

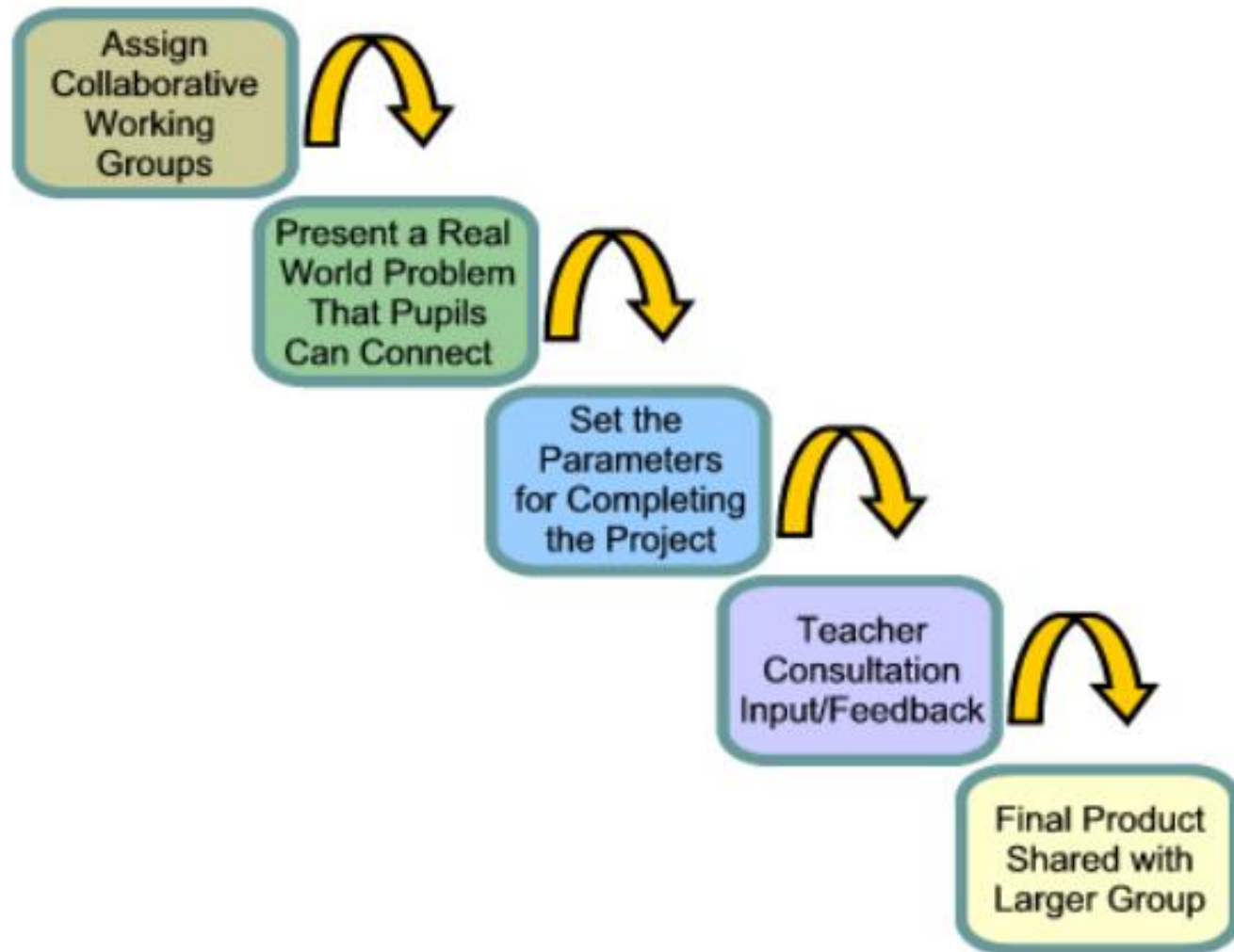


WEB PROGRAMMING 1 - TP

INTRO

- In the course we use *project based learning* techniques; which means that all TP sessions are related to each others.
All contributing in the building of what we call a 'class project'.
- TP sessions are hands-on sessions.
- Each student should work alone on his work station and ask the teacher for consultation and feedback

HOW IT WORKS ?



