδιο (φόρμα) είτε το ένα είτε το άλλο. Επίσης κ**άθε cookie πρέπει να έχει το δικό του όνομα, που να μήν εξαρτάται από την επιλογή του χρήστη, όπως εξηγήθηκε παραπάνω**.

**ΠΑΡΑΔΟΤΕΑ**

Η παράδοση σε ‘γραπτή μορφή’ σύμφωνα με την παραπάνω προθεσμία περιλαμβάνει τον κώδικα και τα σχετικά ‘screenshots’ των φορμών. **Μετά** την παραπάνω παράδοση, θα πρέπει να επιδειχθεί ο κώδικας στο Εργαστήριο. Πρέπει να έχετε διαθέσιμο (π.χ. να κατεβάσετε με mail ή από memory stick) και να στήσετε το σχετικό αρχείο .war στο εργαστήριο. Θα σας ζητηθούν μικροαλλαγές σε αυτό που θα δείξετε, πράγμα που είναι ανθρωπίνως δυνατόν μόνον από αυτόν που πραγματικά έχει ασχοληθεί με την άσκηση αυτή.

σελ. 2/3



σελ. 3/3