Assignment 3

Due: April 6, 2014 (Sunday) 23:59

1. Objective:

Assignment 3 will be a comprehensive exercise which further enhances your knowledge and skills to HTML5, CSS3, and JavaScript. It is based on assignment 2, assignment1, labs, and applied with client-side validation and dynamic display using JavaScript.

2. Introduction:

You are hired by a pizza store to create a webpage for online pizza order. The page will be mainly a form with some styling by CSS3. You need to create a JavaScript validation program to validate user's input for the form based on the requirements below.

3. Content Requirements:

1) Create an HTML5 document (`a3.html`) and place it under the directory "public_html/assign3" (you need to create this directory first).
2) Using JavaScript, display and highlight "This page was last modified on (the actual date you work on this page)" on top or bottom of your a3.html.
3) On your homepage (`public_html/index.html`), create a link (at any format, text link, image link, or a button, easy to spot) to `a3.html`.
4) In `a3.html`, create a form (name: `pizza`) to accept user's input. The specifications and validation rules for each field are in section 3.
5) All JavaScript validation code must be in external file, name it "`myscript.js`". This file is placed under directory "assign3"
6) Recommend to use external CSS for the page. You are allowed to use internal or inline CSS while you feel proper. The external CSS file is placed under directory "assign3"
7) When the client submits the form, the JavaScript must validate the information entered
    - If there are any errors,
        i. Popup a window to show the errors and don't submit the form. Focus the cursor to the field with errors to allow the user to correct the errors.
        ii. Error messages should be meaningful and reflect the exact error condition.
        iii. Make sure to return false
    - If there are no errors,
        i. Change the 1st character in the client Surname (field01) to upper case and the rest of the characters in field01 to lower case.
        ii. Change the initial value in field 13 from N to Y.
        iii. Recalculate the price of the pizza based on what was ordered
        iv. Enable the price – field07 – otherwise the price will not be submitted.

            ```
            document.pizza.field07.disabled = false;
            ```

        v. Make sure to return true.

4. Validation requirements

4.1 General information

- The name of the form is 'pizza',
- Use the field names as described below in the table
- the method to be used for this form is the 'post' method
- the action is:
- action = "http://formpost.azurewebsites.net/home/test"

It's recommended that you review the field validation rules and then implement each validation rule in a separate JavaScript function. Doing so will allow you to resolve any potential JavaScript problems easier.

All students will be using the same validation rules; however, it's very unlikely that two students will come up with the same JavaScript code. Make sure you don't share your design / JavaScript with other students.

4.2 Validation Rules:

<table>
<tr><th>Form Field name</th><th>Validation / Comments</th></tr>
<tr><td>Field01 - Client Surname</td><td>Must be present
Allowable characters
- alphabetic characters lower case (a-z) upper case (A-Z)
- apostrophe (‘)
- hyphen (-)
- Must have at least 1 alphabetic characters (a-z) (A-Z)
when there are no errors, follow the rules in sub-section 3.(6) (“If there are no errors”)</td></tr>
<tr><td>field02 - Client Number</td><td>12 positions

all 12 positions must be present.

One alphabet followed by five (5) digits followed by a hyphen (-) followed by five (5) digits
<table>
<tr><th colspan="12">12 Positions</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td></tr>
<tr><th colspan="12">Format example</th></tr>
<tr><td>A</td><td>N</td><td>N</td><td>N</td><td>N</td><td>N</td><td>-</td><td>N</td><td>N</td><td>N</td><td>N</td><td>N</td></tr>
<tr><th colspan="12">Validation rules</th></tr>
<tr><td>1</td><td colspan="5">2</td><td>3</td><td colspan="4">4</td></tr>
</table>

The User ID (field02) has five (5) rules that must be met before the ID is considered to be valid.

Error message(s) should reflect the rule(s) that are not adhered to.</td></tr>
</table>

	Rule #1 - Must be present and must have a length of 12 **Rule #2 - deals with position 1** **Position 1 must have one of the following alphabets:** **Upper case T or G** **Rule #3 - deals with position 7** **Position 7 must have a hyphen (-).** **Rule #4 - deals with position 2,3,4,5,6** **must be numeric (digits only).** **Rule #5 - deals with positions 8,9,10,11,12** **must be numeric (digits only).**
field03 - Telephone	**12 positions** **all 12 positions must be present.** **Must be in the following format nnn-nnn-nnnn** **n in field03 - Telephone No. - nnn-nnn-nnn must be numeric (digits only)** **The area code nnn-nnn-nnnn can't be all zeros** **The actual phone number nnn-nnn-nnnn can't be all zeros**
field02 - field03 relationship	**This validation should only be done if both, field02 and field03 are valid.** **If position 1 in the client number is the upper case letter T, then the telephone number area code must be 416** **If position 1 in the client number is the upper case letter G, then the telephone number area code must be 905**
field04 - Date of Birth	**7 positions** **all 7 positions must be present.** **Must be in the following format mmmyyyy** **The three letter month mmmyyyy must be a valid three letter abbreviation for a month** **The three letter month mmmyyyy can be upper or lower case or a mix of upper and lower case** **The year mmmyyyy must numeric and must be at least 19 years less than the current year（customer must be at least 19 years old）** **Valid three letter abbreviation for the different months are:** **lower case – jan feb mar apr may jun jul aug sep oct nov dec** **upper case – JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC** **or a mix of lower and upper case**

field05 - Size	**The client must select one of these menu options (small, medium, large, X-large)** **Note: Changes to field05 (size) will require the price to be recalculated** **Note: Remember that there is no default for this entry**
field06 - Crust	**menu** **default = Regular Crust** **No validation required**
field07 - Price	**text type** **Default: disabled** **The price is recalculated and updated on the order form any time the client:** • **select or change a pizza size** • **clicks on the reset button** **The function name for recalculating the price: calculatePrice()** **To format numbers for currency - Use the toFixed(n) method** **Example ----> varName = varName.toFixed(2)** **Rounds varName to the specified number of decimal places**
field08 - Cheeses	**radio type** **default = Mozzarella Cheese** **No validation required**
field09 - Sauces	**radio type** **no default** **Rule: The client/user must click on one of the radio buttons**
Field10	**Toppings** **- checkbox options** **- 6 meaningful toppings** **- no default** **- make sure to include the topping name as a value** **Rule: The client/user must select one topping**
field12 - Instructions	**textarea** **free format - No validation required**
field13 - hidden	**hidden** **when there are no errors, follow the rules in sub-section 3.6 ("If there are no errors").**
field14 - hidden	**Hidden** **No validation required - value = your surname followed by your first name**
field15 - hidden	**Hidden** **No validation required - value = your zenit login account**

5. Evaluation

The evaluation for this assignment will be based on adherence to all of the requirements outlined above including:

- adherence to the coding standards
- quality of the JavaScript technical features/usage/function (syntax, readability, documentation and indentation)
- the content
- the overall appearance, organization and presentation.

6. Submission

Leave your work on zenit account after completion, the instructor will check your work after due date. No explicit submission. No email or printout.

Work on zenit and leave your work untouched after completion/ submission. You need to realize that every modification to the page (while "saving") is changing the date stamp of your work.

Late submissions will be penalized a 10% for each day to a maximum of 5 days. Submission after 5 days will receive the maximum of 50% of the assignment.