

Examen de Fin de Module (Régional)

Module : Développement front-end	Variante N° : 1	
Filière : DEVOWFS	Année : 2025/2026	Durée : 2 heures
Niveau : TS	Mode de formation : Résidentiel	Barème : 40 /40

Partie théorique : 16 points	Note
Dossier 1 :	
1- Quelle instruction JSX est correcte pour afficher une variable message ? A : {message} B : \$message C : { {message} } D : message	(1pt)
2- Dans React, le hook useState sert à ? A : Naviguer entre les pages B : Modifier le DOM directement C : Appeler une AP D : Stocker l'état du composant	(1pt)
3- Pour appliquer un style inline, quelle syntaxe est correcte ? A : style="{color:'blue'}" B : style=(color:"blue") C : style={color: "blue"} D : style={color:"blue"}	(1pt)
4- Quelle bibliothèque est utilisée pour le routage React ? A : react-navigation B : react-dom C : react-router-dom D : redux	(1pt)
5- Quelle méthode est utilisée pour afficher un composant dans la page HTML principale ? A : ReactDOM.display(<App/>) B : React.render(<App/>, root) C : App.render() D : ReactDOM.render(<App/>, root)	(1pt)
6- Quel hook permet d'accéder à une valeur du store Redux ? A : useSelector B : useDispatch C : useEffect D : useStore	(1pt)
7- Dans React, pour exécuter une fonction après le rendu initial, on utilise : A : useDispatch B : useSelector C : useStore D : useEffec	(1pt)
8- Quelle commande permet de créer un projet React ? A : npx create-react-app myapp B : npm install react C : npx react-project D : npm start	(1pt)
Dossier 2 : On souhaite développer une application React permettant de gérer une liste de contacts. L'utilisateur pourra ajouter des contacts, marquer certains contacts comme favoris, et les supprimer.	
1- Donner la commande pour créer un projet React	(1pt)
2- Créer le composant ContactsList.js qui utilise le Hook useState pour gérer l'état des contacts et des entrées de texte et qui : - Permet de récupérer dans une liste les contacts l'API https://api.example.com/contacts - Permet d'ajouter un contact (nom et numéro). - Permet de supprimer un contact de la liste.	(2pts) (2pts) (2pts)
3- Donner le code pour rendre le composant ContactsList dans index.js.	(1pt)
Partie pratique : : 24 points	Note

Nous voulons développer une application React pour gérer une bibliothèque de livres.
Les informations des livres sont récupérées d'une API à l'URL :
https://fake.api.com/books avec la structure suivante :

```
[
  {
    "_id": 1, "title": "Le Petit Prince",
    "category": { "_id": 1, "name": "Littérature" },
    "stock": 12, "price": 50, "favorite": true
  },
  {
    "_id": 2, "title": "Apprendre Python",
    "category": { "_id": 2, "name": "Informatique" },
    "stock": 8, "price": 120,
  },
  ...
]
```

Les composants :

- **Favorite.js** : affiche un cœur indiquant si le livre est favori ou non.
- **Book.js** : une ligne du tableau avec le titre, catégorie, stock, prix, <Favorite/> et un bouton Supprimer.
- **Books.js** : tableau contenant plusieurs <Book/>.
- **Categories.js** : liste des catégories de livres pour filtrer la liste.
- **Paginate.js** : pagination pour naviguer entre les pages de livres.
- **App.js** : composant principal affichant tous les autres.

- 1- Donner Créer le projet React dans le dossier books. (1pt)
- 2- Installer axios. (1pt)
- 3- Créer le composant Favorite.js avec un cœur vide/plein selon le statut. (3pts)
- 4- Gérer le clic sur le cœur pour basculer le favori. (1pt)
- 5- Créer le composant Book.js. (5pts)
- 6- Créer Books.js et récupérer les livres depuis l'API avec axios. (3pts)
- 7- Supprimer un livre via un clic. (1pt)
- 8- Afficher le nombre total de livres. (4pts)
- 9- Créer Paginate.js pour 3 livres par page. (4pts)
- 10- Créer Categories.js pour filtrer par catégorie. (4pts)