

**Name:**Lakshmana Kumar Mettu

**Class ID:**11

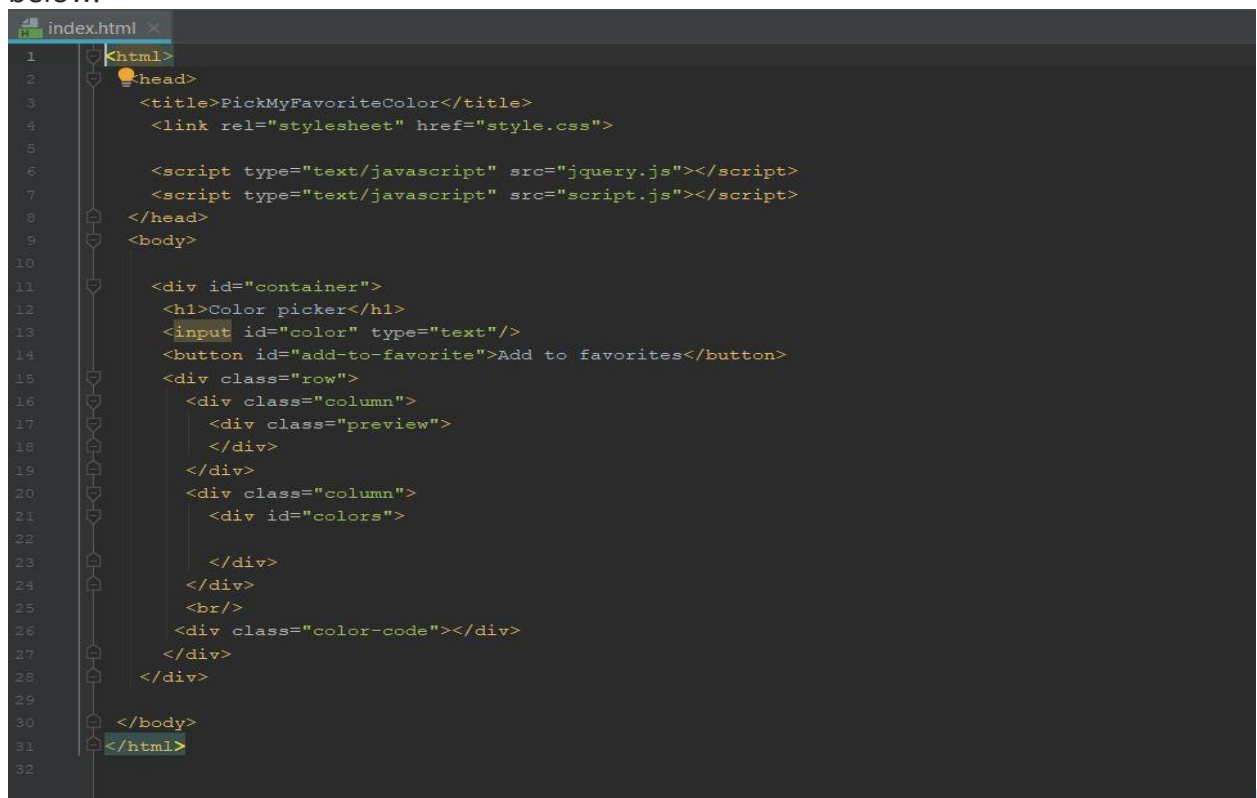
**Introduction:**This ICP is about to learn object oriented JavaScript along with jQuery.

**Programming Elements:** javascript2, jQuery, html5,css

**Tools Required:** JetBrains-webstorm

**Objective-1:**Using JavaScript and jQuery to select a color by typing it in the text box or picking it from a given set of colors.

- I have developed normal html page for this ,shown below.

A screenshot of a code editor window titled 'index.html'. The code is written in HTML and includes jQuery and a custom script. The structure includes a container div with a title 'Color picker', a text input for color selection, a button to add to favorites, a row of two columns. The first column contains a preview div, and the second column contains a div with a list of colors. Below this is a div for the color code.

```
1 <html>
2 <head>
3   <title>PickMyFavoriteColor</title>
4   <link rel="stylesheet" href="style.css">
5
6   <script type="text/javascript" src="jquery.js"></script>
7   <script type="text/javascript" src="script.js"></script>
8 </head>
9 <body>
10
11   <div id="container">
12     <h1>Color picker</h1>
13     <input id="color" type="text"/>
14     <button id="add-to-favorite">Add to favorites</button>
15     <div class="row">
16       <div class="column">
17         <div class="preview">
18         </div>
19       </div>
20       <div class="column">
21         <div id="colors">
22
23         </div>
24       </div>
25     </div>
26     <div class="color-code"></div>
27   </div>
28 </body>
29 </html>
```

- The Javascript page for this one is given below.

