GUÍA

FRAMEWORK

Angular.JS

Conociendo más sobre JavaScript

**alumno:**

quISPE GUEVARA, RENZO DE JESÚS

**CURSO**:

LENGUAJE DE PROGRAMACION ORIENTADO A WEB

**DOCENTE DEL CURSO:**

ING. SAULO PINEDA

2017

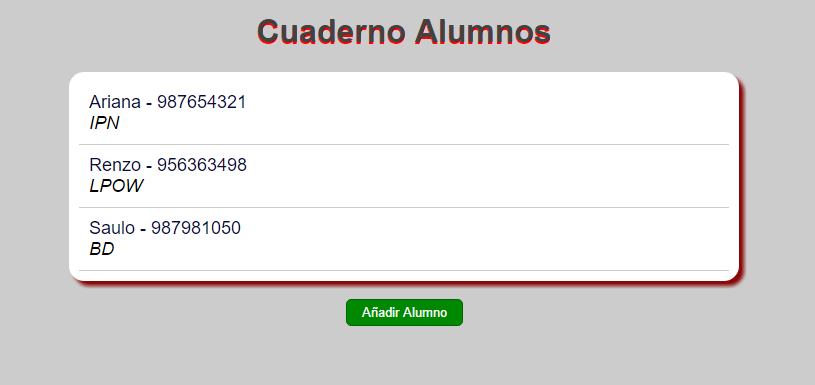
Facultad de Ingeniería Industrial y de Sistemas

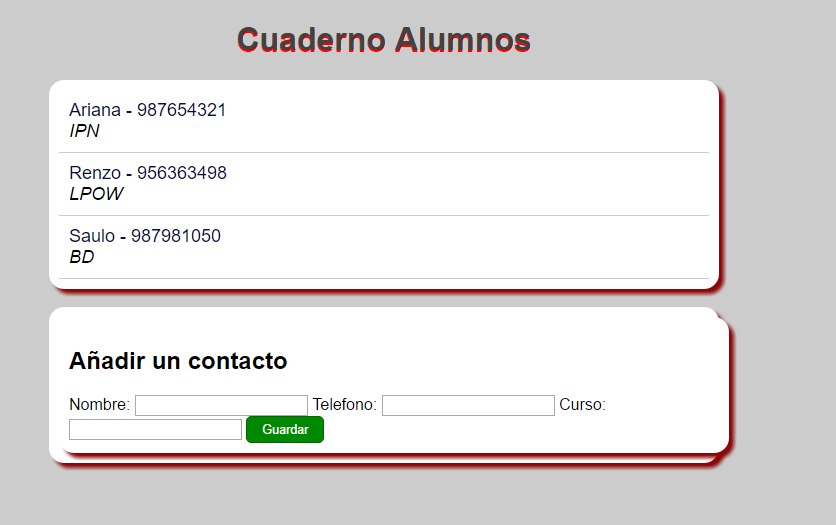
Universidad Nacional Federico Villarreal

03/02/2017

**Guía de Laboratorio**

**Objetivo:** Aprender el framework AngularJS con un ejemplo básico de crear un Formulario de un listado de Alumnos como se muestra a continuación:

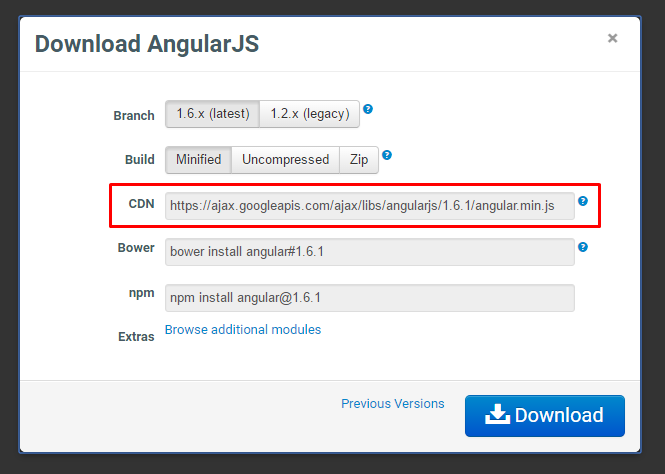




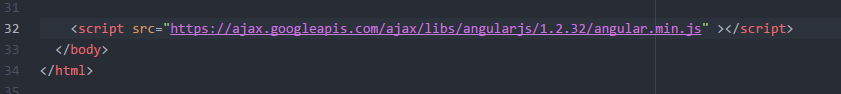
1. Para comenzar a utilizar AngularJS debemos tener la idea de cómo descargarlo para luego usarlo, es para eso se recomienda ir a la página <https://angularjs.org/>
2. Aquí en la página le dan clic al botón de Download.



