# Cour Angular

27/01/2020

Angular = framework JS

Principaux concurrents : React et Vue

Angular JS = Angular 1.0

Angular = Angular 2+

## Historique/avantage Angular :

Grosse réduction du nombre du ligne de code ( ex : Projet google, Google FeedBack initialement codé en JS avait atteint 17000 lignes de code avant que google ne créer Angular en 2012 ce qui a reduit le nombre de ligne a 1500. 2018 sortie de la dernière version d’Angular (6.0)

## Comment ça marche :

* Code en TypeScript.
* Node.JS = serveur qui transcripte le TypeScript qui n’est pas compréhensible par le navigateur en JavaScript compréhensible par ce dernier.
* Le Component  = 3 fichiers :
  + HTML : la vue
  + TS : le code / le model
  + CSS : le design (=annexe)
* Binding directive = interactions des fichiers TS <-> HTML
* Principe MVVM = Model-Vue-Vue-Model.

## Le component

Une page peut contenir plusieurs components imbriqués. Un component = toujours 3 fichiers ( html,css et TS)

## Arborescence

* Node\_module/ : l’ensemble des modules qui seront utilisé par le projet
* Angular.json : la où on met les fichiers bootstrap/fontossom (le chemin) si on l’utilise
* Src/ :
  + Un dossier par component
  + 4 fichiers
    - Les 3 cités plus haut
    - App\_module.ts : fichier de conf, pour importer des module dans le component
* e2e/ : pour les test unitaire

## Les notions :

### L’interpolation : {{nomVar}}

Dans le fichier app-component.ts :

Ajout de variable et de fonction appelables par leur nom dans la vue avec {{ nomVar/nomFonction() }}

### **Event binding** : liaison par évènement

Dans la vue : <balise (click)= « nomFonction() »></balise>

Dans le TS : nomFonction(){return « plop » ;

### 2 ways binding : liaison bidirectionnelle

* Dans app-module.ts  (1ere etape a ne pas oublier !! )

1. FormsModule dans imports[] de @ngModule
2. *import* {FormsModule} *from* "@angular/forms";

* Dans la vue :

<input [(ngModel)]= « nomVar » >

* Dans le app-component.ts

nomVar :typeVar ;

### Directive structurelle (\*ngFor / \*ngIf )

* \*ngFor= «  …. » : équivalent du for (bla of tab ou bla in tab)
* \*ngIf= « a==42 »

### Directive par attribut

* [ngClass]= « » class = « bleu »
  + <https://angular.io/api/common/NgClass>
* [ngStyle]= « » style= « color :red »
  + <https://angular.io/api/common/NgStyle>

## Le typage :

nomVar :type

nomFonction() :type

Différents types :

* + string
  + number
  + string[ ]
  + boolean
  + any = type foure tout
  + Objet

More : [typescriptlang.org/docs/handbook/basic-types.html](http://www.typescriptlang.org/docs/handbook/generics.html)

## Les tableaux

* Déclaration :

nomTab :type[]=[val,val1,val2..]

* Parcours :
* For classique
* Foreach :

for (let x of tab){

Console.log(x) //affiche le contenue du tableau (x = tab[0] puis tab[1] etc..)

}

* Foreach 2 :

for(let i in tab){

console.log(tab[i]) ; // affiche le contenue du tableau (x = 0 puis 1 puis 2etc…)

}

## Classes d’objet

Dans le terminal : ng g class model/nomClass (🡪 chemin depuis app/) : genère la classe nomClass en créant le dossier Model également

Paramètres du constructeur en public : pas besoin de les déclarer ni les instancier, cela est fait automatiquement

Visibilité par defaut des attributs : public

Importer une classe : dans le .ts : import {NomClass} from ‘./model/nomclass’

* Installer Font Awesome : npm install --save font-awesome angular-font-awesome

+ : rajouter "node\_modules/font-awesome/css/font-awesome.css",

* + Dans angular.json

## Liaison Angular ⬄ API

MySQL plus efficace lorsque les données sont liées entre elles .

NoSQL mieux lorsqu’elles sont indépendantes.

## API RESTFull

Mettre dans le dossier www/mon\_projet/ un fichier .htaccess qui contient la redirection pour toute url inconnue sur index.php.

Contenu index.php => ErrorDocument 404 /demo-api/index.php

Ensuite dans index.php : mettre les règles de gestion/redirection en fonction de l’url entrée.

## Pour aller plus loin / les outils utiles

* Electron.js 🡪 Builder application web (angular) en client lourd (.exe)
  + <https://www.electronjs.org/>
* Ionic 🡪 Builder applications web (angular) en application Android et IOS
  + <https://openclassrooms.com/fr/courses/5098931-developpez-une-application-mobile-multiplateforme-avec-ionic-3/5103726-decouvrez-ionic>
* Semi-cours présentation d’Angular en FR : <https://books.ninja-squad.com/public/samples/Deviens_un_Ninja_avec_Angular_extrait.pdf>

Plugins a connaitre :

* Toogle bootstrap (checkbox avec un on/off)
* Select2 (= multi selection select list)

Templates:

* Envato (payant)
  + Theme forest : themes wordpress
  + Graphic river : template pour documents (cv,carte de visite)
  + Code canyon : codes de scripts
* Tutsplus.com ->how to (il y en a des gratuits)