

Exercise02: Javascript

1 Objectives

To learn to use Javascript and play with basics

- simple variables (local and global), conditions, loops, methods, and arrays.
- User input validations.
- JS cookies
- Event Handling

Work with your group (or by yourself). Each group should only upload one submission.

1. Warm Up: Try Some Examples









1. First, open blackboard, go to Course Contents, and then download exercise02.zip file into your workspace (U:\workspace or something like that!). Then, unzip.
2. Play with each of the given examples (in the examples directory). Open them using a text editor of your choice and modify parts of the html or js files to learn how the different instructions work. If you want to use eclipse instead of notepad or vim or emacs etc., create a new static web project and create new html file and open it with a browser.
3. Note: w3schools.com is a good site to learn about web technologies.

Note that the assignment assumes you have understood these examples.



2 Form Validation

2.1 Create a form in HTML and validate entries of the form using javascript.

1. Create two files **validation1.html**
2. Change the TITLE of the validation1.html page to "Validation Form".
3. Create a HTML form in validation.html containing the fields as in the table below and also a continue button. Make it look reasonably good. Validation will be in next step.







Field Label	Field Type	Validation rule	Result
First Name	TextField	*Required. Must contain only alphabetical or numeric characters.	 / 
Last Name	TextField	*Required. Must contain only alphabetical or numeric characters.	 / 
Gender	Dropdown(male,female)	*Required.	 / 
State	Dropdown(California,Florida, New York,Texas,Hawaii,Washingt on,Colorado,Virginia, Iowa,Arizona)	*Required, select from all given list and save the selected one to JSCookies	 / 

***Required field = Cannot be Empty.**



4. Create a file named **validation1.js**. Remember to include validation1.js in the head section of validation.html
5. Write Javascript code in validation1.js so that when user clicks **continue button** it does the following:
 - a) It validates the entries and displays  image (for each entry) if the validation was successful, else it displays  image. These images are included in the lab's zip file (correct.png and wrong.png).
 - b) Once the validation is successful, it uses [JS cookies](#) (click on link to find out more) to store the entered data.
 - c) next, go to the next page (i.e. validation2.html)

2.2 Write another html file to enter Contact information Form

1. Create two files **validation2.html** and **validation2.js**
2. Change the TITLE of the validation2.html page to "Contact information".
3. Create a HTML form in validation2.html with the following Fields and a submit button. Make it look reasonably good.

Field Label	Field Type	Validation rule	Result
Email	TextField	*Required . Must be in the form <u>xxx@xxx.xxx</u> x should be alphanumeric (e.g. no special symbols).	 / 
Phone	TextField	Must be in the form xxx-xxx-xxxx or xxxxxxxxxx x should be numeric	 / 
address	TextField	*Required . Must have city, &state. example: Ames, IA	 / 

***Required field = Cannot be Empty.**

4. Write Javascript code in validation2.js to validate the above form when user clicks Submit button. Your code should display  image if the validation was successful, or if there was an error, display  image. Remember to include validation.js in the head section of validation.html
5. When user clicks on submit:
 - a) if all field are valid, use HTML [local storage](#) (click on link to find out more) to store the address only. Make sure to use text/string arguments when using localStorage.setItem.
 - b) delete the JS cookies that were created previously.
 - c) go to a new page.
6. On the new page, retrieve local storage to show it on google map. The following link ["markers" style](#) (click on link to find out more) has information on how to use Google map API to display a map with a location indicated by a circular marker.

3 Event Handling

Write a Javascript and HTML code to implement the functionality shown in 'Problem2Output.mp4' included in the .zip file.

Hint : It uses HTML5 Canvas commands (see w3schools.com).

4 Submission:

Make sure your solutions work on Chrome (which is what TAs will use to grade the assignment). Zip your html, js files, and participation file (i.e. who worked on which part or if you worked together). Then, submit this zip file on black board. Remember there is only one submission per group. Make sure to include all the files that are needed in order to run your program.

Participation file is a simple txt file, which clarifies the specific participation of two members.