Curso Polymer Softtek 

Febrero 2019

**Temario**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Temas | Horas | Planeación |
| 1 | **Introducción**   * Objetivo del curso * Prerequisitos   + Javascript, HTML, Css basico * Conceptos   + Progressive Web Applications, SOLID, TDD * Desarrollo de aplicaciones web   + Ciclo de Request Response * Tips de trabajo Verifica, Valida, Happy path. Siéntete orgulloso del código que has hecho. Trabajo en equipo. Comparte conocimiento. | Día 01  2 horas | 12-feb |
| 2 | **Ambientación**   * Atom, plugins * Node js * Package manager * Chrome Dev Tools, plugins * Ejercicio: Creación de una página básica * HTML5, Javascript * Página HTML5 con separación de js, css * Crear una página con navegación * Laboratorio * Reproducir una pagina   Reference:  <https://www.w3schools.com/html/html5_intro.asp>  <https://www.w3schools.com/jsref/jsref_obj_array.asp>  <https://bower.io/>  <http://caniuse.com> | Día 01  2 horas | 12-feb |
| 2B | * Introducción a Javascript   Jorge: Curso / Repaso / Ejercicios JavaScript | Día 01  4 horas | 12-feb |
| 3 | **Git**   * Links utiles * Comandos para trabajo con repositorio local * Creación de cuenta en GitHub * Creación de Repositorios remoto * Pull requests * Laboratorio   (Este tema depende de si hay acceso a Github)  Links:  <https://github.com/> | Día 02  4 horas | 13-feb |
| 4 | **Javascript**   * Javascript basico: Tipos, operadores, scope, ciclos, clases, DOM, funciones * Debugger, consola. * Ejercicios de logical de programación * Laboratorio * Sintaxis, Tipos de dato, parseo * Ejercicios   + Ejemplo de página para modificar elementos de HTML   + Tipo de cambio Pesos USD   + Crear objeto persona con datos capturados desde pantalla   + Cifrado Cesar   + Validación de correo   + Elementos repetidos   + Saber si un arreglo tiene elementos repetidos   + Saber si 2 números en un arreglo pueden sumar un numero   + <https://www.testdome.com/questions/javascript/date/8521?visibility=1&skillId=2> * Laboratorio   + Agenda | Día 02  Día 03  8 horas | 13-feb  14-feb |
| 5 | **Javascript Avanzado ECMAScript6**   * Uso de $, $$, \_ * Expresiones regulares * Clases * Funciones incorporadas. * Inline functions, arrow functions, eventos, prototipos, futuros, encadenamientos, callbacks * Clojures**,** Intervalos**,** Timeouts**, Workers** * Ejercicio * Crear Todo App + Remove completed Button * IntervalosyTimers * EjercicioAjaxy REST * Laboratorio * Extra: Cookies, Local Storage   Referencias:  <http://es6-features.org/#StatementBodies>  <https://developers.google.com/web/updates/2015/03/introduction-to-fetch> | Día 03  12 horas | 14-feb |
| 6 | **Responsive Design**   * CSS3, Selectores, layouts, Media queries * Flexbox * Grid layout * Agregar Css a HTML * Cambiar CSS con Javascript * Ejercicio   + CSS   + Tutoriales Css Garden   + Ranita   + Grid * Laboratorio   Links:  <http://cssgridgarden.com/>  <https://flukeout.github.io/>  <https://flexboxfroggy.com/#es>  <https://css-tricks.com/snippets/css/a-guide-to-flexbox/> | Día 04  8 horas | 15-feb |
| 7 | **HTML5 Custom Elements**   * Shadow Dom * Clases * Metodos callback * Custom Events * Links utiles * Ejercicios   + Agenda * Laboratorio   + Agenda   Lunes 18  - Curso / Repaso / Ejercicios Polymer  **Links:**  <https://www.html5rocks.com/en/tutorials/webcomponents/customelements/> | Día 05  16 horas | 18-feb |
| 8 | **Polymer 2.0**   * Ambientación * Elementos   + Properties   + Ciclo de vida   + Data binding   + Observers   + Computed values   + Eventos * Familias * Utilerias   + iron-ajax,   + dom-repeat,   + templates filter, sort, if * Aplicaciones   + App templates   + Routing   + Localization   + App storage   + Service worker * Ejercicios * Laboratorio   **Links:**  <https://www.polymer-project.org/2.0/start/quick-tour>  <https://codelabs.developers.google.com/>  <https://polymer-library.polymer-project.org/2.0/docs/devguide/properties>  <https://polymer-library.polymer-project.org/2.0/docs/devguide/data-system>  <https://polymer-library.polymer-project.org/2.0/docs/devguide/events>  **Laboratorios**  Tu primer app progresiva  <https://codelabs.developers.google.com/codelabs/your-first-pwapp/index.html?index=..%2F..index#0>  Polymer First Element  <https://codelabs.developers.google.com/codelabs/polymer-first-elements/index.html?index=..%2F..index#2>  Adivina Bandera  <https://codelabs.developers.google.com/codelabs/whose-flag/index.html?index=..%2F..index#0>  Migración de App  <https://codelabs.developers.google.com/codelabs/migrate-to-progressive-web-apps/index.html?index=..%2F..index#0>  Starter Kit app  <https://codelabs.developers.google.com/codelabs/polymer-es2015/index.html?index=..%2F..index#0> | Día 06  Día 10  16 horas | 19-feb  25-feb |
| 9 | **Integración con Web Services**   * Llamado a un WS REST * Metodos HTTP * Ejercicio * Laboratorio   **Tema opcional** | Día 11  4 horas | 26-feb |
| 10 | **Proyecto o Examen Final**   * Creación de una aplicación completa. * Simulacion de examen   Examen:  <https://glitch.com/edit/#!/delicious-beard?path=frog-component.html:1:4> | Día 12  Día 13  Día 14  24 horas | 27-feb  28-feb |
| 11 | **Precertificacion**  **Hacker Rank** | Día 15  2 horas |  |

1. **Introducción**

**Objetivo del curso:**

* Aprender los fundamentos de CSS, Javascript, HTML5 y otras herramientas de desarrollo
* Aprender Polymer para crear componentes de la capa Front End de una aplicación web.

