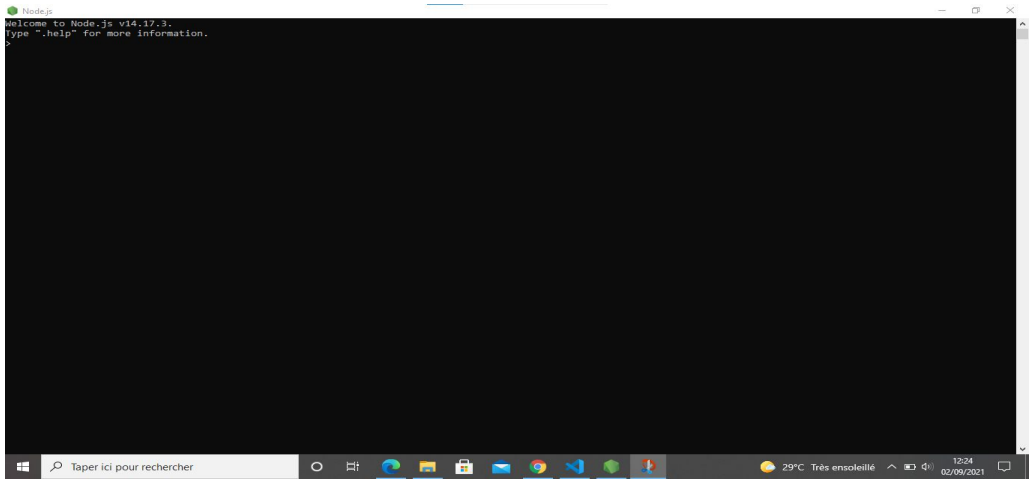
VEILLE SUR ANGULAR ET TYPESCRIPT

- ***Saoudatou DIARRA***
- ***Yacouba DOUMBIA***
- ***Zan COULIBALY***
- ***Seydina Oumar DIARRA***
- ***Ousmane KANE***
- ***Thomas CISSE***
- ***Samba DAOU***
- ***Moussa Sago SIDIBE***
- ***Moussa DIAKITE***
- ***Modibo SAMAKE***
- ***Aminata TRAORE***

a. DÉFINITION

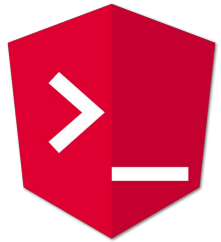
Node.js est un environnement d'exécution JavaScript open source et multiplateforme.

b. ENVIRONNEMENT DE DÉVELOPPEMENT



CLI Angular ?

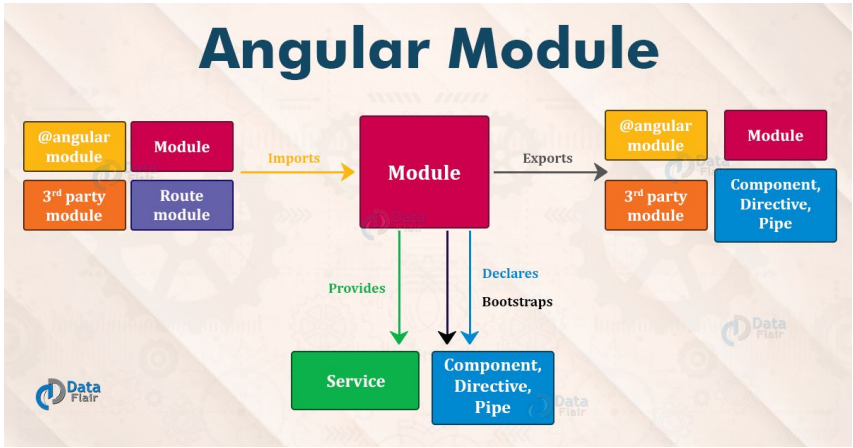
Angular CLI (interface en ligne de commande, en français) est un outil permettant de créer, construire, générer et tester vos applications et librairies Angular.



Un module ?

Un module Angular est un mécanisme permettant de :

- regrouper des composants (mais aussi des services, directives, pipes etc...),
- définir leurs dépendances,
- et définir leur visibilité.



C'est quoi un component?

un composant est un élément réutilisable de l'application, constitué d'une vue et d'un ensemble de traitements associés à cette vue.

C'est quoi un service ?

Un service est une classe TS composée d'attributs et de méthodes, dont l'instanciation est gérée par Angular.

C'est quoi un PIPE ?

Le PIPE a le rôle d'initialiser et modifier l'affichage d'une donnée dans le .component.html

Différents groupes de directives en angular et quelques éléments de celles-ci.

Le fichier “package.json”

Le fichier “package.json” contient des métadonnées pertinentes pour le projet et il est utilisé pour gérer les dépendances du projet, les scripts, les versions.

```
"private": true,
"dependencies": {
  "@angular/animations": "~12.2.0",
  "@angular/common": "~12.2.0",
  "@angular/compiler": "~12.2.0",
  "@angular/core": "~12.2.0",
  "@angular/forms": "~12.2.0",
  "@angular/platform-browser": "~12.2.0",
  "@angular/platform-browser-dynamic": "~12.2.0",
  "@angular/router": "~12.2.0",
  "rxjs": "~6.6.0",
  "tslib": "^2.3.0",
  "zone.js": "~0.11.4"
},
"devDependencies": {
  "@angular-devkit/build-angular": "~12.2.4",
  "@angular/cli": "~12.2.4",
  "@angular/compiler-cli": "~12.2.0",
  "@types/jasmine": "~3.8.0",
  "@types/node": "^12.11.1",
  "jasmine-core": "~3.8.0",
  "karma": "~6.3.0",
  "karma-chrome-launcher": "~3.1.0",
  "karma-coverage": "~2.0.3",
  "karma-jasmine": "~4.0.0",
  "karma-jasmine-html-reporter": "~1.7.0",
```

Création d'un projet Angular

npm install -g @angular/cli : pour installer le CLI angular
ng new nom_projet_angular : pour créer un projet angular
cd nom_projet_angular : pour se positionner dans le répertoire
ng serve --open : pour ouvrir sur le navigateur


TS

C'est quoi TypeScript?

Le TypeScript ou encore 'sur-ensemble de Javascript' est un langage de programmation développé par Microsoft en 2012 permettant d'assigner des types aux variables.

Il représente un "superset" de JavaScript. C'est-à-dire que la syntaxe est la même que celle JavaScript à quelques différences près.

TypeScript vient renforcer le JavaScript en imposant le typage statique des variables.

C'est quoi les normes EcmaScript?

ECMAScript est un ensemble de **normes** concernant les langages de programmation de type script et standardisées par Ecma International dans le cadre de la spécification ECMA-262. Il s'agit donc d'un standard, dont les spécifications sont mises en œuvre dans différents langages de script, comme JavaScript

- ECMAScript(ES) est une norme pour les langages de script.
- Son rôle est d'assurer l'interopérabilité des pages web entre les navigateurs web.

Les types de variables de TypeScript

- **number pour les nombres (entiers, réels, binaires, décimaux, hexadécimaux...) ´**
- **string pour les chaînes de caractère**
- **boolean pour les booléens**
- **array pour les tableaux non-statiques (taille variable)**
- **tuple pour les tableaux statiques (taille et type fixes)**
- **object pour les objets**
- **any pour les variables pouvant changer de type dans le programme**
- **enum pour les énumérations (tableau de constantes)**

Qu'est ce qui explique le typage fort en typescript ?

Un langage est fortement typé si :

1. **La compilation ou l'exécution peuvent détecter des erreurs de typage. Sinon, le langage est dit faiblement typé ;**
2. **Les conversions implicites de types sont formellement interdites. Si de telles conversions sont possibles, le langage est faiblement typé.**

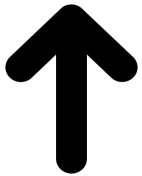
Différence entre les interfaces et les classes

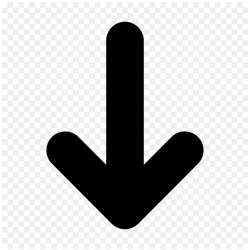
- **Une classe est un plan à partir duquel nous pouvons créer des objets qui partagent la même configuration propriétés et méthodes.**

- **Une interface est un groupe de propriétés et de méthodes associées qui décrivent un objet, mais ne leur fournit ni implémentation ni initialisation.**

Notions : “export” , “import” et “providers”

export devant une classe permet de préparer la classe à être utilisée par d’autres classes





import devant une classe permet d’utiliser la classe dans une classe donnée.

Les fournisseurs (providers) sont utilisés pour enregistrer les classes dans un module angulaire en tant que service. Et puis, ces classes de service peuvent être utilisées par d'autres composants lors de la phase de création elle-même dans le module

C'est quoi un décorateur ?

Un décorateur permet d'ajouter un comportement à un élément lors de l'exécution du code. Les décorateurs prennent la forme de fonctions qui peuvent être par la suite exécutées en utilisant une expression.

Syntaxe : Celle-ci commence avec le caractère @, suivi du nom du décorateur à exécuter

Quelques décorateurs et leurs utilités

@output permet de transmettre les données depuis le composant enfant vers le composant parent

@input permet de transmettre les données depuis le composant parent vers le composant l'enfant

MERCI POUR VOTRE ATTENTION