CONFIGURATION

- Download and install notepad++, dreamweaver, NetBeans, or any IDE you know
- Download latest jquery.js from [jQuery.com](http://jquery.com)
 - place inside scripts folder

```
<head>  
  <script src="myScripts/jquery-1.12.4.min.js"></script>  
</head>
```

- Or use google CDN

```
<head>  
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js">  
  </script>  
</head>
```

CONFIGURATION

- Download and install
 - easyPhp (version 3.1 or higher) or
 - WampServer latest or
 - XAMPP server latest or
 - any application that have an apache server and a MySql database

JQUERY – EXERCICE

```
<body>
  <h1 class="bigText">This is a heading</h1>
  <h2>This is another heading</h2>

  <p class="bigText">This is a paragraph.</p>
  <p>This is another paragraph.</p>

  <div>This is a div element.</div>
  <a href="http://www.google.com/css/">Google</a>

  <button id="btn1">This is a button</button>
</body>
```

This is a heading

This is another heading

This is a paragraph.

This is another paragraph.

This is a div element.

[Google](http://www.google.com/css/)

This is a button

- Write the script that does the following when 'btn1' is clicked:
 1. Hide all p elements
 2. Hide all div elements
 3. Hide all elements that have a class 'bigText'
 4. Hide all links (href)
 5. Hide all elements on the page

JQUERY – EXERCICE

```
<body>
  Name: <input type="text" value="Mickey Mouse">
  <div>
    This is some <b>bold</b> and <i>italic</i> text.
  </div>
  <p>This is some text.</p>
  <a href="http://www.google.com">Google</a>

  <button id="btn1">This is a button</button>
</body>
```

- Which result produces the following operations:
 - \$("input").val();
 - \$("div").html();
 - \$("p").text();
 - \$("a").attr("href");

JQUERY – EXERCICE

○ Create a Zebra Stripes table effect

Student Name	Marks in Science
Janet	85.00
David	92.00
Arthur	79.00
Bill	82.00

On row click :

- Show a popup containing selected student name and grade
- Change row color
 - Green → White
 - White → Green

○ Limit the number of characters to 15 in a text box

Maximum 15 characters

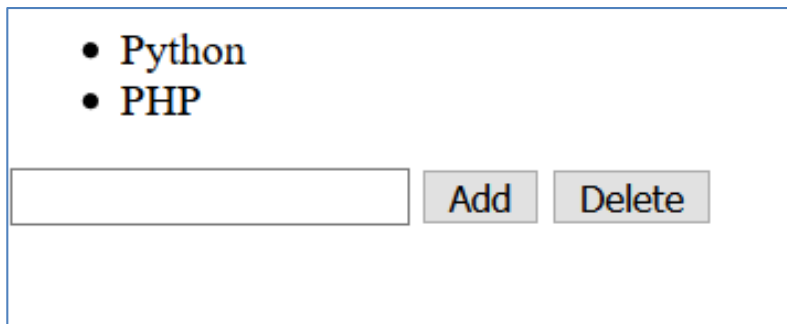
15 Character(s) Remaining

Maximum 15 characters

0 Character(s) Remaining

JQUERY – EXERCICE

- Add an element within an unordered list element on button click.



A screenshot of a web form. It features an unordered list with two items: 'Python' and 'PHP'. Below the list is a text input field, followed by two buttons labeled 'Add' and 'Delete'.

button Add behavior:

1. if textbox is empty ; display an error message
2. if not; add its value to the above list

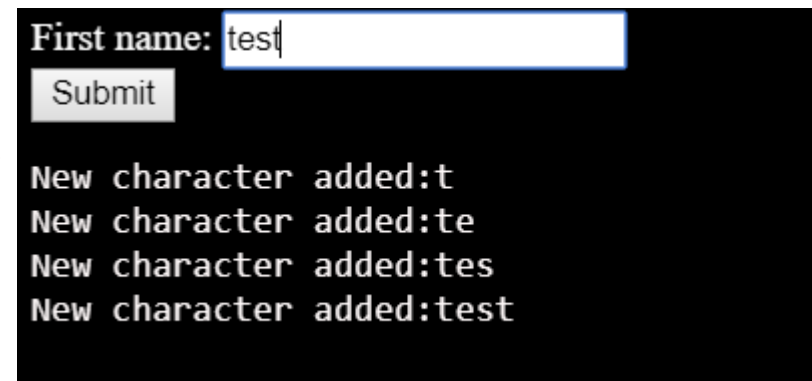
button Delete behavior :