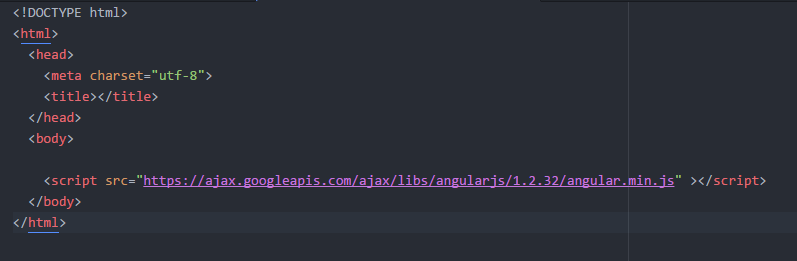
1. Al hacer esto te aparecerá una ventana pequeña donde tienes que copiar el link que se muestra aquí:



1. Lo copiado lo ponemos en nuestro archivo HTML con el nombre “index” al final del <body> y así podremos usar las propiedades y funciones del AngularJS.

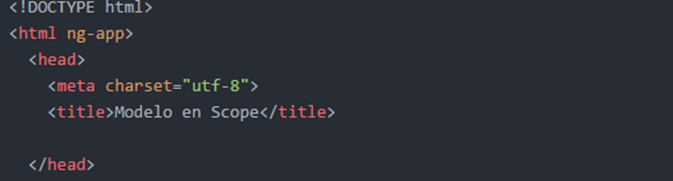


1. Ya declarado esto, podemos empezar a escribir nuestro código con la estructura general:



1. Usamos las propiedades de AngularJS en la cabecera o head:

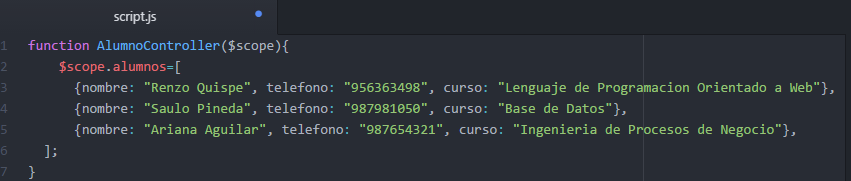
Se incluye dentro de la etiqueta en la que se ejecutará la Aplicación AngularJS



1. En el <body> o cuerpo creamos con más propiedades el ng-controller llamado “AlumnoController” iniciándolo en el mismo body.



1. Ya creado el controlador deberemos crear un modelo y esto lo hacemos con un ng-modelo, creamos un archivo “script.js”, aquí declaramos el modelo y le añadimos sus atributos y valores.



Creamos nuestro modelo dentro del **scope**

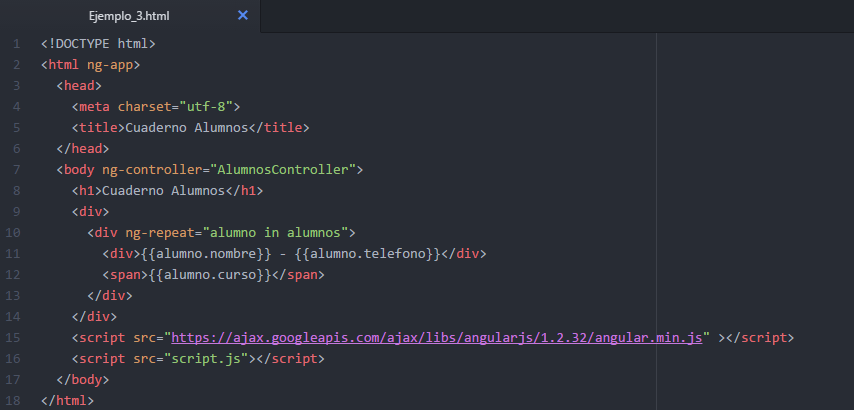
Incluimos la función del controller en un archivo JavaScript

