

Exercice N°1 : Hooks Personnalisé (3 pts)

- Créer une fonction **useToggle** dont la valeur initiale (**true**) est passée comme paramètre et qui retourne un tableau qui contient l'état (**isVal**) et la **méthode modificatrice (toggle)** d'un état (**2pts**)
- Créer un composant qui permet de récupérer les données retournées et de modifier l'affichage du bouton en cliquant sur le bouton (**1pt**)
 - Si la valeur retournée est true => affiche vous avez choisi la valeur true
 - Sinon => affiche vous avez choisi la valeur false

Exercice N°2 : API (4 pts)

- Créer une fonction qui permet de **retourner une liste des données** à partir d'une API donnée en paramètre (url). (**2 pts**)
- Créer un composant fonctionnel qui permet de **récupérer les données** en utilisant le composant précédent dont l'url est : <https://jsonplaceholder.typicode.com/users? limit=10> et de l'afficher dans un tableau (**2 pts**)

Exercice N°3 : Routage (7 pts)

- Créer un composant **NavBar** qui permet d'afficher le menu suivant : (**1pt**)
 - Accueil
 - ListUser
 - UserParId
- Créer le composant **Accueil** contient un titre ' Accueil' (**1pt**)
- Créer le composant **ListUser** qui permet d'afficher le composant mentionné dans Exercice N°2 (**1,5pts**)
- Créer le composant **UserParId** qui permet d'afficher l'utilisateur dont l'id est passé en paramètre (**1,5pts**)
- Créer un composant **App** qui permet de déclarer des routes (**2pts**)

Exercice N°4 : CRUD (6 pts)

Créer un composant **Filiere**:

- Le composant affiche la liste des filières et un formulaire. (**2 pts**)
- Devant chaque filière, afficher un bouton "Supprimer" qui permettra de supprimer une filière (**2 pts**)
- Le formulaire propose d'ajouter une nouvelle filière à la liste (**2 pts**)

NB : la liste des filières est un tableau de chaîne de caractères

["Developpement Digital", "Infrastructure Digital", "Full Stack", "Application Mobile"]

Exercice N°1 : Hooks Personnalisé (3 pts)

- Créer une fonction **useVal** qui dont la valeur initiale (**val**) est passée comme paramètre et qui retourne un tableau qui contient l'état (**val**) et la méthode **modificatrice (changerVal)** d'un état (**2pts**)
 - on modifie la valeur par : val div 2 si val est pair, on multiplié par 3 plus 1 dans le cas contraire
- Créer un composant qui permet de récupérer les données retournées et de modifier l'affichage du bouton en cliquant sur le bouton (**1pt**)

Exercice N°2 : API avec hooks Personnalisés (4 pts)

- Créer une fonction qui permet de **retourner une liste des données** à partir d'une API donnée en paramètre (url). (**2pts**)
- Créer un composant fonctionnel qui permet de **récupérer les données** en utilisant le composant précédent dont l'url est : <https://jsonplaceholder.typicode.com/posts? limit=10> et de l'afficher dans un tableau (**2pts**)

Exercice N°3 : Routage (7 pts)

- Créer un composant **NavBar** qui permet d'afficher le menu suivant : (**1pt**)
 - Accueil**
 - ListPost**
 - PostParId**
- Créer le composant **Accueil** contient un titre 'Accueil' (**1pt**)
- Créer le composant **ListPost** qui permet d'afficher le composant mentionné dans Exercice N°2 (**1,5pts**)
- Créer le composant **PostParId** qui permet d'afficher le post dont l'id est passé en paramètre (**1,5pts**)
- Créer un composant **App** qui permet de déclarer des routes (**2pts**)

Exercice N°4 : CRUD (6 pts)

Créer un composant **Module** :

- Le composant affiche la liste des modules et crée un formulaire. (**2pts**)
- Devant chaque module, afficher un bouton "Supprimer" qui permettra de supprimer un module (**2pts**)
- Le formulaire propose d'ajouter un nouvel module à la liste (**2pts**)

NB : la liste des filières est un tableau de chaîne de caractères

["Developpemnt Front-end", "Developpement Back-end", "Méthode Agile", "Gérer les données"]

Exercice N°1 : Hooks Personnalisé (3 pts)

- Créer une fonction **useVal** qui dont la valeur initiale (**val**) est passée comme paramètre et qui retourne un tableau qui contient l'état (**val**) et la méthode **modificatrice (changerVal)** d'un état (**2 pts**)
 - on modifie la valeur par : val multiplié par 2 si val est multiplié par 3, sinon on garde val dans le cas contraire
- Créer un composant qui permet de récupérer les données retournées et de modifier l'affichage du bouton en cliquant sur le bouton (**1pt**)

Exercice N°2 : API (4pts)

- Créer une fonction qui permet de **retourner une liste des données** à partir d'une API donnée en paramètre (url). (**2pts**)
- Créer un composant fonctionnel qui permet de **récupérer les données** en utilisant le composant précédent dont l'url est : <https://jsonplaceholder.typicode.com/photos? limit=10> et de l'afficher dans un tableau (**2pts**)

Exercice N°3 : Routage (7 pts)

- a. Créer un composant **NavBar** qui permet d'afficher le menu suivant : (**1pt**)

Accueil

ListPhoto

PhotoParId

- b. Créer le composant **Accueil** contient un titre 'Accueil' (**1pt**)
- c. Créer le composant **ListPhoto** qui permet d'afficher le composant mentionné dans Exercice N°2(**1,5pts**)
- d. Créer le composant **PhotoParId** qui permet d'afficher la photo dont l'id est passé en paramètre(**1,5pts**)
- e. Créer un composant **App** qui permet de déclarer des routes (**2pts**)

Exercice N°4 : CRUD (6pts)

Créer un composant **Loisir**:

- Le composant affiche la liste des loisirs et crée un formulaire. (**2pts**)
- Devant chaque module, afficher un bouton "Supprimer" qui permettra de supprimer un loisir (**2pts**)
- Le formulaire propose d'ajouter un nouvel loisir à la liste (**2pts**)

NB : la liste des filières est un tableau de chaîne de caractères **["Foot-ball","Lecture","Voyage"]**