**NOTAS ANGULAR 18**

# Instalación

## Instalar Node.js

Descargar e instalar la última versión LTS de Node.js para Windows desde el sitio oficial: <https://nodejs.org/en/>

## Abrir PowerShell

Abrir ventana de comandos ***PowerShell*** de Windows con permisos de administrador.

## Habilitar la ejecución de scripts en Windows

Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy RemoteSigned

## Instalar Angular CLI

npm install -g @angular/cli

# Comandos generales

## Versión de NodeJS

npm --version

## Versión de Angular

ng --version

## Ejecutar

ng serve

## Compilar

ng build

# Comandos comunes de un proyecto

## Crear nuevo proyecto independiente(Sin necesidad de módulos)

ng new nombre\_app

## Crear nuevo proyecto compartible con módulos

ng new nombre\_app –-standalone=false

## Recrear carpeta ***node\_modules*** o volver a instalar todas las dependencias del proyecto.

npm install

## Crear componente

Ng generate component nombre\_componente

Ó

Ng g c nombre\_componente

## Crear componente en línea(Sin archivo Style y sin archivo Template)

Ng g c nombre\_componente -s -t

## Instalar Bootstrap

npm install bootstrap

Agregar configuración de Bootstrap al archivo Angular.json

"styles": [ "src/styles.css", "node\_modules/bootstrap/dist/css/bootstrap.min.css" ],

"scripts": [ "node\_modules/bootstrap/dist/js/bootstrap.bundle.min.js" ]

# Interpolación | Texto dinámico en HTML

En el siguiente caso se declaran las propiedades y sus tipos, mas no siempre es necesario declarar u tipo.

## Propiedades

Component.ts:

import { Component } from '@angular/core';

@Component({

  selector: 'app-persona',

  templateUrl: './persona.component.html',

  styleUrl: './persona.component.css'

})

export class PersonaComponent {

  nombre: string = 'Juan';

  apellido: string = 'Pérez';

  edad: number = 28;

}

Component.html:

<p>

    Nombre: {{nombre}}

    Apellido: {{apellido}}

    edad: {{edad}}

</p>

## Métodos

Component.ts:

import { Component } from '@angular/core';

@Component({

  selector: 'app-persona',

  templateUrl: './persona.component.html',

  styleUrl: './persona.component.css',

})

export class PersonaComponent {

  private nombre: string = 'Juan';

  private apellido: string = 'Pérez';

  private edad: number = 28;

  getNombre(): string {

    return this.nombre;

  }

  getApellido(){

    return this.apellido;

  }

  getEdad(): number {

    return this.edad;

  }

}

Component.html:

<p>

    Nombre: {{getNombre()}}

    Apellido: {{getApellido()}}

    edad: {{getEdad()}}

</p>

## Template Reference Variable (Repetidor de escritura)

Component.html:

<p>

    Nombre: {{nombre}}

    Apellido: {{apellido}}

    edad: {{edad}}

</p>

<div (keyup)="0">

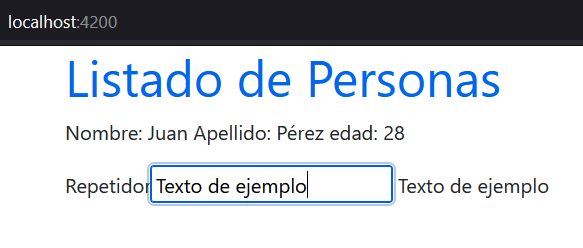
    <label>Repetidor</label>

    <input #entrada />

    {{ entrada.value}}

</div>

Resultado:



## Property Binding en Angular. Asignar valores dinámicos a propiedades HTML

Component.ts

import { Component } from "@angular/core";

@Component({

    selector: 'app-personas',

    templateUrl: './personas.component.html',

    styleUrl: './personas.component.css'

})

export class PersonasComponent{

    deshabilitar = false;

}

Component.html

<div class="container">

    <div class="row">

        <div class="col">

            <h1>Listado de Personas</h1>

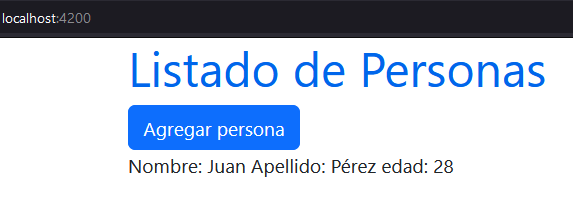
            <button class="btn btn-primary" [disabled]="deshabilitar" Agregar persona</button>

            <app-persona></app-persona>

        </div>

    </div>

</div>

Resultado:

## Event Binding. Escuchar eventos de componentes HTML

Component.ts

import { Component } from "@angular/core";

@Component({

    selector: 'app-personas',

    templateUrl: './personas.component.html',

    styleUrl: './personas.component.css'

})

export class PersonasComponent{

    deshabilitar = false;

    mensaje = 'No se ha agregado ninguna persona';

    agregarPersona(){

        this.mensaje= 'Persona agregada'

    }

}

Component.html

<div class="container">

    <div class="row">

        <div class="col">

            <h1>Listado de Personas</h1>

            <button class='btn btn-primary'

            (click)='agregarPersona()'>Agregar persona</button>

            <p>{{mensaje}}</p>

            <app-persona></app-persona>

        </div>

    </div>

</div>

## Event Binding. Eventos con parámetros

Component.ts

import { Component } from "@angular/core";

@Component({

    selector: 'app-personas',

    templateUrl: './personas.component.html',

    styleUrl: './personas.component.css'

})

export class PersonasComponent{

    deshabilitar = false;

    mensaje = 'No se ha agregado ninguna persona';

    titulo= '';

    agregarPersona(){

        this.mensaje= 'Persona agregada'

    }

    modificarTitulo(event: Event){

        this.titulo = (<HTMLInputElement>event.target).value;

    }

}

Component.html

<div class="container">

    <div class="row">

        <div class="col">

            <h1>Listado de Personas</h1>

            <button class='btn btn-primary'

            [disabled]='deshabilitar'

            (click)='agregarPersona()'>Agregar persona</button>

            <p>{{mensaje}}</p>

            <app-persona></app-persona>

            <label>Titulo persona</label>

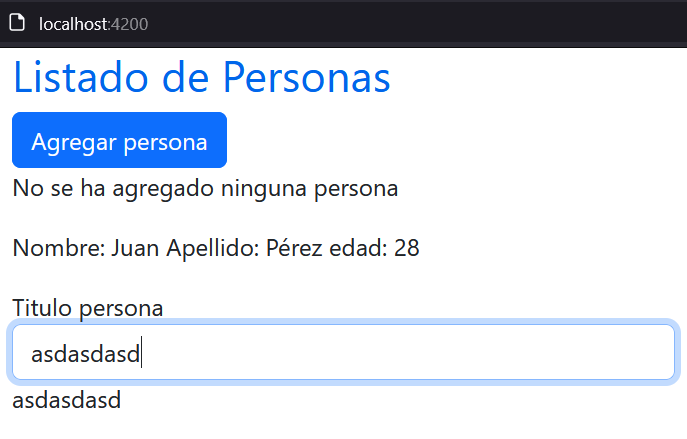
            <input type="text" class="form-control" (input)="modificarTitulo($event)">

            {{titulo}}

        </div>

    </div>

</div>

Resultado:

## Two Way Binding. Enlazar HTML 🡨🡪Componente.

App.module.ts

import { NgModule } from '@angular/core';

import { BrowserModule } from '@angular/platform-browser';

import { AppRoutingModule } from './app-routing.module';

import { AppComponent } from './app.component';

import { PersonasComponent } from './personas/personas.component';

import { PersonaComponent } from './persona/persona.component';

import { FormsModule } from '@angular/forms';

@NgModule({

  declarations: [

    AppComponent, PersonasComponent, PersonaComponent

  ],

  imports: [

    BrowserModule,

    AppRoutingModule,

    FormsModule

  ],

  providers: [],

  bootstrap: [AppComponent]

})

export class AppModule { }

Component.ts

import { Component } from "@angular/core";

@Component({

    selector: 'app-personas',

    templateUrl: './personas.component.html',

    styleUrl: './personas.component.css'

})

export class PersonasComponent{

    deshabilitar = false;

    mensaje = 'No se ha agregado ninguna persona';

    titulo= 'Ingeniero';

    agregarPersona(){

        this.mensaje= 'Persona agregada'

    }

}

Component.html

<div class="container">

    <div class="row">

        <div class="col">

            <h1>Listado de Personas</h1>

            <button class='btn btn-primary'

            [disabled]='deshabilitar'

            (click)='agregarPersona()'>Agregar persona</button>

            <p>{{mensaje}}</p>

            <app-persona></app-persona>

            <label>Titulo persona</label>

            <input type="text" class="form-control"

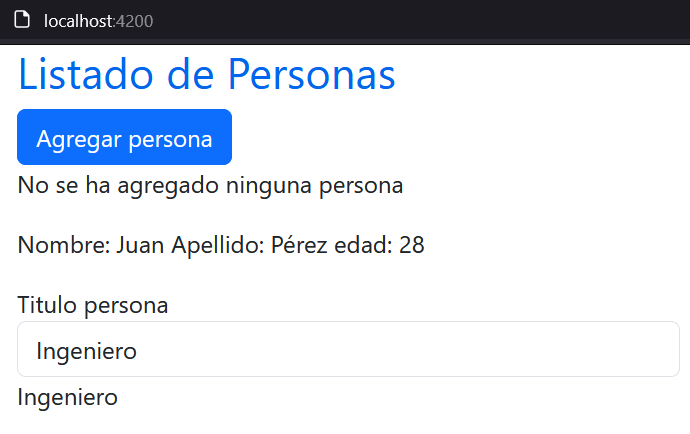
            [(ngModel)]="titulo">

            {{titulo}}

        </div>

    </div>

</div>

Resultado: