Curso Polymer Softtek 

Febrero 2019

**Temario**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Temas | Horas | Planeación |
| 1 | **Introducción**   * Objetivo del curso * Prerequisitos   + Javascript, HTML, Css básico * Conceptos   + Progressive Web Applications, Principios Diseño, SOLID * Conceptos de Orientacion a objetos * Desarrollo de aplicaciones web   + Ciclo de Request Response * Reglas del curso:   + Trabajo en equipo   + Investigación por su cuenta (No todo va a estar digerido)   + Presentar su mejor trabajo. En el proyecto se deben hacer validaciones | Día 01  4 horas |  |
| 2 | **Ambientación**  **HTML5, CSS**   * Atom, plugins: file-icons, emmet, color-picker, highlight-selected * Node js, Git, Apache Cordova, * Package manager: npm, bower * Chrome Dev Tools   + Validar plugins postman, Chrome web server, responsive   + Debugger, application, cache, shadow dom, service worker * Ejercicios   + Folder 01Html5   + HTML5, Javascript   + Página HTML5 con separación de js, css * Laboratorio: Creación de una página básica   + Crear una página con navegación   Referencias:  <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  4 horas |  |
| 3 | **Javascript Básico**   * Javascript basico: Sintaxis, Tipos de variable, Operadores, Scope, Ciclos, Objetos, API, Clases, DOM, funciones, arreglos, eventos, Excepciones, Parseo * Debugger, consola. * Ejercicios de logical de programación * 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   + Testdome   + HackerRank | Día 02  8 horas |  |
| 4 | **Javascript Avanzado**  **ECMAScript6**   * Uso de $, $$, \_ en métodos y propiedades. * Expresiones regulares * Clases * JSON * Funciones incorporadas y de utilería. * Inline functions, arrow functions, eventos, prototipos, encadenamientos, callbacks * Clojures**,** Intervalos**,** Timeouts**, Service Workers** * Fetch, Promesas, * IntervalosyTimers * Ajaxy REST * Ejercicio   + Crear Todo App + Remove completed Button * Laboratorio   + Agenda que agrega y consulta contactos   + Extra: Cookies, Local Storage, Session Storage * Examen:   + Ejercicios Testdome de Javascript   Referencias:  <http://es6-features.org/#StatementBodies>  <https://developers.google.com/web/updates/2015/03/introduction-to-fetch> | Día 03  12 horas |  |
| 5 | **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/>  Responsive patterns  <https://developers.google.com/web/fundamentals/design-and-ux/responsive/patterns> | Día 04  4 horas |  |
| 7 | **HTML5 Custom Elements**   * Shadow Dom * Clases * Metodos callback * Custom Events * Links utiles * Ejercicios   + Ejemplos * Laboratorio   + Agenda   **Links:**  <https://www.html5rocks.com/en/tutorials/webcomponents/customelements/> | Día 06  8 horas |  |
| 8 | **Polymer 2.0 Basico**   * Ambientación * Conceptos   + Properties   + Ciclo de vida   + Data binding   + Observers   + Computed values   + Eventos   + Shadow DOM * Elementos   + Iron elements, paper elements, API elements * Utilerias   + iron-ajax, iron-form,   + dom-repeat, dom-if, dom-bind, filtering and sorting   + Actualizar estilos desde Polymer this.updateStyle()   + this.set(), this.splice(), this.push() * Aplicaciones   + App templates   + Routing   + Localization   + App storage   + Service worker * Ejercicios * Laboratorio   **Temas pendientes de incluir:**   * : Composición, Mixing, Herencia, Slot, Fallback, * Styling, theming (variables css), mixing properties * Complex observers * **Pollyfills** * **Apache Cordoba**   **Ejercicio Scrum**  **Proyecto Codigos postales**  3 componentes:   * Campos de búsqueda y tabla de resultados * Componente de consulta * Backend de información * Clase y estructura de datos   **Al final intercambiar el componente de consulta para validar**  **Links:**  <https://platzi.com/blog/caracteristicas-polymer/>  <https://bbvaopen4u.com/es/actualidad/web-components-presente-y-futuro-en-el-desarrollo-web>  <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>  <https://bbvaopen4u.com/es/actualidad/web-components-presente-y-futuro-en-el-desarrollo-web>  <https://platzi.com/blog/web-components-polymer/>  <https://developers.google.com/web/fundamentals/architecture/app-shell>  <https://developers.google.com/web/fundamentals/design-and-ux/responsive/patterns>  **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>  **Codigos postales**  <https://github.com/Munett/API-Codigos-Postales>  <https://api-codigos-postales.herokuapp.com/v2/codigo_postal/09880>  <https://github.com/redrbrt/sepomex-zip-codes> | Día 07  40 horas | 19-feb  25-feb |
|  | **Polymer 2.0 Avanzado**  Shadow DOM & styling  Shadow DOM concepts  DOM templating  Style shadow DOM  Custom CSS properties  Events  Handle and fire events  Gesture events |  |  |
| 9 | **Extras**   * Cambios en Polymer 3 * Seguridad * Patrones de Diseño * Uso de Sesiones * OAuth   **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 |  |
| 3 | **Git**   * Links utiles * Comandos para trabajo con repositorio local * Creación de cuenta en GitHub * Creación de Repositorios remoto * Pull requests * Laboratorio   Este Tema se deja como opcional porque no hay permiso de usar un repositorio Git  Links:  <https://github.com/> | Día 02  4 horas | 13-feb |

Manejo de configuracion desde archivos

Carga de archivos

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**

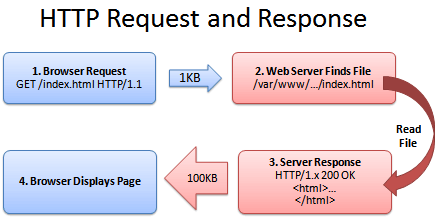
3:02 PM

<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>

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**

# Instalacion de SW base y Verificacion de productos instalados

* Node

node –v

* Atom
* Web Server Chrome
* Git

npm –v

git --version

# Instalacion de herramientas de Polymer para ambiente de Polymer

npm install -g bower

npm install -g polymer-cli

git config --global –edit

# Configurar esta variable para

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

set ANDROID\_SDK=C:\Users\jose.iturbide\AppData\Local\Android\Sdk

../bin/sdk licenses

bower install jquery

bower install --save Polymer/polymer

polymer init

# Para Polymer 3.0

polymer serve --npm

polymer serve --open

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

# Instalar

npm install –g python-shell

npm install –g python

**# Install Gulp**

npm install –g gulp-cli

npm install gulp –D

npx -p touch nodetouch gulpfile.js

gulp –help

Polymer 3

**NO** npm install --save PolymerElements/paper-button

npm install --save @polymer/paper-button

npm install --save @polymer/iron-demo-helpers

#Cordova

npm install -g cordova

Firebase

* Cuidar API key
* Usar realtime o base
* <https://picsum.photos/300/300?random>
* Regexp Test

npm install --save-dev sw-precache

<https://bbva-files.s3.amazonaws.com/cells/bbva-catalog/index.html#/?v=1.7.0>