1. if textbox is empty; delete last entry in the list
2. if not; search for value in the list;(value inside textbox)
 1. if found , delete it
 2. else display error message

- Detect the change on a textbox and add a message containing the added values



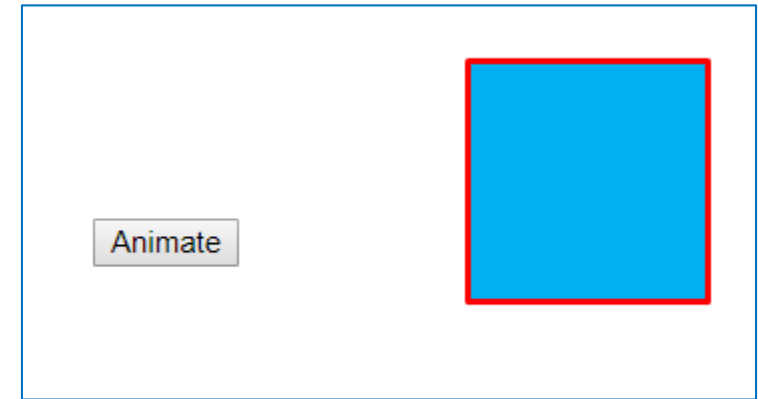
The initial state of the form. It shows a label 'First name:' next to an empty text input field. Below the input field is a 'Submit' button.



The final state of the form after multiple submissions. The 'First name:' label is followed by a text input field containing the text 'test'. Below the input field is a 'Submit' button. Below the button, there is a list of messages: 'New character added:t', 'New character added:te', 'New character added:tes', and 'New character added:test'.