- As the page loads add each color in the colors array to the div '#colors'

snippets

```
// an array of colors and assign it to a variable colors
var colors = [ "22ac5e", "d68236", "71b5c2", "af2655", "b34de7", "e6bd01", "104393", "ca4d94", "4a772d", "c180a7", "958112", "8d2f8d" ]

// sets the preview color to the one entered in the input and display its color code using setPreviewColor function
function setPreviewColor(color) {
    $('#preview').css('background-color', color);
    $('#color-code').text($('#preview').css('background-color'));
}

//adds color boxes to the favorite colors
function addBox(color) {
    $('#colors').prepend( 'nodes: "<div class='item' style='background-color:' + color + "><div>";' );
}

$(document).ready(function(){

    //1.As the page loads add each color in the colors array to the div '#colors'
    colors.forEach(addBox)

    //set the preview color to one of the colors in the colors array randomly
    setPreviewColor(colors[Math.floor(Math.random()*colors.length)]);

    // an event handler for the key up event i.e. when the user types the color in the input and releases the key on the keyboard
    //The event should set the preview color to the color typed in the input
    $(document).on('keydown keyup keypress', '#color', function(){
        color = $(this).val();
        setPreviewColor(color);
    })

    //2.Write an event handler to handle the click the event on the add to favorite button so that the color gets added to the list of favorite color
    // the content of the input gets cleared and the focus gets back on the input
    $('#add-to-favorite').on('click', function(){
        color = $('#color').val();
        if (color != "") {
            addBox(color);
        }
    })
});
```

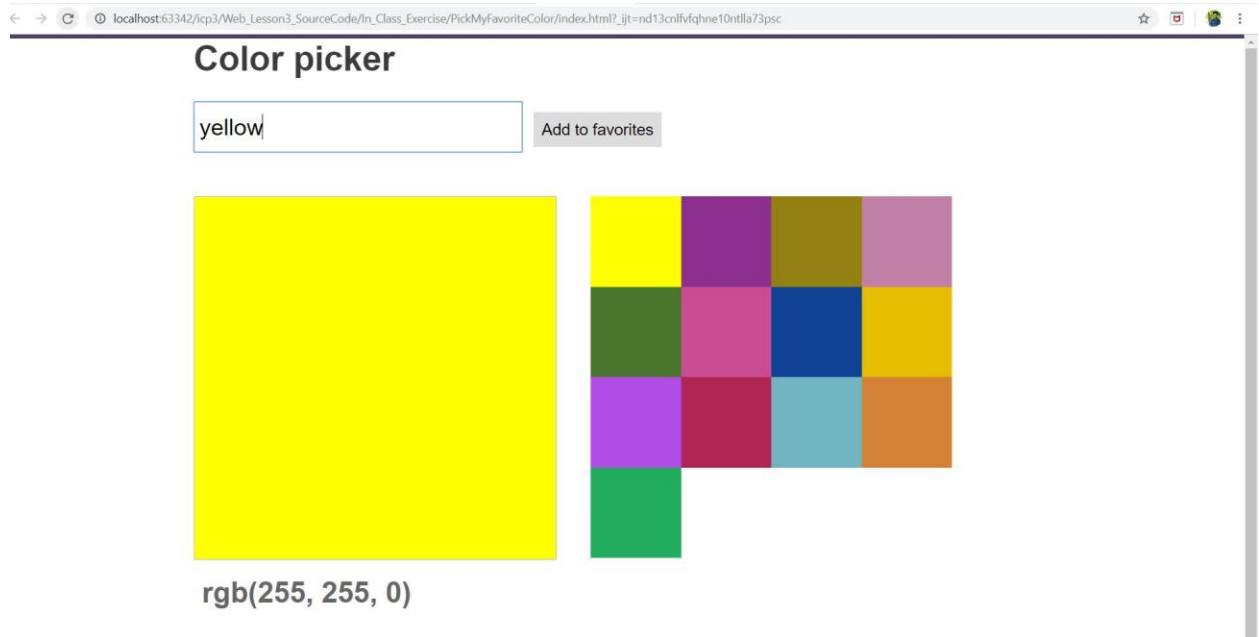
```
//3.Write events handlers such that whenever any item in the favorite colors is clicked or hovered, the color gets displayed in the p
$('#colors').on('click mouseover', '.item', function(){

    color = $(this).css('background-color');
    setPreviewColor(color);
});
```

- The JQuery page to pick favorite color is given below.

```
/*! jQuery v2.0.3 | (c) 2005, 2013 jQuery Foundation, Inc. | jquery.org/license
// @ sourceMappingURL=jquery-2.0.3.min.map
*/
(function(e,undefined){var t,n,r=typeof undefined,i=e.location,o=e.document,s=o.documentElement,a=e.jQuery,u=e.$,l={},c=[],p="2.0.3",f=c.concat,h=c.push;
"inprogress"===i&&(i=n.shift(),i--),i&&("fx"===t&&n.unshift({items:"inprogress"}),delete o.stop,i.call(e,s,o),!r&&o.empty.fire(),_queueHooks:func
},delegate:function(e,t,n,r){return this.on(t,e,n,r)},undelegate:function(e,t,n){return 1===arguments.length?this.off(e,"*"):this.off(t,e||"*",n)}});
```

- Output image is given below.



**Objective-2: To retrieve details of a GitHub user like the name, ID, profile picture and link to the user's account.**

- The HTML page for this one is given below.

```

index.html
1  <html>
2  <head>
3      <title>Github User Finder</title>
4      <link rel="stylesheet" href="style.css">
5      <script type="text/javascript" src="jquery.js"></script>
6      <script type="text/javascript" src="script.js"></script>
7  </head>
8  <body>
9      <header>
10         <h1>Find Me!</h1>
11     </header>
12     <div id="container">
13         <div class="search">
14             <input id="username" placeholder="enter a github username..." type="text"></input>
15         </div>
16         <div id="profile">
17             <h2></h2>
18             <div class="username"></div>
19             <div class="userid"></div>
20             <div class="avatar"></div>
21             <div class="information"></div>
22         </div>
23     </div>
24 </body>
25 </html>

```

- The javascript page along with codesnippets and comments are shown below.

```

function getGithubInfo(user) {
    //1. Create an instance of XMLHttpRequest class and send a GET request using it. The function should finally return the object(it now contains the response)
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
        if (this.readyState == 4) {
            if (this.status == 200) {
                var response = JSON.parse(this.responseText);
                $('#username').html('Username: ' + response['login']);
                $('#userid').html('ID: ' + response['id']);
                $('#avatar').html('<img src=' + response['avatar_url'] + ' alt="img">');
                $('#information').html('<a href=' + response['html_url'] + '>Link to profile</a>');
            }
            else {
                $('#username').html('Invalid User');
            }
        }
    };

    xhttp.open( method: "GET", url: "https://api.github.com/users/"+user, async: false);
    xhttp.send();
}

function showUser(user) {
    //2. set the contents of the h2 and the two div elements in the div '#profile' with the user content
}

function noSuchUser(username) {
    //3. set the elements such that a suitable message is displayed
}

```

```

<script type="text/javascript">
var $ = jQuery;
jQuery(document).ready(function($) {
    $(document).on('keypress', '#username', function(e) {
        //check if the enter(i.e return) key is pressed
        if (e.which == 13) {
            //get what the user enters
            username = $(this).val();

            //reset the text typed in the input
            $(this).val("");

            //get the user's information and store the response
            response = getGithubInfo(username);

            //if the response is successful show the user's details
            if (response.status == 200) {
                showUser(JSON.parse(response.responseText));
            }
            //else display suitable message
            else {
                noSuchUser(username);
            }
        }
    });
});
</script>

```

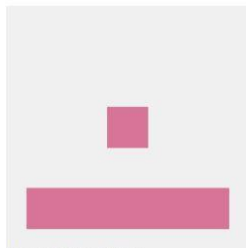
- The JQuery page is given below.

```
(function(e,undefined){var t,n,r=typeof undefined,i=e.location,o=e.document,s=o.documentElement,a=e.jQuery,u=e.$,l={},c=[],p="2.0.3",f=c.concat,h=c.push;
});"inprogress"===i&&(i=n.shift(),r--),i&&("fx"===t&&n.unshift({item:"inprogress"}),delete o.stop,i.call(e,s,o)),l&&o&&o.empty.fire(),_queueHooks:func
},delegate:function(e,t,n,r){return this.on(t,e,n,r)},undelegate:function(e,t,n){return l===arguments.length?this.off(e,"*"):this.off(t,e||"*",n)}})
```

- Output:

## Find Me!

lakshmanamettu



### User Information

Login Name : null  
 Login ID : 42656807  
 GitHub URL : Git Hub URL  
 GitHub Public Repos Of the User : 7

**Obejective-3:Using the To-do List use case and add a functionality of actions in form of the drop down and also add delete the item from list.**

- The html page for this given below.

```
index.html
<html>
<head>
  <title>ToDo List</title>
  <link rel="stylesheet" href="style.css">
  <script type="text/javascript" src="jquery.js"></script>
  <script type="text/javascript" src="script.js"></script>
</head>
<body>

  <div id="container">
    <h1>Get it done!</h1>
    <ol id="items">

    </ol>
    <span class="total"></span>
    <br/>
    <input type="text" id="item"/>
    <button id="add-to-list">Add to list</button>
  </div>

</body>
</html>
```

- Adding drop-down list and delete functionalities using java script are given along with comments.

```
function addToList(item) {
  $('#items').append("<li>" + item + "<span class='label actionlist'><select id='actions'><option value='pending'>Pending</option><option value='su
");
}

function updateTotal() {
  completed = $('#.success').length;
  pending = $('#.pending').length;
  deleted = $('#.removed').length;

  if (completed > 0 || pending > 0 || deleted > 0) {
    $('#total').text(" Pending: " + pending + " Completed: " + completed + " Deleted: " + deleted);
  }
}

$(document).ready(function() {
  wishes.forEach( callbackfn: function(element) {
    addToList(element);
  });
  updateTotal();

  $(document).on('click', '#add-to-list', function() {
    item = $("#item").val();

    $("#item").val(""); /* clear value */
    $("#item").focus();

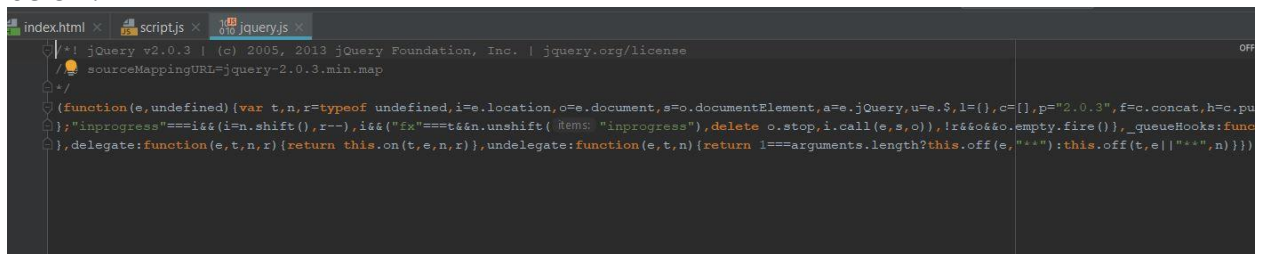
    addToList(item);
    updateTotal();
  });
});
```

```

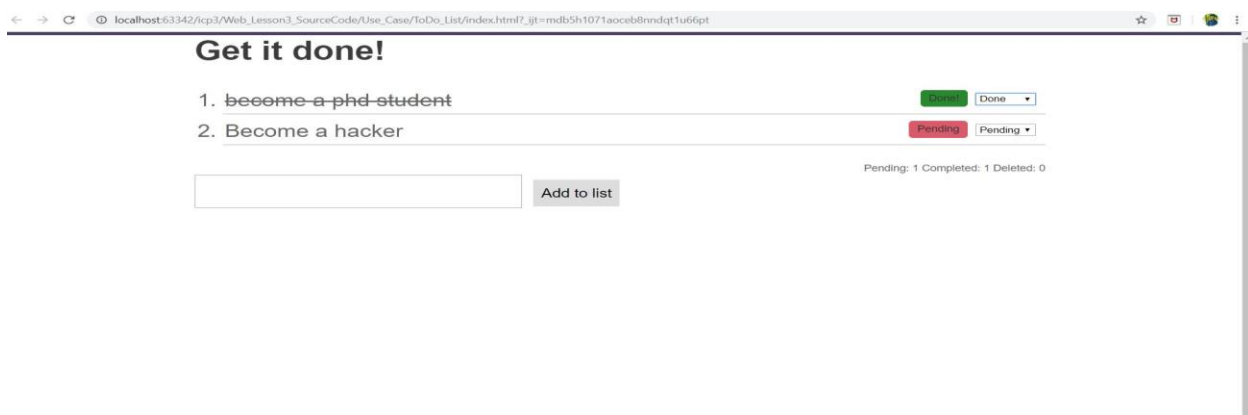
$(function() {
  $(document).on('click', '#actions', function(event) {
    var action = this.options[this.selectedIndex].value;
    if( action == "delete" ) {
      $(this).parent().siblings('.removed').remove();
      $(this).parent().parent().append("<span class='label removed'>Removed</span>");
      $(this).parent().parent().attr("class", 'removed');
      $(this).parent().siblings('.pending').remove();
      $(this).parent().siblings('.success').remove();
      $(this).parent().siblings('.completed').remove();
      $(this).parent().parent().remove();
      updateTotal();
    }
    if( action == "success" ) {
      $(this).parent().siblings('.success').remove();
      $(this).parent().parent().append("<span class='label success'>Done!</span>");
      $(this).parent().parent().attr("class", 'completed');
      $(this).parent().siblings('.pending').remove();
      $(this).parent().siblings('.removed').remove();
      updateTotal();
    }
    if( action == "pending" ) {
      $(this).parent().siblings('.pending').remove();
      $(this).parent().parent().append("<span class='label pending'>Pending</span>");
      $(this).parent().siblings('.success').remove();
      $(this).parent().siblings('.removed').remove();
      $(this).parent().siblings('.completed').remove();
      updateTotal();
    }
  });
});

```

- The JQuery page for this task is given below.



- Output:





## Objective-4: Using Hangman use case to provide hint to the user in form of picture.

- The html page for hangman.js is given here.

```
<html>
<head>
<link rel="stylesheet" href="hangman.css" />
<link href='http://fonts.googleapis.com/css?family=Arvo' rel='stylesheet' type='text/css'>
<script type="text/javascript" src="hangman.js"></script>
</head>
<body>
<div id="hangm">

<div style="height: 2em; margin-bottom: 3em;">
<div id="hangm_end" class="h"></div>
</div>

<div id="hangm_animation">
<div id="hangm_2" class="h pole_h"></div>
<div id="hangm_1" class="h pole_v"></div>

<div id="hangm_3" class="h rope"></div>

<div id="hangm_4" class="h man_head"></div>
<div id="hangm_5" class="h man_upper"></div>
<div id="hangm_6" class="h man_lower"></div>
</div>

<div style="height: 2em;">
<div id="hangm_guesses" class="h"></div>
</div>

<form>
<div class="row">
<div class="column col-sm-3">
<button type="button" id="btn-start" class="btn-lg btn-info" onclick="Hangman.reset(); return false;" >New Game</button>
</div>
<div id="hangm_guessbox" style="height: 8em;" class="h">
<div style="height: 4em;">
Guess the word:<div id="hangm_word"></div>
</div>
</div>
</form>
```

```
<form>
<div class="row">
<div class="column col-sm-3">
<button type="button" id="btn-start" class="btn-lg btn-info" onclick="Hangman.reset(); return false;" >New Game</button>
</div>
<div id="hangm_guessbox" style="height: 8em;" class="h">
<div style="height: 4em;">
Guess the word:<div id="hangm_word"></div>
</div>
<input id="guess" onkeyup="Hangman.guess(this.value); this.value=''; return false;" type="text" placeholder="">
</div>

<div class="column hint col-sm-3">
<button type="button" onclick="Hangman.hint();" >Hint!</button>
<div id="hinttext"></div>
<div id="hintimage"></div>
</div>
</div>
</form>

</div>

</body>
</html>
```