Este modelo contiene 3 atributos: nombre, teléfono y curso

z

1. En el archivo html añadimos:

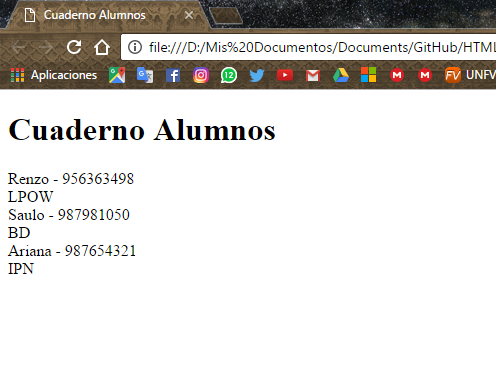
Muestra la lista de Alumnos registrados



Llamado a nuestros script en nuestra carpeta

Llama alumno por alumno de toda la lista

1. Nuestro contenido debe mostrarse así:



1. Para poder añadir diseño a nuestra lista Agregamos un archivo css a nuestro trabajo.



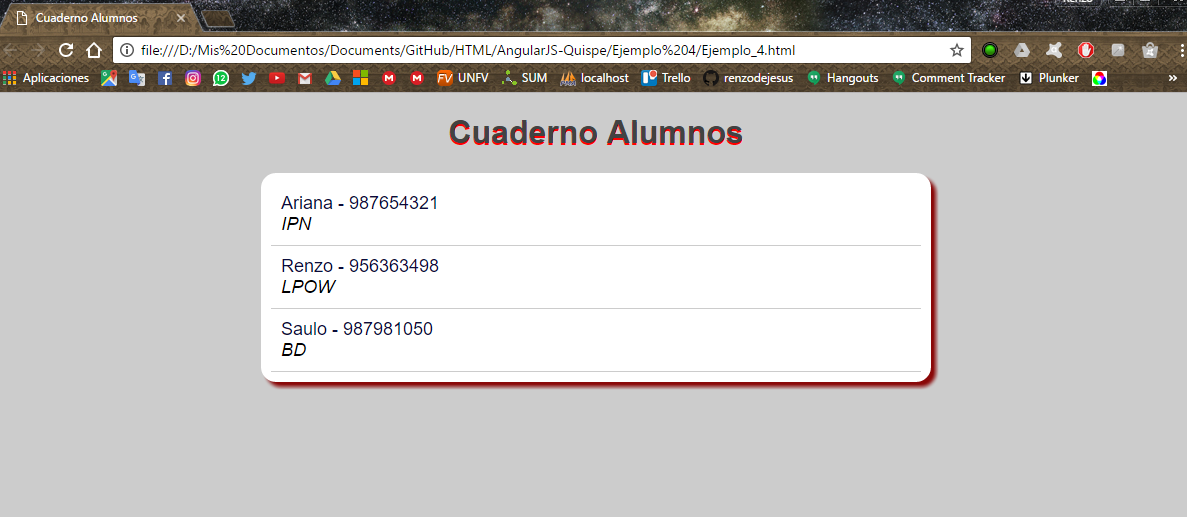
1. En nuestro archivo html agregamos las clases “wraper” y “contact-item“ para añadir el diseño del css y también creamos las clases “nombre” y “curso” para añadir el filtro de ordenar alfabéticamente los nombres como se indica:

Filtro para ordenar alfabéticamente



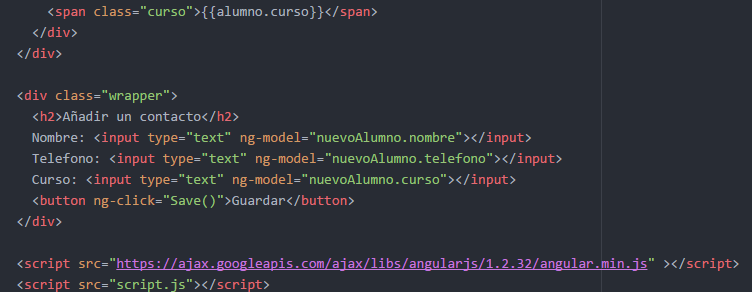
Se crea las clases nombre y curso para poder utilizar el filtro a base de éstas mismas

1. Nuestro contenido debe mostrarse así:



1. Ahora creando el formulario le añadimos la parte de Añadir alumno:

En el html se agrega en un espacio al final del listado, el código para el crear el formulario

