



Angular first steps

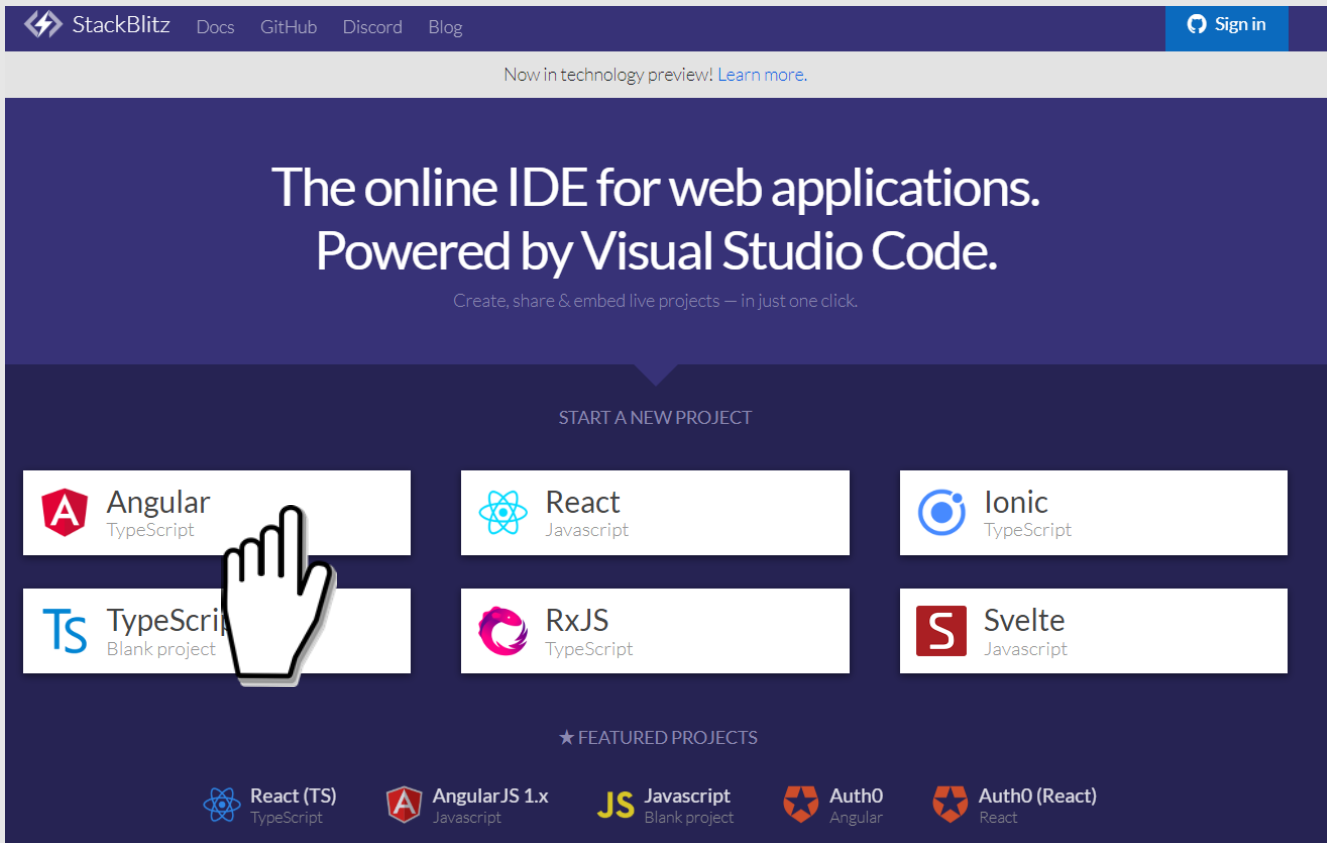
GFT INTERNAL TRAINING

INNOVATE. TRANSFORM. DELIVER.

Agenda

- 1.first steps with stackblitz**
- 2.new app with angular-cli**
- 3.new component**
- 4.new component with angular-cli**
- 5.using directives**
- 6.two-way binding**

first steps with StackBlitz



StackBlitz Docs GitHub Discord Blog


Sign in


Now in technology preview! [Learn more.](#)


The online IDE for web applications. Powered by Visual Studio Code.


Create, share & embed live projects — in just one click.


START A NEW PROJECT


**Angular**
TypeScript

**React**
Javascript


**Ionic**
TypeScript


**TypeScript**
Blank project


**RxJS**
TypeScript


**Svelte**
Javascript


★ FEATURED PROJECTS

**React (TS)**
TypeScript

**AngularJS 1.x**
Javascript

**JS** Javascript
Blank project

**Auth0**
Angular

**Auth0 (React)**
React

The screenshot shows a web-based IDE interface. At the top, there's a navigation bar with a lightning bolt icon, the username '@anonymous', the project name 'angular-z3hdxx', and icons for Save, Fork, and Share. On the right of this bar are links for 'Sign in' and 'Open in New Window' with a 'LIVE' indicator. Below the navigation bar is a sidebar on the left containing icons for file explorer, search, and settings. The file explorer shows a project structure with 'src/app' containing 'app.component.css', 'app.component.html', 'app.component.ts' (selected), 'app.module.ts', and 'hello.component.ts'. The main editor area displays the code for 'app.component.ts' with line numbers 1 through 15. The code defines an Angular component with a selector 'my-app', a template with a heading and two paragraphs, and styles for the heading and paragraph. It also defines an 'AppComponent' class with 'name' and 'twitter' properties. On the right side, there's a panel titled 'GFT Angular Course' with the text 'Hola Brad Green' and 'Twitter account: @bradlygreen'. Below this is a 'Console' section with a 'Clear console on reload' checkbox and a message: 'Console was cleared' and 'Angular is running in the development mode. Call enableProdMode() to the production mode.'

PROJECT

INFO

FILES

- src
 - app
 - app.component.css
 - app.component.html
 - app.component.ts
 - app.module.ts
 - hello.component.ts
- index.html
- main.ts
- polyfills.ts
- styles.css
- angular.json
- New File
- package.json

```
1 import { Component } from '@angular/core';
2
3 @Component({
4   selector: 'my-app',
5   template: `<h1>GFT Angular Course</h1>
6               <p>Hola {{name}}</p>
7               <p>Twitter account: {
8                 {twitter}}</p>`,
9   styles: [ `h1 { color: blue }
10             p { color: green } ` ]
11 })
12 export class AppComponent {
13   name: string = 'Brad Green';
14   twitter: string = '@bradlygreen'
15 }
```

GFT Angular Course

Hola Brad Green

Twitter account: @bradlygreen

Console

☒ Clear console on reload

Console was cleared

Angular is running in the development mode. Call enableProdMode() to the production mode.

new app with angular cli

angular-cli installation

```
console -> npm install -g @angular/cli
```

```
console -> ng --version
```

angular-cli custom installation

```
console -> npm install -g @angular/cli@7.0.2
```

```
console -> ng --version
```


angular-cli upgrading

```
console -> npm uninstall -g @angular/cli
```

```
console -> npm cache clean (if npm <=5)
```

```
console -> npm cache verify (if npm >5)
```

```
console -> npm install -g @angular/cli@latest
```

app prueba1 with angular cli

console -> ng new prueba1

```
? Would you like to add Angular routing? No
? Which stylesheet format would you like to use? (Use arrow keys)
> CSS
SCSS [ http://sass-lang.com ]
SASS [ http://sass-lang.com ]
```

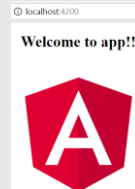
```
create e2e\app.e2e-spec.ts
create e2e\app.po.ts
create e2e\tstconfig.e2e.json
create .gitignore
create karma.conf.js
create package.json
create protractor.conf.js
create tsconfig.json
create tslint.json
Successfully initialized git.
Installing packages for tooling via npm.
Installed packages for tooling via npm.
Project 'prueba1' successfully created.
```

C:\Users\ [redacted] inicio\examples>

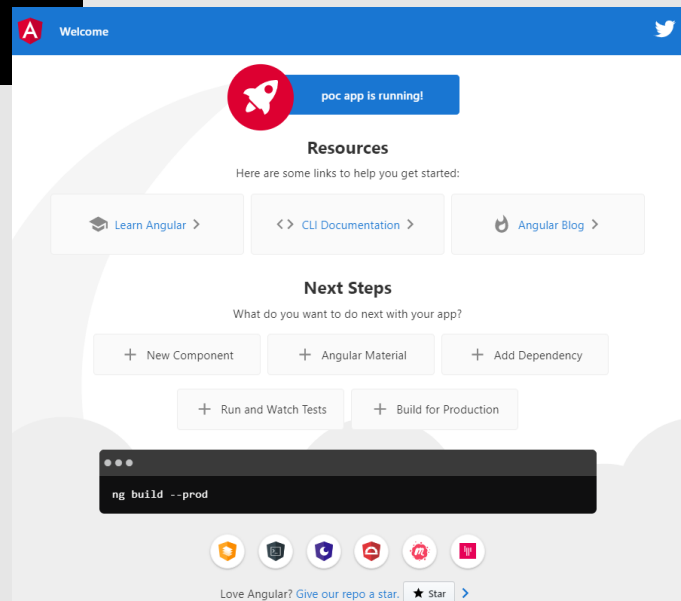
console -> cd prueba1

console -> ng serve

angular 8.0 and older versions



since angular cli 8.3.5



creating a new component

new component saludo

- create file -> saludo.component.ts
- declare component in a module (app.module.ts)
- use its tag

saludo.component.ts

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

@Component({
  selector: 'saludo',
  template: `<h1>Hola {{name}}</h1>`
})

export class SaludoComponent {
  name: string = "Misko Hevery";
}
```


declaration in app.module.ts

```
import { HttpClientModule } from '@angular/http';

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

@NgModule({
  declarations: [
    AppComponent,
    SaludoComponent
  ],
  imports: [
    BrowserModule,
    FormsModule,
    HttpClientModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

use tag in app.component.html

```
<h1>
  {{title}}
</h1>
<saludo></saludo>
```



creating a new component with angular-cli

console -> ng generate component *name*

console (short version) -> ng g c *name*

```
>ng g c control
installing component
  create src\app\control\control.component.css
  create src\app\control\control.component.html
  create src\app\control\control.component.spec.ts
  create src\app\control\control.component.ts
  update src\app\app.module.ts
```

learn by doing

apps works

Hola Misko Hevery

0



```
add() {  
  this.value++;  
}
```



behaviour

```
<button (click)=add()>++</button>
```



event-binding

using directives

learn by doing

Add to DOM/Remove from DOM

Always code as if the guy who ends up
maintaining your code will be a violent
psychopath who knows where you live

Marting Golding

```
<button (click)=toggle()>Add to DOM/Remove from DOM</button>
```

```
<blockquote *ngIf=inDOM> → *ngIf = expression
```

```
  <p>Always code as if the guy  
  who ends up maintaining your code will be a violent psychopath  
  who knows where you live</p>
```

```
  <cite>Marting Golding</cite>
```

```
</blockquote>
```

learn by doing (use *ngFor)

```
superheroes = [
  { name: "batman",
    alterEgo: "bruce wayne",
    superpowers: ["genius intellect", "vast wealth", "indomitable will"]
  },
  { name: "sprocketman",
    alterEgo: "",
    superpowers: ["throws sprockets"]
  },
  { name: "wonder woman",
    alterEgo: "Princess Diana of Themyscira",
    superpowers: ["superhuman strength", "fly", "uses boomerang tiara"]
  },
  { name: "northstar",
    alterEgo: "Jean-Paul Beaubier",
    superpowers: ["superhuman speed", "fly", "project photonic blasts"]
  },
  { name: "superlopez",
    alterEgo: "Juan López Fernández",
    superpowers: ["superhuman strength", "flight", "X-ray vision"]
  }
];
```

Superheroes List

- batman
 - aka bruce wayne
 - Superpowers
 - genius intellect
 - vast wealth
 - indomitable will
- sprocketman
 - aka
 - Superpowers
 - throws sprockets
- wonder woman
 - aka Princess Diana of Themyscira
 - Superpowers
 - superhuman strength
 - fly
 - uses boomerang tiara
- northstar
 - aka Jean-Paul Beaubier
 - Superpowers
 - superhuman speed
 - fly
 - project photonic blasts
- superlopez
 - aka Juan López Fernández
 - Superpowers
 - superhuman strength
 - flight
 - X-ray vision

<https://stackblitz.com/edit/angular-ngfor-todo?file=src/app/app.component.ts>

two-way binding

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

```
@NgModule({  
  declarations: [  
    AppComponent  
  ],  
  imports: [  
    BrowserModule,  
    FormsModule  
  ],  
})
```

in order to use two-way
data binding you must
import FormsModule

Enter a superhero name:

wolfer

Enter a superhero name: `<input [(ngModel)] = name>
`



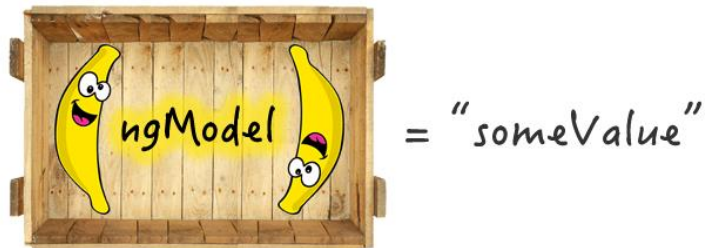
`<h2>{{name}}</h2>`

two-way data binding

two way data binding

[()] = BANANA IN A BOX

`[(ngModel)]` = "someValue"



learn by doing (use ngSwitch)

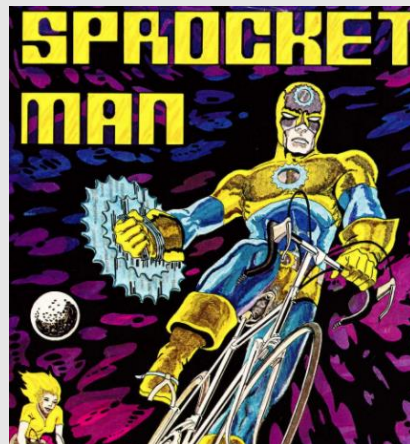
Enter a superhero name:

wolverine



Enter a superhero name:

sprocketman



Shaping the future of digital business

GFT Internal Technical Training

Eduardo García Ibaseta

eduardo.garcia-ibaseta@gft.com
+34 935 639476

GFT IT Consulting, S.L.

Av. Alcalde Barnils, 69-71

08174 Sant Cugat del Vallès (BARCELONA)