- Javascript page to provide hint in the form of picture for hangman.js are shown below along with comments

```

var Hangman = (function () {

    'use strict';

    function Hangman(elId) {

        // Dom is ready
        this.elId = elId;
        this.words = ['SPARKLING', 'RAINBOW', 'CODER', 'WATERFALL'];
        this.hints = {'SPARKLING': ['glistening, glittering, charming'], 'RAINBOW': ['full colors'], 'CODER': ['techie', 'ethical hacker', 'programmer'], 'WATERFALL': ['nature', 'scenic', 'peaceful']};
        this.hintimages = {'SPARKLING': './charming.jpg', 'RAINBOW': './rainbow.jpg', 'CODER': './coder.jpg', 'WATERFALL': './waterfall.jpg'};

    }

    Hangman.prototype.reset = function () {

        // Reset variables
        this.STOPPED = false;
        this.MISTAKES = 0;
        this.GUESSES = [];
        this.WORD = this.words[Math.floor(Math.random() * this.words.length)];

        // Reset Elements
        this.hideElementByClass('h');
        this.showElementByIdWithContent( elId: this.elId + "_guessbox", content: null);
        this.showElementByIdWithContent( elId: this.elId + "_word", this.getGuessedfWord());

    };

    Hangman.prototype.hint = function() {
        this.HINTS = this.hints[this.WORD];
        document.getElementById( elementId: 'hinttext').innerHTML = "<p>" + this.HINTS[0] + "</p>";
        this.HINTIMAGES = this.hintimages[this.WORD];
        document.getElementById( elementId: 'hintimage').innerHTML = "<img src=\"\" + this.HINTIMAGES + \"\" width='100px' height='100px'></img>";
    };
}

```

```

// Add the letter to array GUESSES
this.GUESSES.push(guess);
// Update the word hint
this.showElementByIdWithContent( elId: this.elId + "_word", this.getGuessedfWord());
// Update the guessed letter list
this.showElementByIdWithContent( elId: this.elId + "_guesses", this.GUESSES.join(''));

if (this.WORD.indexOf(guess) < 0) {

    // Incorrect guess
    this.MISTAKES++;

    // Show next part of hangman character
    this.showElementByIdWithContent( elId: this.elId + "_" + this.MISTAKES, content: null);

    if (this.MISTAKES === 6) {
        // Game Over
        this.showElementByIdWithContent( elId: this.elId + "_end", content: "GAME OVER!<br/>The word was: " + this.WORD);
        this.STOPPED = true;
        return;
    }

} else if (this.WORD.indexOf(this.getGuessedfWord()) !== -1) {
    // Victory
    this.showElementByIdWithContent( elId: this.elId + "_end", content: "You made it!<br/>The word was: " + this.WORD);
    this.STOPPED = true;
    return;
}

};

Hangman.prototype.showElementByIdWithContent = function (elId, content) {
    if (content !== null) {
        document.getElementById(elId).innerHTML = content;
    }
}

```

```

Hangman.prototype.showElementByIdWithContent = function (elId, content) {
    if (content !== null) {
        document.getElementById(elId).innerHTML = content;
    }
    document.getElementById(elId).style.opacity = 1;
};

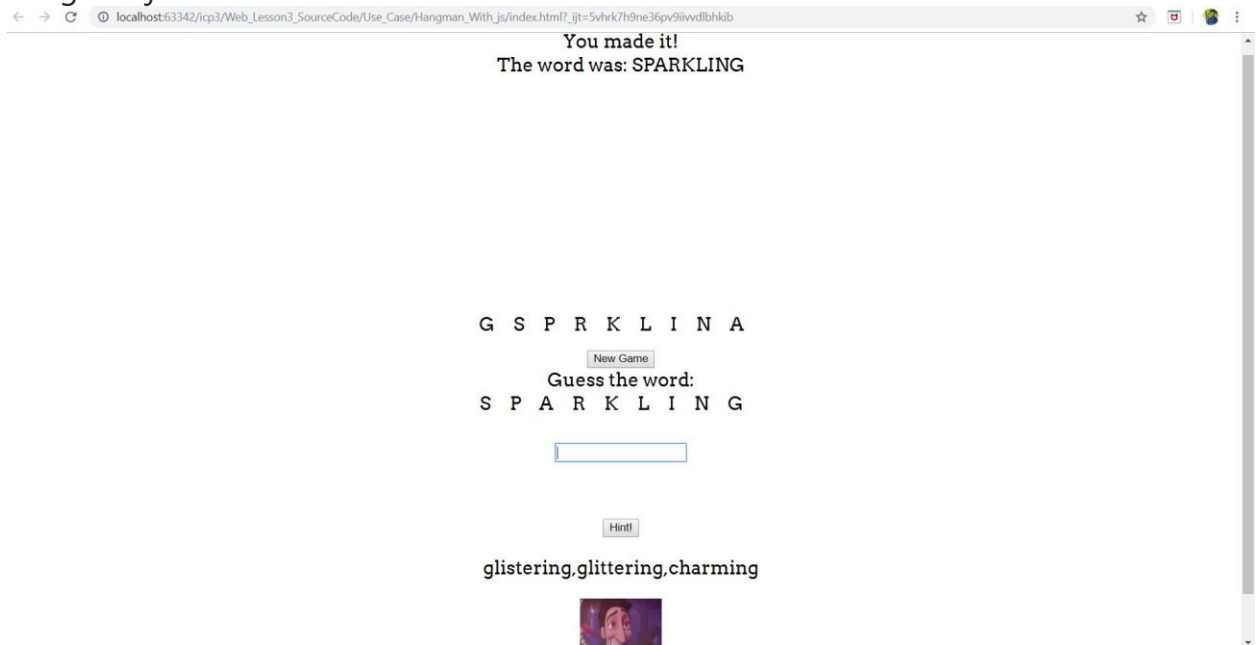
Hangman.prototype.hideElementsByClass = function (elClass) {
    var elements = document.getElementsByClassName(elClass), i;
    for (i = 0; i < elements.length; i++) {
        elements[i].style.opacity = 0;
    }
};

Hangman.prototype.getGuessedWord = function () {
    var result = "", i;
    for (i = 0; i < this.WORD.length; i++) {
        // Word characters
        result += (this.GUESSES.indexOf(this.WORD[i]) > -1) ?
            this.WORD[i] : "_ ";
    }
    return result;
};

return new Hangman('hangman');
})();

```

- Output for hangman.js:



## Objective-4.1(Using Hangman API obtaining the given word):

- The code snippets for hangman API using jquery,java script and html to provide word using hint shown .

```
<html>
<head>
  <link rel="stylesheet" href="style.css">
  <title>Hangman</title>
  <script type="text/javascript" src="jquery.js"></script>
  <script type="text/javascript" src="script.js"></script>
</head>
<body>

<div id="container">
  <div class="board">
    <div class="hangman-word"> </div>
    <div class="remaining-guesses">
      <span class="remaining">7</span> guesses left
    </div>

    <div class="console">
      <input type="hidden" class="token"> </input>
      <input type="text" maxlength=1 class="letter"> </input>
      <button id="guess">Guess</button>
    </div>
    <div class="attempts"></div>
    <button id="new-game">Start a new game</button>
  </div>
  <br/>
</div>
</body>
</html>
```

```

/* sourceMappingURL=jquery-2.0.3.min.map
*/
(function(e,undefined){var t,n,r=typeof undefined,i=e.location,o=e.document,s=o.documentElement,a=e.jQuery,u=e.$,l={},c=[],p="2.0.3",f=c.concat,h=c.push;
"inprogress"===i&&(i=n.shift(),r--),i&&("fx"===t&&n.unshift( 0&&"inprogress"),delete o.stop,i.call(e,s,o)),!r&&o.empty.fire(),_queueHooks:func
},delegate:function(e,t,n,r){return this.on(t,e,n,r)},undelegate:function(e,t,n){return l===arguments.length?this.off(e,"**"):this.off(t,e||"**",n)}})

function newGame() {
$.ajax({
  type: "POST",
  url: "http://hangman-api.herokuapp.com/hangman",
}).done(function(data) {
  $('#hangman-word').text(data.hangman);
  $('#token').text(data.token);
}).fail(function(data) {
  console.log(data)
});
}

function guess(token, letter) {
$.ajax({
  type: "PUT",
  dataType: 'json',
  url: "http://hangman-api.herokuapp.com/hangman",
  data: { "token": token, "letter": letter},
  beforeSend: function() {
    $('#letter').prop('disabled', true);
  }
}).done(function(data) {
  $('#hangman-word').text(data.hangman);
  $('#token').text(data.token);
  if (!data.correct) {
    failures = $('#wrong').length+1;
    handleFailure(failures);
  } else {
    if (data.hangman.indexOf("_") == -1) {
      $('#console').hide();
    }
  }
});
}

```

- Output for hangman API:

h i s t o p a t h o l o g y

t, e, w, q, u, i, k, a, v, h, r

Start a new game

## Conclusion:

- Hence I got insights about to develop web pages using jquery and java script.