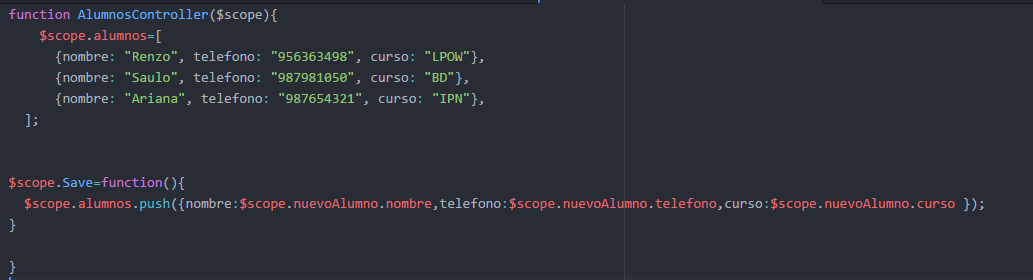


Barra de texto donde se escribirá los datos nuevos

Directiva para función de dar click

Función de Guardar del controller

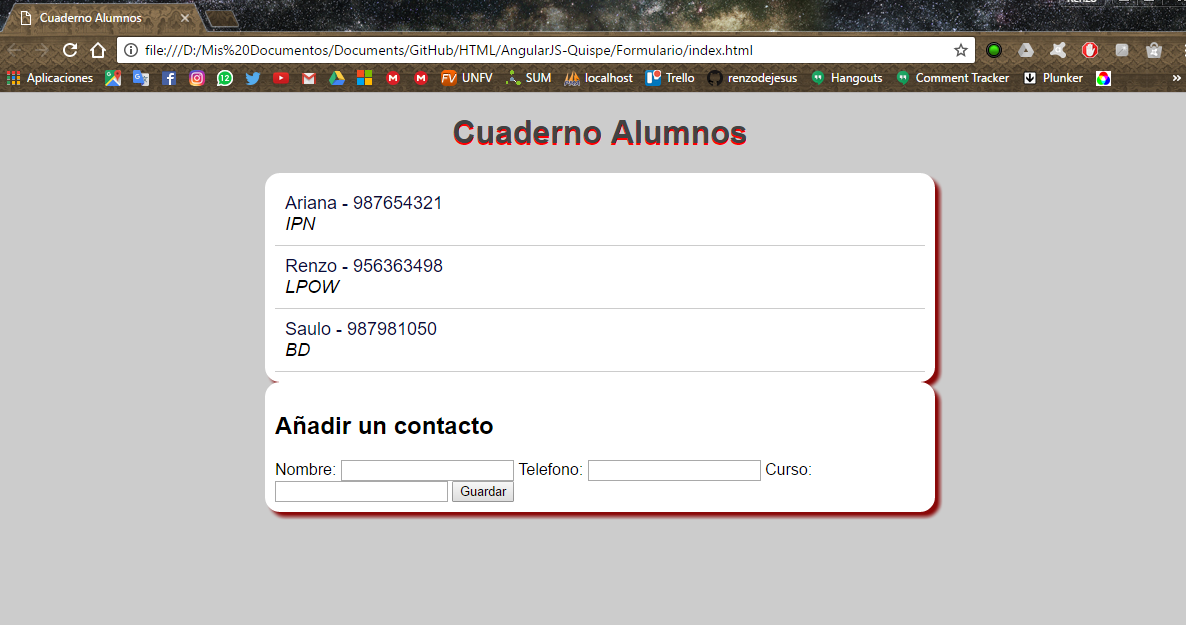
1. Y en el archivo script.js creamos la función “Save()” dentro del Controller que declaramos en el archivo html.



Y en el archivo JS se describe la función o método de guardar, dentro de la función Controller

Es el código que se utiliza para añadir nuevos datos

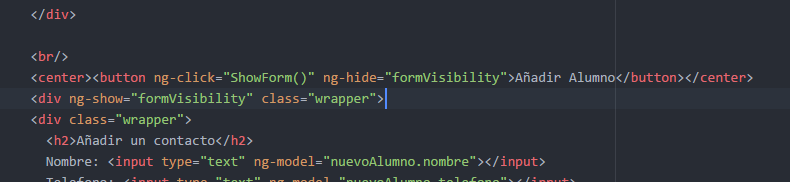
1. Nuestro contenido debe mostrarse así:



1. Y si queremos esconder esta parte de “Añadir contacto” con un solo botón que cuando le damos clic este nos da el formulario, pues debemos añadir las directivas ng-show y ng-hide en el archivo html.

Función del controller para mostrar

Modelo Visibilidad en modo boolean

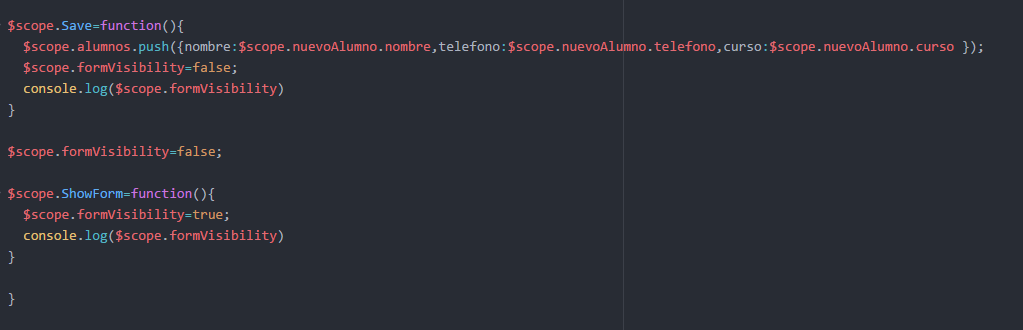


Directiva de Ocultar

Directiva de Mostrar

1. En nuestro archivo script.js describimos la función ShowForm().

Cada vez que se guarde un contacto desaparezca la barra



Función de mostrarse (true) y ocultarse (false)

1. Por último también añadimos en el archivo css diseño para el botón:



1. Finalmente al correr el archivo html se nos debe mostrar así:

