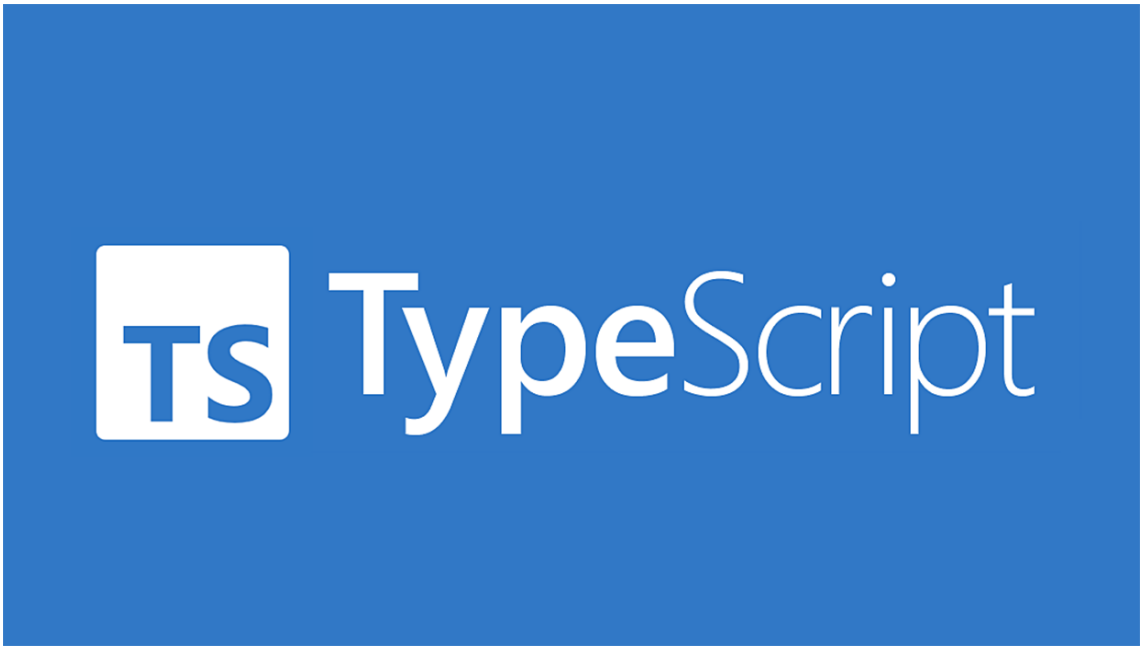

TP2

Nous allons, à travers ce TP, manipuler tous les concepts de **classes, interfaces, héritage et génériques** proposés par `TypeScript`. Des **tests unitaires** sont déjà disponibles, permettant de **tester les différentes étapes de ce TP** dans le dossier `tests`.

Pour chaque étape du TP, on exécutera les tests pour valider que le code répond aux spécifications.
On parle dans ce cas de **TDD** / **Test Driven development**

Toutes les entités sont à créer dans un fichier `src/index.ts`. **Il faudra préfixer toutes les classes, interface, enum ... par le mot clé** `export`

Exemple :

```
export class Musician {
  ...
}

export enum Music {
  ...
}
```

Ceci permet de pouvoir tester notre code depuis les différents fichiers de tests.
Pour le moment, faite abstraction de ce mot clé nous verrons ça dans le prochain chapitre.

La classe Musician

Définissez une `enum Music` lien contenant les clés **JAZZ** et **ROCK**.

Définissez une `classe Musician` lien ayant les propriétés suivantes :

- **firstName** (type string)
- **lastName** (type string)
- **age** (type number)
- **style** (type Music ou undefined)

Implémentez, dans la `classe Musician` une **méthode toString** qui doit retourner une chaîne de la forme **firstName lastName**.

Exécutez la commande `npm run test musician`, et vérifier que tous les tests sont vert.

Les classes JazzMusician et RockStar

Définissez deux nouvelles `classes JazzMusician et RockStar` qui doivent hériter de la `classe Musician`.

Ces classes permettront de **définir la propriété style** dans leur **constructeurs** en utilisant les valeurs **Music.JAZZ et Music.ROCK**.

Modifiez le scope de la propriété **style**. Elle doit être définie avec le scope **private**.

Implémentez l'accesseur `get` lien pour cette propriété.

Modifiez la **méthode toString** pour qu'elle retourne une chaîne de caractère de la forme **firstName lastName plays style** lorsque style est défini.

Exécutez la commande `npm run test musician rock jazz`, et vérifier que tous les tests sont vert.

La classe Album

Créez une nouvelle classe Album. Cette classe aura une propriété title de type string et une méthode toString qui retournera title.

Ajoutez à la classe Musician une propriété privée de type Album[] par défaut à [] et implémentez ses accesseurs (get/set).

Nous allons également définir une méthode addAlbum(album: Album). Cette méthode sera définie dans une interface IMusician qui sera implémentée par la classe Musician.

Exécutez la commande `npm run test musician rock jazz album`, et vérifiez que tous les tests sont vert.

Méthode générique display

Pour terminer, nous allons implémenter une **méthode générique** lien `display` (à l'extérieur de la classe Musician) :

Créez une `interface ElementToString` lien qui a une `méthode toString` retournant une **string**.

La **méthode display** prendra un **tableau d'objets de type générique** qui **héritera de ElementToString** en paramètre.

Pour chaque élément du tableau, nous afficherons sur la console le retour de la méthode toString pour l'objet courant.

Exécutez la commande `npm run test`, 🎉 Tous les tests unitaires doivent passer.