**Dirigido a:**

* Desarrolladores de aplicaciones Web que necesiten aprender a desarrollar con Polymer y Cells

**Prerequisitos**

* Conocimientos básicos de desarrollo web con javascript, css, html.
* Conocimientos en programación orientada a objetos, principios de diseño.

**Material del curso requerido**

* Todo el material de referencia esta en el internet, se proporcionan links para acceso durante la clase.

**Equipo requerido**

* Laptop por cada participante con al menos 4GB de RAM con permisos de administrador, WiFi habilitado.
* Software :
  + Node JS versión 8.11 (que incluye npm)<https://nodejs.org>
  + Git versión 2.17<https://git-scm.com/downloads>
  + Atom versión 1.27<https://atom.io/>
* Monitor ó proyector, pizarron,
* Acceso a todos los sitios listados es INDISPENSABLE

**Metodología**

* Presentaciones
* Ejemplos
* Ejercicios
* Laboratorio
* Investigación por su cuenta

**Software:**

7zip

Notepad++

Git

NodeJs

Atom

**Acceso a recursos y subrecursos en internet:**

**Sugerencias para el instructor**

Las instalaciones deben tener acceso a todos los links listados al final del documento

**NOTAS**

**Día 01**

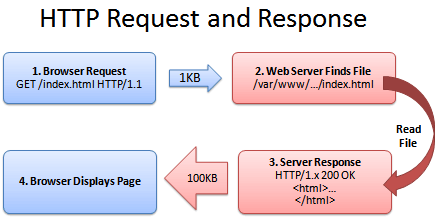
3:02 PM

<https://brianur.info/cifrado-caesar/>

var ascii = "cadena".charCodeAt(0);

var car = String.fromCharCode(65);

Function cifrarCesar(var mensaje, var despl){}



var $ = document.getElementById;

var myElement = $('targetElement');

**Laboratorios**

Tu primer app progresiva

<https://codelabs.developers.google.com/codelabs/your-first-pwapp/index.html?index=..%2F..index#0>

Polymer First Element

<https://codelabs.developers.google.com/codelabs/polymer-first-elements/index.html?index=..%2F..index#2>

Adivina Bandera

[https://codelabs.developers.google.com/codelabs/whose-flag/index.html?index=..%2F..index#0](https://codelabs.developers.google.com/codelabs/whose-flag/index.html?index=..%2F..index" \l "0)

Migración de App

<https://codelabs.developers.google.com/codelabs/migrate-to-progressive-web-apps/index.html?index=..%2F..index#0>

Starter Kit app

<https://codelabs.developers.google.com/codelabs/polymer-es2015/index.html?index=..%2F..index#0>

**Ambientación**

# Verificacion de productos instalados

node -v

Atom

Web Server Chrome

npm -v

git --version

# Preparacion del ambiente Polymer

npm install -g bower

npm install -g polymer-cli

set PATH=C:\Users\jose.iturbide\AppData\Roaming\npm;%PATH%

bower install jquery

bower install --save Polymer/polymer

polymer init

polymer serve --npm

open http://127.0.0.1:<port>/demo/polymer serve --npm

open http://127.0.0.1:<port>/demo/

**Para ejecutar un ejemplo descargado desde Git.**

git clone https://github.com/PolymerLabs/polymer-2-first-element.git

cd polymer-2-first-element

bower install

**Proyecto Final**

Desarrollar una aplicación de Contactos con Polymer 2

La aplicación se debe ejecutar localmente

|  |  |
| --- | --- |
| Resultado de imagen para polymer logo | Aplicación Contactos  IS: JLIL Curso de Polymer, Febrero 2019 |
|  | |
|  | |

Características de la aplicación y que se evaluarán:

**Funcionalidad**

1. Consultar Contacto
   * Se muestra en el panel central, todos los datos deben ser visibles en la 1ª pantalla
2. Agregar contacto
   * Datos a guardar: Nombre, correo, teléfono, IS, superpoder, Imagen. La imagen del superhéroe debe poder elegirse de un catálogo.
3. Eliminar contacto
   * Debe confirmar el borrado antes de realizarlo.
4. Persistencia
   * Se debe persistir los datos en un local storage. Se tendrá una mejor nota una base de datos accedida por WS-REST o
5. Cargar contactos existentes que se hayan dado de alta en la sesión
   * Con Iron Ajax desde archivo o bd remota
6. Persistencia
7. Validaciones
   * Mensajes claros
   * Validación de entrada de texto

**Uso de elementos aprendidos en el curso**

1. Uso de JS, Css, Conceptos:
   * Javascript: Clases, ‘use strict’, arrow functions, modificación de Dom por javascript, Uso de JSON. Expresiones regulares.
   * Css y Responsividad, Media Query. La aplicación debe ser reponsiva, debe mostrar apariencia diferente dependiendo de los siguientes cortes de media query: Movil < 480px, Desktop y Tablet, después de 480px
   * Canceptos: Modularidad imports, Consistencia en el nombre de identificadores y atributos
2. Responsive

El proyecto se debe presentar el día 4-marzo-2019 a las 9:00 hrs