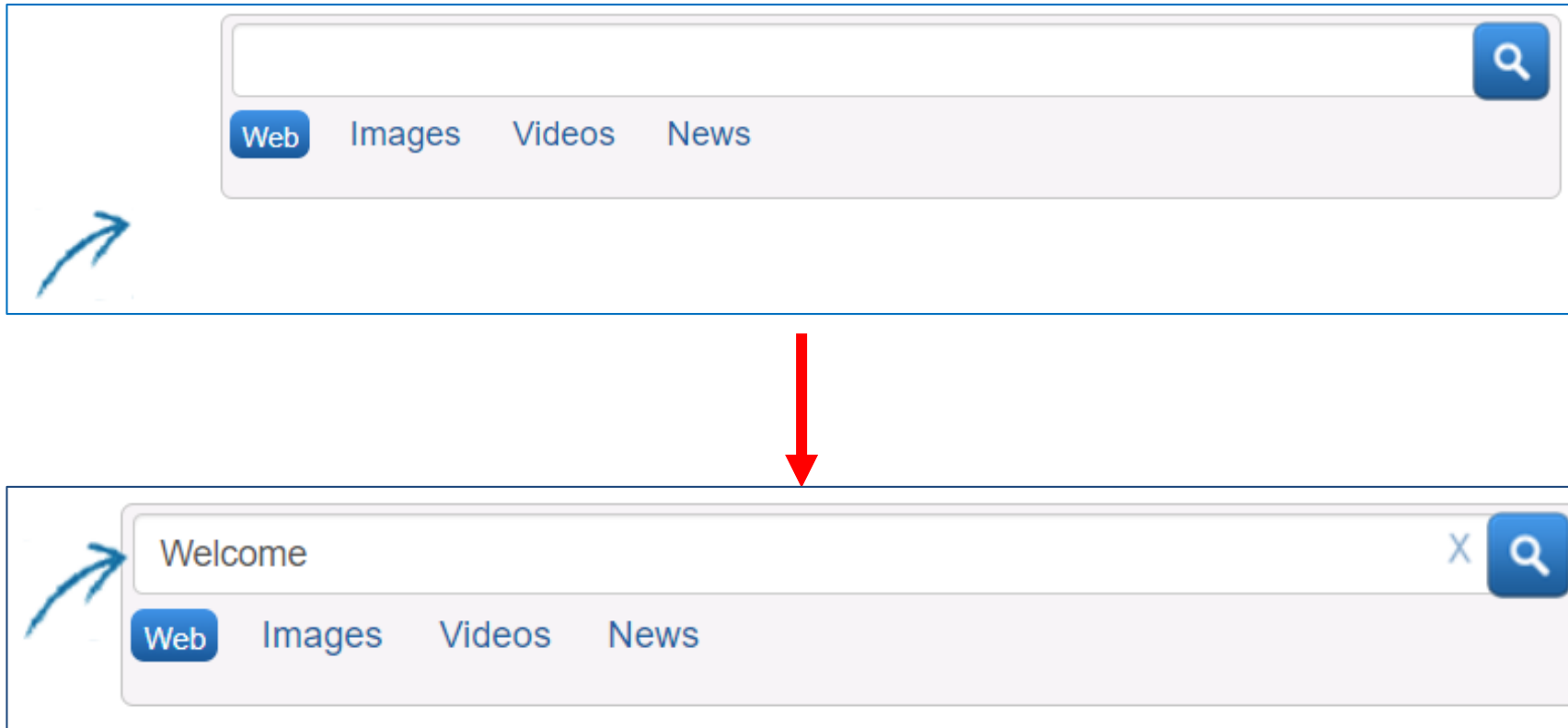
JQUERY – EXERCICE

- On button click :
 - Move square to the right 100px and
 - Change its color to green and
 - Change border color to white
- On right click on the square:
 - Move it up left slowly
 - Change square shape to circle



JQUERY – EXERCICE

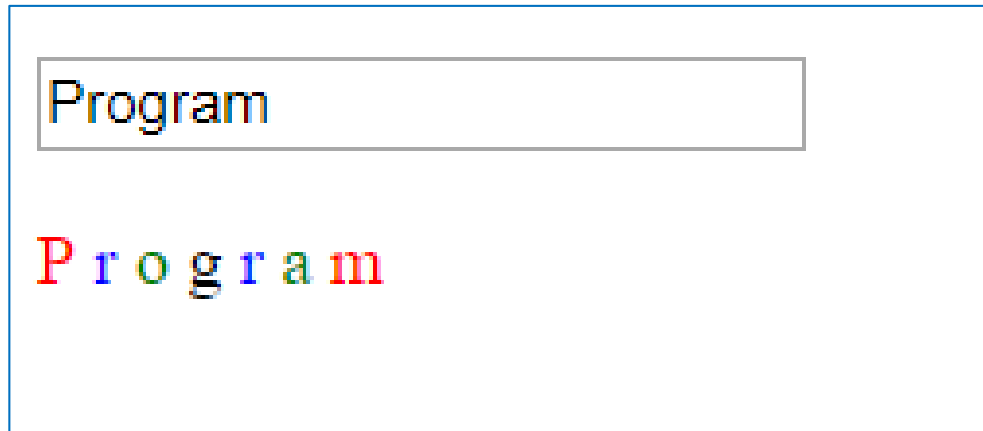
- On page load, move the **blue** arrow in the image below to top right using a fast animation
- Write *Welcome* in the search bar



JQUERY – EXERCICE

In the image below:

1. Each letter typed in the textbox should also appear in the div below it in a random color
 - i.e. when the user types the letter 'P' in the textbox, 'P' will appear in the div below in a random color (here in red)
2. Hit Enter key on the keyboard
 - Show popup containing the value in the textbox minus the letter 'a' and 'A'
3. Hit escape key
 - Empty the display



CLASS PROJECT

The purpose of this section is to have a mini class project that will cover all the needed concepts for this course.

We assume we want to build a small website that is used to sell cars online.

User can log in / signUp, search for cars using different filters, add cars to his wish list and buy them.

Admin should be able to manage all data needed from his back office.

