jQuery Elements

// Comments show the vanilla JavaScript equivalent statements

// var rootEl = document.getElementById("root");

var rootEl = $('#root');

// var titleEl = document.createElement("h1");

var titleEl = $('<h1>');

// titleEl.textContent = "Hello friends";

titleEl.text('Hello friends');

// titleEl.className = 'fancy';

titleEl.attr('class', 'fancy');

// titleEl.classList.add('p-5') - (`p-5` is for padding)

titleEl.addClass('p-5');

// titleEl.style.border = "rgb(122, 242, 242) 3px solid";

titleEl.css('border', 'rgb(122, 242, 242) 5px solid');

// rootEl.appendChild(titleEl);

rootEl.append(titleEl);

// titleEl.append("Welcome to jQuery");

rootEl.append('<h2>With jQuery we can:</h2>');

var abilities = [

  'Select',

  'Create',

  'Style',

  'Animate',

  'Traverse',

  'Event Listen',

  'much more',

];

for (var i = 0; i < abilities.length; i++) {

  // Create a new `<div>` for each ability and its text content

  var abilityEl = $('<li>');

  // abilityEl.textContent = abilities[i];

  abilityEl.text(abilities[i]);

  // `my-3` class adds margin on top and bottom

  abilityEl.addClass('my-3');

  // Alternatively

  // var abilityEl = $("<div>" + abilities[i] + "</div>");

  // Add this new `<div>` to the abilities `<div>` container element.

  rootEl.append(abilityEl);

}

// select all `<div>` elements on the page

// document.querySelectorAll('div');

console.log($('div'));

// Alternatively use the `$.each` method

// $.each(abilities, function(i, ability) {

//    abilityEl.append("<div>" + ability + "</div>");

// })

Click Events

var alertButtonEl = $('#alert-btn');

var themeButtonEl = $('#theme-btn');

var lotteryButtonEl = $('#lottery-btn');

var lotteryNumberEl = $('#lottery-number');

var refreshButtonEl = $('#refresh-btn');

// light theme state

var isDark = true;

// Click event causes alert "Hello World"

// Vanilla JS equivalent: `addEventListener`

alertButtonEl.on('click', function () {

  alert('Hello World');

});

// Click event causes alert light theme toggle

themeButtonEl.on('click', function () {

  if (isDark) {

    $('body').css({ 'background-color': '#d9e9e8', color: '#1a1a1a' });

    isDark = !isDark;

  } else {

    $('body').css({ 'background-color': '#1a1a1a', color: '#d9e9e8' });

    isDark = !isDark;

  }

});

// Click event causes random number

lotteryButtonEl.on('click', function () {

  var random = Math.floor(Math.random() \* 100000000) + 10000000;

  console.log(random);

  lotteryNumberEl.text(random);

});

// Click event causes refresh

refreshButtonEl.on('click', function () {

  location.reload();

});

Form Elements

var formEl = $('#pizza-form');

var firstNameEl = $('input[name="first-name"]');

var lastNameEl = $('input[name="last-name"]');

var emailEl = $('input[name="email"]');

var githubEl = $('input[name="github"]');

function handleFormSubmit(event) {

  // Prevent the default behavior

  event.preventDefault();

  console.log('First Name:', firstNameEl.val());

  console.log('Last Name:', lastNameEl.val());

  console.log('Email:', emailEl.val());

  console.log('GitHub:', githubEl.val());

  // Select all checked options

  var checkedEl = $('input:checked');

  var selected = [];

  // Loop through checked options to store in array

  $.each(checkedEl, function () {

    selected.push($(this).val());

  });

  console.log('Toppings: ', selected.join(', '));

  // Clear input fields

  $('input[type="text"]').val('');

  $('input[type="email"]').val('');

  $('input[type="checkbox"]').prop('checked', false);

}

// Submit event on the form

formEl.on('submit', handleFormSubmit);

Dom Traversal

// Select elements in the document programmatically

// Starting from the `<div id="top">` element, place a blue border around the title of the website, Develop Community Center

$('#top').siblings().first().addClass('p-5').css('border', '#1bbc9b solid 2px');

// Starting from the `<div id="bottom">` element, get a list of its sibling elements and target the fifth one (the `<ul>` element), then change the text on the last item on the list to say "Meetups", not "GitHub"

$('#bottom').siblings().eq(5).children().last().text('Meetups');

// Starting from `<div id="top">`, get a list of its sibling elements and target the fourth one (the `<ul>` element), and append an `<li>` to add "Classmates" to bottom of the list as a new resource.

$('#top').siblings().eq(4).append($('<li>Classmates</li>'));

// Starting from the `<li id="inside">` element, find the closest parent `<ul>` element and add the classes to all of its child `<li>` elements

$('#inside')

  .closest('ul')

  .children('li')

  .addClass('bg-info text-dark mb-3 p-3');

Event Delegation

var displayEl = $('#display');

var showLettersBtnEl = $('#show-letters-btn');

var buttonListEl = $('#buttons');

var clearEl = $('#clear');

// Immediately hide the clear button

clearEl.hide();

function renderLetters() {

  var letters = [

    'A',

    'B',

    'C',

    'D',

    'E',

    'F',

    'G',

    'H',

    'I',

    'J',

    'K',

    'L',

    'M',

    'N',

    'O',

    'P',

    'Q',

    'R',

    'S',

    'T',

    'U',

    'V',

    'W',

    'X',

    'Y',

    'Z',

    '\_',

  ];

  // Dynamically create buttons

  // Create a for-loop to iterate through the letters array.

  for (var i = 0; i < letters.length; i++) {

    // Create button

    var letterBtn = $('<button>');

    // Assign style to the button

    letterBtn.addClass('letter-button btn btn-info');

    // Assign the letter to the data-letter attribute

    letterBtn.attr('data-letter', letters[i]);

    // Display the letter

    letterBtn.text(letters[i]);

    // Attach the letter element

    buttonListEl.append(letterBtn);

  }

}

// Delegate event listener to the parent element, <div id="buttons">

buttonListEl.on('click', '.letter-button', function (event) {

  var displayLetterEl = $('<div>');

  displayLetterEl.addClass('letter');

  // get letter from clicked letter button's `data-letter` attribute and use it for display

  displayLetterEl.text($(event.target).attr('data-letter'));

  displayEl.append(displayLetterEl);

});

// Renders the letter buttons on click

showLettersBtnEl.on('click', function () {

  // render letters

  renderLetters();

  // hide show letters button

  showLettersBtnEl.hide();

  // display clear button

  clearEl.show();

});

// Clears the message board of letters on click

clearEl.on('click', function () {

  displayEl.empty();

});

Moment Format

// Use Moment.js to format the following variables:

// 1. What is today's date in the following format: Jan 1st, 1999?

var today = moment();

$("#1a").text(today.format("MMM Do, YYYY"));

// 2. What is the day of the week today?

var dayWeek = today.format("[Today is] dddd")

$("#2a").text(dayWeek);

// 3. Parse the following date, 11/3/2020, and convert it into the following format: Sunday, February 14th 2010, 3:25:50 pm.

var reformatDate = moment("11/3/20", "MM-DD-YY").format("dddd, MMMM Do YYYY, h:mm:ss a");

$("#3a").text(reformatDate);

// 4. I need to place my recycling bin on the curb on every odd week of the year for collection. Do I need to put out my recycling bin out this week?

var weekNum = today.format("w");

var takeOut;

// Check odd, then assign boolean

if(weekNum % 2) {

  takeOut = true;

} else {

  takeOut = false;

}

$("#4a").text(takeOut + ", because it's currently week " + weekNum);