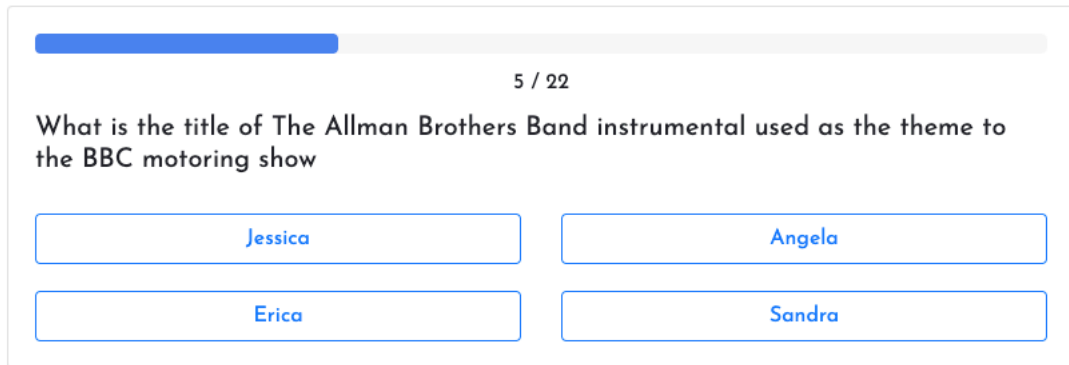


TP-08

ANGULAR-TEMPLATES-BASICS

Objectif du TP:

- Création d'une application Quizz qui affiche et fait défiler les questions / réponses



A screenshot of a quiz application interface. At the top, there is a progress bar and the text "5 / 22". Below this, a question is displayed: "What is the title of The Allman Brothers Band instrumental used as the theme to the BBC motoring show". At the bottom, there are four buttons representing possible answers: "Jessica", "Angela", "Erica", and "Sandra".

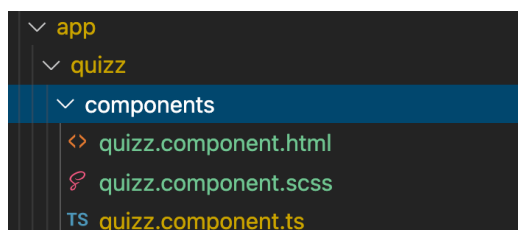
Etapes à suivre :

- Au niveau de l'arborescence du projet, créer le répertoire `src/app/quizz`
- Créer deux sous répertoires de quizz `components` et `core`.
- Créer deux models de donnée dans le fichier `quizz/core/quizz.models.ts`

```
export interface Quizz {  
  question: string;  
  answers: QuizzAnswer[]  
}
```

```
export interface QuizzAnswer {  
  answer: string;  
  isCorrect: boolean;  
}
```

- initialiser les différents fichiers (scss, ts, html) du component QuizzComponent comme suit :



- Au niveau du fichier HTML et SCSS, essayer de reproduire l'interface graphique correspondante au Quizz

- Au niveau de QuizzComponent, créer un attribut public de type Quizz[], et initialiser le avec une liste de questions/reponses (source d'inspiration : https://opentdb.com/api_config.php)
- Au niveau de la Template HTML, parcourir et afficher l'ensemble des question/réponses créés.
- Afficher / cacher les question selon un index que vous allez gérer au niveau de votre componenet (ex : currentQuestionIndex : number)
- Implementer la logique de clique sur les réponses et stocker :
 - Si réponse correcte totalPoints +=10
 - Si le compteur currentQuestionIndex correspond à la fin de la liste des questions, quitter
 - Sinon, incrémenter le compteur currentQuestionIndex, pour afficher la question suivante.
- Mettre en place la logique qui calcule et met à jours la barre de progression ($\text{pourcentage} = \text{currentQuestionIndex} / \text{questions.length} * 100$)