CLASS PROJECT

In order

- download and install any application that have an apache server and a MySql database
 - easyPhp /WampServer /XAMPP server, ...
- move all the TPs we did so far to the root folder (www / htdocs) on the Apache server
- from your browser, open localhost or 127.0.0.1 to browse your old files

SIGN UP — CLIENT SIDE VALIDATION

Implement client-side validation respecting the **rules below**.
If form not valid, a message should popup indicating the reason

Welcome to our site

Last name	<input type="text"/>	<i>not empty</i>
First name	<input type="text"/>	<i>not empty</i>
Gender	<input type="radio"/> Male <input type="radio"/> Female	<i>one selected</i>
Email	<input type="text"/>	<i>valid email format</i>
Password	<input type="password"/>	<i>respect password rules</i>
Confirm Password	<input type="password"/>	<i>equal to password</i>
<input type="button" value="Register"/>		

8+ characters / contains at least 1 special characters and 1 number

SIGN UP — SERVER SIDE VALIDATION

Implement server-side validation respecting the **rules below**.
If form not valid, a response message should be returned to the client. Show a message indicating the reason.

Welcome to our site

Last name	<input type="text"/>	<i>not empty</i>
First name	<input type="text"/>	<i>not empty</i>
Gender	<input type="radio"/> Male <input type="radio"/> Female	<i>one selected</i>
Email	<input type="text"/>	<i>valid email format</i>
Password	<input type="password"/>	<i>respect password rules</i>
Confirm Password	<input type="password"/>	<i>equal to password</i>
<input type="button" value="Register"/>		

8+ characters / contains at least 1 special characters and 1 number

SIGN UP — DATABASE

- Using MySql, create
 - the table 'User' :
 - Id:INT
 - Firstname : VARCHAR
 - Lastname : VARCHAR
 - GenderId : INT
 - Email : VARCHAR
 - Password (Hashed and Encrypted using Salt) : VARCHAR
 - ProfileImageUrl : VARCHAR
 - Salt : VARCHAR
 - TokenVerifyEmail : VARCHAR
 - TokenExpiryDate :DATE
 - IsDeleted (soft delete) :BIT (boolean)
 - the table 'Gender' :
 - Id : INT
 - Description : NVARCHAR
- Prepare the queries needed to add, update and delete a user
- Set connection to the DB from PHP inside the file config.php

SIGN UP — IMPLEMENTATION

Implement the code for sign-up respecting the following :

- Apply 3-tiers architecture
 - Divide the code into the 3 layers (PL – BLL – DAL)
 - Respect the responsibility of each layer
- Use for :
 - Passwords:
 - the salting and peppering algorithm for passwords saved in the DB (encryption and hashing algorithm)
 - Salt : a 5 digit token (letter and numbers) randomly generated
 - ValidationExpiryDate: Today's date plus 7 days
 - IsDeleted : Set to false on user creation

SIGN UP — EMAIL

After a successful sign-up, a welcome email should be sent to the user containing :

- a greeting message
- the user's first name ,last name and gender (not gender id)
- the number of days remaining until account expiry :
 - Today – ValidationExpiryDate

PROFILE — UPDATE

User should be able to navigate to a page where he can update his profile.

He should be able to update his:

- Profile pic (upload only jpg and jpeg image)
- First name
- Last name
- Gender
- Password

From this page, the user can also delete his account.

Note: both client side and server side validations are needed

RESET PASSWORD

From the login page; user should be able to reset his password.

Once user clicks on reset password url, he will be redirected to a page where he enter his email and click ***Reset***.

On click an email should be sent to the user with a reset token (inside a url).

When the user clicks on this url, he is prompted to enter his new password and to confirm it before he can submit.

DASHBOARD — SEARCH

User should be able to navigate to the dashboard page, where he can search for cars using the filters below:

- Brand
 - Drop down list containing the list of brands
 - Filled from the database table **Brand**
 - i.e. Toyota, Nissan, BMW , etc.
- Model
 - Drop down list containing the list of models
 - Filled from the database table **Model** based on the user selection of Brand
 - i.e.
 - if user selected Toyota as brand, the list of Model should be filled with the *Land Cruiser, Rav4, Corolla , etc.*
- Production year
- Price

The results should be displayed as shown in the snapshot :

- Car image
- Description
- Year of production
- Price
- The click on [View details](#) goes to this item's the details page

DASHBOARD — SEARCH

Brand

Toyota

Model

Land cruiser

Production Year

2017

Price


Between

10.000


And

50.000


Search




Description
Year of production
Price
[View details](#)




Toyota , Grey , New
2017
35.0000
[View details](#)




Toyota , White , Used
2017
35.0000
[View details](#)



Description



Description



Description